Diffusion Module (DICTRA) Documentation Set

Version 2016a

Thermo-Calc Software

Thermo-Calc Software AB or Thermo-Calc Software, Inc..

Norra Stationsgatan 93, SE-113 64 Stockholm, Sweden

+46 8 545 959 30

info@thermocalc.com

www.thermocalc.com

Contents

Console Mode Command Reference 1

Getting Started with the Diffusion Module (DICTRA)

Thermo-Calc Version 2016a

Introduction to the Diffusion Module (DICTRA)

The Diffusion Module (DICTRA) is an add-on module to the Thermo-Calc software package. It is used for simulation of diffusion controlled transformations in multicomponent systems. The simulations are both time- and space-dependent.

The Diffusion Module (DICTRA) is suitable for solving diffusion problems that include a moving boundary (so-called *Stefan problems*). The multicomponent diffusion equations in the various regions of a material are solved under the assumption that thermodynamic equilibrium holds locally at all phase interfaces. The concentration fields only vary along one spherical coordinate, but the geometry may be either planar, cylindrical or spherical.

In this section:

About this Quick Start Guide

The following introduces you to the basic concepts, commands and theory to start using DICTRA. An understanding of this information is necessary to ensure accurate calculations. Use this guide in combination with the online help, *Diffusion Module (DICTRA) User Guide* and the *Diffusion Module (DICTRA) Command Reference*. More details about DICTRA can be found in Andersson et. al[1] and Borgenstam et al.[2] and extensive information about using Thermo-Calc is in the *Help Resources* on page 5.

About the File Formats

The Diffusion Module (DICTRA) uses different file formats: *log*, *macro* and *workspace* files.

Log Files

Log files (*.LOG) are plain text files used to save a sequence of commands. Log files can be edited in a text editor.

Macro Files

Macro files (*.DCM) are plain text files used to save a sequence of commands that can be loaded and executed. Macro files can be edited in a text editor. When creating a command sequence, you add comments to the file by starting a line with @@.

Workspace Files

Workspaces files (*.DIC) allow you to save all the data in your current workspace. A workspace contains all the data, specified settings and the result of any calculations performed. The saved data includes original and modified thermodynamic data, the last set of conditions and options, and the results of any simulations.

Opening the Module and Macro Files

You can go to into the Command Mode to explore the command options, or continue with this quick start guide, which uses three example macro files to demonstrate how to use the software.

Opening the DICTRA Module in Thermo-Calc

1. Open Thermo-Calc in Console Mode.

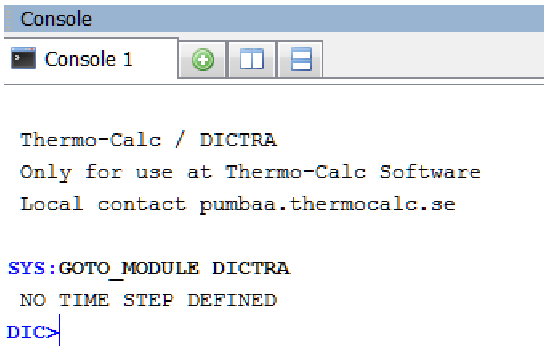
By default, Thermo-Calc opens in the most recently used state. If it opens in Graphical Mode, on the toolbar, click **Switch to Console Mode**.

---

[1] J.-O. Andersson, T. Helander, L. Höglund, P. Shi, and B. Sundman, “Thermo-Calc & DICTRA, computational tools for materials science,” Calphad, vol. 26, no. 2, pp. 273–312, Jun. 2002

[2] A. Borgenstam, L. Höglund, J. Ågren, and A. Engström, “DICTRA, a tool for simulation of diffusional transformations in alloys,” J. Phase Equilibria, vol. 21, no. 3, pp. 269–280, May 2000

2. In the `SYS` module, type `GOTO_MODULE DICTRA`. For a list of all the available commands in the current module, at the prompt type a question mark (?) and press <Enter>. For a description of a specific command, type `Help` followed by the name of the command.



Opening Macro Files

The default directory where the software and the examples are installed vary by platform and user type. On Windows, once Thermo-Calc is installed, go to **Start → All Programs >Thermo-Calc** and click **Examples** to open the folder.

The examples are in the first folder called **Beginner Guide**.

For other platforms, search the online help for *Default Directory Locations*.

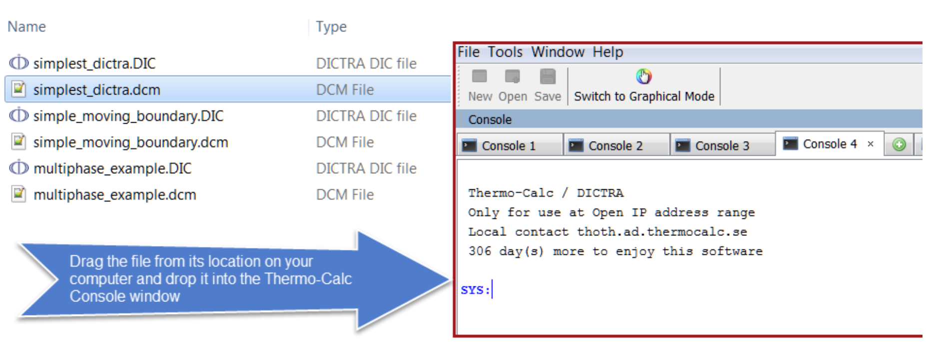
You can open the example macro files in these ways.

Using a Command in Console Mode

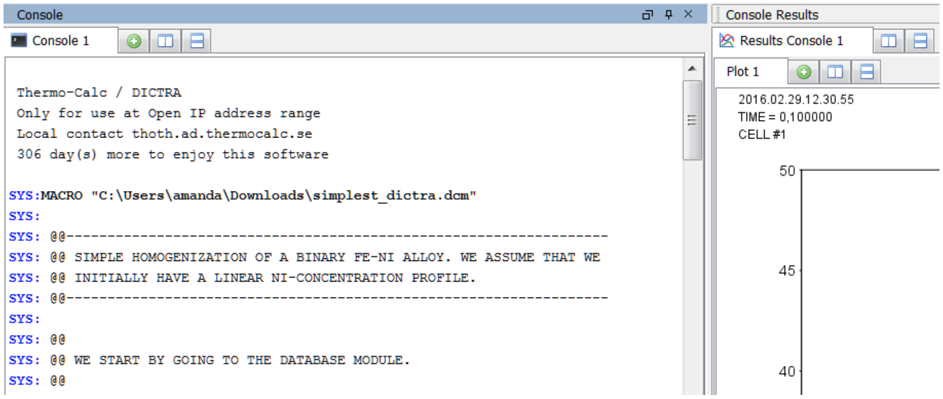
At the `SYS` prompt, type MACRO_FILE_OPEN and press <Enter>. In the file dialogue window that opens, locate and select the macro file on your computer.

Double-click to open or drag and drop a file into Console Mode

Navigate to the **Beginner Guide** folder and then drag the file from its location into Thermo-Calc Console window.



The macro automatically runs after you drop it into the Console window.



Basic Terminology

The following definitions are useful to help you understand the examples and commands in this guide. For more detailed information search or browse the online help (**Help →Online Help**).

Cells and Regions in a System

A system normally consists of one *cell*. This cell contains one or several *regions* in which the diffusion problem is to be solved. A system of diffusion equations is solved for each region. The size of the system is the sum of sizes of all the regions in all the system's cells.

Regions and Grids

A *region* contains one or more phases. If a region contains several phases one of those must be entered as a *matrix phase* whereas the others are entered as *spheroid*.

A region must also contain a number of *grid points*. The composition is only known at these grid points and the software assumes that the composition varies linearly between them. The amount and composition of all the phases present at a single grid point in a certain region are those given by thermodynamic equilibrium keeping the over-all composition at the grid point fixed.

Phases

Each region contains one or more *phases*. A phase can be of the *matrix*, *spheroid* or *lamellar* type. A phase can be introduced with the status *active* (the default) or *inactive*.

Composition

Two types of composition variables are used: site-fractions and u-fractions, although you can also use weight fraction or mole fraction, for example.

The *site-fractions* are used to set up the problem and interface with POLY-3.

The *u-fractions* are used in the diffusion equations.

Geometry and Coordinates

The Diffusion Module (DICTRA) can only handle diffusion problems where composition vary along one spatial coordinate. The geometry of your system can be *planar*, *cylindrical* or *spherical*. The examples in this guide use the default planar geometry.

Planar Geometry

This geometry corresponds to an infinitely wide plate of a certain thickness. If the system has a planar geometry, then the lower boundary (the zero coordinate) is at the left side of the system. The upper boundary (the coordinate with the highest value) is at the right side of the system.

Boundary Conditions

By default the system is closed, which means that matter cannot move across its boundaries. However, you can change this setting for both the lower boundary (left side/centre) and the upper boundary (right side/surface) of the system. The examples in this guide use the default closed system.

Setting up a Simulation

Although there are several commands to learn to correctly set up a DICTRA module simulation, the sequence in which to proceed is basically the same:

1. Define and read **thermodynamic** and **kinetic** data.
2. Set **global conditions** (i.e. temperature, T, and pressure, P).
3. Enter **region(s)**.
4. Enter **grid(s)** in region(s).
5. Enter **phase(s)** in region(s).
6. Enter **composition(s)** of phase(s).
7. Set **geometry** (optional – the geometry is planar by default).
8. Set **boundary conditions** (optional – closed boundaries are the default).
9. Set simulation **time**.
10. **Run** the simulation.

For definitions of the above, see *Basic Terminology* on the previous page. The following sections describe each step in more detail for these simulation types:

- *A Single Phase Simulation* on page 9
- *Moving Phase Boundary Simulation* on page 20
- *A Multiphase Simulation* on page 29

The Diffusion Module (DICTRA) References

1. J.-O. Andersson, T. Helander, L. Höglund, P. Shi, and B. Sundman, “Thermo-Calc & DICTRA, computational tools for materials science,” *Calphad*, vol. 26, no. 2, pp. 273–312, Jun. 2002.
2. A. Borgenstam, L. Höglund, J. Ågren, and A. Engström, “DICTRA, a tool for simulation of diffusional transformations in alloys,” *J. Phase Equilibria*, vol. 21, no. 3, pp. 269–280, May 2000.
3. J.-O. Andersson and J. Ågren, “Models for numerical treatment of multicomponent diffusion in simple phases,” *J. Appl. Phys.*, vol. 72, no. 4, pp. 1350–1355, Aug. 1992.
4. S. Crusius, L. Höglund, U. Knoop, G. Inden, and J. Ågren, “On the growth of ferrite allotriomorphs in fe-c alloys,” *Zeitschrift für Met.*, vol. 83, no. 10, pp. 729–738, 1992.
5. J. Ågren, “Numerical Treatment of Diffusional Reactions in Multicomponent Alloys,” *J. Phys. Chem. Solids*, vol. 43, no. 4, pp. 385–391, 1982.
6. S. Crusius, G. Inden, U. Knoop, L. Höglund, and J. Ågren, “On the numerical treatment of moving boundary problems,” *Zeitschrift für Met.*, vol. 83, no. 9, pp. 673–678, 1992.
7. L. Höglund, “Computer simulation of diffusion controlled transformations in multicomponent alloys,” PhD Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 1997.
8. Z. Hashin and S. Shtrikman, “A Variational Approach to the Theory of the Effective Magnetic Permeability of Multiphase Materials,” *J. Appl. Phys.*, vol. 33, no. 10, pp. 3125–3131, 1962.
9. A. Engström, “Interdiffusion in multiphase, Fe-Cr-Ni diffusion couples,” *Scand. J. Metall.*, vol. 24, no. 1, pp. 12–20, 1995.
10. H. Larsson and A. Engström, “A homogenization approach to diffusion simulations applied to α + γ Fe–Cr–Ni diffusion couples,” Acta Mater., vol. 54, no. 9, pp. 2431–2439, May 2006.
11. H. Larsson and L. Höglund, “Multiphase diffusion simulations in 1D using the DICTRA homogenization model,” *Calphad Comput. Coupling Phase Diagrams Thermochem.*, vol. 33, no. 3, pp. 495–501, 2009.

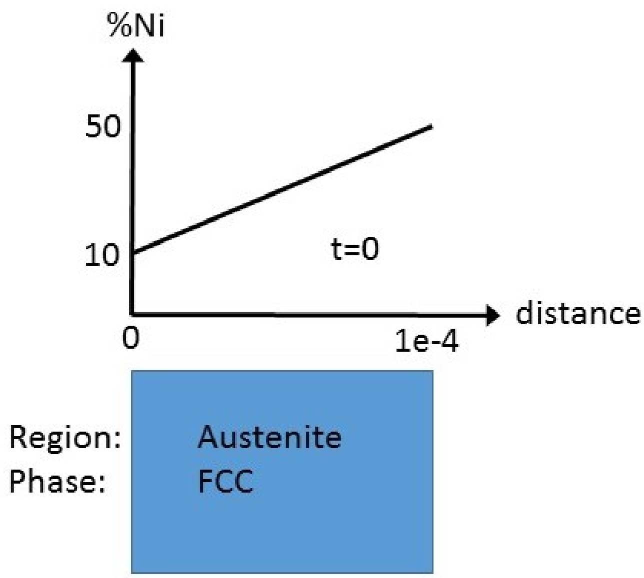
Single Phase Simulations

In this section:

A Single Phase Simulation

In a 100 µm wide planar domain, the single phase example you will run simulates the diffusion of Fe and Ni at a temperature of 1400 K. At this temperature the material is fully austenitic, i.e. the only phase present is the so-called fcc (face centered cubic) phase (in Thermo-Calc the name of this phase is FCC_A1). Initially, there is a linear variation in Ni going from 10 mass-% on the left-hand side to 50 mass-% on the right-hand side.

The simulation time is 10^5 s, or about 28 h. The boundaries are by default closed. The plot generated by the macro file show the initial and final Ni profile.

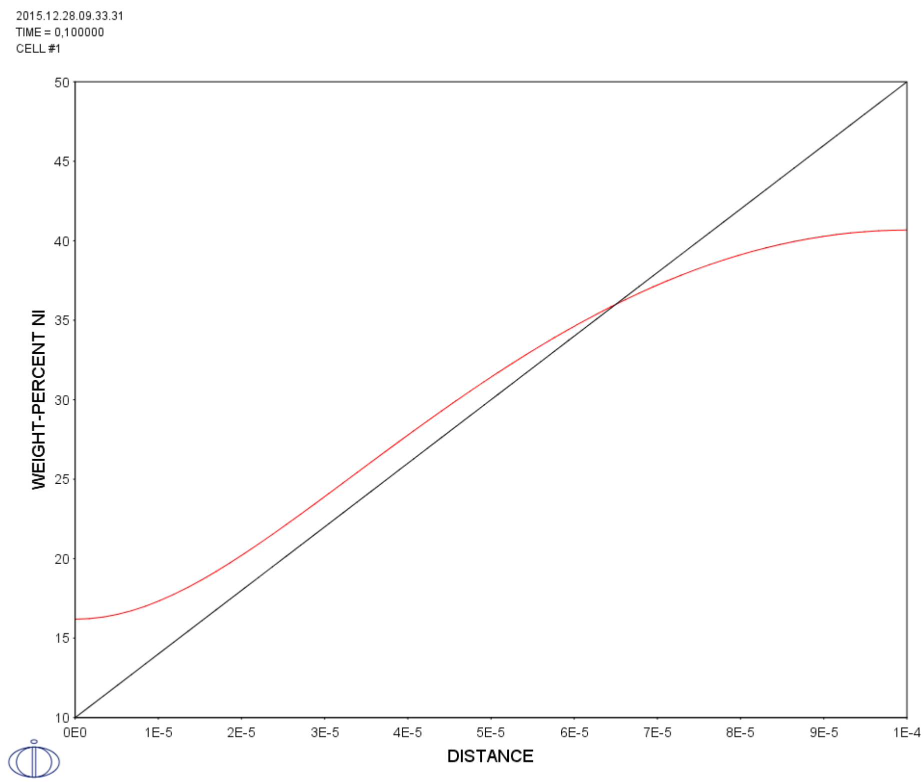


Schematic view of the initial state of example simplest_dictra. The width of the domain is 100 µm and there is a linear gradient in composition going from 10 to 50 mass-% Ni. There is a single region named austenite that consists of an fcc phase.

Running the Single Phase Simulation

For the single phase simulation, you use an example macro file called `simplest_dictra.dcm`. Open this file in Thermo-Calc in Console Mode as described in *Opening the Module and Macro Files* on page 3.

The simulation should automatically be set up, executed and finished within a matter of seconds. Results, as shown below, are plotted in the **Console Results** plot window.

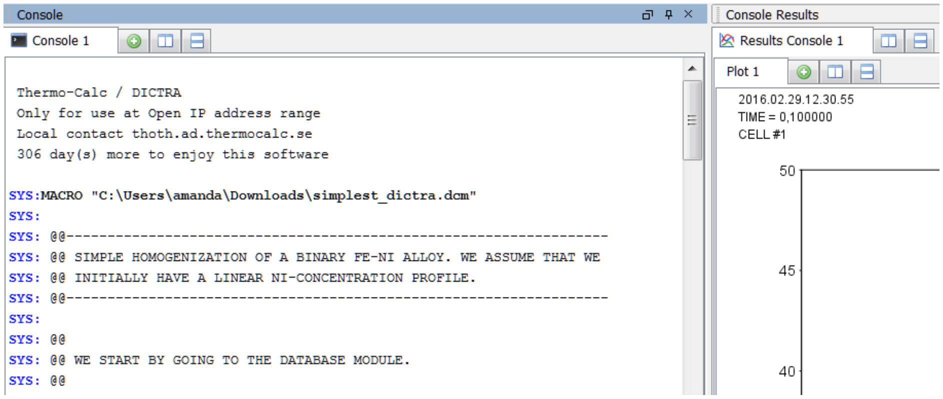


Output from the example simplest_dictra showing the initial and final Ni profile.

Single-Phase Command Details

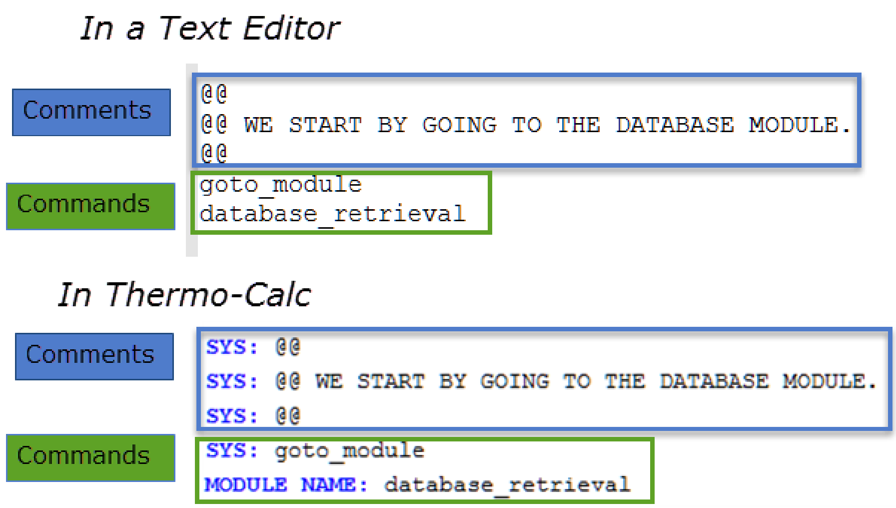
To examine the commands in the macro file, you can either open the file `simplest_dictra.dcm` in a text editor or scroll to the top of the **Console** window.

```

@@-----------------------------------------------------------------
@@ SIMPLE HOMOGENIZATION OF A BINARY FE-NI ALLOY. WE ASSUME THAT WE
@@ INITIALLY HAVE A LINEAR NI-CONCENTRATION PROFILE.
@@-----------------------------------------------------------------

@@
@@ WE START BY GOING TO THE DATABASE MODULE.
@@
goto_module
database_retrieval

@@
@@ USE THE FEDEMO DATABASE FOR THERMODYNAMIC DATA
@@
switch_database
fedemo

@@
@@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
@@
define_system
fe ni
```



In the following sections, the commands are discussed in the order they are executed. The first time a command appears it is listed in its general form in UPPERCASE and then using `code font` to detail the specific form of the example. For subsequent examples, only the new commands are detailed. All commands may be abbreviated as long as they are unambiguous.

You can also review the comments included in the example files. The first command you will use is shown in the macro file as follows:



Single Phase - Specifying the Thermodynamic System

Thermo-Calc Console Mode has several modules used for different purposes. In order to define the thermodynamic system the DATABASE_RETRIEVAL module is used.

GOTO_MODULE <MODULE>

```
goto_module database_retrieval
```

There are many databases for different purposes and material types. In this example the *fedemo* database is used. This is the thermodynamic iron demonstration database.

SWITCH_DATABASE <DATABASE>

```
switch_database fedemo
```

Determine what elements to use in the simulation, in this case `Fe` and `Ni`.

DEFINE_SYSTEM <LIST OF ELEMENTS>

```
define_system fe,ni
```

Only the FCC_A1 phase takes part in this simulation and therefore all phases are initially rejected (the wildcard * means *all phases* in this context) and then, with the next command, the FCC_A1 phase is restored.

REJECT <ELEMENTS, SPECIES, PHASES, CONSTITUENT OR SYSTEM> <LIST OF THE RELEVANT TYPE>

```
reject phases *
```

RESTORE <ELEMENTS, SPECIES, PHASES OR CONSTITUENTS> <LIST OF THE RELEVANT TYPE>

```
restore phases fcc_a1
```

The thermodynamic system now consists of the elements `Fe` and `Ni` and the `fcc_a1` phase.

To actually read the data from file into computer memory the GET_DATA command must be executed.

GET_DATA

```
get_data
```

At this stage only the thermodynamic data is defined and read. The kinetic data (mobilities) are, in this case, stored in another database. This is normally the case though it is possible to have both thermodynamic and kinetic data in a single database.

To add data from a different database the APPEND_DATABASE command is used. The kinetic data in this example is stored in the *mfedemo* database. This is the kinetic iron demonstration database.

APPEND_DATABASE <DATABASE>

```
append_database mfedemo
```

The same sequence of commands used for thermodynamic data is used to read kinetic data from this database.

```
define_system fe,ni
reject phases *
restore phases fcc_a1
get_data
```

Setting up the Single-Phase Simulation

The simulation set-up is performed in the DICTRA_MONITOR module.

```
goto_module
```

```
dictra_monitor
```

In this example only one *global* condition is set, temperature. The other global condition that can be set is pressure, but that is rarely done as the default value (1 atm) is set automatically and usually accepted.

SET_CONDITION <GLOBAL OR BOUNDARY>

```
set_condition global
```

The SET_CONDITION command is immediately followed by these sub-prompts:

<T OR P>

```
T
```

The T is for temperature. Then follows a standard procedure for entering temporally piece-wise functions:

<LOW TIME LIMIT> <FUNCTION> <HIGH TIME LIMIT> <Y/N> <IF Y, THEN NEXT FUNCTION> <...>

```
0 1400; * N
```

These prompts start with the *low time limit*, here zero (`0`), then the *function* (here the constant value `1400`) closed by a semicolon (`;`), then the *high time limit* (here a wildcard `*` is entered), then a `Y` or a `N` depending on whether more functions will be entered (in this case `N` because only a single function is entered).

The wildcard * used for the upper time limit means that the entered function is valid for the whole simulation. What this means is that a constant temperature equal to 1400 K is used during the whole simulation.

In any simulation at least one *region* must be defined. This is a named container that designates a certain part of, in this case the whole, domain. The name of a region is arbitrary and specified by you.

ENTER_REGION <NAME OF REGION>

```
enter_region austenite
```

The use of regions becomes clearer in the context of so-called moving phase boundary simulations (see *Moving Phase Boundary Simulation* on page 20).

A region must contain a numerical *grid* specified by width and type.

ENTER_GRID_COORDINATES

<NAME OF REGION FOR WHICH GRID IS TO BE ENTERED>

<WIDTH OF REGION>

<TYPE OF GRID>

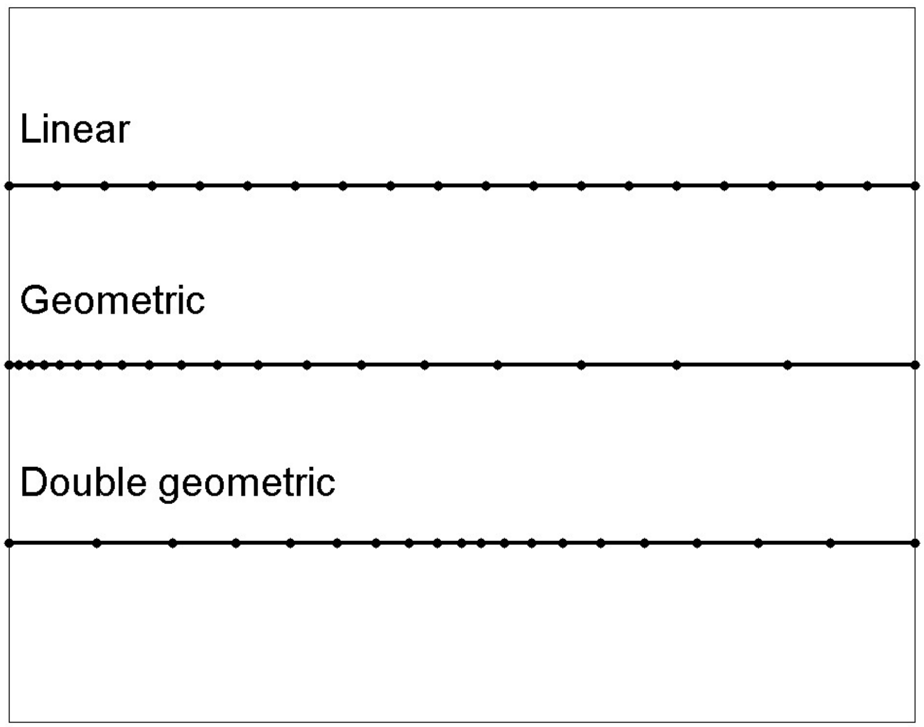
<NUMBER OF GRID POINTS>

```
enter_grid_coordinates
austenite
1e-4
linear
60
```

In the entries above, a 10^{-4} m wide (`1e-4`), `linear`, i.e. equidistant, grid is used with `60` grid points. There are many types of grids, see below for examples, and for other types than the linear there are additional sub-prompts.

Grid Types

Three different types of numerical grids are shown: linear, geometric and double geometric. A region must contain a grid.



A region must contain one *active* phase of type *matrix*. The meaning of active/inactive and phase type is explained in the next examples.

ENTER_PHASE_IN_REGION

<ACTIVE OR INACTIVE>

<NAME OF REGION>

<PHASE TYPE IN REGION>

<PHASE NAME>

```
enter_phase_in_region
active
austenite
matrix
fcc_a1
```

In this example only one phase participates, the `fcc_a1` phase, and it must therefore be `active` and of type `matrix`.

The initial composition profile must be specified for all phases that take part in a simulation. Since there are only two components, Fe and Ni, in the present simulation the initial *composition profile* must only be given for one of them.

ENTER_COMPOSITIONS

<REGION NAME>

<PHASE NAME>

<DEPENDENT COMPONENT>

<COMPOSITION TYPE>

<COMPONENT NAME>

<TYPE OF COMPOSITION PROFILE>

<COMPOSITION PROFILE DEPENDENT INPUT>

```
enter_compositions
austenite
fcc_a1
fe
weight_percent
ni
linear
10
50
```

In the above entries, `Fe` is chosen as the dependent component and the initial profile for Ni is a `linear` variation going from `10` mass-% on the lower/left hand side of the system to `50` mass-% on the upper/right hand side of the system. The composition type can be chosen among a number of different types, e.g. `weight percent` or mole fraction. There are also many different ways of specifying the composition profile, e.g. an arbitrary position dependent function or reading it for each grid point from file. Here the simplest possible type is used a `linear` function.

Now you set the simulation time. If not specified the initial time is set to zero.

SET_SIMULATION_TIME

<SIMULATION END TIME>

<AUTOMATIC TIMESTEP CONTROL Y/N>

<MAXIMUM TIMESTEP>

<INITIAL SIZE OF TIMESTEP>

<SMALLEST ALLOWED TIMESTEP>

```
set_simulation_time

1E5

y

1E4

1E-7

1E-7
```

Here the simulation time is set to 10^5 [s] (`1E5`), the maximum timestep size to 10^4 (`1E4`) and the initial and smallest acceptable to 10^{-7} (both `1E-7`). The *automatic timestep control* should normally always be turned on, which it is by default (you answer `Y` above).

Now you save the workspace:

SAVE_WORKSPACES <FILENAME> <OVERWITE EXISTING FILE Y/N>

```
save_workspaces simplest_dictra y
```

The command name SAVE_WORKSPACES can be misleading. The meaning of this command is roughly set result file. The set-up is saved immediately, but all results during the simulation are also saved to the selected file.

Running and Plotting the Simulation

The following command starts the simulation.

SIMULATE_REACTION

```
simulate_reaction
```

In order to plot results it is necessary to enter the post processor sub-module (note that the GOTO_ MODULE command is not used).

POST_PROCESSOR

```
post_processor
```

In most cases, either a specific position in the domain or one or more specific times must be specified with the SET_PLOT_CONDITION command, depending on whether time or spatial position is chosen as independent (x-axis) variable.

SET_PLOT_CONDITION <CONDITION WITH ARGUMENTS>

```
set_plot_condition time 0,1e5
```

Here, the composition profile of Ni at the initial and final time will be plotted and thus `time` is chosen as plot condition.

In other cases distance can be given relative to, for example, a phase interface, hence the global specification in the following.

SET_DIAGRAM_AXIS <X OR Y> <AXIS QUANTITY>

```
set_diagram_axis x distance global
```

In this example the only meaningful measure of distance is relative to the domain as a whole, i.e. the leftmost position is zero and the rightmost position corresponds to the width of the region (10^{-4} m).

The dependent property to be plotted is mass-% Ni.

```
set_diagram_axis y weight_percent ni
```

Then plot the diagram:

PLOT_DIAGRAM

```
plot_diagram
```

In the **Console Results** plot window, graphs of the initial and final Ni profile display.

Finally, the SET_INTERACTIVE_MODE command is executed to return control to the user. If the command is absent in a macro the program terminates when the end of the macro file is reached.

SET_INTERACTIVE_MODE

```
set_interactive_mode
```

Theory for the Single Phase Simulation

The flux of a component k in the z-direction in an isobarothermal system is in general given by

$$J_k = \sum_{i=1}^{n} L_{ki} \frac{\partial \mu_i}{\partial z}$$

where the are kinetic coefficients and is the chemical potential of component i. The correlation effects, i.e. the coupling of the flux of component k to the chemical potential gradients of the other components, are normally neglected

$$(L_{ki} = 0, k \neq i)$$

and thus

$$J_k = L_{kk}\frac{\partial \mu_k}{\partial z} = M_k c_k \frac{\partial \mu_k}{\partial z}$$

where is the concentration and the mobility of component k.

The equation for the flux is combined with the equation of continuity, which takes the following form in a planar domain,

$$\frac{\partial c_k}{\partial t} = \frac{\partial}{\partial z}\left(-J_k\right)$$

which relates the local evolution of the concentration of k to the divergence of the flux.

The expression for the flux can be expanded in terms of concentration gradients

$$J_k = -\sum_{i=1}^{n} M_k c_k \frac{\partial \mu_k}{\partial c_j}\frac{\partial c_j}{\partial z} = -\sum_{i=1}^{n} D_{kj}\frac{\partial c_j}{\partial z}$$

where the diffusion coefficent of component k with respect to the concentration gradient of component j has been introduced.

$$D_{kj} = M_k c_k \frac{\partial \mu_k}{\partial c_j}$$

The flux expressions above are given in the so-called lattice-fixed frame of reference. In practical calculations it is more common to use a volume-fixed frame of reference. For a discussion of these concepts, see Andersson and Ågren (1992)[1].

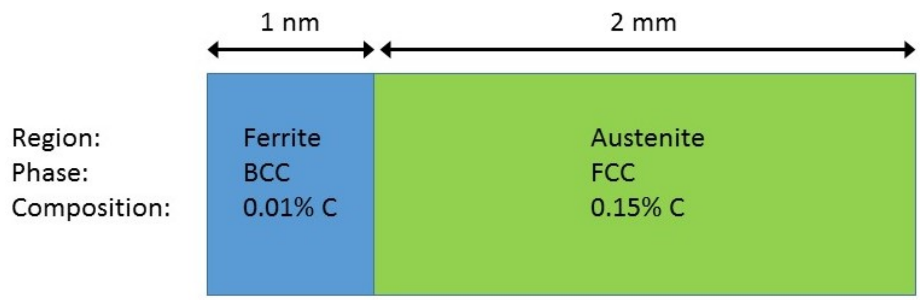
[1] J.-O. Andersson and J. Ågren, “Models for numerical treatment of multicomponent diffusion in simple phases,” J. Appl. Phys., vol. 72, no. 4, pp. 1350–1355, Aug. 1992.

Moving Phase Boundary Simulations

In this section:

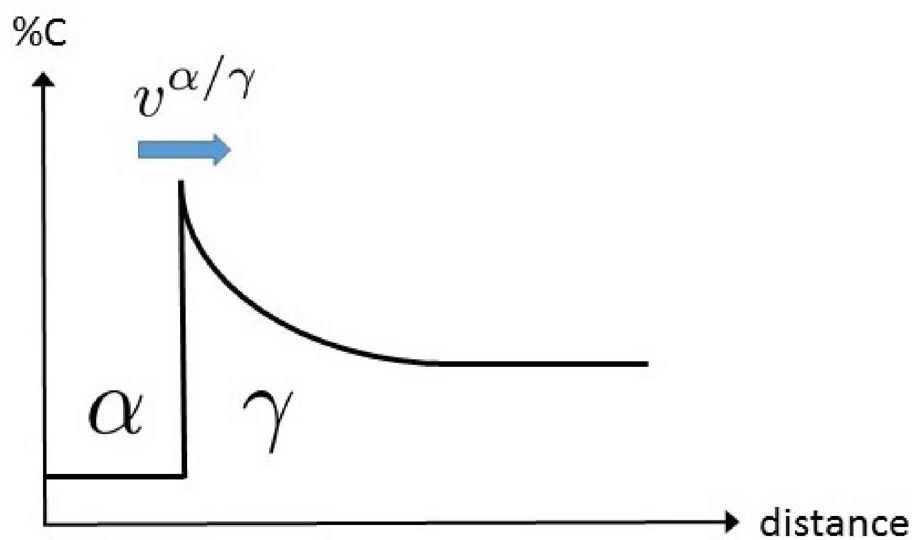
Moving Phase Boundary Simulation

The growth of ferrite (bcc) into austenite (fcc) is simulated in this example. The austenite is assumed to be initially homogeneous with the composition Fe - 0.15 mass-% C. The transformation temperature is 1050 K. The initial thickness of the austenite is 2 mm and an initially very thin ferrite (1 nm) is also present at the start of the simulation. The initial state is as below.



The set-up of the Simple Moving Boundary example macro. There are two regions, Ferrite and Austenite, consisting of bcc and fcc, respectively.

A schematic graph of the carbon profile during the transformation is shown in the image below. Crusius et al (1992)[1] is recommended as further reading.

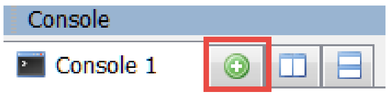


Schematic view of the carbon profile during the transformation from austenite (γ) to ferrite (α) in the Simple Moving Boundary macro example.

[1] S. Crusius, L. Höglund, U. Knoop, G. Inden, and J. Ågren, “On the growth of ferrite allotriomorphs in fe-c alloys,” Zeitschrift für Met., vol. 83, no. 10, pp. 729–738, 1992.

Run the Macro File and Learn the Commands

If the single phase example has just been run, then either restart the software program or open another Console window by clicking the green button with the plus sign.



At the `SYS` prompt, type `macro` (short for the MACRO_FILE_OPEN command) and press <Enter>. Locate and select the macro file called `simple_moving_boundary.dcm` as described in *Opening the Module and Macro Files* on page 3. The commands in the macro file set up, run and plot the simulation in the Console Results window.

In the macro file some abbreviations of the commands are used. You can shorten command names in an arbitrary manner as long as the abbreviations are unambiguous.

In the macro file most commands are first given as comments with the full name and then abbreviated. Commands that were not used in the single-phase simulation, or are used differently, are explained in more detail.

Comments in the macro file can be entered after two "at" symbols (@@). In most cases the arguments of a command can be entered on the same line as the command itself. See *Single-Phase Command Details* on page 10 for more information.

Moving Phase - Specifying the Thermodynamic System

This set of commands is the same as in the single-phase simulation example. Thermodynamic and kinetic data is read for Fe and C and the bcc and fcc phases.

```
@@ goto_module database_retrieval
go da
@@ switch_database fedemo
sw fedemo
@@ define_system fe,c
def-sys fe c
@@ reject phase *
rej ph *
@@ restore phase fcc_a1,bcc_a2
rest ph fcc,bcc
@@ get_data
get
@@ append_database mfedemo
```

```
app mfedemo
@@ define_system fe,c
def-sys fe c
@@ reject phase *
rej ph *
@@ restore phase fcc_a1,bcc_a2
rest ph fcc,bcc
@@ get_data
Get
```

Setting Up the Moving Phase Boundary Simulation

The first region to be entered is entered in the same way as in the single-phase simulation.

```
@@ goto_module dictra_monitor
go d-m
@@ set_condition global T 0 1050; * N
set-cond glob T 0 1050; * N
```

A constant temperature equal to `1050`K is used in this simulation.

```
@@ enter_region ferrite
ent-reg ferrite
```

The phase interface between ferrite and austenite is created by introducing a region called austenite that is attached to the ferrite region. Regions are always separated by phase interfaces and must therefore always contain different *matrix* type phases.

ENTER_REGION

<NAME OF REGION>

<NAME OF EXISTING REGION TO ATTACH TO>

<ATTACH TO THE RIGHT OF EXISTING REGION Y/N>

```
enter_region
austenite
ferrite
y
```

The domain now consists of two regions: `ferrite` and `austenite`.

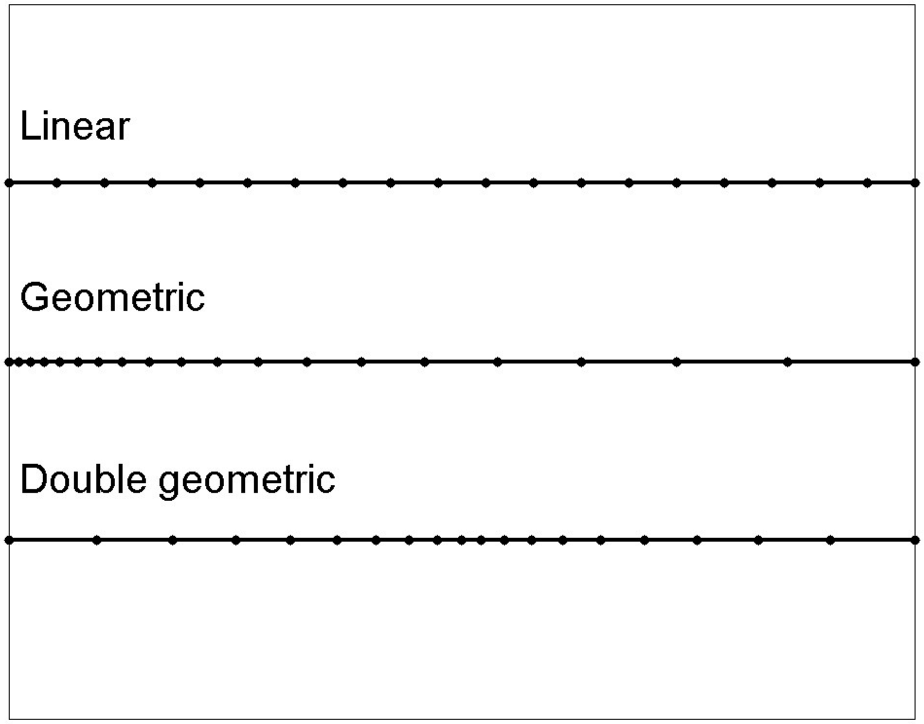
As in the single-phase simulation example, the regions must contain both grids and one *active* phase of type *matrix*. These are entered separately for both regions.

```
enter-grid
ferrite
1e-9
linear
10
enter-grid
austenite
2e-3
geometric
50
1.05
```

In the above entries, the grid type for the `ferrite` region is the same as in the single-phase simulation example, i.e. equidistant. For the `austenite` region the grid type is called `geometric` and for these grids the grid point spacing changes by a constant factor between every grid point. In this case the spacing increases by 5%, i.e. a factor `1.05`, from the lower/left side for each grid point.

Grid Types

Three different types of numerical grids are shown: linear, geometric and double geometric. A region must contain a grid.



Now the initial composition of both phases is entered. In this case Fe is automatically chosen to be the dependent component.

```
enter-phase
active
ferrite
matrix
bcc
enter-phase
active
austenite
matrix
fcc
```

Both phases are assumed to be initially homogeneous, the bcc phase having `0.01` mass-% C and the fcc phase `0.15` mass-% C, as shown in the following:

```
enter-composition
ferrite
bcc
w-p
C
```

```
linear
0.01
0.01
enter-composition
austenite
fcc
w-p
C
linear
0.15
0.15
```

Set the simulation time and other parameters related to the time-step size.

```
set-simulation-time
1e5
y
1e4
1E-7
1E-7
```

Finally, save the simulation set-up to file.

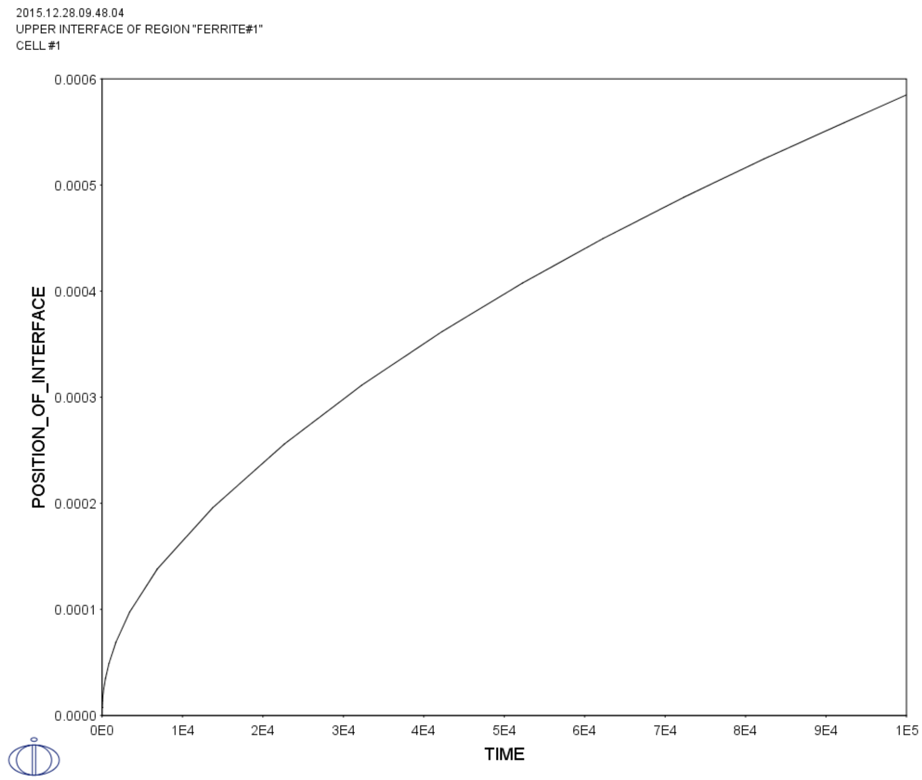
```
save simple_moving_boundary yes
```

Running and Plotting the Moving Phase Simulation

These commands are the same as in the single-phase example.

```
@@simulate_reaction
sim
@@ post_processor
post
```

The ferrite/austenite phase interface position as a function of time is plotted.



Output from the example simple_moving_boundary showing the phase interface position as a function of time.

As mentioned in the single phase example, in general either a specific spatial position or a specific time must be set as a plot condition. Here the phase `interface` is set as plot condition and it can be referenced as the lower interface of the austenite region or as the `upper` interface of the `ferrite` region.

SET_PLOT_CONDITION

<TYPE OF PLOT CONDITION>

<FOR INTERFACE CONDITION, SPECIFY REGION>

<LOWER OR UPPER INTERFACE OF REGION>

```
set-plot-condition

interface

ferrite

upper
```

Set time as x-axis variable

```
@@set-diagram-axis x time

s-d-a x time
```

Now set the interface position as y-axis variable

```
s-d-a y
```

POSITION_OF_INTERFACE

<SPECIFY REGION>

<LOWER OR UPPER INTERFACE OF REGION>

```
pos-of-int ferrite upper
@@ plot_diagram
plot
```

The plot has the parabolic appearance characteristic for many types of diffusion problems.

Theory for the Moving Phase Boundary Simulation

It is assumed that local equilibrium holds at the phase interface, i.e. the chemical potentials of all components are continuous across the boundary, but the chemical potential gradients are in general discontinuous.

In order to maintain mass balance a set of flux balance equations must be satisfied at the phase interface:

$$v^{\alpha/\gamma}\left(c_k^{\alpha} - c_k^{\gamma}\right) = J_k^{\alpha} - J_k^{\gamma} \qquad k = 1, \ldots, n-1$$

where is the interfacial velocity. The concentrations and the fluxes are those at the interface on the α and γ sides, respectively.

In the present case there are no degrees of freedom at the interface and the concentrations can be taken directly from a phase diagram. There is only one flux balance equation and the interface velocity can be evaluated once the fluxes at the interface are known. However, for ternary and higher systems the state at the interface must be found by some iterative procedure.

For more information on moving phase boundary problems, see, for example, Ågren (1982)[1],Crusius et al. (1992)[2] and Höglund (1997)[3].

---

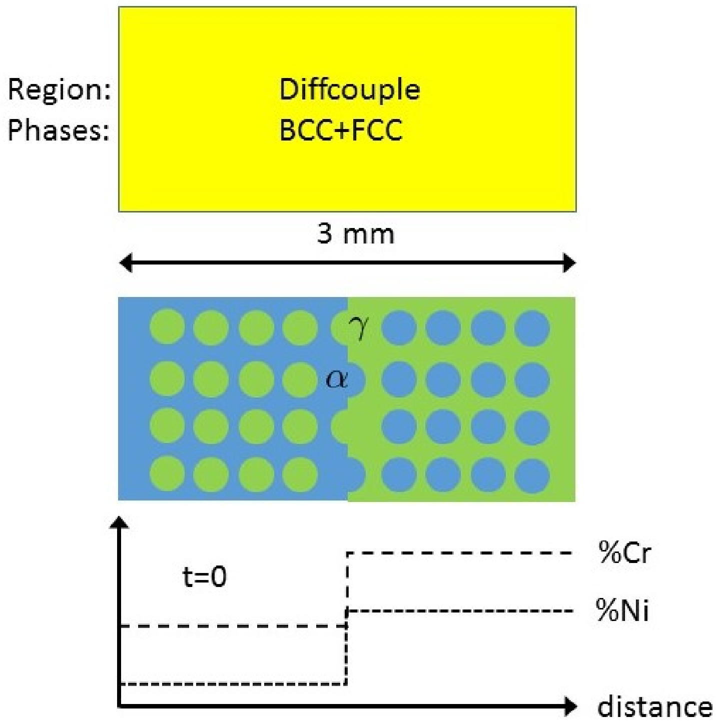
[1]J. Ågren, "Numerical Treatment of Diffusional Reactions in Multicomponent Alloys," J. Phys. Chem. Solids, vol. 43, no. 4, pp. 385–391, 1982.
[2]S. Crusius, G. Inden, U. Knoop, L. Höglund, and J. Ågren, "On the numerical treatment of moving boundary problems," Zeitschrift für Met., vol. 83, no. 9, pp. 673–678, 1992.
[3]L. Höglund, "Computer simulation of diffusion controlled transformations in multicomponent alloys," PhD Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 1997.

Multiphase Simulations

In this section:

A Multiphase Simulation

This example simulates the evolution of an Fe-Cr-Ni diffusion couple during a 100 h heat treatment at . Both end members of the diffusion couple are duplex ferrite + austenite, but the majority phase is ferrite on the left hand side and austenite on the right. With this type of simulation it is assumed that the material is fully equilibrated at each grid point, i.e. the local phase fractions, phase compositions and so forth are obtained from an equilibrium calculation with the local overall composition as condition.



Schematic view of the initial state of multiphase_example. Two different ternary Fe-Cr-Ni alloys form a diffusion couple. There is a single region named diffcouple in which both the bcc and fcc phases are entered. Both alloys are duplex bcc+fcc, but the majority phase is bcc (α) in the left hand alloy and fcc (γ) in the right.

Run the Macro File and Learn the Commands

At the `SYS` prompt, type `macro` and pres <Enter>. Locate and select the macro file called `multiphase_example.dcm` as described in *Opening the Module and Macro Files* on page 3. The commands in the macro file should set up the simulation, run it and then produce a graph in the Console Results window.

Most of the set-up is the same as the single phase and moving phase boundary examples. Detailed comments are only given for the new commands for this simulation. See *Single-Phase Command Details* on page 10 for more information.

Specifying the Multiphase Thermodynamic System

Data for Fe, Cr and Ni and the fcc and bcc phases are read from the *fedemo* (thermodynamics) and *mfedemo* (kinetics) databases.

```
go da
sw fedemo
def-sys fe cr ni
rej ph *
rest ph bcc,fcc
get
app mfedemo
def-sys fe cr ni
rej ph *
rest ph bcc,fcc
get
```

Setting Up the Multiphase Simulation

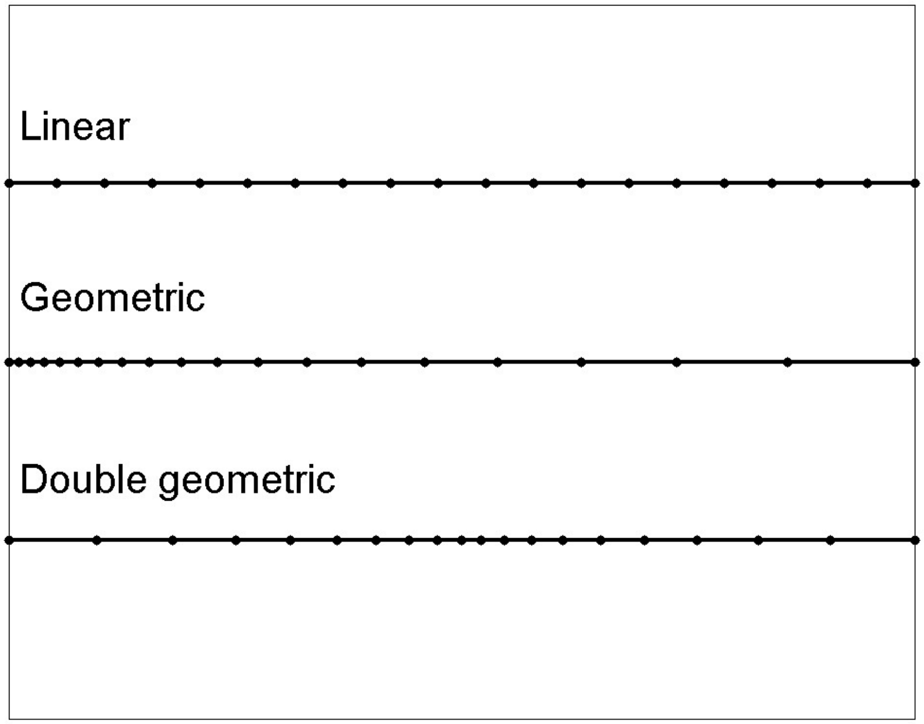
Set the temperature `T` during the simulation to 1100°C (`1373.15` K) and enter a region `diffcouple`.

```
go dictra-monitor
set-cond glob T 0 1373.15; * N
ent-reg diffcouple
ent-grid
diffcouple
3e-3
double
60
0.85
1.15
```

In the above example a so-called `double` geometric grid is used. This is similar to the geometric grid used in the moving phase boundary example, but instead two geometrical factors are entered, one for the lower and one for the upper half of the region. In the lower half of the region the distance between subsequent grid points decreases by 15 % (`0.85`) and in the upper half it increases by 15 % (`1.15`). The resulting grid has a much larger grid point density toward the middle of the domain.

Grid Types

Three different types of numerical grids are shown: linear, geometric and double geometric. A region must contain a grid.



As always, a region must contain exactly one `active` phase that is of type `matrix`. In this example it is arbitrary whether it is the bcc or the fcc phase that is entered as matrix.

```
enter-phase
active
diffcouple
matrix
fcc
```

The bcc phase is entered in the same manner as the fcc phase, except that the type is set as `spheroid`.

```
enter-phase
active
diffcouple
spheroid
bcc
```

The initial composition profile of the fcc phase is entered as a `function` of distance.

```
enter-composition
diffcouple
fcc
```

```
fe
m-f
cr
function
2.57e-1+(4.23e-1-2.57e-1)*hs(x-1.5e-3);
ni
function
6.47e-2+(2.75e-1-6.47e-2)*hs(x-1.5e-3);
```

In the above entries, the `hs` denotes the Heaviside step function that has the following properties:

$$hs\left(x\right)=\begin{cases}0 & x<0\\ 1 & x\geq 0\end{cases}$$

The resulting Cr profile is thus `0.257` on the left half of the domain and `0.423` on the right with a sharp step at the center.

As shown in *A Multiphase Simulation* on page 29

The initial composition of the fcc phase is actually the initial overall composition. This is due to the choice of initial composition of the spheroid bcc phase.

ENTER_COMPOSITIONS

<REGION NAME>

<PHASE NAME>

<USE EQUILIBRIUM VALUE Y/N>

```
enter-comp
diffcouple
bcc
y
```

When entering the composition for a type spheroid phase it is possible to either, as here, let the composition of the matrix phase be the overall composition or manually enter a start composition and volume fraction.

The local kinetics of the multiphase mixture must be estimated by means of some function that may depend on the local phase fractions, phase compositions and the mobilities of the individual phases. The command for choosing the function to use is ENTER_HOMOGENIZATION_FUNCTION. There are many

[8][2]) is a good choice.

ENTER_HOMOGENIZATION_FUNCTION

<ENTER FUNCTION NUMBER>

```
enter-homo-fun
1
```

Finally, set simulation time (100 h) and select the name of the result file:

```
set-sim-time
3.6e5
yes
3.6e4
1e-7
1e-7
save multiphase_example y
```

Running and Plotting the Multiphase Simulation Results

Run the simulation.

```
simulate
```

Go to the post processor.

```
post
```

Plot the mole fraction fcc phase as a function of distance at the end of the heat treatment.

```
set-plot-condition time last
set-diagram-axis x distance global
set-diagram-axis y npm(fcc)
```

Instead of explicitly entering the final time it is possible to use the keyword `last`. The `np` of npm stands for *number of moles* of the phase argument and the `m` is the normalizing quantity (moles). The limits of an axis can be set with the SET_SCALING_STATUS command. You manually set the scaling of the axis and then restore automatic scaling:

SET_SCALING_STATUS

<X OR Y AXIS>

[1] Z. Hashin and S. Shtrikman, "A Variational Approach to the Theory of the Effective Magnetic Permeability of Multiphase Materials," J. Appl. Phys., vol. 33, no. 10, pp. 3125–3131, 1962.

[2] Z. Hashin and S. Shtrikman, "A Variational Approach to the Theory of the Effective Magnetic Permeability of Multiphase Materials," J. Appl. Phys., vol. 33, no. 10, pp. 3125–3131, 1962.

<USE AUTOMATIC SCALING Y/N>

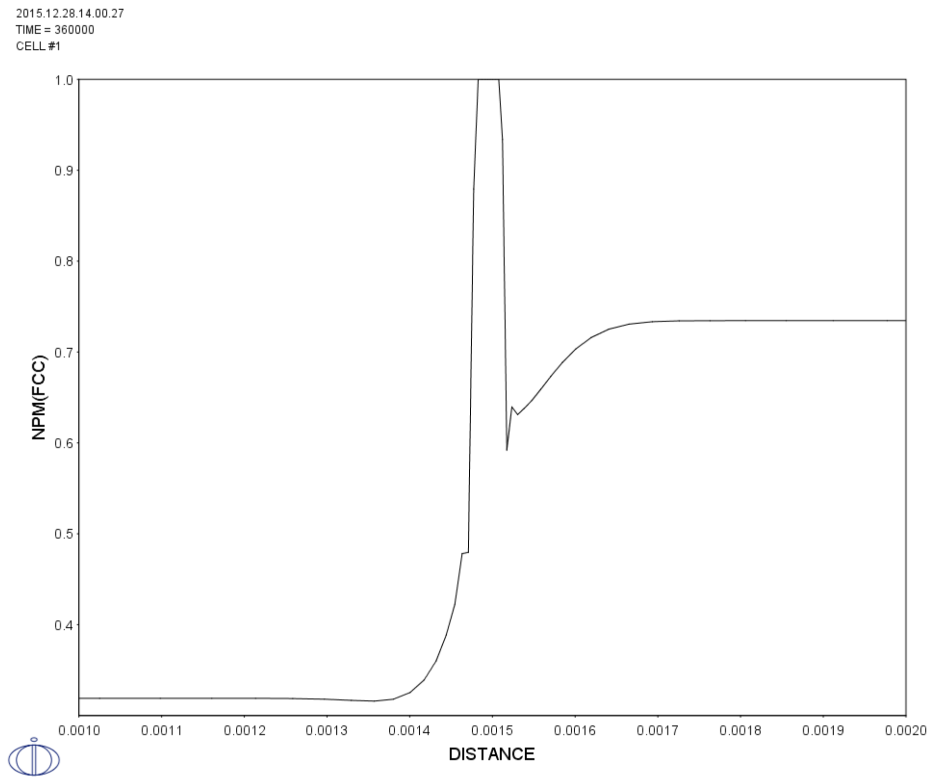
<IF MANUAL SCALING, ENTER LIMITS>

```
set-scaling-status x n 1e-3 2e-3
```

Plot the diagram.

```
plot
```

The plot below shows that a single phase fcc zone has formed and this was also observed experimentally, see Reference [9][1].



Output from the multiphase_example showing the mole fraction fcc phase as a function of distance at the end of the heat treatment.

Theory for the Multiphase Simulation

More details on the so-called homogenization model for multiphase simulations can be found in Larsson and Engström (2006)[2] and Larsson and Höglund (2009)[3]. As mentioned above, it is assumed that the

[1] A. Engström, "Interdiffusion in multiphase, Fe-Cr-Ni diffusion couples," Scand. J. Metall., vol. 24, no. 1, pp. 12–20, 1995.

[2] H. Larsson and A. Engström, "A homogenization approach to diffusion simulations applied to α + γ Fe–Cr–Ni diffusion couples," Acta Mater., vol. 54, no. 9, pp. 2431–2439, May 2006.

[3] H. Larsson and L. Höglund, "Multiphase diffusion simulations in 1D using the DICTRA homogenization model," Calphad Comput. Coupling Phase Diagrams Thermochem., vol. 33, no. 3, pp. 495–501, 2009.

material is locally fully equilibrated and that the local phase fractions, phase compositions and so forth is obtained from an equilibrium calculation with the local overall composition as condition. From a numerical point of view the homogenization model treats the multiphase material as a single phase having the “average”, or “effective”, properties of the local phase mixture.

When estimating the effective kinetics of a multiphase mixture the product of solubility and mobility in each phase is considered. It is therefore convenient to define

$$\Gamma_k^{\phi} = M_k^{\phi} c_k^{\phi}$$

for each phase ϕ. The effective kinetics of the multiphase mixture is denoted and in this example it was evaluated using the lower Hashin-Shtrikman bound. The default method is a simple rule of mixtures

$$\Gamma_k^{\star} = \sum_{\phi} f^{\phi} \Gamma_k^{\phi}$$

where is the volume fraction of ϕ. The flux is obtained as

$$J_k = -\Gamma_k^{\star} \frac{\partial \mu_k}{\partial z}$$

which replaces the single-phase equation

$$J_k = L_{kk} \frac{\partial \mu_k}{\partial z} = M_k c_k \frac{\partial \mu_k}{\partial z}$$

for multiphase simulations.

Next Steps

The Diffusion Module (DICTRA) Examples collection contain a variety of different simulation types and is a good starting point when thinking of creating a new simulation.

When setting up a new simulation there are two recommended ways to proceed:

- Copy and gradually adjust an existing macro file (that can be taken from the Examples collection).
- Type in the set-up directly while recording a log file. The command for recording a log file is SET_LOG_FILE and this should be done immediately when starting the program in the SYS module. The log file can then be used as a macro file by changing the file extension from LOG to DCM.

Diffusion Module (DICTRA) User Guide

Version 2016a

Introduction to theDiffusion Module (DICTRA)

In this section:

About the Diffusion Module (DICTRA)

The Diffusion Module (DICTRA) is an add-on module to the Thermo-Calc software. It is used for simulation of diffusion controlled transformations in multicomponent systems. The simulation calculations are both time- and space-dependent.

The Diffusion Module (DICTRA), which is often referred to as DICTRA, is suitable for solving diffusion problems that include a moving boundary (so-called *Stefan problems*). The multicomponent diffusion equations in the various regions of a material are solved under the assumption that thermodynamic equilibrium holds locally at all phase interfaces. Simulations are one-dimensional and three different geometries can be performed: planar, cylindrical and spherical.

You can set the boundary conditions in many ways, enabling you to solve different types of problems.

Types of Diffusion Problems

Problem type	Examples
One-phase problems	• Homogenization of alloys • Carburising and decarburising of e.g. steel in austenitic state
Moving boundary problems	• Growth or dissolution of precipitates • Coarsening of precipitates • Microsegregation during solidification • Austenite to ferrite transformations • Growth of intermediate phases in compounds
Long-range diffusion in multi-phase systems	• Carburizing of high-temperature alloys • Interdiffusion in compounds, e.g. coating systems • Nitriding and nitrocarburization
Cooperative growth	• Growth of pearlite in alloyed steels
Deviation from local equilibrium	• Calculations under paraequilibrium conditions

Typographical Conventions

The following typographical conventions are used throughout the documentation:

Convention	Definition
Forward arrow symbol →	The forward arrow symbol → instructs you to select a series of menu items in a specific order. For example, **Tools → Options** is equivalent to: From the **Tools** menu, select **Options**.
Boldface font	A **boldface** font indicates that the given word(s) are shown that way on a toolbar button or as a menu selection. For example, if you are told to select a menu item in a particular order, such as **Tools→Options**, or to click **Save**.
Italic font	An *italic* font indicates the introduction of important terminology. Expect to find an explanation in the same paragraph or elsewhere in the guide.
COMMAND	For features in Thermo-Calc that use the command line, this font and all capital letters indicates that this is a COMMAND used in the Console Mode terminal. Examples of how you can use a command are written with code font: Use DEFINE_ELEMENTS followed by a list of the elements that you want in your system. (To list the elements that are available in your current database, use LIST_DATABASE and choose `Elements`).
HELP	Text in blue and underline and a page number is a link to another topic in the current or referenced guide. Often command names are also topics. Clicking the link takes you to more detail about a particular command or subject in the PDF.
<Enter>	Text with <angle brackets> indicates a keyboard entry. Usually to press <Enter> (or Return).
`code` and **`code bold`**	A `code` font shows a programming code or code example. The **`code bold`** font highlights the entry.
Important	Provides important information and indicates that more detail is located in the linked or named topic.
Note	The information can be of use to you. It is recommended that you read the text or follow the link to more information.
Also see	Go to more information about the topic being discussed.
Examples	Go to the example collection to learn more.

Help Resources

Online and Context Help

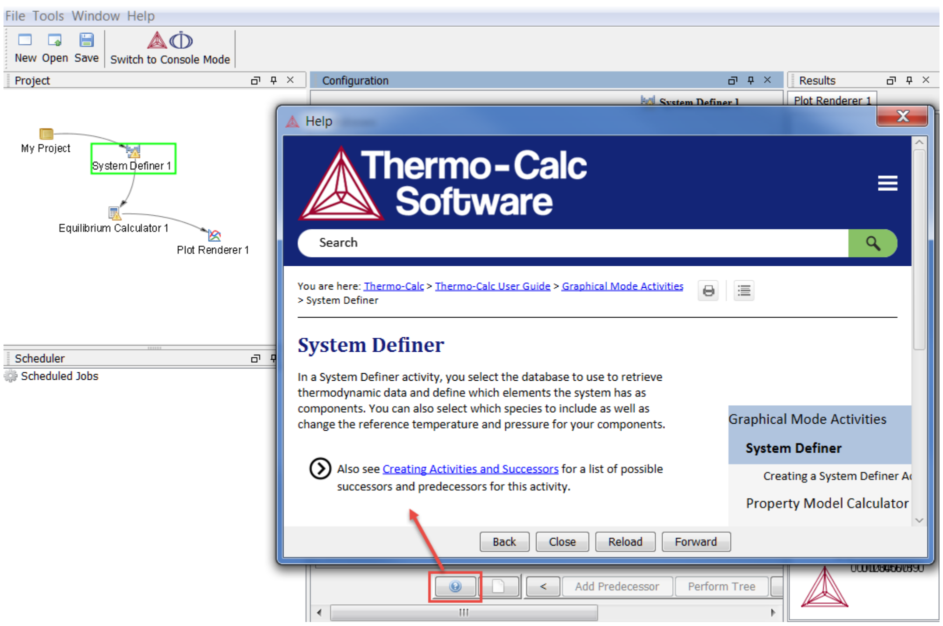
Online Help

To access online help in a browser, open Thermo-Calc and select **Help →Online Help**. The content opens in a browser but uses local content so you don't need an internet connection.

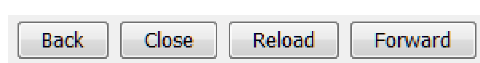
Context Help

When you are in Graphical Mode, you can access feature help (also called *topic-sensitive* or *context help*) for the activity nodes in the tree.

1. In the Project window, click a node. For example, **System Definer**.
2. In the lower left corner of the **Configuration** window, click the help button .
3. The **Help** window opens to the relevant topic.



The window that opens has the same content as the help you access in the browser. There are these extra navigation buttons in this view.

- Use the buttons on the bottom of the window,  , to navigate **Back** and **Forward** (these are only active once you have started using the help to load pages and create a history), to **Close** the window, and **Reload** the original content.

Console Mode Help

Console Mode is for Thermo-Calc and the Diffusion Module (DICTRA).

In Console Mode at the command line prompt, you can access help in these ways:

- For a list of all the available commands in the current module, at the prompt type a question mark (`?`) and press <Enter>.
- For a description of a specific command, type `Help` followed by the name of the command. You can only get online help about a command related to the current module you are in.
- For general system information type `Information`. Specify the subject or type `?` and the available subjects are listed. This subject list is specific to the current module.

Diffusion Module (DICTRA) Examples Collection

To learn more about how to use the add-on software you can open and run the example files. These macro files (with the extension *.`DCM`) include comments, which you can either run in Thermo-Calc or open and read in a text editor. If you read the macro file in a text editor, you do not see the output that Thermo-Calc gives in response to the commands stored in the macro file.

You can find these examples on the website under **Examples**. The files are also included in the installation. Refer to the *Thermo-Calc Installation Guide* for details about default folders where manuals and examples are installed for your operating system.

The pdf document *Diffusion Module (DICTRA) Examples* contains the contents of the macro files and the output with which theDICTRA module responds.

Example Descriptions

Example	Description
One-Phase problems	
a1	Homogenization of a binary Fe-Ni alloy (Initially a linear concentration is assumed).
a2a	Homogenization of a binary Fe-Ni alloy (Initially a step profile is assumed).
a2b	Simple homogenization of a binary Fe-Ni alloy. This example is identical to a2a however, it uses implicit time integration instead of the trapetzoidal method for solving the PDEs.
a3	Uphill diffusion in an Fe-Si-C alloy. Setup file for the simulation of uphill diffusion in a ternary single phase austenite matrix due to the classical darken experiment published by I.S. Darken: Trans. Aime, v. 180 (1949), pp. 430-438. In this setup two pieces of austenite (3.80 wt%Si, 0.49 wt%C) and (0.05 wt%Si, 0.45 wt%C) are put together and are subsequently annealed at 1050 C for 13 days. As both pieces are austenite they must be

Example	Description
	entered into the same region. we can accomplish this by giving the compositions of Si and C in each gridpoint individually. For convenience we store these data on file.
a4a	Carburization of binary Fe-C allow (comparison with an analytical erf-solution). This is a simple binary simulation with just one single phase region. The numerical simulation is compared with an analytical erf-solution. A special database is created, erf.tdb, in which the diffusion coefficient is set to a concentration independent value.
a5	Simulation of carburization of a binary Fe - 0.15 wt% C alloy. A mixture of 40% n2 and 60% cracked methanol was used as carrier gas. The carburizing "carbon potential" in the gas is 0.85 wt%. A surface reaction controls the flux of c at the surface.
a6	C-diffusion through a tube wall. A simple example on diffusion through a tube wall. The tube-material is an Fe-0.6%Mn-0.7%Si-0.05%C alloy. On the inside wall a carbon activity of 0.9 is maintained whereas on the outside the C-activity is very low. This example demonstrates the use of the command SET-FIRST-INTERFACE as well as the use of MIXED boundary conditions.
a7	Simulation of a homogenization heat treatment. The initial segregation profile is created from a Scheil calculation (see macro create_initial_profile.TCM). The command INPUT_ SCHEIL_PROFILE in the DICTRA module performs most of the setup. Only time and temperature must be entered after the INPUT_SCHEIL_PROFILE command is executed.
Moving boundary problems	
b1a	Austenite to ferrite transformation in a binary Fe-C alloy. Setup file for calculating a ferrite (BCC)/austenite (FCC) transformation in a binary Fe-C alloy. The initial state is an austenite of 2 mm thickness. The composition of the austenite is Fe-0.15wt%C. After austenitisation the specimen is quenched down to 1050K. The system is assumed closed, so we do not set any boundary conditions (closed system is default). Ferrite is expected to grow into the austenite. For this reason we start with a thin region with ferrite adjacent to the austenite.
b1b	Austenite to ferrite transformation in a binary Fe-C alloy (Inactive ferrite). Same problem as in exb1a but now we set up the problem with ferrite as an inactive phase adjacent to the initial austenite.
b1c	Austenite to ferrite transformation in a binary Fe-C alloy (gradual cooling down). Same problem as in exb1a and exb1b but now we start the simulation at a higher temperature and assume a gradual cooling down to 1050 K. When 1050 K is reached, the temperature is kept constant and thus have an isothermal transformation. As in exb1b we have ferrite as an inactive phase adjacent to the initial austenite.
b2	Cementite dissolution in an Fe-Cr-C alloy. Setup file for calculating the dissolution of a spherical cementite particle in an austenite matrix. This case is from Z.-K. Liu, L. Höglund,

Example	Description
	B. Jönsson and J. Ågren: Metall. Trans. A, v. 22A (1991), pp. 1745-1752. In order to achieve the correct average composition in the calculation it is necessary to take into account the fact that the calculation is setup using the volume fraction of the phases. To calculate the initial state at the heat treatment temperature we need first to determine the state at the normalizing temperature. To calculate the volume fraction of the phases we need to enter a number of functions that calculate these quantities.
b3	Shows the dissolution of 23-carbide in an austenitic matrix. Use this to calculate the dissolution of an M23C6 particle in an austenite matrix. A film of ferrite is allowed to nucleate around the carbide during the precipitation.
b4a	Solidification path of a Fe-18%Cr-8%Ni alloy (eutectic reaction). This examples demonstrates the solidification path of an Fe-18%Cr-8%Ni alloy. A eutectic reaction is assumed, LIQUID -> BCC + FCC. Hence the BCC and FCC regions should be on separate sides of the liquid region. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.
b4b	Solidification path of an Fe-18%Cr-8%Ni alloy (peritectic reaction). Same as exb4a but, now a peritectic reaction is assumed, LIQUID + BCC -> FCC. Hence the FCC region should appear in between the LIQUID and the BCC. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.
b4c	Solidification path of an Fe-18%Cr-8%Ni alloy (peritectic reaction, homogeneous liquid). Same as exb4b but, now we will amend the diffusivity data for the LIQUID and use a very high value for the diffusivity in order to simulate a case where it is assumed the composition in the LIQUID is always homogeneous. This case should be considered less realistic.
b4d	Solidification path of an Fe-18%Cr-8%Ni alloy (peritectic reaction, heat-flux controls the temperature). Same as exb4b but, instead of controlling the temperature the amount heat extracted is given. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.
b5	Diffusion couple of Fe-Ni-Cr alloys. This example demonstrates the evaluation of a ternary Fe-Cr-Ni diffusion couple. A thin slice of ALPHA phase (38%Cr,0%Ni) is clamped between two thicker slices of GAMMA phase (27%Cr, 20%Ni). The assembly is subsequently heat treated at 1373K. This setup corresponds to diffusion couple in M. Kajihara, C.-B. Lim and M. Kikuchi: ISIJ International, v. 33 (1993), pp. 498-507. Also see M. Kajihara and M. Kikichi: Acta. Metall. Mater., v. 41 (1993), pp. 2045-2059.
b6	Microsegregation of phosphorus. This example illustrates the effect of microsegregation

Example	Description
	of phosphorus during peritectic solidification in steel.
b7	This example shows how to enter dispersed phases on either side of a phase interface. The particular case shows how the kinetics of a ferrite to austenite transformation is affected by simultaneous precipitation of niobium carbide. The transformation is caused by carburization.
Cell calculations	
c1	Carbon cannon in ferrite/austenite Fe-C system, two-cell calculation. This example simulates what happens to a FERRITE plate that has inherited the carbon content of its parent AUSTENITE. The FERRITE plate formed is embedded in an AUSTENITE matrix. This setup corresponds to a proposed mechanism for formation of WIDMANNSTÄTTEN FERRITE or for the FERRITE phase of the BAINITE structure. It is assumed that the phase boundary between FERRITE and AUSTENITE is immobile, this is achieved in the simulation by putting the FERRITE and the AUSTENITE in two different cells. Also see M. Hillert, L. Höglund and J. Ågren: Acta Metall. Mater., v.41 (1993), pp. 1951-1957.
c2	Cementite dissolution in an Fe-Cr-C alloy (three different particle sizes). Setup file for calculating the dissolution of CEMENTITE particles in an AUSTENITE matrix. This case is identical to exb2 except that we here have three different particle sizes. Altogether six particles are considered using three different cells. This in order to be able to represent some size distribution among the CEMENTITE particles. Also see Z.-K. Liu, L. Höglund, B. Jönsson and J. Ågren: Metall.Trans.A 22A (1991), pp. 1745-1752.
Diffusion in dispersed systems	
d1a	Carburization of Ni-25%Cr alloy. Setup file for carburization of a Ni-25Cr alloy. In this case the M3C2 and M7C3 carbides are entered as spheroid phases in a FCC matrix. This case is from A. Engström, L. Höglund and J. Ågren: Metall. Trans. A, v. 25A (1994), pp. 1127-1134. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. In this example the DISPERSED SYSTEM MODEL is used, which requires that the default HOMOGENIZATION MODEL is disabled. With the DISPERSED SYSTEM MODEL the command ENTER_LABYRINTH_FUNCTION is used to take into account the impeding effect of dispersed phases on long-range diffusion. For the HOMOGENIZATION MODEL the command ENTER_HOMOGENIZATION_FUNCTION should be used.
d1b	Carburization of Ni-25%Cr alloy, using homogenization model. Setup file for carburization

Example	Description
	of a Ni-25Cr alloy. In this case the M3C2 and M7C3 carbides are entered as spheroid phases in a FCC matrix. This case is from A. Engström, L. Höglund and J. Ågren: Metall. Trans. A, v. 25A (1994), pp. 1127-1134.This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. Here the default HOMOGENIZATION MODEL is used and then ENTER_HOMOGENIZATION_FUNCTION should be used instead of ENTER_LABYRINTH_FUNCTION.
d2a	Diffusion couple of Fe-Cr-Ni alloys (initially with a step-profile). Setup file for calculating the interdiffusion in a diffusion couple between a two-phase (FCC+BCC) and a single-phase (FCC) Fe-Ni-Cr alloy. This case is from A. Engström: Scand. J. Met., vol. 24, 1995, pp.12-20. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. In this example the DISPERSED SYSTEM MODEL is used, which requires that the default HOMOGENIZATION MODEL is disabled. With the DISPERSED SYSTEM MODEL the command ENTER_LABYRINTH_FUNCTION is used to take into account the impeding effect of dispersed phases on long-range diffusion. For the HOMOGENIZATION MODEL the command ENTER_HOMOGENIZATION_FUNCTION should be used.
d2b	Diffusion couple of Fe-Cr-Ni alloys, using homogenization model (initially with a step-profile). Setup file for calculating the interdiffusion in a diffusion couple between a two-phase (FCC+BCC) and a single-phase (FCC) Fe-Ni-Cr alloy. This case is from A. Engström: Scand. J. Met., vol. 24, 1995, pp.12-20. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. The default HOMOGENIZATION MODEL is used and then ENTER_HOMOGENIZATION_FUNCTION is used instead of ENTER_LABYRINTH_FUNCTION.
d3	Diffusion couple of Fe-Cr-Ni alloys (using the new model). This example shows the use of the homogenization model. It is taken from H. Larsson and A Engström, Acta Mater, v.54 (2006), pp. 2431-2439. Experimental data from A Engström, Scand J Metall, v.243 (1995), p. 12. The homogenization model can be used for multiphase simulations like the dispersed system model, but unlike the dispersed system model there is no need to have a single continuous matrix phase and, furthermore, there is no need to limit the size of time-steps. The set-up is performed in the same manner as for the dispersed system model, which means that a certain phase is entered as the matrix phase and the other phases are entered as spheroidal, but the choice of matrix phase does not affect the simulation.
Cooperative growth	
e1	Growth of pearlite in an Fe-Mn-C alloy. This is an example file for a setup and a calculation of PEARLITE growth in an Fe - 0.50wt%C - 0.91wt%Mn steel.
Coarsening	
f1	Coarsening of M6C precipitate in an Fe-Mo-C alloy. Setup file for calculating the Ostwald-

Example	Description
	ripening of a spherical M6C carbide in an AUSTENITE matrix.
Kinetic data	
g1	Kinetic data file for checking mobilities and diffusivities in an Fe-Ni alloy.
g2	Optimization of mobilities in Ni-Al fcc alloys. Kinetic data file for reading thermodynamic data and setting up the kinetic parameters which are needed for an optimization of the FCC phase in the binary Ni-Al system. Also see A. Engström and J. Ågren: Assessment of Diffusional Mobilities in Face-Centered Cubic Ni-Cr-Al Alloys, in the February issue of Z. Metallkunde, 1996.
Deviation from local equilibrium	
h1	Ferrite/austenite diffusion couple with interface mobility. Setup file for calculating the growth of FERRITE into AUSTENITE with a limited interface mobility. This is achieved by adding a Gibbs-energy contribution to the FERRITE using the SET-SURFACE-ENERGY command.
h2	Ferrite/austenite para-equilibrium in an Fe-Ni-C alloy. Setup file for calculation of the growth of FERRITE into AUSTENITE in an Fe-2.02%Ni-0.0885%C alloy using the para-equilibrium model. The results are compared with experimental information from Hutchinson, C. R., A. Fuchsmann, and Yves Brechet. "The diffusional formation of ferrite from austenite in Fe-C-Ni alloys." Met. Mat. Trans A 35.4 (2004): 1211-1221.
h3	Diffusion in a temperature gradient. This calculation shows how a temperature gradient induces diffusion.
Diffusion in complex phases	
i1	Diffusion in a system with B2-ordering. Diffusion in including effects from chemical ordering. The datafile AlFeNi-data.TDB contains both a thermodynamic and kinetic description for the ordered and disordered bcc.
i2	Diffusion of carbon in cementite. This example demonstrates the use of the model for calculation of diffusion through a stoichiometric phase. The flux of a component in the stoichiometric phase is assumed to be proportional to the difference in chemical potential at each side of the stoichiometric phase multiplied with the mobility for the component in the phase. The mobility is assessed from experimental information and is basically the tracer diffusivity for the component. This calculation is compared with experimental data where a sample of pure iron has been exposed to a gas atmosphere with a certain carbon activity. The weight gain is then measured as a function of time. The experimental data is obtained from Ozturk B., Fearing V. L., Ruth A. Jr. and Simkovich G.,

Example	Description
	Met Trans A, vol 13A (1982), pp 1871-1873.
i3a	Diffusion in iron oxide. Oxidation of iron sample and consequent growth of an oxide layer.
i3b	Diffusion in iron oxide with a GB contribution. Oxidation of iron sample and consequent growth of an oxide layer using the grain boundary diffusion contribution model.

Opening a Macro File

Opening Macro Files

The default directory where the software and the examples are installed vary by platform and user type. On Windows, once Thermo-Calc is installed, go to **Start → All Programs >Thermo-Calc** and click **Examples** to open the folder.

The examples are in the first folder called **Beginner Guide**.

For other platforms, go to *Help Resources* on page 5 and search the online help for *Default Directory Locations*.

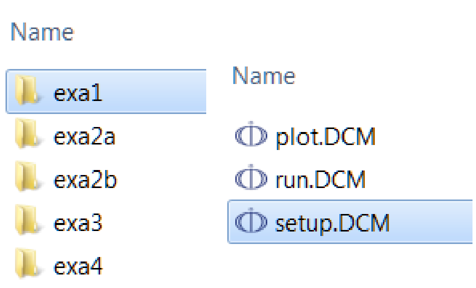
You can open the example macro files in these ways.

Using a Command in Console Mode

At the `SYS` prompt, type MACRO_FILE_OPEN and press <Enter>. In the file dialogue window that opens, locate and select the macro file on your computer.

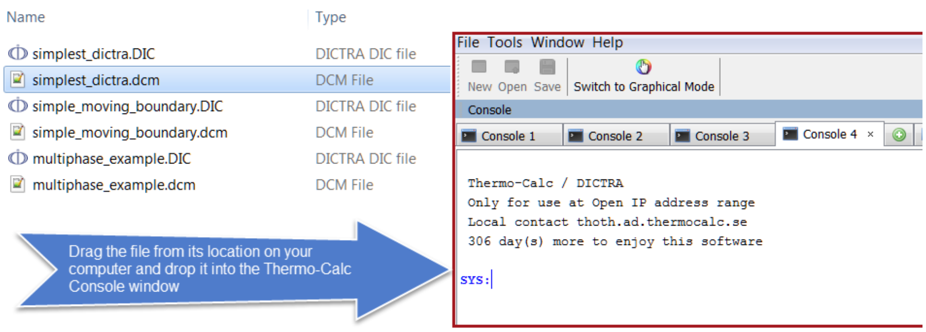
Double-click to open or drag and drop a file into Console Mode

Navigate to the file location, where each set of file examples is in its own folder. Each example consists of three macro files: a **setup** macro, a **run** macro and a **plot** macro. You must run these three macros in order.

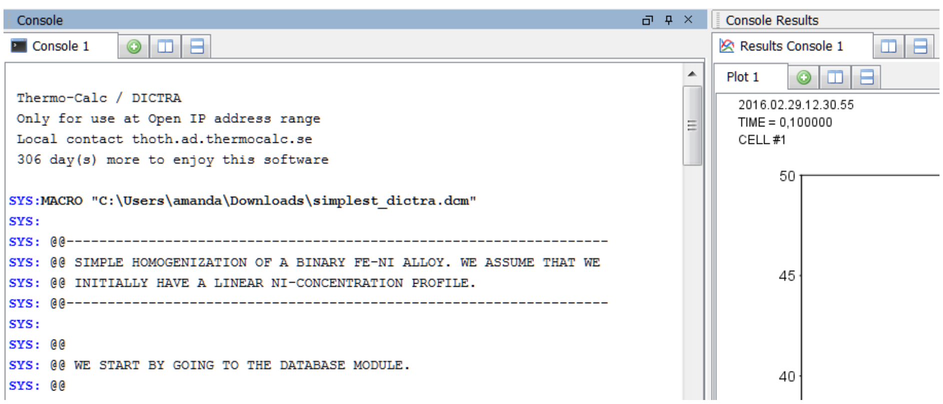


To begin, double-click the **setup.DCM** file to open it. It launches a new session of Thermo-Calc and opens in Console Mode. You need to start with the setup.DCM file.

Navigate to the **Beginner Guide** folder and then drag the file from its location into Thermo-Calc Console window.



The macro automatically runs after you drop it into the Console window.



Installed Example Locations

On Windows, once Thermo-Calc is installed, you can also locate the Examples and Materials folders, plus all the Manuals using the shortcuts located in the Start menu. Go to **Start → All Programs >Thermo-Calc** and click **Examples**, **Manuals**, or **Materials** as required to open the applicable folder.

In the table, *<user>* stands for the username and *<version>* for the version of Thermo-Calc, for example 2016a.

OS	User type	Default directory
Windows	Normal user	Users*<user>*\Thermo-Calc*<version>* Users*<user>*\Documents\Thermo- Calc*<version>* My documents
	Administrator	Program Files\Thermo-Calc*<version>* Users\Public\Documents\Thermo- Calc*<version>* Public documents
Mac	Administrator (user name and password required)	Examples and manuals in /Users/Shared/Thermo-Calc/*<version>* To go to this folder, in Finder, from the **Go** main menu select **Go to folder**. Enter the above file path and click **Go**.
Linux	Non root user	home/*<user>*/Thermo-Calc/*<version>*
	Root user	usr/local/Thermo-Calc/*<version>*

Diffusion Module (DICTRA) Licenses

You can start (and install) the Thermo-Calc software without a valid license but you cannot do any calculations. To show information about the available and installed licenses, from the main menu select **Help → Show License Info**.

The Diffusion Module (DICTRA) requires a separate license. If you are using a network client installation of Thermo-Calc, then you may not be able to use the Diffusion Module (DICTRA) even if you have access to a license server with a valid network license file. This is because other clients that are part of your network installation may have checked out all instances of the network license allowed to run simultaneously. A Diffusion Module license is checked out when you enter the DICTRA module and it is checked back in when you exit.

If you try to enter DIC_PARROT after having been in the PARROT module earlier in the session, a message displays: `DICTRA PARROT disabled after using PARROT`. To enter DIC_PARROT relaunch Thermo-Calc and enter these commands starting in the SYS module: GOTO_MODULE DIC_PAR.

Search the online help or see the Thermo-Calc Installation Guide for more about license types.

Using this Guide

> Before using this guide, review the *Thermo-Calc User Guide* for details about using Console Mode, as well as the *Console Mode Command Reference* and *Diffusion Module (DICTRA) Command Reference* all included with this documentation set.

This guide includes an overview of the program and its components, and describes in generic terms how you can define your system of components, set up and perform simulations, as well as visualize the results.

Working in the Diffusion Module (DICTRA) on page 16 introduces you to the terminology, workflow and file formats in this add-on module.

The next topic is an overview of *Basic Concepts* on page 22 including descriptions of cells, regions, phase, and phases, for example. The topic also gives examples of the different simulation types to prepare you for defining your own system.

The next topics walk you through the concepts and general workflow to set up a simulation.

System Definition: Retrieving Data on page 29 describes how you define the system and retrieve thermodynamic and kinetic data about the elements.

Configuring a Diffusion Module (DICTRA) Calculation on page 31 and *Diffusion Module (DICTRA) Simulations* on page 40 describes how you configure and run simulations, and *Plotting Simulations in the POST Module* on page 47 explains how to plot the simulation results.

The *Troubleshooting* on page 52 topic has some recommendations and guidelines to adjust configurations and settings if you have difficulty with any of the steps.

Working in the Diffusion Module (DICTRA)

In this section:

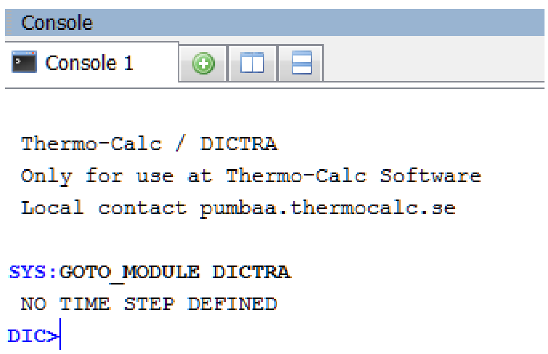
Opening DICTRA and Terminology

Opening the DICTRA Module in Thermo-Calc

1. Open Thermo-Calc in Console Mode.

By default, Thermo-Calc opens in the most recently used state. If it opens in Graphical Mode, on the toolbar, click **Switch to Console Mode**.

2. In the `SYS` module, type `GOTO_MODULE DICTRA`. For a list of all the available commands in the current module, at the prompt type a question mark (?) and press <Enter>. For a description of a specific command, type `Help` followed by the name of the command.



Terminology

The Diffusion Module (DICTRA) has two Console Mode modules: the DICTRA_MONITOR module and the DIC_PARROT module. In this guide, *DICTRA* and *DICTRA module* are used interchangeably to refer to the DICTRA_MONITOR module.

As of 2016, DICTRA is considered an add-on module to Thermo-Calc and is referred to as the Diffusion Module in some cases. The functionality is the same. It is only the name that has changed to better describe what the module does.

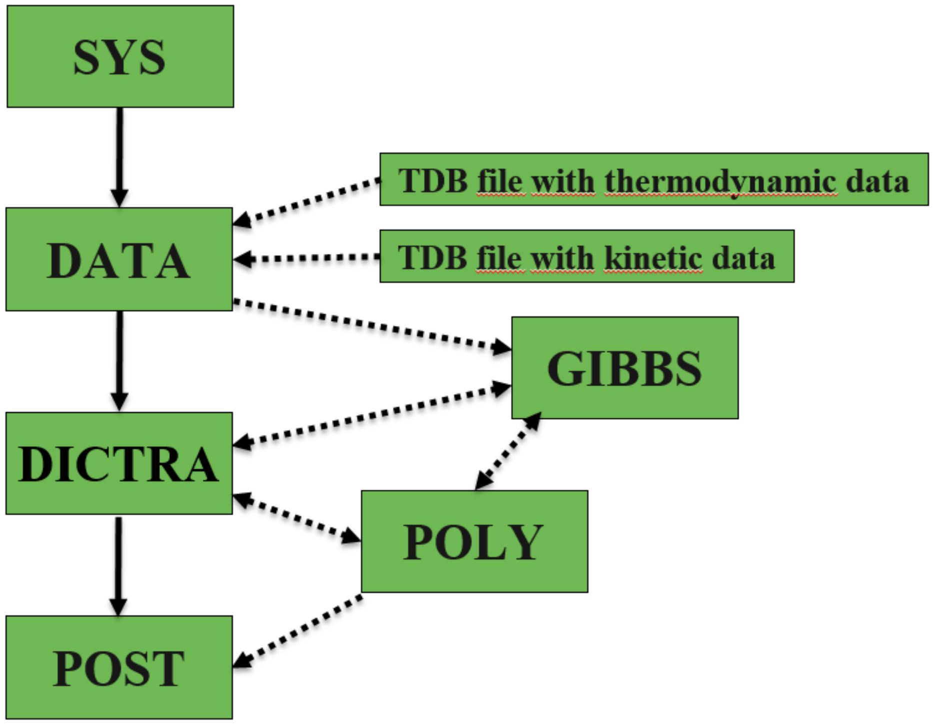
The DIC_PARROT module assesses experimental data and uses this data to optimize calculations.

Also see *PARROT Commands* on page 86 in the *Console Mode Command Reference*.

Typical Workflow

Click the link to see an example of the typical workflow when you set up a problem and run a simulation in the Diffusion Module (DICTRA). The solid arrows represent the typical movements between the modules. The dashed arrows represent movement of data within the DICTRA module.

An example of a typical workflow



The basic workflow is as follows.

1. You are in the **SYS** module when you start Thermo-Calc in Console Mode.

   In the **DATA** module define the system. Before performing a calculation, you must define your system and retrieve thermodynamic and kinetic data. The data is retrieved from database files (*.TDB).

   The data needed to describe a system is sent to the **GIBBS** module. When the data is in GIBBS, the software can access the retrieved data. The retrieved data is accessible to the **POLY** module, which is involved in simulations that require that equilibrium calculations are performed.

2. In the DICTRA module:

- Set the initial state of the system and define its geometry and boundary conditions. The simplest system has one cell that contains one region, but you may put additional regions in a cell and put additional cells in your system.
- Define which phases each region contains.
- Set various parameters for the numerical calculations.
- Run the simulation from within the DICTRA module.

3. In the **POST** module plot and visualize your data.

- Choose the data to plot
- Change the appearance of the diagram
- Save the diagram as an image file (in many different formats).

Diffusion Module (DICTRA) File Formats

The Diffusion Module (DICTRA) uses different file formats: *log* files (*.LOG), *macro* files (*.DCM) and *workspace* files (*.DIC).

Log Files

Log files (*.LOG) are plain text files used to save a sequence of commands. Log files can be edited in a text editor.

- To start saving your input into a log file, in the SYS module type `SET_LOG_FILE`, followed by the name of the file that you want to save your command sequence to.
- When creating a command sequence, and if you want to save the output in the log file, type `SET_ECHO` before *SET_LOG_FILE* on page 242. This is useful if you later want to use the log file as a macro file.

Macro Files

Macro files (*.DCM) are plain text files used to save a sequence of commands that can be loaded and executed. Macro files can be edited in a text editor. When creating a command sequence, you add comments to the file by starting a line with @@.

- To run a macro file, in the SYS or DICTRA module type `MACRO_FILE_OPEN`, followed by the name of the macro file. DICTRA runs the *MACRO_FILE_OPEN* on page 5 sequence that the file contains.
- To remain in the console and keep working in the DICTRA module, in the SYS, DICTRA, POLY, or POST module add the command *SET_INTERACTIVE* on page 8 at the end of the macro file. Alternatively, use `EXIT` as the final command.

Workspace Files

Workspaces files (*.DIC) allow you to save all the data in your current workspace. A workspace contains all the data, specified settings and the result of any calculations performed. The saved data includes original and modified thermodynamic data, the last set of conditions and options, and the results of any simulations.

- At the `DIC` prompt use *SAVE_WORKSPACES* on page 30.

- To load the data and calculation results of a workspace file, at the `DIC` prompt use *READ_WORKSPACES* on page 160.

Operators and Functions

Sometimes you need to specify functions when setting up a system. For example, to make global and boundary conditions vary as a function of time. The following operators can be used in writing functions. Use a semi-colon (`;`) to mark the end of a function.

Available Operators for Writing Functions

Operator	Description
+, -, *, /	addition, subtraction, multiplication, division
**	exponentiation
SQRT(X)	square root
EXP(X)	Exponential
LOG(X)	natural logarithm
LOG10(X)	base 10 logarithm
SIN(X), COS(X), ATAN(X)	sinus, cosinus and arctangent
SIGN(X)	sign function SIGN(X) is -1 when X<0, is 0 when X=0 and is 1 when X>0.
ERF(X)	error function

Specifying Time-Dependent Profiles

In the DICTRA module you are often prompted to specify a time-dependent profile for a condition.

Time-Dependent Profiles

To define these profiles or intervals, enter information as follows and refer to the *Diffusion Module (DICTRA) Command Reference*:

Command profile	Description
LOW TIME LIMIT	To determine the time when the function starts to apply.
HIGH TIME LIMIT	To determine the time when the function stops applying. Use an asterisk (*) to set the high limit to the time when the simulation ends.

Command profile	Description
ANY MORE RANGES	Enter `Y` to specify an additional range with a time-dependent function, which starts to apply at the time step after the high time limit of the previous range is reached.

Basic Concepts

In this section:

System, Cells and Regions

A system normally consists of one *cell*. This cell contains one or several *regions* in which the diffusion problem is to be solved. A system of diffusion equations is solved for each region. The size of the system is the sum of sizes of all the regions in all the system's cells.

The interface between two regions typically moves as the result of a simulation, that is, the regions grow or shrink. Such an interface is called an *inner interface*. The boundary of a cell is called an *outer interface* and the condition of this type of interface may be affected by the choice of boundary conditions.

For some simulations it is useful to have several cells in a system. Matter can move between cells. However, it is assumed that there is diffusional equilibrium between all cells (this means that there are no differences in diffusion potentials between cells). In other words, the size of a cell is fixed during the simulation.

By default this is a *closed system*, which means that matter cannot move across its boundaries. However, you can change this setting for both the *lower boundary* (left side/centre) and the *upper boundary* (right side/surface) of the system. You can also set these boundary conditions to vary as a function of time, temperature and/or pressure.

Global Conditions

There are several global parameters that influence the diffusion rate in simulations: temperature, pressure and heat content removal. These parameters are defined as functions of time. The temperature can also be a function of the spatial position, and heat content removal can be a function of temperature or pressure.

For heat content removal, the amount of extracted heat per time unit is normalized and the size of the system is normalized to one mole of atoms.

Also see *Setting Global Conditions* on page 32 in the *Thermo-Calc User Guide*.

Geometry and Coordinates

The Diffusion Module (DICTRA) can only handle diffusion problems where composition vary along one spatial coordinate. The geometry of your system can be *planar*, *cylindrical* or *spherical*. The examples in this guide use the default planar geometry.

Planar Geometry

This geometry corresponds to an infinitely wide plate of a certain thickness. If the system has a planar geometry, then the lower boundary (the zero coordinate) is at the left side of the system. The upper boundary (the coordinate with the highest value) is at the right side of the system.

Cylindrical and Spherical Geometries

Cylindrical geometry corresponds to an infinitely long cylinder of a certain radius.

Sypherical geometry corresponds to a sphere with a certain radius.

For both geometries, then the lower boundary (the zero coordinate) is by default at the centre of the cylinder/sphere. The upper boundary (the coordinate with the highest value) is at the *surface* of the cylinder/sphere. You can enter a leftmost coordinate that is larger than zero to allow for tube and hollow sphere geometries.

Also see *Specifying the Geometry* on page 32.

Regions and Grids

A *region* contains one or more phases. If a region contains several phases one of those must be entered as a *matrix phase* whereas the others are entered as *spheroid*.

A region must also contain a number of *grid points*. The composition is only known at these grid points and the software assumes that the composition varies linearly between them. The amount and composition of all the phases present at a single grid point in a certain region are those given by thermodynamic equilibrium keeping the over-all composition at the grid point fixed.

Also see *Regions and Gridpoints* on page 32.

Phases

Each region contains one or more *phases*. A phase can be of the *matrix*, *spheroid* or *lamellar* type. A phase can be introduced with the status *active* (the default) or *inactive*.

An inactive phase is a phase which does not participate in the calculations until it is stable. If a phase is inactive, then the driving force for precipitation of the phase is continuously monitored. When the driving force exceeds a predefined positive value, the phase take parts in the simulation and the phase automatically becomes active. A new region is then created and the formerly inactive phase is entered as the matrix phase of the new region. The name of the region is set to the name of the phase that just became active but with `R_` added as a prefix.

Also see *Phases in Regions* on page 36 and *Compositions in Phases* on page 37.

Region Phase Types

Phase type	Description
Matrix	The region’s primary phase. If there is only one phase in a region, then that phase is automatically a matrix.
Spheroid	Used to treat one or several dispersed phases in a matrix. Unless the homogenization model is invoked, which happens automatically when a spheroid phase is entered, then diffusion is only considered in the matrix phase; the spheroid phases then only act as sources and sinks of the diffusing elements.
Lamellar	Used to simulate the growth of a eutectic or eutectoid region. The software only treats the growth, not the dissolution, of such a region. The *pearlite calculation model* is used for a phase that is lamellar.

Composition Variables

Two types of composition variables are used: site-fractions and u-fractions, although you can also use weight fraction or mole fraction, for example.

The *site-fractions* are used to set up the problem and interface with POLY-3. The site fraction of a species *k*, y_k, is the fraction of equivalent lattice sites occupied by species k. It is the only concentration variable that uniquely defines the composition and the constitution of a phase and it is necessary to use the site fraction in the storage of the data during a simulation.

The *u-fractions* are used in the diffusion equations. The u-fraction of a component *k* is defined as

$$u_k = \frac{x_k}{\sum_{j \in S} x_j}$$

The summation is only performed over the substitutional elements. The choice of the volume fixed frame of reference in the calculations make it convenient to use a composition variable which is related to the volume. The definition of the u-fraction is based upon the assumption that a species may or may not contribute to the volume of a phase. An interstitial element is assumed to not contribute to the volume. Substitutional elements are assumed to contribute equally to volume.

Diffusion Module (DICTRA) Simulation Types

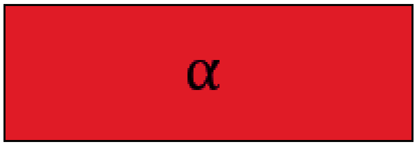
The Diffusion Module (DICTRA) is used to simulate diffusion-controlled transformations in multicomponent alloys. The following are some of the types of simulations it can handle.

Also see *Diffusion Module (DICTRA) Examples Collection* on page 6.

Diffusion in a Single-Phase System

The simplest simulation is one where certain components in one phase diffuse over time in that single phase. To simulate this, you create one region and enter one phase into that region. You can set up the system under various conditions by defining profiles for how temperature and pressure change over time, or for how the boundary conditions of the region change over time.

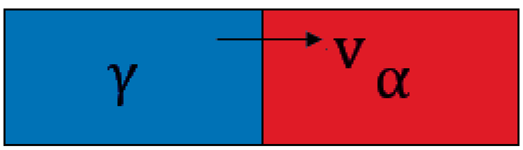
Simulate diffusion in a single-phase system.



Moving Boundary Multi-Phase System

With two regions in a single cell you can simulate how diffusion causes phase transformations. For example, you can simulate how an individual particle grow or dissolve as a function of time. The result is a simulation of how the boundary between the regions migrate over time. As in the case of a one-phase simulation, you can set up the system under various conditions by defining profiles for how temperature changes overtime, or for how the boundary conditions of the region change overtime.

Simulate a moving boundary multi-phase system

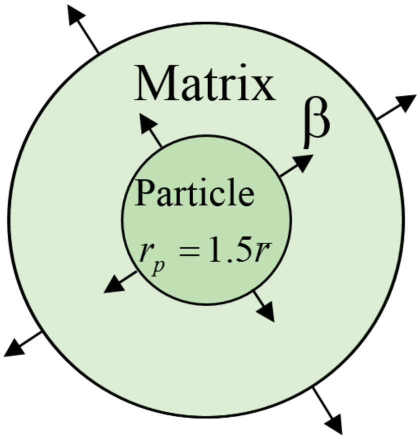


Normally, simulations are performed under the assumption that thermodynamic equilibrium holds locally at all phase interfaces. However, it is also possible to simulate a moving phase boundary problem while assuming that para-equilibrium conditions apply at the phase boundary.

Coarsening Processes

You can run a simulation of a coarsening process. In accordance with Lifshitz-Slyozov-Wagner theory, this is done by assuming that the coarsening can be simulated by calculating the growth of a single particle of the maximum size in the particle size distribution.

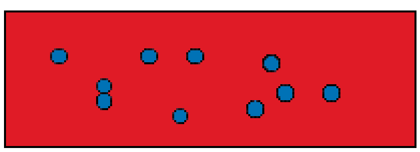
Simulate a coarsening process



Diffusion in Disperse Systems

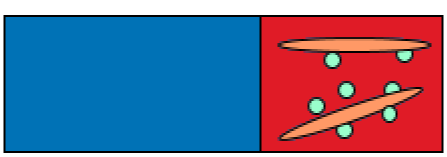
You can perform simulations where one or more dispersed phases are present in a matrix phase. For example, you can simulate how carburization causes precipitation of carbides in an austenitic matrix phase. The dispersed phases act as point sinks or sources of solute atoms in the simulation and the fraction and composition is calculated from the average composition in each node, assuming that equilibrium holds locally in each volume element.

Simulate diffusion in a disperse system



There are two ways in which you can simulate diffusion in a system that contains a dispersed phase, but it is strongly recommended that you use the *homogenization model*. The homogenization model allows you to take into account diffusion in all phases for which you have kinetic data; the homogenization model treat all phases in the same way regardless of which phase was entered as type matrix and which phases were entered as type spheroid.

Simulate diffusion with the homogenization model



Cooperative Growth

You can simulate the growth of pearlite in alloyed steels by calculating the Eutectoid reaction γ→α+β, where α and β grow cooperatively as a lamellar aggregate into the γ matrix.

Defining the System

In this section:

System Definition: Retrieving Data

In order to perform a simulation in the Diffusion Module, both a thermodynamic database and a kinetic database is needed. The thermodynamic database is a normal Thermo-Calc database, whereas the kinetic database contains information about the atomic mobility of individual components in various phases. The next topic, *Defining a System* below is a short summary of the available commands.

To avoid making the diffusion problem unsolvable, it is recommended that you make the problem as simple as possible. Therefore, define your system with as few components and phases as possible.

Defining a System

See the *DATABASE_RETRIEVAL Commands* on page 11 in the *Thermo-Calc Console Mode Command Reference*.

1. Open Thermo-Calc in Console Mode. At the `SYS` prompt, type `GOTO_MODULE DATA`.
2. Use SWITCH_DATABASE to select databases. Unless you directly specify the name of the database as a parameter to the command, the available databases are listed. Specify the one to use.
3. Use DEFINE_ELEMENTS followed by a list of the elements that you want in your system. (To list the elements available in the current database, type `LIST_DATABASE ELEMENTS`.)
4. Use REJECT with the keyword, `phases` to make sure that unnecessary phases are not retrieved from the database.
5. Use RESTORE with the keyword, `phases` to restore any phases that have been excluded.
6. Use GET_DATA to read from the database and send the thermodynamic data about your system to the GIBBS and DICTRA workspace.
7. Use APPEND_DATABASE to select the database from which you want to retrieve kinetic data.
8. Define your elements and specify whether to reject and restore any phases. Do this in exactly the same way as you would for the thermodynamics data (see the preceding step 2, 3 and 4).
9. Use GET_DATA to read from the database and add the kinetic data to the thermodynamic data that already exists in the GIBBS and DICTRA module workspace.

10. Use APPEND_DATABASE again to add thermodynamic or kinetic data from another database. When all the data needed is retrieved, you can start using the DICTRA module.

Configuring a Diffusion Module (DICTRA) Calculation

These topics describe in general terms what options you have when setting up your problem.

In this section:

Setting Global Conditions

Use *SET_CONDITION* on page 34 and choose `Global`. Global conditions include temperature (T), heat content removal (Q) and pressure (P), and these reduce the degrees of freedom for equilibrium calculations.

Also see *Global Conditions* on page 23.

If the heat content removal is specified then only the initial temperature should be entered with the *SET_INITIAL_TEMPERATURE* on page 36 command.

- In practice, the pressure is rarely specified. By default the pressure is assumed to be constant and equal to 1 bar.
- For each condition, you can define a profile over time that consists of several intervals of time, or *ranges*, with each range having its own function to define the value of the condition.
- Temperature can be entered as a function of time (TIME) and/or spatial coordinate (X). The pressure condition can be defined as a function of time (TIME). The heat content removal can be defined as a function of temperature (T), pressure (P) and/or time (TIME).

  For example, the following function could specify a cooling rate if it is used to specify how the global condition temperature (T) changes with time (TIME) during a certain range: `1900-1*TIME;`

Specifying the Geometry

By default a DICTRA module system has a planar geometry. To change the geometry of the system, use *ENTER_GEOMETRICAL_EXPONENT* on page 11 followed by a number `0` (planar), `1` (cylindrical) or `2` (spherical).

Also see *Geometry and Coordinates* on page 23.

With a cylindrical or spherical geometry, the system's zero coordinate (left boundary) is at the centre of the cylinder or sphere. The highest coordinate (right boundary) is defined by the cylinder's or sphere's radius. If you want a geometry corresponding to a tube or a hollow sphere, then use *SET_FIRST_INTERFACE* on page 36 to locate the system's left boundary at a coordinate that is greater than zero.

Regions and Gridpoints

Enter a new region by specifying its name and, unless it is the first region, how it should be located with respect to one already existing region. The size of a region is determined when a grid is entered into the region.

Also see *System, Cells and Regions* on page 23 and *Regions and Grids* on page 24 and *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

Creating a Region

1. Complete *Defining a System* on page 29.
2. Use ENTER_REGION to enter a region into the system.
3. Enter an arbitrary name for the region at the `REGION NAME` prompt.
4. If this is the first region you create in your system, then the procedure stops here. Otherwise, continue to the next step.
5. Enter the name of an already existing region to which the new region is attached.
6. Specify whether the new region should be `ATTACHED TO THE RIGHT OF` the existing region. Enter `Y` to attach to the existing region's right side or `N` to attach it to its left side.
7. If the system has a cylindrical or spherical geometry, then this is a question of whether to put the new region outside the existing region, so that the new region surrounds the existing one, or to put it at the existing region's centre, so that the existing region surrounds the new region.

Adding Gridpoints to a Region

This topic describes how you specify the size and grid point distribution to use in a region.

1. Complete *Defining a System* on page 29 and Creating a region.
2. Use ENTER_GRID_COORDINATES followed by the name of the region.
3. Specify the width of the region in meters. .
4. At the `NUMBER OF POINTS` prompt enter the number of grid points for the region.
5. Specify the type of grid point distribution to have in the region.

- `LINEAR`, for an equally spaced grid.
- `READ_POINT_BY_POINT` to read a number of specific grid points from a data file. The input source from which to read the points when entering values point by point. To enter the grid points at the terminal, enter `READ_POINT_BY_POINT TERMINAL`.
- `GEOMETRIC` for a grid that yields a varying density of grid points in the region. Enter the `VALUE OF R IN THE GEOMETRICAL SERIES`, which determines the distribution. A

geometrical factor larger than one yields a higher density of grid points at the lower end of the region and a factor smaller than one yields a higher density of grid points at the upper end of the region.

- `DOUBLE_GEOMETRIC` to have a high number of grid points in the middle or at both ends of a region. Specify two geometrical factors. `VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION` for the distribution in the lower (left) part of a region, and `VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF SUBREGION` for the distribution in the upper (right) part of a region.

Boundary Conditions

Boundary conditions are conditions that define how matter behaves at the boundaries of your system. By default, matter is not allowed to cross the system boundaries.

By default the system is closed, which means that matter cannot move across its boundaries. However, you can change this setting for both the lower boundary (left side/centre) and the upper boundary (right side/surface) of the system. The examples in this guide use the default closed system.

Also see *SET_CONDITION* on page 34.

Boundary Condition Descriptions

Boundary condition	Description
FIX_FLUX_VALUE	Enter functions that yield the flux times the molar volume for the independent components. Can be a function of time, temperature and pressure. Use this boundary condition with caution as it may cause unrealistic compositions.
STATE_VARIABLE_ VALUE	A legal equilibrium condition expression in POLY-3 syntax that reduces the degrees of freedom. This type should be used with care as no checks are done in advance to determine if it is a legal expression. Do not specify the conditions for temperature and pressure here as these are taken from the global conditions.
POTENTIAL_FLUX_ FUNCTION	Use to take into account the finite rate of a surface reaction. The flux for the independent components must be given in the format: $J_k = f_k(T,P,TIME) * \left[POTENTIAL_k^N - g_k(T,P,TIME)\right]$ The functions *f* and *g* may have time (TIME), temperature (T), and/or pressure (P) as arguments. *N* is an integer. The potentials are those with user defined reference states.

Boundary condition	Description
ACTIVITY_FLUX_ FUNCTION	Use to take into account the finite rate of a surface reaction. The flux for the independent components must be given in the format: $J_k = f_k(T,P,TIME) * \left[ACTIVITY_k^N - g_k(T,P,TIME)\right]$ The functions *f* and *g* may have time (TIME), temperature (T), and/or pressure (P) as arguments. *N* is an integer. The activities are those with user-defined reference states.
ITERATIVE_ACTIVITY_ FLUX_ FUNCTION	Same as ACTIVITY_FLUX_FUNCTION but an iterative scheme is used to determine the flux. Use this if the other command encounters problems.
CLOSED_SYSTEM	Corresponds to a fix flux value, which is set to zero at all times. This is the default boundary condition.
MIXED_ZERO_FLUX_ AND_ ACTIVITY	The flux of selected components is set to zero and the activity of others may be set to a prescribed value.
GAS	This option can cause a change in system volume. The flux of selected components is set to zero and the activity of others can be set to a prescribed value. Use this to treat an expanding system, e.g. the growth of an external oxide scale.

Setting Boundary Conditions

1. Complete *Defining a System* on page 29 and *Regions and Gridpoints* on page 32.
2. Use *SET_CONDITION* on page 34 and choose `Boundary`. Then specify whether you want to set the LOWER or the UPPER boundary.
3. Specify which condition type you want to set the boundary to.
4. Enter any parameter values that are necessary for the condition type you have chosen.
5. Specify the `LOW TIME LIMIT`, when the function you define starts to apply.
6. If required, enter the function that is applied during the time interval. For most condition types, the function may have time, temperature and pressure as arguments.
7. Enter the `HIGH TIME LIMIT` for the interval.
8. If you want to define a different function for another time interval, then enter `Y` when prompted with `ANY MORE RANGES`. Then enter another function for the next range.
9. When you have defined functions for all the ranges you want, enter `N` at the `ANY MORE RANGES` prompt.

Phases in Regions

There is no supersaturation locally within a given region.

When defining a region you need to enter one or more phases into the region. The number and type of phases depends on the simulation you want to run.

- To simulate diffusion of components within one phase, enter one matrix into the region.
- To simulate how an interface between regions moves, enter two regions with at least one phase in each.
- To simulate growth of one or more dispersed phases in a matrix phase, enter one matrix phase and one or more spheroid phases.

Also see *Phases* on page 24 and *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

Entering Phases in a Region

1. Complete *Defining a System* on page 29 and *Regions and Gridpoints* on page 32.
2. Use ENTER_PHASE_IN_REGION. Specify whether the phase is to be `ACTIVE` or `INACTIVE`.

3. For an inactive phase, specify the following:

- Enter the name of the region and the side of that region which the phase is attached to.
- Enter the required driving force (evaluated as DGM (phase) in POLY-3) to be used for determining whether an inactive phase is stable.
- Enter the `CONDITION TYPE` that the phase boundary condition should be set to if the inactive phase becomes stable. (Inactive phases are only considered in the simulation calculations once it is stable.)

4. For an active phase specify the following:

- `REGION NAME` for the region in which the phase is to be entered.
- `PHASE TYPE` -`MATRIX`, `LAMELLAR` or `SPHEROID`. `SPHEROID` is only available if a `MATRIX` phase is created first. You can only enter one matrix phase in each region.
- Enter the `PHASE NAME`. You can append a hash sign (`#`) and a digit at the end of the phase name to designate the composition set number (e.g. `FCC#2`). If you do not append this and the phase has more than one composition set, then you are prompted to specify a composition set number.

Compositions in Phases

When you specify the composition of a phase, you do this by specifying the distribution of the phase within its region—a composition profile.

Also see *Phases* on page 24 and *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

Specifying Phase Compositions in a Region

1. Complete *Defining a System* on page 29, *Regions and Gridpoints* on page 32 and *Phases in Regions* on the previous page.

2. Use ENTER_COMPOSITIONS to enter a composition of a phase in a certain region.

3. Enter the `REGION NAME` of the region in which the phase has previously been entered.

4. Enter the `PHASE NAME` of the phase whose composition you want to specify.

5. If the phase is spheroid, then for `USE EQUILIBRIUM VALUE` enter `Y` to automatically calculate the equilibrium fractions of the phase and its constitution at the start of the

simulation. Specify the initial `VOLUME FRACTION OF` the spheroid phase.

6. Enter the `DEPENDENT COMPONENT`. Sometimes the dependent component is automatically set and there is no prompt.
7. Enter a `COMPOSITION TYPE` for the constitution of the phase. The options are:

   SITE_FRACTION

   MOLE_FRACTION

   MOLE_PERCENT

   WEIGHT_FRACTION

   WEIGHT_PERCENT

   U_FRACTION
8. Enter the `TYPE` of composition distribution (that is, the type of composition profile). The options are:

   - `LINEAR` to have an equally spaced composition distribution. Specify the composition at the first (leftmost) grid point (`VALUE OF FIRST POINT`) and at the last (rightmost) grid point (`VALUE OF LAST POINT`).
   - `READ_POINT_BY_POINT` to read the composition at each grid point. The specification of these can either be read from a DATA-file that you are prompted to select or entered directly at the terminal. If you want to enter them at the terminal, then enter `READ_POINT_BY_POINT TERMINAL`.
   - `GEOMETRIC` for a distribution that yields a higher concentration of the component at the lower end of the region if a geometrical factor larger than one is given and a higher concentration at the upper end of the region if the factor is smaller than one. You are prompted to specify the `VALUE OF R IN THE GEOMETRICAL SERIES` which determines the distribution.
   - `FUNCTION` for the global distance denoted ‘gd’. Useful functions include the error function (`erf(gd)`) and the heavy-side step function (`hs(gd)`). For example, the function `3+2hs(x-1e-4)` specifies a concentration of 3 at the region’s left side and 5 at the region’s right side, with a sharp step in the concentration profile at 1e-4m=100µm.

Adding (Creating) Cells

If you add cells to a system, then each new cell is always placed next to the system's right outer boundary. Use *CREATE_NEW_CELL* on page 9.

Diffusion Module (DICTRA) Simulations

These topics describe the typical order in which the system in the DICTRA module is set up to run a simulation.

Setting up a problem and running the simulation so that it successfully converges may require some fine-tuning and *Troubleshooting* on page 52.

In this section:

Tips for a Successful Simulation

Simplify your problem as much as possible

For example, if you have an alloy with five or six components, first run a simulation with only the main components. When this simulation works, you can then add more components as necessary.

Start with a low accuracy and refine it on a second run

If you start with a high accuracy setting, then computing the simulation takes much longer, and problems are more likely. Check the profiles and the overall mass balance, and if the accuracy is insufficient then you can improve the accuracy values. To change the accuracy, use *SET_ACCURACY* on page 33 and increase or decrease `MAX RELATIVE ERROR` and `MAX ABSOLUTE ERROR`.

Modify the grid or time step

If you get large fluctuations in the composition or phase boundary profiles, then you can try modifying the grid or the time step of the simulation. You can increase the time step by lowering the accuracy if automatic time step control is used (the default). If you increase the tolerated relative and/or absolute error with *SET_ACCURACY* on page 33, then the automatic time step control increases the time step.

Integrate the PDEs

The composition or phase boundary profiles can often also be improved if the partial differential equations (PDEs) are integrated fully implicitly. To integrate the PDEs in this way, use *SET_SIMULATION_CONDITION* on page 39 to set the parameter `DEGREE OF IMPLICITY WHEN INTEGRATING PDEs` to a value of `1`.

Debug

To receive more detailed information about what is happening during the simulation, you can use DEBUG (see *DEBUGGING* on page 9. If your simulations fails to converge, then this may give you some clues about the source of the failure.

Setting Up and Running a Simulation

See *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

1. Before you can set up a problem and start a simulation, you must have retrieved thermodynamic and kinetic data and defined your system. See *System Definition: Retrieving Data* on page 29.
2. Set conditions that hold globally in the system using SET_CONDITION `Global`. Normally, this is the temperature condition.

3. Enter a region into the current cell using ENTER_REGION. Give the region an arbitrary name. If the region that you create is not the cell's first region, then by default it is attached to the right side of the last region created in the cell.
4. For each region, use ENTER_GRID_COORDINATES to specify the region's size (width) as well as the type of grid and the number of grid points to be used in the region. The size is specified in meters.
5. Use ENTER_PHASE_IN_REGION to enter a phase into a region. Specify whether the phase is active/inactive, which type of phase it is, and what the name of the phase is.
6. Use ENTER_COMPOSITIONS to specify the initial composition of a phase that you have entered in a region. You can specify the composition in terms of site fraction, mole fraction, mole percent, weight fraction, weight percent or u-fraction. The composition profile may be entered by a user or be read from a file.
7. Use SET_CONDITION `Boundary` to set non-default boundary conditions.
8. Use ENTER_GEOMETRICAL_EXPONENT to enter the geometrical exponent that defines the geometry of the system. If you do not specify the geometrical exponent, then it is assumed that the system has a planar geometry.
9. Use SET_SIMULATION_TIME to set the simulation time.
10. If you want to use a specific model for the simulation calculations, then set the DICTRA module to use that model. See *Simulation Models* below.
11. You are now ready to run the simulation. However, it is recommended that you use SAVE_WORKSPACES to save the setup of your problem before running the simulation.
12. Use SIMULATE_REACTION to start the simulation. To ensure that the simulation is run with default settings (answer `yes`), otherwise you may be prompted to specify certain values during the running of the simulations (if phases appear or disappear in the simulation).

Simulation Models

Based on how the calculation is configured, the software uses different kinds of models to perform the simulation. For some models, you must explicitly set the DICTRA module to use them.

COARSENING_MODEL

You must explicitly set the DICTRA module to use this model with the command *COARSENING_MODEL* on page 8. The model allows you to perform coarsening simulations.

To use this mode, configure the calculation as follows:

- Have one cell only, with a single particle that is 1.5 times larger than the average particle size.

- The matrix phase should be in contact with the maximum size particle on the left/lower side, and then the model specifies a local equilibrium with an average sized particle on the right/upper side.
- The surface energy then results in growth of both the maximum size particle and the matrix phase. Use *SET_SURFACE_TENSION* on page 41 to enter the surface energy to enable coarsening.

PARA_EQUILIBRIUM_MODEL

You must explicitly set the DICTRA module to use this model with *PARA_EQUILIBRIUM_MODEL* on page 30. The model allows you to calculate a moving phase boundary problem while assuming that para-equilibrium conditions hold at the boundary.

You can only simulate a single moving boundary in one cell.

Substitutional components are, in effect, regarded as one composite component. The composition of the substitutional components in the phase that is to be dissolved in several ways. Enter AUTO to set it to the value at the region's lower or upper boundary; enter a value followed by a percentage sign (%) to set it to a certain percentage from the moving boundary inside the region; enter a hyphen (-) followed by a value to set it to a fixed distance from the moving boundary; or simply enter a value to set the composition to that specific value.

HOMOGENIZATION_MODEL

If you enter a spheroid phase when you configure your calculation, then the homogenization model is automatically used. This is useful when you want to simulate long-range diffusion through a multiphase mixture, under the assumption that local equilibrium holds at each node point.

You can also explicitly set the DICTRA module to use the model with *HOMOGENIZATION_MODEL* on page 25. This way, you can use the homogenization model also when calculating a moving boundary problem.

When the homogenization model is used, simulations are run using an implicit finite volume method to increase numerical stability. Because of this, and other factors, homogenization model simulations generally run slower than other comparable simulations. You can increase or decrease the `DEGREE OF IMPLICITY WHEN INTEGRATING PDEs` with *SET_SIMULATION_CONDITION* on page 39.

Disperse Model

This model can be used to calculate diffusion of a phase that is dispersed in a matrix phase. However, it is recommended that you use the *HOMOGENIZATION_MODEL* on page 25 for this type of calculation instead.

The disperse model has the following limitations:

- There must be a continuous matrix phase otherwise the simulation breaks down.
- There is no implicit integration of the partial differential equations (PDEs), leading to a lower accuracy.
- Only diffusion in the matrix phase can be simulated. It is assumed that no diffusion occurs in the dispersed phase.

There is no specific command to activate this model. It is only available if a phase of type "spheroid" is entered in a region. After a spheroid phase has been entered the homogenization model must explicitly be inactivated in order to run the simulation with the dispersed model. This is done by entering HOMOGENIZATION_MODEL NO.

Thermomigration Model

The thermomigration model allows you to simulate thermally induced diffusion in temperature gradients. To use the model, you must set the temperature to depend on the spatial coordinate (x) in your calculation. In addition, you must also enter the heat of transport quantity (Q*) using the ENTER_HEAT_TRANSFER_PARAMETER command. You must enter this parameter separately for each component in each phase.

The model is automatically used once this is done.

See example exh3 *Deviation from local equilibrium*.

Grain-boundary Model (GB_MODEL)

You must explicitly set the DICTRA module to use this model with *GB_MODEL* on page 24. This model allows you to simulate grain-boundary and dislocation-assisted diffusion.

The grain-boundaries and the dislocations contribute to the total amount of diffusion according to the weighted fractions. The parameters are entered separately for each region in order to allow for different expression for different phases.

Pearlite Model

This model is automatically used if there is lamellar phase in a region.

Importing a Scheil Segregation Profile

You can import a previously calculated Scheil segregation profile into the Diffusion Module (DICTRA). A Scheil calculation is a conservative estimate of the segregation that occurs during solidification and is therefore useful when estimating the necessary time/temperature for a homogenizing heat treatment.

From a practical point of view it is also a simple and fast calculation compared to a more accurate, but computationally more difficult, diffusion simulation.

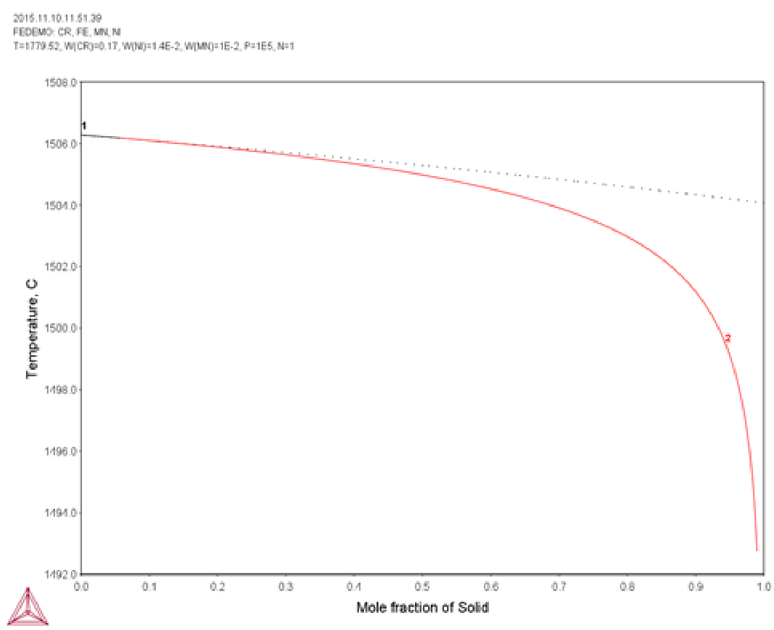
In order to limit the computational effort, this feature cannot be combined with the fast diffusers option in the Scheil module.

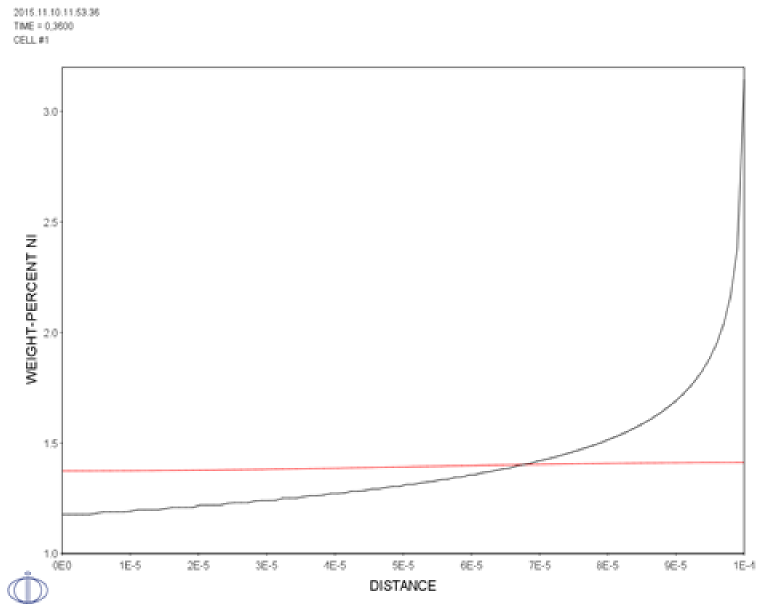
Also see *SCHEIL Commands* on page 235 in the *Thermo-Calc Console Mode Command Reference*, *Scheil Simulations* on page 144 in the *Thermo-Calc Console Mode Command Reference* and *INPUT_SCHEIL_PROFILE* on page 27 in the *Thermo-Calc Console Mode Command Reference*.

Example A7

In the following example, taken from *Diffusion Module (DICTRA) Examples Collection* on page 6 number a7, the Scheil segregation profile for a ferritic stainless steel (Fe-17Cr-1.4Ni-1Mn) was calculated in the Scheil module in Thermo-Calc.

The profile was then imported into the DICTRA module and a one hour homogenizing heat treatment at 1200 °C was simulated. For the simulation a domain size of 100 µm was assumed, which approximately corresponds to the secondary dendrite arm spacing. The plot below show the initial and final Ni profile.




2015.11.10.11:53:36
TIME = 0,3600
CELL #1
WEIGHT-PERCENT NI
DISTANCE
3.0
2.5
2.0
1.5
1.0
0E0
1E-5
2E-5
3E-5
4E-5
5E-5
6E-5
7E-5
8E-5
9E-5
1E-4

Plotting Simulations in the POST Module

You can plot or tabulate the results of your simulation in the POST_PROCESSOR module, also called the POST module and accessed within the DICTRA module. The resulting diagram is on a plot tab in the Console Results window. Once the simulation results are plotted, you can then change its appearance in the Plot Properties window.

In this section:

Plotting Simulation Results

Plotting a diagram is to graphically represent how a dependent variable varies with an independent variable. The independent variable is either distance or time.

See *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

How to Plot Results

1. Complete all the steps to run your simulation (see *System Definition: Retrieving Data* on page 29, *Configuring a Diffusion Module (DICTRA) Calculation* on page 31, and *Diffusion Module (DICTRA) Simulations* on page 40).

2. IAt the `DIC` prompt type `POST_PROCESSOR` (or just `POST`).

3. Use SET_DIAGRAM_AXIS to specify the variables to plot on the X- and Y-axes of the diagram. One of the axes must be set to either TIME or DISTANCE.

   For example, to set the X-axis to plot the distance from the lower boundary of the calculation (rather than from the lower boundary of the current region) and the Y-axis to plot the weight-percent composition of the Si element, at the POST prompt type:

   ```
   SET_DIAGRAM_AXIS X DISTANCE GLOBAL
   SET_DIAGRAM AXIS Y WEIGHT-PERCENT SI
   ```

   You can specify a Z-axis, which is represented as tick marks on the curve in the XY-plane.

   When one of the axes is set to `TIME` or `DISTANCE`, the variable automatically becomes the independent variable. If you do not want the independent variable plotted along an axis, use SET_INDEPENDENT_VARIABLE and set it to `TIME` or `DISTANCE`.

4. Simulation calculations depend on both time and a spatial coordinate; these variables must be fixed during the plotting process. Use SET_PLOT_CONDITION to set the variable to keep fixed. These variables can be set as the fixed plot condition:

   - `TIME`: The integration time.
   - `DISTANCE`: The distance in the system (from the lower boundary).
   - `INTERFACE`: At the location of a certain interface between two phases.
   - `INTEGRAL`: The value of the plot condition is integrated over the phase, region or system volume.

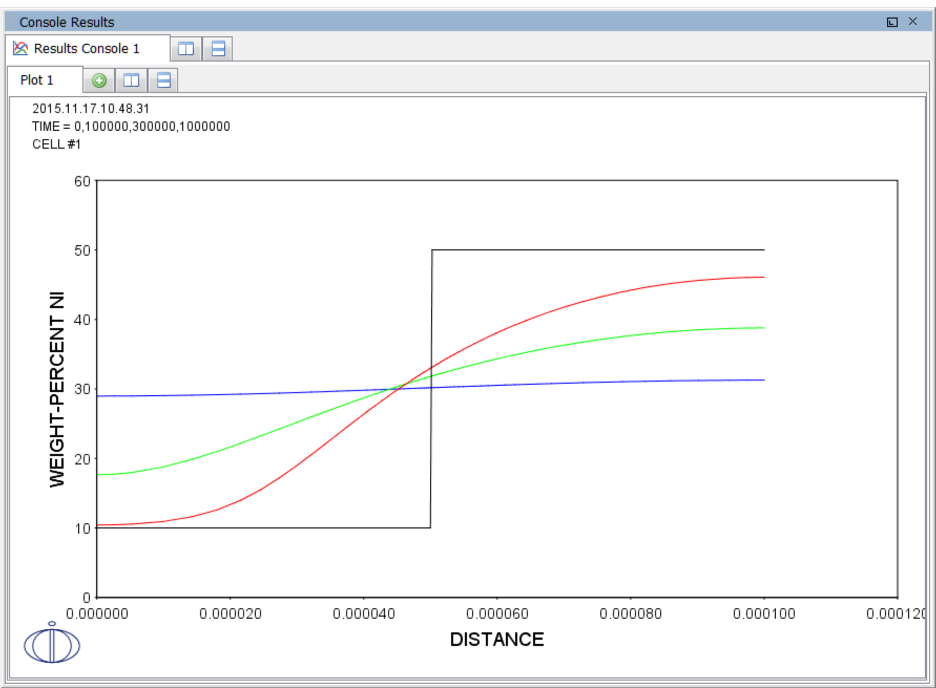
5. For `TIME`, more than one value can be entered using commas to separate the values. One

graph is plotted for each value.

6. Use PLOT_DIAGRAM to plot the diagram.
7. The plot is added as a tab to the **Console Results** window.

The plot example below is from example exa2a in the *Diffusion Module (DICTRA) Examples Collection* on page 6. Distance is set as the independent variable and the weight-percent of Ni is on the Y-axis. The plot condition is `TIME`: the plot shows four graphs, each with the spatial distribution of Ni in the region at the time steps 0, 1e5, 3e5, and 10e5.

8. To print the plotted graphs direct to a postscript-file, type `PLOT` and a filename.
9. In the **Plot** window, right click the diagram and choose **Properties** to adjust its appearance and colors.



Also see *Editing the Plot Properties* below and the *Thermo-Calc User Guide*.

Editing the Plot Properties

To change the appearance of a plotted diagram, right click the diagram and select **Properties**. In the **Plot Properties** window you can change the fonts and colours, and add a title, for example.

You can also change the colour, stroke (solid/dashed/dotted/dash_dot) and line width of a particular series of lines in the plot by double-clicking one of the lines in the series. In this way, you can also toggle whether data points should be shown or not for a series of lines. The crosshair cursor turns into a cursor resembling a pointing hand when it is placed over a line that can clicked but if you hold down Ctrl, this does not happen (the cursor continues to be shown as a crosshair).

To configure the default settings for plotting, from the main menu, select **Tools→Options**. For Console Mode click the **Plotting** tab.

See *Changing Global Settings* on page 187 in the *Thermo-Calc User Guide*.

Saving Diagrams

To save a plotted diagram you can do any of the following:

- Use MAKE_EXPERIMENTAL_DATAFILE to save all the data about the plotted diagram in an EXP-file. An EXP-file is a plain text file that describes the diagram in the DATAPLOT graphical language. Note that information about the underlying calculations that the diagram is based on is not saved in an EXP-file.
- Use DUMP_DIAGRAM to save the diagram image to a PS, PDF, SVG, EMF, GIF, or PNG file. You are asked which format to save it.
- In the **Console Results** window, right click a diagram and select **Save Plot**. From the list, choose the file format to save it in.
- Use PRINT_DIAGRAM to print a hard copy of the diagram.

See *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference* and the *DATAPLOT User Guide* included with this documentation set.

Tabulating Simulation Results

See *DICTRA Commands* on page 5 in the *Diffusion Module (DICTRA) Command Reference*.

1. Complete all the steps to run your simulation.
2. At the DIC prompt type `POST_PROCESSOR` (or just `POST`).
3. Use ENTER_SYMBOL `Table` to specify which variables that should be shown in the table columns. Separate the variables by commas (`,`).

4. Use SET_INDEPENDENT_VARIABLE to specify which variable that the rows in the table should represent.
5. Use TABULATE to generate the table.

Troubleshooting

The following topics contain some general advice to consider in order to avoid having your simulation fail to converge when you calculate different kinds of problems in the Diffusion Module (DICTRA).

In this section:

Moving Boundary Problems

Use the automatic starting values when you run the simulation the first time. If the first time step does not converge try decreasing the starting values for the velocities. Alternatively, increase the duration of the first time step.

If the simulation fails during the first time step, then check how the starting values affect the residuals. You can do this by setting the parameter `NS01A PRINT CONTROL` with SET_SIMULATION_CONDITION to `1`. The software then prints the tried unknown value and the residual for each flux balance equation in each iteration. The equations are organized so that the first unknown value is always the migration rate of the phase interfaces followed by the unknown potentials. All unknown values are scaled by the starting value or the calculated value in the previous time step.

Check the grid distribution at the interfaces. If the grid is too coarse or too fine-grained then the fluxes may not be defined well.

Use activities instead of potentials when the content of one or more of the components becomes very low. If you do this, then the activity approaches zero, which is easier to handle than a potential that approaches infinity. Use SET_SIMULATION_CONDITION to set the parameter `VARY POTENTIALS OR ACTIVITIES` to `ACTIVITIES`.

Cell Calculations

In cell calculations, determine start values for each cell separately and thereafter connect the individual cells. You can even start out simulating in planar and/or cylindrical geometry in order to determine sufficient starting values if you want to use a spherical geometry.

Diffusion in Dispersed Systems

If you simulate diffusion in a dispersed system and the homogenization model is turned off, then time integration is explicit. The maximum time step allowed, or the highest time step actually taken in the calculation, affects the final result of the simulation. This effect is most evident if one or more of the diffusing species have low solubility in the matrix phase. When this is the case, a supersaturation is created in the matrix phase during a diffusion step and if too large a time step is allowed, then too much supersaturation is created. Consequently, always check your results by performing a second simulation with a maximum time step that is ten times lower than in the first run. If there is large difference in results between the first and the second run, then decrease the time step again with a factor of ten and perform a third run. Repeat this process until the results can be trusted.

If a simulation fails because the program is not able to calculate an equilibrium, then try to calculate the initial equilibrium yourself using Thermo-Calc and enter the composition and fraction of each phase separately instead of having the DICTRA module calculate it. Also enter starting values for each phase with the command SET_ALL_START_VALUES. Alternatively, you may force the software to use certain start values using SET_SIMULATION_CONDITION.

Cooperative Growth

If your simulation fails to converge, try calculating the A1e temperature with Thermo-Calc and start your simulation at a temperature slightly below that temperature. Use the equations given in the interactive help in order to estimate S0 and the growth rate.

Diffusion Module (DICTRA) Command Reference

Version 2016a

Introduction to the Diffusion Module (DICTRA) Commands

In this section:

General Commands

These commands are universal in Console Mode. For convenience they are repeated here and described in the *Thermo-Calc Console Mode Command Reference*.

Command/Syntax	Description and Prompts
BACK ON PAGE 4	Return to the previous module. From the POST-processor you go back to the DICTRA-MONITOR module.
EXIT ON PAGE 4	Terminates the program and returns to the operating system. Unless a SAVE_ WORKSPACES command has been given before all data entered is lost.
GOTO_MODULE ON PAGE 4	`MODULE NAME` Select and go to another module. The name of the module must be given. In order to obtain a list of available modules give a return.
HELP ON PAGE 5 ?	Enter HELP or ? to either a list of all commands or specific help for a command by giving the command (abbreviated) `COMMAND` The command for which a description is wanted should be given. If the abbreviation is not unique a list of all matching commands are given.

DICTRA Commands

In this section:

AMEND_CELL_DISTRIBUTION

Use this to amend the cell distribution factor of the currently selected cell.

Syntax	AMEND_CELL_DISTRIBUTION
Prompt	`CELL DISTRIBUTION FACTOR /4.446590813-323/` A parameter that determines the weight of the cell, it can be used to simulate a distribution of different cell sizes.

AMEND_MOBILITY_DATA

Use this command to change an existing parameter.

Also see *ENTER_MOBILITY_DATA* on page 16 for information about the parameter names and prompts.

Syntax	AMEND_MOBILITY_DATA
Prompt	`PARAMETER NAME <IDENTIFIER> (<PHASE>, <COMPONENT ARRAY>; <DIGIT>)` Enter ? to get an overview

CHECK_DIFFUSION_MATRIX

Display the diffusion coefficient matrix for a phase at a given composition, pressure and temperature.

Syntax	CHECK_DIFFUSION_MATRIX
Prompts	`OUTPUT TO SCREEN OR FILE /SCREEN/`
	`PHASE NAME` Name of the phase for which the diffusion coefficient matrix is to be displayed.
	`CONCENTRATION OF` Concentration of the component in U-fraction for which the diffusion coefficient matrix is to be displayed.
	`PRESSURE` Pressure for which the diffusion coefficient matrix is to be displayed.
	`TEMPERATURE` Temperature at which diffusion coefficient matrix is to be displayed.

Syntax	CHECK_DIFFUSION_MATRIX
	Use one or several of DLPBMX0E: • `D`: reduced diffusion matrix • `L:L` matrix (diagonal) • `P:L'` matrix • `B:L"` matrix • `M:MU(k)` array • `X:dMU(k)/dCj` matrix • `0`: unreduced diffusion matrix • `E`: Eigen values of matrix

COARSENING_MODEL

Enable or disable the use of the simplified model for calculating Ostwald-ripening in multicomponent systems. This must be used together with the SET_SURFACE_TENSION command.

The model is based on LSW theory (after Lifshitz and Slyozov *Chem Solids* 19(1961) p. 35 and Wagner Z *Elektrochemie* 65 (1961) p. 581). The model is also described in Gustafson et al *proc Adv heat resistant steel for power gen Spain* (1998) p. 270 and Björklund et al *Acta Met* 20 (1972) p. 867.

LSW theory is strictly only valid for low volume fractions of the particle phase. The theory predicts that the normalized particle size distribution obtains a constant shape where the largest particles have a radius equal to 1.5 times the average particle radius.

The calculations are performed in one cell on a maximum size particle which thus is assumed to be 1.5 times the size of the average particle size. The matrix phase is on one side in contact with the maximum size particle and on the other the matrix phase is in local equilibrium with an average sized particle. The effect of the surface energy is such that both the maximum size particle as well as the matrix phase grows.

Syntax	COARSENING_MODEL
Prompt	`ENABLE COARSENING MODEL /N/` The default is `N`. Enter `Y` to enable the use of this model. For another example of coarsening, see *Property Model Calculator* on page 60 in the *Thermo-Calc User Guide*.

CREATE_NEW_CELL

Create a new cell and attach it to the list of existing cells.

Syntax	CREATE_NEW_CELL
Prompt	CELL DISTRIBUTION FACTOR A parameter that determines the weight of the cell, this parameter can be used in order to simulate a distribution of different cell sizes.

DEBUGGING

Determines the debugging level. Depending on the value given, different amounts of information are written onto the output device. This can be used to determine what causes the program to crash during a simulation.

Syntax	DEBUGGING
Prompt	DEBUG LEVEL (YES,NO,0,1,2,3,4) /NO/ The level of debugging information to be displayed.

DELETE_REGION

Delete a region and all its associated data from the current cell.

Syntax	DELETE_REGION
Prompt	REGION NAME The region name to delete.

ENTER_COMPOSITIONS

Enter the composition into the phases in a region.

Syntax	ENTER_COMPOSITIONS
Prompts	REGION NAME Name of the region into which the compositions are to be entered.
	PHASE NAME Name of the phase in a region into which the compositions are to be entered.
	USE EQUILIBRIUM VALUE This is only for spheroid phases in simulations with dispersed phases. If this option is used

Syntax	ENTER_COMPOSITIONS
	the program automatically calculates the equilibrium fractions of the spheroid phase and its constitution at the start of the simulation.
	`DEPENDENT SUBSTITUTIONAL SPECIES` A dependent substitutional species is required in order to be able to determine which species are independent. The program only queries for the compositions of the independent species. Sometimes the dependent species is chosen by the program and thus this question is never given. This may be due to stoichiometric constraints or to the fact that it has been set already in the kinetics database due to the model selected for the diffusion.
	`DEPENDENT INTERSTITIAL SPECIES` A dependent interstitial species is required in order to be able to determine which species are independent. The program only queries for the compositions of the independent species. Vacancies are always regarded as dependent and therefore if vacancies are present in the phase then this question is never given.
	`COMPOSITION TYPE` Type of composition used for the constitution of the phase. Options are: SITE_FRACTION, MOLE_FRACTION, MOLE_PERCENT, WEIGHT_FRACTION, WEIGHT_PERCENT and U_ FRACTION.
	`TYPE` Type of composition profile to be entered, options are: LINEAR, READ_POINT_BY_POINT, FUNCTION, and GEOMETRIC. If FUNCTION is chosen the composition profile can be given as a function of the global distance denoted X. Some useful functions are the error function denoted `erf(X)` and the Heaviside step-function denoted `hs(X)`. For example the function `3+2hs(x-1e-4)` provides a concentration of 3 at the left side and 5 at the right side with a sharp step in the concentration profile at 1e-4m=100µm.
	`VALUE OF FIRST POINT` Composition in the first gridpoint. Values in between are interpolated linearly if a LINEAR type of profile is specified.

Syntax	ENTER_COMPOSITIONS
	VALUE OF LAST POINT Composition for the last gridpoint. Values in between are interpolated linearly if a LINEAR type of profile is specified.
	INPUT FILE The input source from which to read the points when entering values point by point, default is TERMINAL.
	VALUE OF POINT The value of the point when entering values point by point.
	VALUE OF R IN THE GEOMETRICAL SERIE Use the same geometrical factor as for the geometrical grid.
	VOLUME FRACTION OF Initial volume fraction of a spheroid phase.

ENTER_ENHANCEMENT_FACTOR

Change the mobility of a specific element in a specific phase. The mobility of the element in the phase is multiplied by a factor which is specified as an argument to the command.

Syntax	ENTER_ENHANCEMENT_FACTOR
Prompt	MOBILITY ENHANCEMENT FACTOR FOR PHASE

ENTER_GEOMETRICAL_EXPONENT

Enter the geometrical exponent which defines the geometry of the system. The program handles one-dimensional geometries defined by the geometrical exponent. These geometries are:

- Planar. This corresponds to an infinitely wide plate of a certain thickness.
- Cylindrical. This corresponds to an infinitely long cylinder of a certain radius.
- Spherical. Sphere with a certain radius.

Syntax	ENTER_GEOMETRICAL_EXPONENT
Prompt	GEOMETRICAL EXPONENT Enter an integer value between 0 and 2.

ENTER_GRID_COORDINATES

Enter the size and gridpoint distribution of the grid in each region separately. The size of the region is specified in the units that the diffusion data is entered in. The grid in a specific region may also be subdivided into several parts, where the type of each part is chosen independently, by first entering SUB_REGION as the overall grid type.

Syntax	ENTER_GRID_COORDINATES
Prompts	`REGION NAME` Name of the region into which a grid is to be entered.
	`WIDTH OF REGION` The actual size of the region is entered. The size of the region is specified in units compatible with those of the diffusion data.
Options	**Description**
`TYPE` of grid	`TYPE` Type of grid to entered, options are: • `LINEAR`: For an equally spaced grid. • `READ_POINT_BY_POINT`: Can be done either from the keyboard or from a predefined file. • `GEOMETRIC`: Yields a higher number of gridpoints at the lower end of the region if a geometrical factor larger than one is given and a higher number of gridpoints at the upper end of the region if the factor is smaller than one. • `DOUBLE_GEOMETRIC`: Divides the region in two halves and generates a separate geometrical grid in each half. It gives a high number of gridpoints in the middle or at both ends of a region, two geometrical factors should be entered.
`SUB_REGION`	If `SUB_REGION` is entered when you are prompted to enter the grid type, then the next prompt is `END COORDINATE IN SUB REGION` Enter a value equal to or less than the total width of the region. You are then prompted for the number of grid points and the grid type to be used between the left-hand side of the region and the end coordinate. As long as the end coordinate is less than the total

Syntax	ENTER_GRID_COORDINATES
	width of the region you are prompted for an end coordinate and the grid to be used in the sub region starting at the end of the former sub region. Thus, progressively higher end coordinates must be entered and the final end coordinate must be equal to the total width of the region.

	`NUMBER OF POINTS`
	The number of points present in the region. Please consider the interspacing of the grid when determining the number of points.
	`INPUT FILE`
	The input source from which to read the points when entering values point by point, default is TERMINAL.
	`VALUE OF POINT`
	The value of the point when entering values point by point.
	`VALUE OF R IN THE GEOMETRICAL SERIES`
	Value in the geometrical factor in the series determining the distribution of the grid points. A geometrical factor larger than one yields a higher density of gridpoints at the lower end of the region and a factor is smaller than one yields a higher density of gridpoints at the upper end of the region.
	`VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION`
	The geometrical factor in the series for the lower (left) part of a region in a double geometrical grid
	`VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION`
	The geometrical factor in the series for the upper (right) part of a region in a double geometrical grid.

ENTER_HEAT_TRANSFER_PARAMETER

A model for thermos-migration simulates the thermal induced diffusion in temperature gradients. This model requires that the temperature depends on the length coordinate in the system and that the quantity `heat of transport` (`Q*`), is entered.

Syntax	ENTER_HEAT_TRANSFER_PARAMETER
Prompt	HEAT TRANSFER PARAMETER FOR PHASE The phase for which the parameter is to be entered. This parameter is entered separately for each component in each phase.
	The implementation and functionality of this model is described in Höglund, L. & Ågren, J. "Simulation of Carbon Diffusion in Steel Driven by a Temperature Gradient". *J. Phase Equilibria Diffus.* 31, 212–215 (2010).

ENTER_HOMOGENIZATION_FUN

Use this with the homogenization model for multiphase simulations. The homogenization model is based on the assumption of local equilibrium at each node point, which yields the local chemical potentials at each node point from which the local chemical potential gradients may be estimated. The chemical potential gradients are the driving forces for diffusion. The local kinetics must also be evaluated by some averaging procedure, the choice of which is determined by this command. The local kinetics is evaluated by considering the product of mobility times u-fraction for each component in each phase and the volume fraction of each phase.

About the Homogenization Functions

The geometrical interpretation of the Hashin-Shtrikman bounds are concentric spherical shells of each phase. For the general lower Hashin-Shtrikman bound the outermost shell consists of the phase with the most sluggish kinetics and vice versa for the general upper bound. The geometrical interpretation of the Hashin-Shtrikman bounds suggest further varieties of the bounds, viz. #3 and #4, where the outermost shell consist of a prescribed phase or the phase with highest local volume fraction, respectively.

The geometrical interpretation of the Wiener bounds are continuous layers of each phase either parallell with (upper bound) or orthogonal to (lower bound) the direction of diffusion.

The labyrinth factor functions implies that all diffusion takes place in a single continuous matrix phase. The impeding effect on diffusion by phases dispersed in the matrix phase is taken into account by multiplying the flux with either the volume fraction (#7), or the volume fraction squared (#8), of the matrix phase.

The varieties with excluded phases are useful in several respects. First, if a phase is modelled as having zero solubility for a component, the mobility of that component in that phase is undefined, which causes a (non-terminal) error. Setting a phase as excluded causes the mobility of all components in that phase to be set to zero. Second, often there are some major matrix solid solution phases and some minor precipitate phases. If the mobilities in the minor precipitate phases are zero the lower Hashin-Shtrikman bound is useless as it produces a kinetic coefficient of zero. However, using homogenization function #9 the excluded phases are not considered when evaluating what phase has the most sluggish kinetics.

Syntax	ENTER_HOMOGENIZATION_FUN
Prompt	`ENTER HOMOGENIZATION FUNCTION # /5/` Enter a digit between 1 and 14 (default is 5). The options corresponding to the numbers are listed below.

Homogenization Functions

Enter a digit between 1 and 14 (default is #5) to assign the homogenization function then follow the prompts. The homogenization functions are:

No.	Function name
1	General lower Hashin-Shtrikman bound*
2	General upper Hashin-Shtrikman bound*
3	Hashin-Shtrikman bound with prescribed matrix phase*
4	Hashin-Shtrikman bound with majority phase as matrix phase*
5	Rule of mixtures (upper Wiener bound)
6	Inverse rule of mixtures (lower Wiener bound)

No.	Function name
7	Labyrinth factor f with prescribed matrix phase
8	Labyrinth factor f**2 with prescribed matrix phase
9	General lower Hashin-Shtrikman bound with excluded phase(s) *
10	General upper Hashin-Shtrikman bound with excluded phase(s) *
11	Hashin-Shtrikman bound with prescribed matrix phase with excluded phase(s) *
12	Hashin-Shtrikman bound with majority phase as matrix phase with excluded phase(s) *
13	Rule of mixtures (upper Wiener bound) with excluded phase(s)
14	Inverse rule of mixtures (lower Wiener bound) with excluded phase(s)
	* For the Hashin-Shtrikman bounds, see Hashin, Z. & Shtrikman, S. "A Variational Approach to the Theory of the Effective Magnetic Permeability of Multiphase Materials". *J. Appl. Phys.* 33, 3125–3131 (1962).

ENTER_LABYRINTH_FUNCTION

Enters a constant value or a function of temperature, pressure, or the volume fraction of the phase where diffusion occurs. This function increase or reduces the diffusion coefficient matrix. This function is primarily used when spheroid phases have been entered into a region. It may also be used for increasing or decreasing all diffusion coefficients in a certain matrix phase by a constant factor.

Syntax	ENTER_LABYRINTH_FUNCTION
Prompt	REGION F(T,P,VOLFR,X)=

ENTER_MOBILITY_DATA

Use this command to enter previously non-existing mobility parameters. If a function is already defined, it is deleted.

You can also use:

- *ENTER_ENHANCEMENT_FACTOR* on page 11 to enter a factor by which element mobilities are multiplied with.
- *ENTER_MOBILITY_ESTIMATE* on page 18 to enter a rough mobility estimate for elements in specific phases. This can be useful for phases where there is no data in the database.

A valid parameter should have the general form of:

`<IDENTIFIER> ( <PHASE NAME> , <COMPONENT ARRAY> ; <DIGIT> )`

The `identifier` must be followed by an opening parenthesis, a phase name, a comma and a component array. Optionally, the component array can be followed by a semicolon and a digit. The parameter name is terminated by a closing parenthesis.

These identifiers are legal:

- `MQ` (activation energy for mobility)
- `MF` (frequency factor for mobility)
- `DQ` (activation energy for diffusivity)
- `DF` (frequency factor for diffusivity)
- `OQ` (activation energy for mobility, ordered part)
- `OF` (frequency factor for mobility, ordered part).

The `phase name` must be followed by an & and directly following that the name of the diffusing specie.

The `component array` consists of a list of constituent names. Interaction parameters have two or more constituents separated by a comma. If the phase has sublattices at least one constituent in each sublattice must be specified. The constituents in different sublattices must be given in sublattice order and are separated by a colon.

After the component array a subindex `digit` can be specified after a semicolon. This digit must be in the range 0 to 9. The interpretation of the subindex depends on the excess model used for the phase. If no semicolon and digit is given the subindex value is assumed to be zero.

<table>
<tr><td>Syntax</td><td>ENTER_MOBILITY_DATA</td></tr>
<tr><td>Prompt</td><td>

`PARAMETER: <PARAMETER NAME>`

As explained above, specify a correct and complete parameter name, which should contain all the necessary parts of the general form:

`<IDENTIFIER>(<PHASE>,<COMPONENT ARRAY>;<DIGIT>)`

If a parameter name is not acceptable or <Enter>, is pressed, the error message displays:

`*** Error, please re-enter each part separately`

Examples

- `MQ(FCC&C,Fe:Va)` Mobility of C in fcc Fe with interstitials.
- `MQ(FCC&C,Fe,Cr;0)` The regular parameter for Fe and Cr in fcc.
- `MQ(FCC&C,Fe,Cr;1)` The subregular interaction parameter (Redlish-Kister model).

</td></tr>
<tr><td>Identifier</td><td>

`IDENTIFIER /MQ/ <MQ, MF, DQ, DF, OQ, OF>`

If this command is used one or more times, the previous value on this prompt is set as default. Press <Enter> for the same type identifier or specify a new type.

</td></tr>
<tr><td>Phase Name</td><td>

`PHASE NAME /FCC&C/`

Each parameter is valid for a specific phase. The name of that phase must be supplied. The name must not be abbreviated. The phase name must be followed by an '&' and directly following that, the name of the diffusion specie.

</td></tr>
</table>

ENTER_MOBILITY_ESTIMATE

Use this command to enter a function yielding a mobility estimate for a specific element in a specific phase.

Also see *LIST_MOBILITY_ESTIMATES* on page 28.

Syntax	ENTER_MOBILITY_ESTIMATE
Prompt	`MOBILITY ESTIMATE FOR PHASE` Interactively enter a function yielding a mobility estimate for a specific element in a specific phase. This can be used to enter mobility estimates in phases for which there is no assessed data. Estimates entered here override database values.

ENTER_PHASE_IN_REGION

Enter a phase into an earlier defined region.

Syntax	ENTER_PHASE_IN_REGION
Prompts	`ACTIVE OR INACTIVE PHASE` Type of phase entered. An inactive phase is a phase which does not participate in the calculations until it is stable. This is done by regarding the driving force for precipitation of the phase in an equilibrium calculation. The program then automatically retransforms the inactive phase into an active one.
	`REGION NAME` Name of the region into which the phase is to be entered.
	`PHASE TYPE` Type of phase entered. Legal types are: • MATRIX • LAMELLAR (see *ENTER_PHASE_IN_REGION - Lamellar Prompts* on the next page). • SPHEROID: requires that a MATRIX phase has been previously entered.
	`PHASE NAME` Name of the phase that is to be entered. A #-sign and a digit may append the phase name in order to specify the composition set number, e.g. FCC#2.
	`COMPOSITION SET` Number of the composition set. This is needed if the phase has been amended to have more than one composition set and the number was not given directly on the phase name, see above.
	`ATTACH TO REGION NAMED` Enter the name of the region onto which the inactive phase is to be attached.
	`ATTACHED TO THE RIGHT OF` Enter `Y` to attach the inactive phase on the right side of the region else, enter `N` to attach on the left side.
	`REQUIRED DRIVING FORCE FOR PRECIPITATION` The required driving force (evaluated as DGM (phase) in POLY-3) to be used for

Syntax	ENTER_PHASE_IN_REGION
	determining whether an inactive phase is stable.
	CONDITION TYPE Boundary condition used if the inactive phase becomes stable.

ENTER_PHASE_IN_REGION - Lamellar Prompts

The following are the *ENTER_PHASE_IN_REGION* on the previous page subprompts for Lamellar.

For a LAMELLAR type of phase, the pearlite calculation model is invoked. The following is displayed:

```
eutectoid reaction is "gamma" ==> "alpha" + "beta"
```

This is to clarify what is meant with GAMMA, ALPHA and BETA in the specific sub-prompts that display as follows.

Subprompts	LAMELLAR
	ENTER NAME OF "ALPHA" PHASE Enter the GES phase name for the ALPHA phase of the eutectic/eutectoid decomposition product.
	ENTER NAME OF "BETA" PHASE Enter the GES phase name for the BETA phase of the eutectic/eutectoid decomposition product.
	ENTER NAME OF "GAMMA" PHASE Enter the GES phase name for the GAMMA matrix phase.
	ENTER "ALPHA"/"BETA" SURFACE TENSION: Enter function for the surface tension between the ALPHA and BETA phases, $\sigma\alpha/\beta$
	ENTER "ALPHA"/"GAMMA" SURFACE TENSION: Enter function for the surface tension between the ALPHA and GAMMA phases, $\sigma\alpha/\gamma$
	ENTER "BETA"/"GAMMA" SURFACE TENSION: Enter function for the surface tension between the BETA and GAMMA phases, $\sigma\beta/\gamma$
	OPTIMUM GROWTH CONDITION FACTOR /2/: Enter the Optimum-growth-rate-factor. Due to Zener's maximum growth rate criteria this factor has a value of 2 for volume controlled growth and 3/2 for boundary

Subprompts	LAMELLAR
	controlled growth. Due to Kirkaldy's extreme in entropy production criteria the values are 3 and 2, respectively.
	`NAME OF DEPENDENT ELEMENT` Enter the name of the substitutional element to consider as the dependent one.
	`GROWTH MODEL (VOLUME/BOUNDARY/KIRKALDY) FOR ELEMENT X` Select growth model to be used for element X. Choose between: • `Volume` diffusion model • `Boundary` diffusion model • `Kirkaldy`'s mixed mode diffusion model. Implies MIXED, see below.

	`DF(X) = /VALUE/AUTOMATIC/MIXED/TDB/`
	Either input a numerical value on the pre-exponential factor DF or select one of the keywords:
	• AUTOMATIC
	• MIXED
	• TDB
	AUTOMATIC is only available for element Carbon. It implies a mixed type of calculation where the volume diffusion part is calculated due to J. Ågren "A revised expression for the diffusivity of carbon in binary Fe-C austenite". *Scr. Metall.* 20, 1507–1510 (1986) (volume diffusion of C in Austenite) and the boundary diffusion part due to J. Ågren "Computer simulations of the austenite/ferrite diffusional transformations in low alloyed steels" *Acta Metall.* 30, 841–851 (1982) (boundary diffusion of C is assumed to be the same as C diffusion in Ferrite).
	The k' or k'', as appropriate, are given by B. Jönsson "On the Lamellar Growth of Eutectics and Eutectoids in Multicomponent Systems" *Trita-Mac R. Inst. Technol. Stock.* 478, 27 pages (1992). MIXED means a mixed mode calculation using an effective diffusion coefficient. Coefficient k' or k'' is asked for, see below. TDB means calculate the diffusion coefficient for volume diffusion for element X from the parameters stored in the database.
	`DQ(X)=`
	Input a numerical value on the activation energy DQ for element X.
	`K'=`

Subprompts	LAMELLAR
	`K''=` Input a numerical value on the k coefficient used to calculate the effective diffusion coefficient, use in MIXED mixed mode calculations, see B. Jönsson: *Trita-Mac* 478, 1992.
	`DF_BOUNDARY(X)=` Input a numerical value on DF for boundary diffusion of element X in a mixed mode calculation.
	`DQ_BOUNDARY(X)=` Input a numerical value on DQ for boundary diffusion of element X in a mixed mode calculation.
	`DF_VOLUME(X)=` Input a numerical value on DF for volume diffusion of element X in a mixed mode calculation. N.B. key word TDB may also be used, see TDB above.
	`DQ_VOLUME(X)=` Input a numerical value on DQ for volume diffusion of element X in a mixed mode calculation.
	`AUTOMATIC START VALUES FOR THE S0 DETERMINATION /Y/:` Enter `Y` if you want automatic start values for the unknown parameters in the S0 determination else enter `N`. S0 is the critical lamellar spacing for which the growth rate is zero.
	`CRITICAL THICKNESS OF "ALPHA" LAMELLA:` If you answered `N` this prompt displays. The critical thickness of ALPHA is about 0.9 of S0, which in turn is about 1/3 to 1/2 of the observed lamellar spacing S. For binary Fe-C alloys the observed pearlite lamellar spacing is approximately given by $S = 1.75 \cdot 10^{-5}(A_{1e} - T)^{-1}$ The equation may be used as a start value approx. for alloyed steels. However, use the A_{1e} temperature of the steel.
	`CRITICAL THICKNESS OF "BETA" LAMELLA:` The critical thickness of BETA is about 0.1 of S0, see above.
	`AUTOMATIC START VALUES ON POTENTIALS /Y/:` Enter `Y` if you want automatic start values for the unknown potentials.

Subprompts	LAMELLAR
	`GIVE POTENTIALS FOR "ALPHA"/"GAMMA" EQUIL.` If you answered `N` this prompt displays. Enter start values for the unknown potentials, MU, at the ALPHA/GAMMA phase boundary.
	`GIVE POTENTIALS FOR "BETA"/"GAMMA" EQUIL.` Enter start values for the unknown potentials, MU, at the BETA/GAMMA phase boundary.
	`GROWTH RATE V:` Enter a start guess on the growth rate. As a hint on what value to choose we recognize that for binary Fe-C alloys the pearlite growth rate is approximately given by: $S = 8 \cdot 10^{-9} (A_{1e} - T)^2$ The equation may be used as a start value approx. for alloyed steels. However, use the A_{1e} of the steel.
	`AUTOMATIC START VALUES ON OTHER VARIABLES /Y/:` Enter `Y` if you want automatic start values for the unknowns in the determination of the growth rate.
	`FRACTION OF "ALPHA" PHASE:` Enter a guess on the fraction of the ALPHA phase. For pearlite it is about 0.9.
	`GIVE POTENTIALS FOR "ALPHA"/"GAMMA" EQUIL.` Enter start values for the unknown potentials, MU, at the ALPHA/GAMMA phase boundary.
	`GIVE POTENTIALS FOR "BETA"/"GAMMA" EQUIL.` Enter start values for the unknown potentials, MU, at the BETA/GAMMA phase boundary.

ENTER_REGION

Enter a region into the system. Enter this before entering a grid or any phases.

Syntax	ENTER_REGION
Prompts	REGION NAME Name of a region to be entered. The name of the region can be arbitrarily chosen.
	ATTACH TO REGION NAMED Name of a region to which the new region should be attached.
	ATTACHED TO THE RIGHT OF Relative position of the new region. To attach the new region to the right of the named region answer YES, to attach to the left answer NO.

GB_MODEL

Grain-boundary and dislocation assisted diffusion is implemented by assuming that these contributed to the diffusion by using the same frequency factor and a modified bulk activation energy. The grain-boundaries and the dislocations contribute to the total amount of diffusion according to the weighted fractions.

Syntax	GB_MODEL
	The parameters are entered separately for each region in order to different expression for different phases.
	The used expressions for the grain-boundary and dislocation contributions are: $M^{gb}=M_0^{bulk}\cdot\exp(F_{redGB}\cdot Q^{bulk}/R/T)$ $M^{disl}=M_0^{bulk}\cdot\exp(F_{redDisl}\cdot Q^{bulk}/R/T)$ where • M_0^{bulk}: frequency-factor in the bulk • Q^{bulk}: activation energy in the bulk • F_{redGB}: Bulk diffusion activation energy multiplier (typical value 0.5) • $F_{redDisl}$: the bulk diffusion activation energy multiplier (typical value 0.8) The weighted calculated value for the mobility M^{new} is then evaluated from: $M^{new}=\delta/d\cdot M^{gb}+\rho\cdot b^2\cdot M^{disl}+(1-\delta/d-\rho\cdot b^2)\cdot M^{bulk}$ where • δ: the grainboundary thickness (typical value $0.5\cdot 10^{10}$) • d: the grainsize as a function of time and temperature (typical value $25\cdot 10^{-6}$)

Syntax	GB_MODEL
	• ρ:the dislocation density as a function of time and temperature • b:burgersvector (typical value $1{\cdot}10^{-10}$) • M^{bulk}: the mobility in the bulk (δ/d being the fraction of grain boundaries in the bulk and $\rho{\cdot}b^2$ being the fraction of dislocations in the bulk)

HOMOGENIZATION_MODEL

Enable or disable the use of the homogenization model and its default settings. These commands are used to enable the homogenization model. The homogenization model is used for multiphase simulations assuming that local equilibrium holds at each node point. When entering phases into a region one of them is entered as MATRIX phase and all other phases as SPHEROID, but it does not matter which one is entered as matrix phase and it does not affect simulations. Homogenization model simulations differs from all other DICTRA module simulations in that it is implemented using an implicit finite volume method in order to increase numerical stability (degree of implicity can be chosen using the *SET_SIMULATION_CONDITION* on page 39 command). Because of this, and other factors, homogenization model simulations generally run slower than other comparable simulations.

Syntax	HOMOGENIZATION_MODEL
Prompts	`ENABLE HOMOGENIZATION` `Y` to enable the homogenization model.
	`USE DEFAULT SETTINGS` `Y` to use default settings for the homogenization model. For `N`you are prompted as follows.
	`ADD IDEAL FLUX CONTRIBUTION` Enter `Y` or `N`. For `Y`, enter a fractional ideal flux contribution between zero and one. In multiphase regions the system loses degrees of freedom which may cause fluctuations in the composition profiles. This can be amended by adding a small ideal contribution to the fluxes. The ideal flux contribution should normally not be used.
	`USE INTERPOLATION SCHEME` Enter `Y` or `N`. The interpolation scheme may speed up simulations significantly. For `Y` you are prompted with the following: • Enter the number of steps in composition space. In the limit where an infinite number of steps are used, exactly the same solution is obtained as without the interpolation scheme. However, excellent results can be obtained with a reasonable discretization.

Syntax	HOMOGENIZATION_MODEL
	• The discretization can be either linear or logarithmic. For the linear discretization the scheme is not used at node points where the content of one or more solutes fall below a certain critical value. For such cases, where composition span many orders of magnitude, the logarithmic discretization can be tried. • Enter the fraction of free physical memory to be used by the interpolation scheme.
	`USE GLOBAL MINIMIZATION` Enter `Y` or `N` whether global minimization should be used in equilibrium calculations. In general, using global minimization significantly increases the CPU time for a given simulation, but there is also a significantly reduced risk for non-converged equilibrium calculations.
	`REFRESH JACOBIAN EVERY ITERATION` Entering `Y` increases computational demand for each iteration, but may in some cases improve convergence rate.
	`DEFAULT GRID PARAMETER VALUES` These settings only affect moving phase boundary simulations where the grid changes during the simulation. If you enter No, these you are prompted with the following: • `Geometrical coefficient`. The geometrical coefficient used in each region. • `Fixed interface width`. A value larger than zero makes interface widths fixed to that value. • `Grid fineness away from interface`. A value other than one causes the grid away from the interface to be coarser (>1) or finer (<1) than what would be obtained just by the geometrical coefficient. • `Interface width fraction`. If the interface width isn't fixed the program aims for a width equal to this factor times the cell width. • `Consecutive critical time-steps to delete region`. If the width of a region falls below a certain critical value and shrinks monotonically for this number of time-steps it is deleted.

INPUT_SCHEIL_PROFILE

As per normal procedure, and before entering the DICTRA monitor, you need to read thermodynamic and kinetic data to use this command.

This command takes a previously calculated Scheil segregation profile and performs most of the setup needed to use that profile in a simulation, for example a homogenizing heat treatment.

The command creates a region called SCHEIL_REGION. You then create a linear grid and enter the composition read from the file containing the segregation profile.

After issuing this command, you specify simulation time (*SET_SIMULATION_TIME* on page 40) and simulation temperature (*SET_CONDITION* on page 34, enter `GLOBAL` and variable `T`) to finalize setup.

In order to limit the computational effort, this feature cannot be combined with the fast diffusers option in the Scheil module.

Also search the online help and refer to the *Diffusion Module (DICTRA) User Guide*, *Thermo-Calc User Guide* and *Thermo-Calc Console Mode Command Reference* for detailed information about Scheil simulations.

Syntax	INPUT_SCHEIL_PROFILE
Prompts	`ENTER FILE LOCATION OF SCHEIL SEGREGATION PROFILE` In interactive mode (the command line), a file dialogue window opens. Navigate to the file containing the Scheil profile. Alternately, if you are working from a macro file, the line should contain the name of the file.
	`ENTER WIDTH OF REGION` Enter the width of the region [m] that contains the Scheil profile. This typically corresponds to the secondary dendrite arm spacing.
	`ENTER MAIN SOLID SOLUTION PHASE` Enter the name of one of the main solid solution phases. In the next prompt you can enter other phases that enter the simulation.
	`SHOULD MORE PHASES BE ENTERED IN THE REGION` Enter Y to enter more phases then enter the phase name as prompted next.
	`ENTER PHASE NAME` Enter the name of another phase that should be entered into the region.

LIST_CONDITIONS

Lists the conditions set with the *SET_CONDITION* on page 34 command.

Syntax	LIST_CONDITIONS
Prompt	OUTPUT FILE File where the information is to be written.

LIST_MOBILITY_DATA

Lists the mobility data from the database or entered interactively by the *ENTER_MOBILITY_DATA* on page 16 command.

Syntax	LIST_MOBILITY_DATA
Prompt	OUTPUT FILE File where the information is to be written. This command does not work for encrypted databases.

LIST_MOBILITY_ESTIMATES

Syntax	LIST_MOBILITY_ESTIMATES
	Lists the mobility estimate entered interactively by the *ENTER_MOBILITY_ESTIMATE* on page 18 command.

LIST_PROFILES

Lists the concentration profiles and grid coordinates in the cell.

Syntax	LIST_PROFILES
Prompts	NAME OF REGION Name of region(s) in which the profiles are to be listed.
	OUTPUT FILE File where the information is to be written.
	COMPOSITION TYPE Composition type in which the profiles are to be written. Legal composition types are:

Syntax	LIST_PROFILES
	• SITE_FRACTION • MOLE_FRACTION • WEIGHT_FRACTION • U_FRACTION
	COMPONENTS The output information may be limited to the specified components.
	CONSTITUENTS The output information may be limited to the specified constituents.

LIST_REGION

List the names of the defined regions, active and inactive phases and the global coordinates of the interfaces.

Syntax	LIST_REGION
Prompt	OUTPUT FILE File where the information is to be written.

LIST_TIMESTEPS

Syntax	LIST_TIMESTEPS
	List time steps in the workspace and those stored on file during a simulation. This for the DICTRA module.

MACRO_FILE_OPEN

Use this to predefine sequences of commands on a file and then execute them with the MACRO command. This is useful when the same calculation is made often with just small changes. One good case for applying this is when calculating diagrams from an assessment. With a macro file all commands can be stored on a file and you just type MACRO <filename>.

Syntax	MACRO_FILE_OPEN
Prompt	`MACRO FILENAME` Give the name of the file with the macro commands. Default extension is DCM.
	The macro file can contain any legal DICTRA module commands. The macro must be terminated with EXIT or in the SYS, DICTRA, POLY-3 or POST module with the command *SET_INTERACTIVE* on page 37.

PARA_EQUILIBRIUM_MODEL

Turns on the para-equilibrium model in the simulation. This implies that the local equilibrium assumption is no longer valid and that substitutional components are regarded as one composite component.

The model is limited to treating only one single moving interface in one cell. The composition of the substitutional components in the phase that is to be dissolved may be chosen in these ways: `AUTO`, `value%`, `-value` and `value`.

Syntax	PARA_EQUILIBRIUM_MODEL
Prompts	`ENABLE PARAEQ` Enables or disables the use of this model by using one of the key words YES or NO.
	`AUTO` The value at the far end (upper or lower end) of the region.
	`VALUE %` The value at a certain percentage from the interface in the region.
	`- VALUE` The value at a fixed distance from the interface.
	`VALUE` A specific value.

POLY_COMMAND

This command sends a string to the POLY-3 module interpreter where it is then executed.

An example of the use of this command is as follows. There are some differences in the default settings for equilibrium calculations based on whether the equilibrium calculation is performed from the DICTRA module or POLY. The changes in the defaults are performed automatically every time the DICTRA module is entered. By using this command it is possible to override those changes.

Syntax	POLY_COMMAND
Prompt	`TO POLY` Command line sent to the POLY-3 module.

POST_PROCESSOR

Syntax	POST_PROCESSOR
	Gives control to the post processor which has its own command repertoire. Also see *POST PROCESSOR Commands* on page 46.

READ_WORKSPACES

The DICTRA, POLY-3 and GES5 workspaces can be read from a file where they must be already saved with a SAVE command. This file is not printable.

Syntax	READ_WORKSPACES
Prompt	`FILE NAME` Name of the file where the workspaces shall be read from. The default file extension is `dic`.

SAVE_WORKSPACES

The workspaces in DICTRA, POLY-3 and GES5 are saved on a file. Return to the state before the SAVE command by entering a READ command. The SAVE command should normally be given before starting a simulation with *SIMULATE_REACTION* on page 42.

Syntax	SAVE_WORKSPACES
Prompts	FILE NAME Name of the file where the workspaces shall be saved on. The default file extension is DIC.
	OVERWRITE CURRENT FILE CONTENT
	PROCEED WITH SAVE If there is already a file with this name and if you answer Y the previous content is overwritten. If you have results from SIMULATE_REACTION these are lost when SAVE is used. You may append several results by the SIMULATE_REACTION command without destroying the previous results but SAVE erases them all.

SELECT_CELL

Selects the current cell from the list of existing cells and enables you to enter and display data into that cell. This command is for the DICTRA.

Also see the POST PROCESSOR prompt of the same name *SELECT_CELL* on page 55.

Syntax	SELECT_CELL
Prompt	NUMBER Specify the cell number by giving an integer or one of the key words NEXT or PREVIOUS.

SELECT_TIMESTEP

Select a time step from those stored on file during a simulation. The profiles can be listed and simulation can be continued from this time step.

Syntax	SELECT_TIMESTEP
Prompt	TIMESTEP The time step to be selected, legal syntax is: • FIRST • LAST • time • #nnn • #? When selecting a time no interpolation is performed but the time step closest to the time entered is selected. #nnn can be obtained from the number given by LIST_TIMESTEPS or by typing #?.
	DELETE ALL OTHER TIMESTEPS Clears the current workspace from all other time steps except the one selected. This is necessary if the simulation is to be continued from this time step.

SET_ACCURACY

Enter the accuracy requirements to use in the determination of the time step when using the automatic time step procedure. It may also be necessary to modify the parameters to control the removal of grid points and set manual starting values for velocities and potentials at the phase interfaces.

Syntax	SET_ACCURACY
Prompts	MAX RELATIVE ERROR The maximum allowed relative error of the profile during one time step integration.
	MAX ABSOLUTE ERROR Maximum allowed absolute error of the profile during one time step integration. This parameter must be chosen with some relation to the smallest concentration in the profiles.

SET_ALL_START_VALUES

Enter starting values for various quantities, such as velocities and potentials.

Syntax	SET_ALL_START_VALUES
Prompts	`START VALUE FOR VELOCITY OF INTERFACE` A starting value for velocity at the named interface is required.
	`START VALUE FOR POTENTIAL` A starting value for a potential at the named interface is required. If an AUTOMATIC value is specified the program attempts to determine a starting value and also selects a suitable component for which the potential is varied.
	`VARYING SPECIES IN INTERFACE` The species which potential is treated as unknown.
	`AUTOMATIC STARTING VALUES FOR PHASE COMPOSITIONS` Compositions used as starting values in the equilibrium calculations using POLY-3. When using automatic starting values the compositions are taken from the entered profiles.

SET_CONDITION

Define conditions to reduce the degrees of freedom at equilibrium or defines the boundary conditions at the outer rims of the system. Use it to set the temperature, pressure or heat extracted from the system.

Conditions can be a function of time and different time-dependent functions can be specified at different time intervals. The syntax for this is approximately the same as used in the GES to specify temperature ranges for thermodynamic parameters.

DICTRA uses a constant molar volume, which is included in the flux, i.e. the unit of flux as entered in boundary conditions is:

$$\text{flux*molar volume} = \text{mol*m}^{-2}\text{*s}^{-1}\text{*m}^{3}\text{*mol}^{-1} = \text{m*s}^{-1}$$

Syntax	SET_CONDITION
Prompts	`GLOBAL OR BOUNDARY CONDITION` Type of condition to be specified. A global condition is either pressure (P), temperature (T) or heat content removal (P) and may be specified as a function of time. Boundary conditions determine how the cell interacts with the world outside the cell.
	`VARIABLE` Legal variables are pressure (P), temperature (T) or heat extracted (Q) as a function of time (TIME), or time-temperature-pairs (T-T-P) that specifies temperature at a specific time and lets the program calculate the cooling or heating rate. For Q the amount of extracted heat per time unit is normalized and the size of the system is normalized to 1 mole of atoms.
	`BOUNDARY`

Syntax	SET_CONDITION
	Defines on which side of the system the boundary conditions are to be specified. Options are UPPER (the rightmost side of the system) and LOWER (the leftmost side of the system).
	`CONDITION TYPE` Defines the type of boundary condition to be specified. The options may in most cases be functions of both TIME, T (temperature) and P (pressure). The default is CLOSED_SYSTEM which is equivalent to setting the fluxes of all components to zero at the boundary. Options are: • `FIX_FLUX_VALUE`: Enter functions that yield the flux times the molar volume for the independent components. May be a function of time, temperature and pressure. • `STATE_VARIABLE_VALUE`: A legal expression in POLY-3 syntax that reduces the degrees of freedom. This type of boundary condition should be used with the uttermost care as no checks are done if it is a legal expression in advance. • `POTENTIAL_FLUX_FUNCTION` and `ACTIVITY_FLUX_FUNCTION`: These types of boundary conditions are used to take into account the finite rate of a surface reaction. The flux for the independent components must be given in the format: $J_k = f_k(T,P,TIME) * \left[ACTIVITY_k^N - g_k(T,P,TIME)\right]$ or $J_k = f_k(T,P,TIME) * \left[POTENTIAL_k^N - g_k(T,P,TIME)\right]$ where f and g may be functions of time (TIME), temperature (T), and pressure (P), and N is an integer. The activities are those with user defined reference states. The function f_k is the mass transfer coefficient, g_k is the activity of the corresponding species in the gas and N is a stoichiometric coefficient. For more details see L. Sproge and J. Ågren, “Experimental and theoretical studies of gas consumption in the gas carburizing process” *J. Heat Treat*. 6, 9–19 (1988). • `ITERATIVE_ACTIVITY_FLUX_FUNCTION`: Same as activity flux function above. However, an iterative scheme is used to determine the flux. This method may be used instead of activity flux function when the latter has

Syntax	SET_CONDITION
	problems. • `CLOSED_SYSTEM`: Corresponds to a fix flux value, which is set to zero at all times. • `MIXED_ZERO_FLUX_AND_ACTIVITY`: The flux of selected components is set to zero and the activity of others may be set to a prescribed value. • `GAS`: The flux of selected components is set to zero and the activity of others may be set to a prescribed value. This option is used for treating an expanding system, e.g. the growth of an external oxide scale.
	`LOW TIME LIMIT` The lower time limit to be used when entering a time dependent function.
	`HIGH TIME LIMIT` The upper time limit to be used when entering a time dependent function. An asterisk * indicates the high limit as infinity.
	`ANY MORE RANGES` To specify whether any additional time dependent functions exists or not.
	`TYPE OF CONDITION FOR COMPONENT` The type of condition when setting a boundary condition of the type MIXED. Options are ZERO_FLUX and ACTIVITY.

SET_FIRST_INTERFACE

Set the coordinate of the first interface in the cell, when a value other than zero is required, for instance when simulating diffusion through a tube wall or a hollow sphere. The default value is zero.

Syntax	SET_FIRST_INTERFACE
Prompt	`COORDINATE FOR FIRST INTERFACE` The coordinate to which the first interface is to be set.

SET_INITIAL_TEMPERATURE

Use this when the heat removal rate from the system is specified. It yields the initial temperature of the system. How the temperature then varies during the simulation is a result of the heat removed from the system.

Syntax	SET_INITIAL_TEMPERATURE
Prompt	`INITIAL TEMPERATURE /2000/`

SET_INTERACTIVE

Syntax	SET_INTERACTIVE
	Useful in demonstration or macro files in order to stop the execution of the command file and pass over input focus to the keyboard.
	In the DICTRA module POST_PROCESSOR this is called SET_INTERACTIVE_ MODE but it is the same command.

SET_NUMERICAL_LIMITS

Set parameters to control the integration, the solution of the flux balance equations, diffusion equations and the equilibrium calculation during the simulation.

Syntax	SET_NUMERICAL_LIMITS
Prompts	`REQUIRED SUM OF SQUARES IN NS01A` Required accuracy during the solution of the flux-balance equations.
	`MAX NUMBER OF CALLS TO CALFUN OF NS01A` The maximum number of iterations when solving the flux-balance equations. This number acts as a *safety valve*, a moderate choice is 10*(number_of_interfaces *(number_of_ components - 1)) but use a larger value if necessary.
	`STEP USED BY NS01A` A user supplied parameter which is used to calculate the estimates of the partial derivatives numerically when solving the flux-balance equations.
	`MAX STEP USED BY NS01A` A parameter which must be set to a generous estimate of the 'distance' between the initial approximation and the required solution of the flux-balance equations.
	`MAX NUMBER OF EQUIDISTANT POINTS IN A REGION` The number of equidistant parts in which a region is divided into that is required to describe the profile. This parameter is used by the procedure that removes unnecessary gridpoints from the profile during the simulation; the number of gridpoints is normally not allowed to be less than this number if a linear grid is used.

Syntax	SET_NUMERICAL_LIMITS
	FRACTION OF REGION ASSIGNED TO THE INTERFACE Fraction of a region at the region border that is to be regarded as part of the interface. This parameter is used by the procedure that removes unnecessary gridpoints from the profile during the simulation.
	SMALLEST NUMBER USED IN SCALING FLUX EQUATIONS The flux-balance equations are scaled by the velocities calculated in the previous time step. This number may however decrease to such a small value so that convergence may be affected. The scaling factor is therefore not allowed to decrease below this value.
	DEFAULT DRIVING FORCE FOR INACTIVE PHASES Sets the necessary driving force needed before an inactive phase is allowed to start to precipitate.

SET_REFERENCE_STATE

The reference state for a component is important when using activities, chemical potentials and enthalpies and it is determined by the data. For each component the data must be referred to a selected phase, temperature and pressure the *reference state*. All data in all phases where this component dissolves must use the same reference state. However, different datasets may use different reference states for the same element. Thus one must mix data from different databases with caution.

By default activities etc. are computed relative to the reference state used by the database and this may thus differ depending on the database. You can select the reference state of a component if the reference state in the database is not suitable.

Syntax	SET_REFERENCE_STATE
Prompts	COMPONENT The name of the component must be given.
	REFERENCE STATE The name of a phase that must be either entered or dormant must be given. The component must be a constituent of this phase of course. A subtle problem is if the component exists in several species in the phase, for example oxygen as O, O2 and O3 in a gas. Normally one would like to have the most stable species as reference state of oxygen, i.e. O2 in this case. Therefore the program calculates the Gibbs energy of all possible states with the phase with the pure component at the current temperature and selects the most stable one.
	TEMPERATURE

Syntax	SET_REFERENCE_STATE
	Select the temperature for the reference state. The value * means the temperature used for the calculation.
	PRESSURE Select the pressure for the reference state.

SET_SIMULATION_CONDITION

Set parameters used to control output and certain parameters during the simulation.

Syntax	SET_SIMULATION_CONDITION
Prompts	NS01A PRINT CONTROL Determines whether data should be printed about the iterative procedure to solve flux-balance equations. This parameter is normally set to 0 but can be set to 1 when difficulties with convergence occur. NS01A prints out the values used in the iterations and the residuals.
	FLUX CORRECTION FACTOR This parameter controls if the flux correction scheme should be used in the calculations. The value should normally always be 1.
	NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF This parameter specifies the number of equally large time steps that one time step should be subdivided into.
	CHECK INTERFACE POSITION This parameter determines whether the time step is to be controlled by the phase interface displacement during the simulation.
	VARY POTENTIALS OR ACTIVITIES Determines whether the program should use the potential or the activity of a component in order to find the correct tie line at the phase interface. The potential or the activity is varied by the program and is set in order to reduce the degrees of freedom at the local equilibrium.
	ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT Determines whether the program is allowed to itself switch the component that is used to reduce the degrees of freedom at the local equilibrium. The scheme used is that of choosing the components which activities or potentials varied most during the previous time step.

Syntax	SET_SIMULATION_CONDITION
	`SAVE WORKSPACE ON FILE` This parameter determines whether the workspaces are to be saved on file during the course of the simulation. Options are `Y` to always save on file, `N` to never save on file, or `##` to save every n:th time on file (## is a integer value ranging from 0 to 99).
	`DEGREE OF IMPLICITY WHEN INTEGRATING PDES` Normally a value of 0.5 (trapezoidal rule) should be used. If however, large fluctuations occur in the profiles it may be necessary to use the value 1.0 (Euler backwards). • `0.0` Euler forwards • `0.5` Trapezoidal rule • `1.0` Euler backwards
	`MAX TIMESTEP CHANGE PER TIMESTEP` Factor specifying the maximum increase in the time step taken from one time step to another. If 2 is given the maximum time step is twice as long as the previous time step taken.
	`USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION` This command mainly concerns the calculation of the equilibrium when using the disperse model in the DICTRA module, where the equilibrium calculations sometimes fail due the abrupt changes in the composition over the region. If `Y`, then these equilibrium calculations are performed using forced starting values in POLY_3.
	`ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF` This determines how often the diffusion coefficient matrix is calculated when solving the partial differential equations (PDE) of diffusion problem. The default setting is to calculate the diffusion coefficient matrix, yielding the stiffness matrix, at each iteration. However, when setting this parameter to `N`, it is only calculated at the first iteration and a constant stiffness matrix is used to obtain the solution to the PDEs. This then leads to an implicit solution and therefore the degree of implicitly is automatically set to 1.

SET_SIMULATION_TIME

Enter the time specific conditions for a simulation.

Syntax	SET_SIMULATION_TIME
Prompts	END TIME FOR INTEGRATION The time up to which the simulation is to be carried out.
	AUTOMATIC TIMESTEP CONTROL Determines whether the time step should be controlled by an automatic procedure or not. If you answer N you are prompted for the fixed time step to use during the simulation. The time step determined by the automatic time step control procedure is controlled by the parameters set by the command *SET_ACCURACY* on page 33.
	MAX TIMESTEP DURING INTEGRATION The maximum time step allowed during the simulation. This is required when using the automatic procedure to determine the time step.
	TIMESTEP DURING INTEGRATION Fixed time step used when the automatic time step procedure is disabled.
	INITIAL TIMESTEP Time step used as the initial time step.
	SMALLEST ACCEPTABLE TIMESTEP The smallest time step allowed during the simulation. This is required when using the automatic procedure to determine the time step.

SET_SURFACE_TENSION

This command enters a distance- and velocity-dependent function which is added to the Gibbs energy of the phase at left side of each phase interface. This can then be used to emulate the effect of surface tension on the equilibrium between at the phase interface or limited phase mobility.

Syntax	SET_SURFACE_TENSION
Prompt	FUNCTION Enter a surface energy function which adds to the Gibbs Energy expression for the phase located at the lower (left) side of the interface. The expression is multiplied with the volume per mole substitional atoms. This command is also used to simulate a limited interfacial mobility control where the energy function is a function of the interface velocity. It is used to enter the surface energy to enable coarsening. However, it can also be used to simulate a case with a limited interfacial mobility.
	This function should only be used in spherical geometries if the function is solely dependent of the interface position.

Syntax	SET_SURFACE_TENSION

	Function describing how the (surface) energy function varies with the particle radius and/or the interface velocity.
	
	The classical expression for the surface energy contribution has the form of:
	
	$\frac{2\sigma V_m}{r}$
	
	The molar volume should be given relative to the molar volume used by default in DICTRA, 1 x 10^{-5} (m^3/mole). The volume should also be given per mole of substitutional atoms. For a precipitate of type M_xC_y this means multiplying with a factor (y+x)/x. If we use M_6C as an example with surface tension 0.5 (J/m^2) and molar volume of 0.71 x 10^{-5} (m^3/mole), the entered function should be 2*0.5*0.71*(7/6)/X;
	
	Commonly the surface tension function is defined as:
	
	`delta-G = 2 * SIGMA * Vm / R`
	
	The function to be entered will then be:
	
	`2*SIGMA/X;`
	
	A limited interface mobility can be expressed as:
	
	`delta-G = v * Vm /const`
	
	The function to be entered is then:
	
	`v/"const";`
	
	The expression is multiplied by the partial molar volume of the substitutional components.

SIMULATE_REACTION

Start the simulation. If given without any argument, you are prompted for certain values during simulations where phases appear or disappear.

Syntax	SIMULATE_REACTION
Prompt	The command can also be given with the argument YES typed on the same line: `SIMULATE_REACTION YES` With the `YES` argument, default values are used during simulation and no user input can be given. This is especially useful when using batch mode.

STORE_HOMOGENIZATION_DATA

Read/write the interpolation scheme data to file. The interpolation scheme is an option of the homogenization model that is used to speed up simulations. Use if this option can speed up simulations further.

Syntax	STORE_HOMOGENIZATION_DATA
Prompt	`STORE/READ HOMOGENIZATION MODEL INTERPOLATION DATA <Y/N>` Answering `Y` or `N` enables/disables read/write of interpolation scheme data to file.

SWITCH_MODEL

Select if the *HOMOGENIZATION_MODEL* on page 25 should attempt to solve the problem if the classic model fails.

There are two solvers for moving phase boundary problems. The classic computationally efficient model and the more robust, but computationally more demanding, homogenization model. By default, the classic solver is tried first. If this fails, then the new solver is tried for a few time-steps before switching back to the classic solver, and so on.

Syntax	SWITCH_MODEL
Prompts	`ENABLE AUTOMATIC SWITCH OF MODEL (Y/N)` Determine if automatic switching from the classic solver to the homogenization model is allowed.

UTILITIES_HOMOGENIZATION

Use this with the homogenization model for the available utilities.

Syntax	UTILITIES_HOMOGENIZATION
Prompts	`SET TEMPERATURE ACCORDING TO SOLIDUS TEMPERATURE` Set the global temperature such that it goes to a certain value T {target}=T_{sol}-X, where X is an entered value. Also enter the rate by which the temperature approaches T {target}. The solidus temperature is determined with a +/- 1 K accuracy. Input guesses of the minimum and maximum temperature to occur during the simulation. This utility can be useful to optimize homogenization heat treatments. The liquid phase must be entered into the system.
	`DUMP RESULTS TO TEXT FILES`

Syntax	UTILITIES_HOMOGENIZATION
	Select `Y` to save simulation results to various text files. The names of these files are fixed and are saved to the current working directory.
	`READ INITIAL COMPOSITION FROM TEXT FILE` Select `Y` to read initial composition from a text file `XF.TXT` that must be present in the working directory. The file should contain the mole fractions of all elements, in alphabetical order, starting from the first grid point, in the first region, in the first cell.
	`ENTER GHOST PHASE` Select `Y` to force the so-called ghost phase to be created. This phase has full solubility of all components and zero diffusivity. The Gibbs energy surface of the ghost phase is set above all other phases. The name of this phase is ZZDICTRA_GHOST. It can be used for numerical reasons. It is created automatically if there is a phase that lack solubility range of one or more components and then used internally by the program.
	`ENTER INACTIVE PHASES INTO ONE REGION` Select `Y` to make all inactive phases at a given interface to be entered into a single multiphase region when any one of the phases becomes stable.
	`EXPLICITLY SET SUBSTITUTIONAL/INTERSTITIAL` Select `Y` to explicitly select for each element whether it should be substitutional or interstitial. There must be at least one substitutional element in each region. These settings only have effect for the homogenization model and results viewed in the post processor must be interpreted with care; use `DUMP RESULTS TO TEXT FILES` to get results corresponding to the settings entered here.
	`SAVE AVERAGE FINITE VOLUME COMPOSITION` Save the average finite volume composition instead of a converted piece-wise linear composition.
	`SET CONSTANT PHASE ADDITION` Add constant Gibbs energy contributions to phases (in J/mol formula unit).
	`USE ELEMENT MOBILITY PREFACTOR` Enter constant factors which the mobilities of elements are multiplied with in all phases.
	`ENTER ELEMENT MOBILITY ESTIMATE` Interactively enter a constant mobility estimate for specific elements in specific phases. This can be used to enter mobility estimates in phases for which there is no assessed data. Estimates entered here override database values.

Syntax	UTILITIES_HOMOGENIZATION
	`ENTER MINIMUM SAVE INTERVAL` Enter a minimum time interval (in seconds) that must pass between subsequent saved time-steps. This is useful in cases where a large number of small time-steps may occur in order to keep down the size of the result file.
	`SIMPLIFIED EVALUATION OF ACTIVITY BOUNDARY CONDITION` With this setting enabled the fluxes on boundaries are evaluated directly from the activity gradient, i.e. without taking into account the change of state on the boundary caused by the prescribed boundary activities.
	`ENTER EXPLICIT LIMITS ON COMPOSITION` Use this setting with caution to explicitly enter upper and lower limits on composition for all components (in mole fraction).
	`ENTER CHEMICAL POTENTIAL CONTRIBUTION` Enter a function (in valid poly-3 syntax) that is evaluated separately at all grid points and added to the chemical potential of all elements. The contribution affects the flux between grid points.
	`DO NOT BALANCE EXTERNAL BOUNDARY FLUXES` For all boundary conditions except type GAS, the flux of substitutional elements into the domain is balanced with the flux of the dependent substitutional elements such that the size of the domain is preserved. Enabling this setting disables the balancing of the substitutional fluxes.
	`FIX EXTERNAL BOUNDARY FLUX OF DEPENDENT COMPONENT` Enter a constant flux on the boundary of the dependent substitutional component. For all boundary conditions except type GAS, this flux is otherwise set to a value such that the net substitutional flux is zero.
	`MODIFIED KINETICS ON BOUNDARY` Use this to modify the kinetics on the boundary and enter a factor. If the factor is greater/smaller than one, the mobilities of all elements on the boundary is set equal to the maximum/minimum mobility of all elements times the factor. This can be useful in order to quickly saturate an outermost finite volume.

POST PROCESSOR Commands

In this section:

Purpose and Methodology

The purpose of the POST PROCESSOR is to read and process data from the internal data structure of the DICTRA module in order to present the result of a simulation in either* GRAPHICAL (command *PLOT_DIAGRAM* on page 54) or* TABULAR FORM (command *MAKE_EXPERIMENTAL_DATAFILE* on page 53).

Before plotting/printing, you have to specify what variables should be plotted on the diagram axis or printed in the table columns. For this purpose there is a command *SET_DIAGRAM_AXIS* on page 57. After having SET both X- and Y-axis variables, you have to SET also plot condition and independent variable. It is also possible to specify a Z-axis. Its values appear as tick marks on the XY-curve.

Plot Condition and Independent Variable

There are two free variables after a simulation done by the DICTRA module. One is the simulation TIME, the other is a DISTANCE in the system.

In general, when plotting a diagram you MUST set either type as fixed, i.e. the *plot condition*, use command *SET_PLOT_CONDITION* on page 60. The plot condition chosen is printed on a separate line above the diagram.

The other variable is then the independent variable, use command *SET_INDEPENDENT_VARIABLE* on page 59. The independent variable is used as stepping variable. It varies along the calculated curve.

Legal Plot Conditions

- `TIME`: Integration time
- `DISTANCE`: Distance in system. It may be a GLOBAL distance counted from the rightmost interface of the system. Alternatively, it may be LOCAL. The distance is then counted from the LOWER interface of the specific region prompted for.
- `INTERFACE`: At a certain interface. An interface is identified by the NAME of the adjacent region, with the addition that you will be prompted to specify if the interface is at the upper or lower end of the region.
- `INTEGRAL`: Is automatically SET when using an INTEGRAL VARIABLE as axis variable.
- `TIE_LINE`: At a certain interface. Allows you to plot a certain quantity from both sides of an interface. The typical application is for plotting tie-lines.

Legal Independent Variables

- `TIME`: Integration time. Is automatically SET when using an INTEGRAL VARIABLE' or TIME as axis variable.
- `DISTANCE`: Space coordinate. It may be GLOBAL. It is then counted from the rightmost interface of the system. Alternatively, it may be LOCAL. The distance is then counted from the LOWER interface of the specific region prompted for.

APPEND_EXPERIMENTAL_DATA

Add experimental data and text on a calculated diagram. The experimental data and text are added to a file prepared according to the syntax of the DATAPLOT graphical language. The picture generated from the data is superimposed on the ordinary graphical output from the POST PROCESSOR. The experimental data file can be created with an ordinary text editor. Another use of the APPEND_EXPERIMENTAL_DATA command is to superimpose plots from several independent calculations. For this purpose, there is a command *MAKE_EXPERIMENTAL_DATAFILE* on page 53 which dumps a calculated diagram on a file according to the DATAPLOT syntax. With the aid of a basic text editor many such files may be merged. Remember to have only one prologue section on the file (see below).

Also see *About DATAPLOT Files* on page 66.

Syntax	APPEND_EXPERIMENTAL_DATA
Prompts	`USE EXPERIMENTAL (Y OR N)` Specify whether the data from an experimental data file should be included in the next plot. If `N`, no experimental data is plotted.
	`EXPERIMENTAL DATAFILE`

Syntax	APPEND_EXPERIMENTAL_DATA
	Specify the name of the file with the experimental data. Default file extension is exp.
	PROLOGUE NUMBER Select which prologue to use. In a prologue one may e.g. give the scaling of an axis, the axis texts, and so on. -1 gives a list of all prologues on the file. Read more about prologues below.
	DATASET NUMBER(S) Select from which dataset(s) data should be read. Several datasets may be given separated with commas or spaces. -1 gives a list of all datasets on the file.

DETERMINE_KIRK_PLANE

To use this command a certain time must be set as plot condition. You are prompted for the position of a plane at time zero. The position of the plane at the plot condition time is then calculated. The difference in position is equal to the Kirkendall shift, i.e. how much an inert marker would drift in the material due to a net flux of vacancies.

Syntax	DETERMINE_KIRK_PLANE
Prompt	INITIAL ZERO PLANE /-1/

DIFFERENTIATE_VALUES

Syntax	DIFFERENTIATE_VALUES
	Differentiate the plotted curve.

DUMP_DIAGRAM

An alternative way to create plots. Supported graphical formats are PNG, BMP, PDF, JPEG and TIFF. The plot is saved to a file.

Syntax	DUMP_DIAGRAM
Prompt	OUTPUT FORMAT (PNG,BMP,PDF,JPEG,TIFF) Specifies which graphical format to use.
	RESOLUTION (LOW,MEDIUM,HIGH) Specifies the resolution of the plot.

ENTER_SYMBOL

Define a symbolic name to represent either a table or an 'arbitrary' function. Legal variables in functions and columns in tables are state variables, auxiliary variables, integral variables or previously defined functions. Functions are a useful feature of the POST PROCESSOR to define quantities.

Syntax	ENTER_SYMBOL
Prompts	`FUNCTION OR TABLE /FUNCTION/` Select what kind of symbol to enter.
	`NAME` Each symbol has a unique name that must start with a letter and can have maximum 8 characters. If one wishes to enter the name and the value on the same line they must be separated with an equal sign =.

	`FUNCTION`
	
	Functions are evaluated from an expression of state variables, auxiliary variables, integral variables or previously defined functions. The expression is a Fortran-like expression and operators +, -, *, / and ** can be used (** only with integer powers). Unary functions like LOG, LOG10, EXP, SIN, COS, ABS and ERF can also be used. An expression can be continued on more than one line. An expression should be terminated by a semicolon or an empty line.
	
	Examples of functions:
	
	`POI(CEM,U)-POI(CEM,L);`
	The thickness of the region named CEMENTITE
	
	`AC(CR)/X(FCC,CR);`
	The activity coefficient for Cr in phase FCC
	
	`SQRT(TIME);`
	The square root of the simulation time
	
	`&`
	
	The ampersand & sub-prompt displays if the function was not terminated by a semicolon. It allows a user to continue to write the function on the new line if one line is not enough for the function. If one has finished the function just press return again.

	`VARIABLE(S)` When entering a table, specify what variables are to be in the various columns. Separate the variables with commas or space characters. At present a maximum of 15 columns are allowed.

Syntax	ENTER_SYMBOL
	The independent variable is always printed in the first column.

INFORMATION

Some general information about the POST PROCESSOR module is given.

Syntax	INFORMATION
Prompt	`WHICH SUBJECT` Select which subject you want additional information about. Type ? to get a list of topics. • PURPOSE • STATE VARIABLES • INTEGRAL VARIABLES • AUXILIARY VARIABLES • PLOT CONDITION AND INDEPENDENT VARIABLE • BASIC METHODOLOGY

INTEGRATE_VALUES

Syntax	INTEGRATE_VALUES
	Integrate the plotted curve.

LABEL_CURVES

Create labels. Each label is explained with to the right of the diagram.

Syntax	LABEL_CURVES
Prompt	`LABEL CURVES` Select `Y` or `N`.
	For example: *1. X: X(CR); Y: TIME; PC:3* is read as *1 is a curve with X(CR) on the x-axis, TIME on the y-axis and plot condition PC according to value 3 specified on the plot condition line printed above the diagram*.

LIST_PLOT_SETTINGS

Syntax	LIST_PLOT_SETTINGS
	Lists the present values of most parameters specifying the type of diagram to be plotted.

LIST_REGION_NAMES

Lists all region names defined.

Syntax	LIST_REGION_NAMES
	All region names have a #-sign and the cell number appended.

LIST_SYMBOLS

Lists a specific or all symbols defined.

Syntax	LIST_SYMBOLS
Prompt	NAME Give the name of a symbol or an asterisk * to list all defined symbols.

LIST_TIME_STEPS

This for the POST PROCESSOR module.

Syntax	LIST_TIME_STEPS
	Lists all integration time steps.

MAKE_EXPERIMENTAL_DATAFILE

Save graphical information in a file with the DATAPLOT format. To merge two or more diagrams from separate calculations, use this command to export it and add them together with a normal text editor.

Also see *APPEND_EXPERIMENTAL_DATA* on page 49 and *About DATAPLOT Files* on page 66

Syntax	MAKE_EXPERIMENTAL_DATAFILE
Prompt	`OUTPUT FILE` File where the graphical information is written. Default file extension is `exp`

PLOT_DIAGRAM

Plot graphical information on the specified device using the plot format set by *SET_PLOT_FORMAT* on page 61.

Syntax	PLOT_DIAGRAM
Prompt	`PLOT FILE` The name of the file or graphical device.

PRINT_DIAGRAM

Syntax	PRINT_DIAGRAM
	Print the diagram using the printers defined in Windows. It is only available for a Windows operating system.

QUICK_EXPERIMENTAL_PLOT

Defines a pair of axes, sets the axis labels to X and Y, and scales both x- and y-axis between 0.0 and 1.0 unless a prologue is read from the data file.

It is similar to the APPEND_EXPERIMENTAL_DATA command but can be used when there is no graphical information to be plotted in the DICTRA module workspace.

Syntax	QUICK_EXPERIMENTAL_PLOT
	Also see *APPEND_EXPERIMENTAL_DATA* on page 49 for information about the format of the data file.

REINITIATE_PLOT_SETTINGS

Syntax	REINITIATE_PLOT_SETTINGS
	All parameters describing the diagram are given as default values.

SELECT_CELL

Select the cell data to be processed. You can only plot data from one cell at a time. Type `?` to get a list of valid cell numbers. This command is for the POST PROCESSOR.

Also see the DICTRA module prompt of the same name *SELECT_CELL* on page 32.

The current cell number is displayed as a part of the POST PROCESSOR prompt.

Syntax	SELECT_CELL
Prompt	`NUMBER` Number of the cell to be selected. Specify cell number by giving an integer or one of the key words NEXT or PREVIOUS.
	`TABULATE` Tabulate a named table. The independent variable is always printed in the first column.
	`NAME` Give the symbolic name of the table.
	`OUTPUT FILE /SCREEN/` Select output device/file. Press Enter to get output on the screen.

SET_AXIS_LENGTH

Change the relative length of an axis, i.e. the number of tic-marks on the axis. The default number of tic-marks on an axis is 10 when the relative length is 1. The number of units per tic-mark must be a multiple of 1, 2, or 5 to obtain a reasonable scaling of an axis.

Syntax	SET_AXIS_LENGTH
Prompts	`AXIS` Specify the axis to set the axis length.
	`AXIS LENGTH` Specify the relative axis length. The relative length 1 corresponds to 10 tic-marks on the axis.

SET_AXIS_PLOT_STATUS

Specify to plot a diagram axis. Use it to merge different diagrams on a pen-plotter or to obtain the diagram faster. The default to plot the axes.

Syntax	SET_AXIS_PLOT_STATUS
Prompt	`AXIS PLOT` `Y` or `N` to plot axis.

SET_AXIS_TEXT_STATUS

Change the axis text from the automatic text given by the axis specification to another text.

Syntax	SET_AXIS_TEXT_STATUS
Prompts	`AXIS (X, Y OR Z)` Specify which axis text status to change (if the axis type is INVERSE, X2 or Y2 may be used to set the corresponding opposite linear axis text).
	`AUTOMATIC AXIS TEXT (Y OR N)` Specify if automatic axis text is to be used or not.
	`AXIS TEXT` Enter axis text.

SET_AXIS_TYPE

Select a linear, logarithmic or inverse axis.

Syntax	SET_AXIS_TYPE
Prompts	`AXIS (X, Y OR Z)` Specify which axis to change the axis type.
	`AXIS TYPE` Specify which axis type to set. Select `LIN`ear (default), `LOG`arithmic or `INV`erse. Only the three first characters are needed.

SET_COLOR

On devices that support colors/(line types), select different colors/(line types).

Syntax	SET_COLOR
Prompt	• Text and axis Color • Diagram Color • DATAPLOT Color

SET_DIAGRAM_AXIS

Specify the axis variables of a plot. At least two axis variables (x and y) must be specified.

Syntax	SET_DIAGRAM_AXIS
Prompts	AXIS (X,Y OR Z) Specifies the axis to set a variable.
	VARIABLE Specifies the variable to plot along this axis. A variable is specified by its mnemonic or name. The valid variable mnemonics and names are explained below. The variable types are: • NONE: To clear an axis setting. • INTEGRAL VARIABLES: A quantity obtained by integration in space over the whole system or over a specific region. In a planar geometry values are given per unit area, in a cylindrical geometry they are given per unit length, and in a spherical geometry they are absolute values. • FUNCTIONS: Identified by a name which is entered with enter function. Use the list_symbols command to get a list of valid function names. • AUXILIARY VARIABLES: As a complement to state variables and integral variables the auxiliary variables are defined. The variables can be called by their mnemonic names, which are shown in the rightmost column. These are useful in user-defined functions. • STATE VARIABLES: State variables in the POST module are similar to those defined in POLY-3. Examples of state variables are temperature, mole fraction, enthalpy, etc. In POLY-3 a general notation method based on character mnemonics is designed for a predefined set of state variables.

Syntax	SET_DIAGRAM_AXIS
	Also see *State, Integral and Auxiliary Variables* on page 70. Depending on the axis variable some sub-prompts may be available.
	`FOR COMPONENT` When an activity, flux, mole-, weight- or U-fraction or percent is plotted the name of the component must be supplied.
	`IN REGION` When the lamellar spacing, e.g. for pearlite, is plotted the name of the region must be supplied.
	`FOR PHASE` When the lamellar thickness, e.g. for ferrite in pearlite, is plotted the name of the phase must be supplied.
	`INTERFACE` When the velocity or the position of an interface is plotted the name of the interface must be given. An interface is identified by the name of the region on its upper side.
	`TYPE` When choosing distance as variable one has additionally to specify what type of distance. A distance may be GLOBAL, i.e. counted from the leftmost interface, or LOCAL, i.e. counted from the interface to the left of the region whose name you are prompted for.
	`REGION NAME` Name of the region wherein the local distance is measured.

SET_DIAGRAM_TYPE

Set the diagram to a square (the default) or triangular plot (Gibbs triangle).

Syntax	SET_DIAGRAM_TYPE
Prompts	`TRIANGULAR DIAGRAM` Select `Y` for a triangular plot.
	`PLOT 3:RD AXIS` For triangular plots is selected specify if a 3:rd axis, connecting the end points of the x- and y-axis is plotted.

Syntax	SET_DIAGRAM_TYPE
	`CLIP ALONG THE 3:RD AXIS` Remove all lines outside the region limited by a line joining the end points of the X- and Y-axis.

SET_FONT

Select the font to use for labels and numbers when plotting the diagram. For some devices (e.g. PostScript) there may be other fonts available and these are selected by the *SET_PLOT_FORMAT* on page 61 command.

Syntax	SET_FONT
Prompts	`SELECT FONTNUMBER` Give the number for the font to select. Type ? to get an online list of the available fonts.
	`FONT SIZE` A value of 0.3 is recommended.

SET_INDEPENDENT_VARIABLE

There are two free variables after a simulation is done in theDiffusion Module (DICTRA). One is the simulation TIME, the other is a DISTANCE in the system. When plotting a diagram you must choose either one to vary along the curve, i.e. the independent variable, the other one is then the plot condition.

When plotting integral quantities TIME should be the independent variable.

Syntax	SET_INDEPENDENT_VARIABLE
Prompts	`VARIABLE` Select which variable TIME (integration time) or DISTANCE (space coordinate) to treat as the independent variable.
	`TYPE` When choosing distance as independent variable specify the type of distance. A distance may be GLOBAL, i.e. counted from the leftmost interface, or LOCAL, i.e. counted from the interface to the left of the region whose name you are prompted for.
	`NAME OF REGION`

Syntax	SET_INDEPENDENT_VARIABLE
	For a LOCAL distance supply the name of the region within which the distance is measured. Type a ?, to get a list of valid region names.

SET_PLOT_CONDITION

There are two free variables after a simulation is done in theDICTRA module. One is the simulation TIME, the other one is a DISTANCE in the system. In general, when plotting a diagram you must choose either type as fixed (i.e. the plot condition), the other one is then the independent variable. However, when plotting integral quantities, plot condition is automatically set to INTEGRAL and TIME is chosen as independent variable. N.B. You cannot mix different kinds of plot conditions. However, for TIME and DISTANCE you may supply up to 15 different condition values, see sub-prompt VALUE(S) below. The plot condition chosen is printed on a separate line above the diagram.

Syntax	SET_PLOT_CONDITION
Prompts	CONDITION Specify the type of condition to set. Valid conditions are: • NONE: Condition not set. • TIME: Integration time. • DISTANCE: Distance in system. • INTERFACE: At a certain interface. • INTEGRAL: Value is integrated over the phase/region/system volume • TIE_LINE: Pair of values from both sides of certain interface. Depending on the condition some or none of these sub-prompts may display.
	INTERFACE Specify at which interface the condition should be set. FIRST and LAST refer to the leftmost and the rightmost interface of the system, respectively. Type ? to get a full list of relevant region names.
	TYPE When distance is chosen as plot condition specify the type of distance. A distance may be GLOBAL, i.e. counted from the leftmost interface, or LOCAL, i.e. counted from the interface to the left of the region whose name you are prompted for.
	NAME OF REGION Give the name of the region within which the distance is measured.

Syntax	SET_PLOT_CONDITION
	AT UPPER INTERFACE OF REGION Give the name of the region which upper interface is specified for the TIE_LINE plotting. This plot-condition allows you to plot a certain quantity from both sides of an interface. The typical application is for plotting tie-lines.
	UPPER OR LOWER INTERFACE OF REGION Specify if the condition is at the LOWER or UPPER interface of a region. The LOWER interface is placed on the left side of a region and the UPPER interface on its right side.
	VALUE(S) Supply up to 15 numerical values separated by commas or spaces for the condition set. If plot condition TIME was chosen FIRST, LAST or #n (where n is an integer number) may be specified. #? provides a list of time steps.

SET_PLOT_FORMAT

Adjust the format of the graphical output to another graphical device. Usually the default device is a Tektronix-4010 terminal. This default can be changed with the *SET_PLOT_ENVIRONMENT* on page 243 command in the SYSTEM MONITOR or by your TC.INI file, see separate documentation.

Syntax	SET_PLOT_FORMAT
Prompt	GRAPHIC DEVICE NUMBER Depending on the available hardware different plot formats may be available. These are listed with a ?. With some formats there can be additional sub-prompts asking for e.g. font type and size.

SET_PLOT_OPTIONS

Toggle on/off the plotting options on the diagram.

Syntax	SET_PLOT_OPTIONS
Prompts	PLOT HEADER Toggle the plot of the text above the diagram.
	PLOT LOGO Toggle the plot of the logo at lower-left corner of the diagram.
	PLOT FOOTER Toggle the plot of the footer identifier text (only on postscript devices).

Syntax	SET_PLOT_OPTIONS
	WHITE COUNTOURED PS CHARS Toggle the option of having a thin white contour around postscript characters (only on postscript devices).
	PLOT REMOTE EXPONENTS Toggle the plot of the remote exponents on the axis.
	PLOT CELL# Toggle the plot of the cell number text at the upper-right corner of the diagram.
	PLOT-CONDITION STATUS Toggle the plot of the plot-condition status line above the diagram.

SET_PLOT_SIZE

Specify a relative scale factor to change the size of the diagram. The default value of the scaling factor depends on what output device is chosen by the *SET_PLOT_FORMAT* on the previous page command. The default plot size is adjusted to the chosen device.

Syntax	SET_PLOT_SIZE
Prompt	RELATIVE PLOT SIZE Enter the relative scaling factor.

SET_PREFIX_SCALING

When prefix scaling is enabled the remote exponent for an axis is automatically chosen to have a value which is a multiple of three, i.e. ..., -6, -3, 0, 3, 6,...

Syntax	SET_PREFIX_SCALING
Prompts	AXIS (X OR Y) Specify X- or Y- axis.
	USE PREFIX SCALING Enables or disables prefix scaling, Y or N. Give an integer value to select a prefix power of your own.

SET_RASTER_STATUS

Set to have a raster plotted in the diagram. Default is no raster plotted.

Syntax	SET_RASTER_STATUS
Prompt	`RASTER PLOT (Y OR N) /Y/` Enables (`Y`) or disables (`N`) the raster plot.

SET_SCALING_STATUS

Choose between manual or automatic scaling on a specified axis. If manual scaling is chosen specify a minimum and a maximum value. Manual scaling can be used to magnify interesting parts of a diagram. When an axis variable is selected by the *SET_DIAGRAM_AXIS* on page 57 command the scaling status for the axis is always set to automatic scaling.

Syntax	SET_SCALING_STATUS
Prompts	`AXIS(X, Y OR Z)` Specify what axis to set the scaling status.
	`AUTOMATIC SCALING (Y OR N)` Automatic (`Y`) and manual (`N`) scaling.
	`MIN VALUE` For manual scaling specify the value of the starting point of the specified axis.
	`MAX VALUE` For manual scaling specify the value of the end point of the specified axis.

SET_TIC_TYPE

Change the size of the tic marks and the placement either inside or outside of the axis.

Syntax	SET_TIC_TYPE
Prompt	`TIC TYPE` Place on either the inside or the outside of the diagram axis. The absolute value of TIC TYPE determines the length. • `TIC TYPE > 0` (outside the axis) • `TIC TYPE < 0` (inside the axis)

SET_TITLE

Specify a title that displays on all listings and diagrams from the POST PROCESSOR.

Syntax	SET_TITLE
Prompt	`TITLE` Enter the title to include on all output. There is a maximum length of about 60 characters.

SET_TRUE_MANUAL_SCALING

Set to manual scaling instead of the default where the tic marks on the axis are placed in even intervals using the whole axis length. The scaling routine adjusts the given minimum and maximum values slightly to accomplish this. The command works like a toggle. To reset the scaling behavior just repeat the command a second time.

Syntax	SET_TRUE_MANUAL_SCALING
Prompt	`AXIS (X OR Y)` Specify the axis you want to toggle between automatic adjustment or avoiding the adjustment of the given maximum and minimum values.

DATPLOT Files

In this section:

About DATAPLOT Files

A DATAPLOT file can be divided into two sections. The prologue, which contains directives for manipulating the default settings on the diagram layout, and the dataset, which contains the actual data and any text to be plotted in the diagram. The sections may contain multiple PROLOGUE and DATASET statements, respectively.

Also see the *DATPLOT User Guide* included with this documentation set.

All prologues must be placed before the first DATASET statement.

The syntax must be used for the DATAPLOT file.

In general, each line in a DATAPLOT file must consist of a legal keyword plus its parameters or an XY coordinate pair.

An XY coordinate pair is two real numbers and an optional graphical operation code (GOC).

Keywords for Prologues

Keyword	Parameters
PROLOGUE	inumb text
XSCALE	min max
YSCALE	min max
XTEXT	text
YTEXT	text
XTYPE	LIN, LOG or INV
YTYPE	LIN, LOG or INV
XLENGTH	rnumb
YLENGTH	rnumb
DIAGRAM_TYPE	TRIANGULAR or SQUARE
TIC_TYPE	rnumb
TITLE	text

Keywords for Datasets

Keyword	Parameters
DATASET	inumb text
FONT	inumb
CHARSIZE	rnumb
COLOR	inumb
GLOBALSIZE	rnumb
LFSIZE	rnumb
LINETYPE	inumb
BLOCK GOC=C3,DEFGOC; X=C1+273.15; Y=LOG10(C2);	Select one X and one Y variable
BLOCK GOC=C3,DWR; X1=C1+273.15; X2=C1; Y=C1+C2*4.184;	or multiple X and one Y variable
BLOCK GOC=C4,MWAS; X=C1+273.15; Y1=C2*4.184; Y2=C3;	or one X and multiple Y variables
BLOCK GOC=C5,DWA; X1=C1; X2=C4; Y1=C2*4.184; Y2=C3;	or multiple XY pairs
BLOCKEND	
DRAWLINE	x,y x,y
TEXT	text or ~name (of a string, see below)
ATTRIBUTE	TOP, CENTER or BOTTOM
STRING	name text The text string may contain text formatting codes namely: ^S# to set font size, ^G for Greek font, ^F# to set font type, ^U# for positioning the text upwards, ^D# for positioning the text downwards, ^R#for positioning the text to the right, ^L# for positioning the text to the left, and finally ^N for printing without updating current position to the end of the character. These text formatting codes only work for soft fonts not hardcopy postscript fonts.
INCLUDE	filename

Keyword	Parameters
CLIP	ON or OFF
SYMBOLSIZE	rnumb
FUNCTION Y=203(X); or X=203(Y);	<start end steps GOC> The < > denotes an optional parameter.
PAINT	<code> <video> <mode> The <> denotes optional parameters. Paint the area enclosed by the current path in the current pattern. The current path starts at the last 'moveto' given and includes all subsequent 'draws'. Also see PCFUNCTION below. Default is <code>=0, <video>=NORMAL and <mode>=TRANSPARENT. To set a new current pattern supply any or all of the optional parameters. <code> is a single letter 0-9, A-Z or a-t (if <code>=t supply also a number in the range 0.00 - 1.00, e.g. PAINT t 0.25.) <video> is a string reading NORMAL or INVERSE. <mode> is a string reading TRANSPARENT or OPAQUE.
PCFUNCTION Y=203(X); or X=203(Y);	<start end steps GOC> The <> denotes an optional parameter. May be used to add a function to the current path.

Available GOCs

These are the available GOCs:

- `W`: World coordinates (* DEFAULT)
- `V`: Virtual coordinates
- `N`: Normalized plot box coordinates (NPC)
- `M`: Move to this XY (*)
- `D`: Draw to this XY
- `A`: XY is absolute values (*)
- `R`: XY are relative values
- `S`: Plot current symbol at XY
- `B`: Apply soft spines on the drawn curve (use only on BLOCK data)

- `S2`: Change current symbol to 2 and plot at XY
- `'`: Plot the following text at XY

Example

```
700 2.54
50 1.91 WRDS
1.1 0.9 NS3
100 20'This is a text
BLOCK GOC=C3,MWAS; X=C1+273.15; Y=C2*4.184;
600 -1400 S1'text
700 -1500
BLOCKEND
```

State, Integral and Auxiliary Variables

In this section:

State Variables

The numerical values of some state variables, e.g. activity, chemical potential, and the Energetic Extensive properties, are given relative to some reference state which has been defined in the thermodynamic database file. Sometimes a user may have specified a reference state of his own, using *SET_REFERENCE_STATE* on page 169. To get numerical values relative to the user-defined reference state, append an R to the mnemonic names as defined.

Intensive Properties

Mnemonic	Description
T	temperature
P	pressure
AC(component)	activity
MU(component)	chemical potential

Extensive Properties

NORMALIZATION of extensive properties: For all extensive properties a suffix can be added to the mnemonic name to indicate a normalized extensive property.

Extensive Properties Mnemonic Suffix

Mnemonic	Description
Extensive property `Z at each grid point:	
Z	ext. prop. `Z at each grid point.
ZM	ext. prop. `Z per moles of atoms.
ZW	ext. prop. `Z per mass (gram).
ZV	ext. prop. `Z per volume (m3).
Extensive property `Z for a phase at each grid point:	
Z	ext. prop. `Z for the current amount of the phase.
ZM	ext. prop. `Z per mole of atoms of the phase.
ZW	ext. prop. `Z per mass (gram) of the phase.
ZV	ext. prop. `Z per volume (m3) of the phase.
ZF	ext. prop. `Z per mole formula unit of the phase.

Mnemonic	**Description**
Energetic Extensive properties (note: suffixes M, W, V and F can be applied)	
S	entropy
S(phase)	entropy of a phase
V	volume
V(phase)	volume of a phase
G	Gibbs energy
G(phase)	Gibbs energy of a phase
H	enthalpy
H(phase)	enthalpy of a phase
A	Helmholtz energy
A(phase)	Helmholtz energy of a phase
Amount of components (note: suffixes M, W and V can be applied)	
N(component)	Number of moles of a component.
N(phase,component)	Number of moles of a component in a phase.
B(component)	Mass of a component.
B(phase,component)	Mass of a component in a phase.
N	Number of moles
B	Mass (Note that the combination BW is not very interesting as it is always unity. BV is the density.)
Amount of a phase (note: suffixes M, W and V can be applied). The normalizing properties are calculated at each grid point.	
NP(phase)	Number of moles of a phase.
BP(phase)	Mass of a phase.
VP(phase)	Volume of a phase.
Y(phase,species#sublattice)	site fraction (this quantity is dependent upon the model chosen for the phase)

Integral Variables

The variable mnemonics are constructed in the following way.

- The first letter is always I for INTEGRAL VARIABLE.
- The second letter specifies quantity.
- The third letter is OPTIONAL and specifies the normalizing quantity.

First and Second Letter

Class	Quantity	Description
I	N	for number of moles
I	W	for mass
I	V	for volume
I	U	for number of moles of volume-contributing elements
II	S	for entropy
II	H	for enthalpy
II	G	for Gibbs energy
II	A	for Helmholtz energy

Class I

Integral quantities of CLASS=I may take 0-3 arguments.

The arguments MUST be given in 'falling' order of significance.

1. Region name
2. Phase name
3. Component name

Class II

Integral quantities of CLASS=II may take 0-2 arguments.

The arguments MUST be given in 'falling' order of significance.

1. Region name

2. Phase name

Third Letter - Normalizing Quantity

Quantity	Description
N	for total number of moles in system
W	for total mass of system
V	for total volume of system
U	for total number of moles of volume-contributing elements in system

Examples

- IW(PEARLITE,BCC,CR) is the mass of CR in the BCC phase in region PEARLITE.
- IW(PEARLITE,BCC) is the mass of BCC phase in region PEARLITE. IW is the total mass in the system.
- IVV(Austenite) is the volume fraction of Austenite in a single cell calculation.
- IVV(2, Austenite) is the volume fraction of Austenite in Cell 2 for a multi cell calculation.

Auxiliary Variables

The following table lists auxiliary variables that are associated with both names and mnemonics.

Interface names are the same as the region names with the addition that you also have to specify if the interface is at the U(pper) or L(ower) end of the regions.

Name and Mnemonic Auxiliary Variables

Name	Mnemonic	Argument
ACTIVITY	AC(component)	component
DISTANCE		
FLUX	JV(component)	component
LAMELLAR-SPACING	LS(name)	region name
LAMELLAR-THICKNESS	LT(name)	phase name
MOLE-FRACTION	X(component)	component
MOLE-PERCENT		component
POSITION-OF-INTERFACE	POI(name,U/L)up*)	interface name

Name	Mnemonic	Argument
TEMPERATURE-KELVIN	T	
TEMPERATURE-CELSIUS	-	
U-FRACTION	UF(component)	component
VELOCITY-OF-INTERFACE	VOI(name,U/L)up*)	interface name
WEIGHT-FRACTION	W(component)	component
WEIGHT-PERCENT		component

Mnemonic Auxiliary Variables

There are also auxiliary variables that are only associated with a mnemonic.

Mnemonic	Description
GD	global distance
LD	local distance
M(phase,J)	mobility coefficient where J=diffusing specie
LOGM(phase,J)	^{10}log of the mobility coefficient
DT(phase,J)	tracer diffusion coefficient where J=diffusing specie
LOGDT(phase,J)	10log of the tracer diffusion coefficient
DC(phase,J,K,N)	chemical diffusion coefficient where K=gradient specie, and N=reference specie
LOGDC(phase,J,K,N)	^{10}log of chemical diffusion coefficient
DI(phase,J,K,N)	intrinsic diffusion coefficient
LOGDI(phase,J,K)	^{10}log of intrinsic diffusion coefficient
JV(phase and/or specie)	flux in volume fixed frame of reference
JL(phase and/or specie)	flux in lattice fixed frame of reference
QC(phase,J,K,N)	Q=R(ln (DC{T1}) - ln (DC{ T1+e}))/(1/(T1+e) - 1/ T1)
QT(phase,J)	Q=R(ln (DT{T1}) - ln (DT{T1+e}))/(1/(T1+e) - 1/ T1)
QI(phase,J,K,N)	Q=R(ln (DI{T1}) - ln (DI{T1+e}))/(1/(T1+e) - 1/T1)

Mnemonic	Description
FC(phase,J,K,N)	D0=exp(ln (DC{T1})+Q/R/ T1)
FT(phase,J)	D0=exp(ln (DT{T1})+Q/R/T1)
FI(phase,J,K,N)	D0=exp(ln (DI{T1})+Q/R/T1)

Diffusion Module (DICTRA) Examples Guide

Thermo-Calc Software

Version 2016a

About the Diffusion Module (DICTRA) Examples Guide

This guide is available as a PDF.

For descriptions of each example see *Diffusion Module (DICTRA) Examples Collection* on page 6.

Installed Examples

The PDF and example files are also included with your installation in a folder location based on the operating system.

On Windows, once Thermo-Calc is installed, you can also locate the Examples and Materials folders, plus all the Manuals using the shortcuts located in the Start menu. Go to **Start → All Programs >Thermo-Calc** and click **Examples**, **Manuals**, or **Materials** as required to open the applicable folder.

In the table, *<user>* stands for the username and *<version>* for the version of Thermo-Calc, for example 2016a.

OS	User type	Default directory
Windows	Normal user	Users*<user>*\Thermo-Calc*<version>* Users*<user>*\Documents\Thermo- Calc*<version>* My documents
	Administrator	Program Files\Thermo-Calc*<version>* Users\Public\Documents\Thermo- Calc*<version>* Public documents
Mac	Administrator (user name and password required)	Examples and manuals in /Users/Shared/Thermo-Calc/*<version>* To go to this folder, in Finder, from the **Go** main menu select **Go to folder**. Enter the above file path and click **Go**.
Linux	Non root user	home/*<user>*/Thermo-Calc/*<version>*
	Root user	usr/local/Thermo-Calc/*<version>*

Download from the Website

You can also download the PDF and the example files from the website

1. Go to http://www.thermocalc.com/.

2. From the menu, select **Support →Documentation**.
3. Click **Examples** to expand the list.
4. To the right of the example files you want to download, click **Read More**.
5. Click the links to either open a PDF in a browser or to download zipped files containing the examples.

Diffusion Module (DICTRA)
Examples Guide
Version 2016a

exa1

One-phase problem. Simple homogenization of a binary Fe-Ni alloy. It is assumed there is initially a linear Ni-concentration profile.

exa2a

One-phase problem. Simple homogenization of a binary Fe-Ni alloy. We have put together a Ni rich and a Ni lean alloy.

exa2b

One-phase problem. Simple homogenization of a binary Fe-Ni alloy. We have put together a Ni rich and a Ni lean alloy. This example is identical to a2a. However, in this example implicit time integration is used instead of the trapetzoidal method for solving the PDEs.

exa3

One-phase problem. Setup file for the simulation of uphill diffusion in a ternary single phase austenite matrix due to the classical darken experiment published by L.S. Darken: Trans. Aime, v.180 (1949), pp. 430-438. In this setup two pieces of austenite (3.80 wt%Si, 0.49 wt%C) and (0.05 wt%Si, 0.45 wt%C) are put together and are subsequently annealed at 1050 C for 13 days. As both pieces are austenite they must be entered into the same region. we can accomplish this by giving the compositions of Si and C in each gridpoint individually. For convenience we store these data on file.

exa4

One-phase problem. This is a simple binary simulation with just one single phase region. We will compare our numerical simulation with an analytical erf-solution. For this reason a special database erf.tdb is created in which the diffusion coefficient is set to a concentration independent value.

exa5

One-phase problem. Simulation of carburization of a binary Fe-0.15 wt% C alloy. A mixture of 40% N2 and 60% cracked methanol was used as carrier gas. The carburizing "carbon potential" in the gas is 0.85 wt%. A surface reaction controles the flux of C at the surface.

exa6

One-phase problem. A simple, perhaps not realistic, example on diffusion through a tube wall. The tube-material is an Fe-0.6%Mn-0.7%Si-0.05%C alloy. On the inside wall a carbon activity of 0.9 is maintained whereas on the outside the C-activity is very low. This example demonstrates the use of the command SET-FIRST-INTERFACE as well as the use of MIXED boundary conditions.

exa7

DICTRA simulation of a homogenization heat treatment

exb1a

Moving boundary problems. Setup file for calculating a ferrite(BCC)/austenite(FCC) transformation in a binary Fe-C alloy. The initial state is an austenite of 2 mm thichness. The composition of the austenite is Fe-0.15wt%C. After austenitisation the specimen has been quenched down to 1050K. The system is assumed closed, so we do not set any boundary conditions (closed system is default). Ferrite is expected to grow into the austenite. For this reason we start with a thin region with ferrite adjacent to the austenite.

exb1b

Moving boundary problem. Same problem as in exb1a but now we set up the problem with ferrite as an inactive phase adjacent to the initial austenite.

exb1c

Moving boundary problems. Same problem as in exb1a and exb1b but now we start the simulation at a higher temperature and we assume a gradual cooling down to 1050 K. When we reach 1050 K we keep the temperature constant and thus have an isothermal transformation. As in exb1b we have ferrite as an inactive phase adjacent to the initial austenite.

exb2

Moving boundary problems. Setup file for calculating the dissolution of a spherical cementite particle in an austenite matrix. This case is from Z.-K. Liu, L. Höglund, B. Jönsson and J. Ågren: Metall. Trans.A, v.22A (1991), pp. 1745-1752.

exb3

exb3_setup.DCM

exb4a

Moving boundary problems. This examples demonstrates the solidification path of an Fe-18%Cr-8%Ni alloy. A eutectic reaction is assumed, LIQUID -> BCC + FCC. Hence the BCC and FCC regions should be on separate sides of the liquid region. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.

exb4b

Moving boundary problems. The same as exb4a but now a peritectic reaction is assumed, LIQUID + BCC -> FCC. Hence the FCC region should appear inbetween the LIQUID and the BCC. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.

exb4c

Moving boundary problems. Same as exb4b but, now the diffusivity data is amended for the LIQUID and a very high value for the diffusivity is used in order to simulate a case where we assume that the composition in the LIQUID is always homogeneous. This case should be considered less realistic than exb4b. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.

exb4d

Moving boundary problems. Same as exb4b but instead of controlling the temperature the amount of heat extracted is given. Comparison is made with both a Scheil-Gulliver simulation and equilibrium solidification conditions, both made with Thermo-Calc.

exb5

Moving boundary problems. This example demonstrates the evaluation of a ternary Fe-Cr-Ni diffusion couple. A thin slice of ALPHA phase (38%Cr, 0%Ni) is clamped between two thicker slices of GAMMA phase (27%Cr, 20%Ni). The assembly is subsequently heat treated at 1373K. This setup corresponds to diffusion couple A in M. Kajihara, C.-B. Lim and M. Kikuchi: ISIJ International 33 (1993), pp. 498-507. See also M. Kajihara and M. Kikichi: Acta Metall.Mater. 41 (1993), pp.2045-2059.

exb6

Moving boundary problems. This example illustrates the effect of microsegregation of phosphorus during peritectic solidification in steel.

exb7

Moving boundary problems. This example shows how to enter dispersed phases on either side of a phase interface. The particular case shows how the kinetics of a ferrite to austenite transformation is affected by simultaneous precipitation of niobium carbide. The transformation is caused by carburization.

exc1

Cell calculations. This example simulates what happens to a FERRITE plate that has inherited the carbon content of its parent AUSTENITE. The FERRITE plate formed is embedded in an AUSTENITE matrix. This setup corresponds to a proposed mechanism for formation of WIDMANNSTÄTTEN FERRITE or for the FERRITE phase of the BAINITE structure. It is assumed that the phase boundary between FERRITE and AUSTENITE is immobile, this is achieved in the simulation by putting the FERRITE and the AUSTENITE in two different cells. See also M. Hillert, L. Höglund and J. Ågren: Acta Metall. Mater. 41 (1993), pp.1951-1957.

exc2

Cell calculations. Setup file for calculating the dissolution of CEMENTITE particles in an AUSTENITE matrix. This case is identical to exb2 except that we here have three different particle sizes. Altogether six particles are considered using three different cells. This is in order to represent some size distribution among the CEMENTITE particles. See also Z.-K. Liu, L. Höglund, B. Jönsson and J. Ågren: Metall.Trans.A, v. 22A (1991), pp. 1745-1752.

exd1a

Diffusion in dispersed systems. Setup file for carburization of a Ni-25Cr alloy. In this case the M3C2 and M7C3 carbides are entered as spheroid phases in a FCC matrix. This case is from A. Engström, L. Höglund and J. Ågren: Metall.Trans.A v. 25A (1994), pp. 1127-1134. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. In this example the DISPERSED SYSTEM MODEL is used, which requires that the default HOMOGENIZATION MODEL is disabled. With the DISPERSED SYSTEM MODEL the command ENTER_LABYRINTH_FUNCTION is used to take into account the impeding effect of dispersed phases on long-range diffusion. For the HOMOGENIZATION MODEL the command ENTER_HOMOGENIZATION_FUNCTION should be used.

exd1b

Diffusion in dispersed systems. Setup file for carburization of a Ni-25Cr alloy. In this case the M3C2 and M7C3 carbides are entered as spheroid phases in a FCC matrix. This case is from A. Engström, L. Höglund and J. Ågren: Metall.Trans. A, v.25A (1994), pp. 1127-1134. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. The default HOMOGENIZATION MODEL is used and then ENTER_HOMOGENIZATION_FUNCTION should be used instead of ENTER_LABYRINTH_FUNCTION.

exd2a

Diffusion in dispersed systems. Setup file for calculating the interdiffusion in a diffusion couple between a two-phase (FCC+BCC) and a single-phase (FCC) Fe-Ni-Cr alloy.@@ This case is from A. Engström: Scand. J. Met., v. 24, 1995, pp.12-20. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. In this example the DISPERSED SYSTEM MODEL is used, which requires that the default HOMOGENIZATION MODEL is disabled. With the DISPERSED SYSTEM MODEL the command ENTER_LABYRINTH_FUNCTION is used to take into account the impeding effect of dispersed phases on long-range diffusion. For the HOMOGENIZATION MODEL the command ENTER_HOMOGENIZATION_FUNCTION should be used.

exd2b

Diffusion in dispersed systems. Setup file for calculating the interdiffusion in a diffusion couple between a two-phase (FCC+BCC) and a single-phase (FCC) Fe-Ni-Cr alloy. This case is from A. Engström: Scand. J. Met., v. 24, 1995, pp.12-20. This simulation can be run with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL. Here the default HOMOGENIZATION MODEL is used and then ENTER_HOMOGENIZATION_FUNCTION should be used instead of ENTER_LABYRINTH_FUNCTION.

exd3

Diffusion in dispersed systems. This example shows the use of the homogenization model. It is taken from H. Larsson and A. Engström, Acta. Mater. v.54 (2006), pp. 2431-2439. Experimental data from A. Engström, Scand J Metall, v.243 (1995), p.12. The homogenization model can be used for multiphase simulations like the dispersed system model, but unlike the dispersed system model there is no need to have a single continuous matrix phase and, furthermore, there is no need to limit the size of time-steps. The set-up is performed in the same manner as for the dispersed system model, which means that a certain phase is entered as the matrix phase and the other phases are entered as spheroidal, but the choice of matrix phase will not affect the simulation.

exe1

Cooperative growth. This is an example file for a setup and a calculation of PEARLITE growth in an Fe-0.50wt%C - 0.91wt%Mn steel.

exf1

Coarsening. Setup file for calculating the Ostwald-ripening of a spherical M6C carbide in an AUSTENITE matrix.

exg1

Kinetic data. A file for checking mobilities and diffusivities in an Fe-Ni alloy.

exg2

Kinetic data. A file for reading thermodynamic data and setting up the kinetic parameters which are needed for an optimization of the FCC phase in the binary Ni-Al system. See also A. Engström and J. Ågren: ("Assessment of Diffusional Mobilities in Face-Centered Cubic Ni-Cr-Al Alloys" in Z. METALLKUNDE, Feb. 1996).

exh1

Deviation from local equilibrium. Setup file for calculating the growth of FERRITE into AUSTENITE with a limited interface mobility. This is achieved by adding a Gibbs-energy contribution to the FERRITE using the SET-SURFACE-ENERGY command.

exh2

Deviation from local equilibrium Setup file for calculation of the growth of FERRITE into AUSTENITE in an Fe-2.02%Ni-0.0885%C alloy using the para-equilibrium model. The results are compared with experimental information from Hutchinson, C. R., A. Fuchsmann, and Yves Brechet. "The diffusional formation of ferrite from austenite in Fe-C-Ni alloys." Metall. Mat. Trans. A 35.4 (2004): 1211-1221.

exh3

Deviation from local equilibrium. This calculation shows how a temperature gradient induces diffusion.

exi1

Diffusion in complex phases. Diffusion in including effects from chemical ordering. The datafile AlFeNi-data.TDB contains both a thermodynamic and kinetic description for the ordered and disordered BCC.

exi2

Diffusion in complex phases. This example demonstrates the use of the model for calculation of diffusion through a stoichiometric phase. The flux of a component in the stoichiometric phase is assumed to be proportional to the difference in chemical potential at each side of the stoichiometric phase multiplied with the mobility for the component in the phase. The mobility is assessed from experimental information and is basically the tracer diffusivity for the component. This calculation is compared with experimental data where a sample of pure iron has been exposed to a gas atmosphere with a certain carbon activity. The weight gain is then measured as a function of time. The experimental data is obtained from Ozturk B., Fearing V. L., Ruth A. Jr. and Simkovich G., Met. Trans A, vol 13A (1982), pp. 1871-1873.

exi3a

Diffusion in complex phases. Oxidation of iron sample and consequent growth of an oxide layer.

exi3b

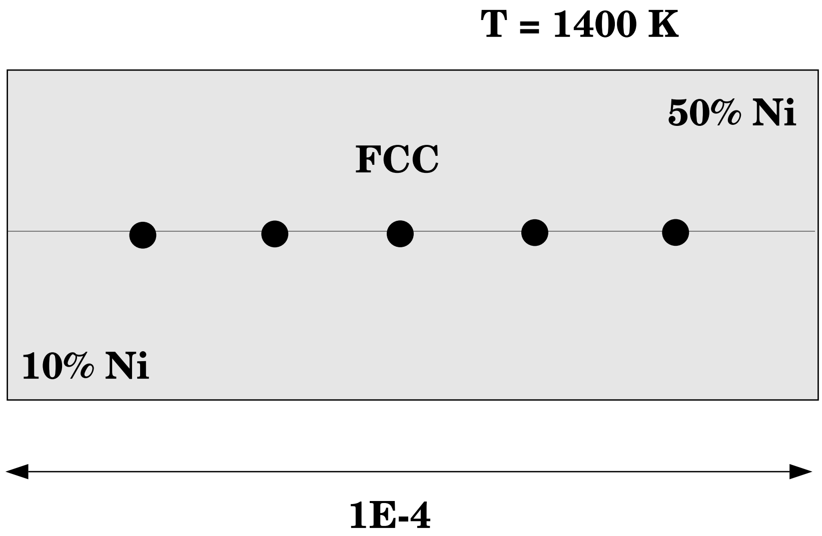
Diffusion in complex phases. Oxidation of iron sample and consequent growth of an oxide layer using the grain boundary diffusion contribution model.

One-phase problems

Example a1

Homogenisation of a binary Fe-Ni alloy

(Initially we assume a linear concentration profile)



Results

exa1-setup

```
SYS:About
 Copyright Foundation for Computational Thermodynamics,
 Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa1\setup.DCM"SYS: @@
SYS: @@ One-phase problem.
SYS: @@ Simple homogenization of a binary Fe-Ni alloy. It is assumed
SYS: @@ there is initially a linear Ni-concentration profile.
SYS: @@
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: goto_module
MODULE NAME: data
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: switch_database
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
```

```
 TCAQ2    =  Aqueous Solution  v2.5
 AQS2     =  TGG Aqueous Solution Database v2.5
 GCE2     =  TGG Geochemical/Environmental TDB v2.3
 PURE5    =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO   =  Aluminum Demo Database
 FEDEMO   =  Iron Demo Database
 NIDEMO   =  Nickel Demo Database
 SLDEMO   =  Solder Demo Database
 OXDEMO   =  Oxide Demo Database
 SUBDEMO  =  Substance Demo Database
 PTERN    =  Public Ternary Alloys TDB v1.3
 PAQ2     =  Public Aqueous Soln (SIT) TDB v2.4
 PG35     =  G35 Binary Semi-Conductors TDB v1.2
 MOB2     =  Alloys Mobility  v2.4
 MOB1     =  Alloys Mobility  v1.3
 MOBFE1   =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2   =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3   =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4   =  Ni-Alloys Mobility  v4.0
 MOBNI3   =  Ni-Alloys Mobility  v3.1
 MOBNI2   =  Ni-Alloys Mobility  v2.4
 MOBNI1   =  Ni-Alloys Mobility  v1.0
 MOBAL3   =  Al-Alloys Mobility  v3.0
 MOBAL2   =  Al-Alloys Mobility  v2.0
 MOBAL1   =  Al-Alloys Mobility  v1.0
 MOBCU1   =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1   =  Mg-Alloys Mobility  v1.0
 MOBSI1   =  Si-Alloys Mobility  v1.0
 MOBTI1   =  Ti-Alloys Mobility  v1.0
 MALDEMO  =  Al-Alloys Mobility demo database
 MFEDEMO  =  Fe-Alloys Mobility demo database
 MNIDEMO  =  Ni-Alloys Mobility demo database
 USER     =  User defined Database

DATABASE NAME /TCFE8/: fedemo
 Current database: Iron Demo Database

 VA                       /-   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: @@
TDB_FEDEMO: define_system
ELEMENTS: fe ni
 FE                       NI   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: @@
TDB_FEDEMO: reject
ELEMENTS, SPECIES, PHASES, CONSTITUENT OR SYSTEM: /PHASES/: phase
PHASES: *
 LIQUID:L                 BCC_A2                   FCC_A1
 HCP_A3                   LAVES_PHASE_C14  REJECTED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RESTORE THE THERMODYNAMIC DATA FOR THE FCC PHASE
TDB_FEDEMO: @@
TDB_FEDEMO: restore
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /ELEMENTS/: phase
PHASES: fcc
 FCC_A1  RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: @@
TDB_FEDEMO: get_data
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_FEDEMO: @@
TDB_FEDEMO: append_database
 Use one of these databases

 TCFE8    =  Steels/Fe-Alloys  v8.0
 TCFE9    =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1   =  FROST database v1.0
 TCFE7    =  Steels/Fe-Alloys  v7.0
 TCFE6    =  Steels/Fe-Alloys  v6.2
 TCFE5    =  Steels/Fe-Alloys  v5.0
 TCFE4    =  Steels/Fe-Alloys  v4.1
 TCFE3    =  Steels/Fe-Alloys  v3.1
 TCFE2    =  Steels/Fe-Alloys  v2.1
 TCFE1    =  Steels/Fe-Alloys  v1.0
 FEDAT    =  TCS/TT Steels Database v1.0
 TCNI9    =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8    =  Ni-Alloys  v8.0
 TCNI7    =  Ni-Alloys  v7.1
 TCNI6    =  Ni-Alloys  v6.0
 TCNI5    =  Ni-Alloys  v5.1
 TCNI4    =  Ni-Alloys  v4.0
 TCNI1    =  Ni-Alloys  v1.3
 TCAL4    =  Al-Alloys  v4.0
 TCAL3    =  Al-Alloys  v3.0
 TCAL2    =  Al-Alloys  v2.0
 TCAL1    =  Al-Alloys  v1.2
 TCMG5    =  Mg-Alloys  v5.0 SNAPSHOT
```

```
 TCMG4    =  Mg-Alloys  v4.0
 TCMG3    =  Mg-Alloys  v3.0
 TCMG2    =  Mg-Alloys  v2.0
 TCMG1    =  Mg-Alloys  v1.1
 TCTI1    =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1    =  Copper  v1.0 SNAPSHOT
 TCCC1    =  Cemented carbide  v1.0
 TCHEA1   =  High Entropy Alloy  v1.0
 SSOL5    =  SGTE Alloy Solutions Database v5.0
 SSOL4    =  SGTE Alloy Solutions Database v4.9f
 SSOL2    =  SGTE Alloy Solutions Database v2.1
 SSUB5    =  SGTE Substances Database v5.1
 SSUB4    =  SGTE Substances Database v4.1
 SSUB3    =  SGTE Substances Database v3.3
 SSUB2    =  SGTE Substances Database v2.2
 SNOB3    =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2    =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1    =  SGTE Nobel Metal Alloys Database v1.2
 STBC2    =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1    =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1    =  SGTE Molten Salts Database v1.2
 SNUX6    =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2    =  TC Semi-Conductors  v2.1
 SLAG4    =  Fe-containing Slag  v4.0 snapshot
 SLAG3    =  Fe-containing Slag  v3.2
 SLAG2    =  Fe-containing Slag  v2.2
 SLAG1    =  Fe-containing Slag  v1.2
 TCOX7    =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6    =  Metal Oxide Solutions  v6.1
 TCOX5    =  Metal Oxide Solutions  v5.1
 TCOX4    =  Metal Oxide Solutions  v4.1
 ION3     =  Ionic Solutions  v3.0
 ION2     =  Ionic Solutions  v2.6
 ION1     =  Ionic Solutions  v1.5
 NOX2     =  NPL Oxide Solutions Database v2.1
 TCSLD3   =  Solder Alloys  v3.1
 TCSLD2   =  Solder Alloys  v2.0
 TCSLD1   =  Solder Alloys  v1.0
 TCSI1    =  Ultrapure Silicon  v1.1
 TCMP2    =  Materials Processing  v2.5
 TCES1    =  Combustion/Sintering  v1.1
 TCSC1    =  Super Conductor  v1.0
 TCFC1    =  SOFC Database v1.0
 TCNF2    =  Nuclear Fuels  v2.1b
 NUMT2    =  Nuclear Materials  v2.1
 NUOX4    =  Nuclear Oxides  v4.2
 NUTO1    =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1    =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10   =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11   =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2    =  Aqueous Solution  v2.5
 AQS2     =  TGG Aqueous Solution Database v2.5
 GCE2     =  TGG Geochemical/Environmental TDB v2.3
 PURE5    =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO   =  Aluminum Demo Database
 FEDEMO   =  Iron Demo Database
 NIDEMO   =  Nickel Demo Database
 SLDEMO   =  Solder Demo Database
 OXDEMO   =  Oxide Demo Database
 SUBDEMO  =  Substance Demo Database
 PTERN    =  Public Ternary Alloys TDB v1.3
 PAQ2     =  Public Aqueous Soln (SIT) TDB v2.4
 PG35     =  G35 Binary Semi-Conductors TDB v1.2
 MOB2     =  Alloys Mobility  v2.4
 MOB1     =  Alloys Mobility  v1.3
 MOBFE1   =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2   =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3   =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4   =  Ni-Alloys Mobility  v4.0
 MOBNI3   =  Ni-Alloys Mobility  v3.1
 MOBNI2   =  Ni-Alloys Mobility  v2.4
 MOBNI1   =  Ni-Alloys Mobility  v1.0
 MOBAL3   =  Al-Alloys Mobility  v3.0
 MOBAL2   =  Al-Alloys Mobility  v2.0
 MOBAL1   =  Al-Alloys Mobility  v1.0
 MOBCU1   =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1   =  Mg-Alloys Mobility  v1.0
 MOBSI1   =  Si-Alloys Mobility  v1.0
 MOBTI1   =  Ti-Alloys Mobility  v1.0
 MALDEMO  =  Al-Alloys Mobility demo database
 MFEDEMO  =  Fe-Alloys Mobility demo database
 MNIDEMO  =  Ni-Alloys Mobility demo database
 USER     =  User defined Database

DATABASE NAME /FEDEMO/: mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: define_system
ELEMENTS: fe ni
 FE                       NI  DEFINED
APP: reject
ELEMENTS, SPECIES, PHASES, CONSTITUENT OR SYSTEM: /PHASES/: phase
PHASES: *
 BCC_A2                   FCC_A1  REJECTED
APP:
APP: restore
ELEMENTS, SPECIES, PHASES OR CONSTITUENTS: /ELEMENTS/: phase
PHASES: fcc
 FCC_A1  RESTORED
APP:
APP: get_data
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
 -OK-
```

```
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: goto_module
MODULE NAME: dictra_monitor
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set_condition

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: global
VARIABLE : T
LOW TIME LIMIT /0/: 0
T(TIME,X)= 1400;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC> @@
DIC> @@ WE START BY ENTERING A REGION
DIC> @@
DIC> enter_region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID INTO THE REGION
DIC> @@
DIC> @@ FOR SIMPLICITY WE USE AN EQUIDISTANT GRID
DIC> @@
DIC> enter_grid_coordinates
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASE INTO REGION
DIC> @@
DIC> enter_phase_in_region
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL NI COMPOSITION INTO THE PHASE. WE ASSUME A LINEAR
DIC> @@ VARIATION IN THE REGION.
DIC> @@
DIC> enter_compositions
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: weight_percent
PROFILE FOR /NI/: ni
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 10
VALUE OF LAST POINT : /10/: 50
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set_simulation_time
END TIME FOR INTEGRATION /.1/: 1E6
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON FILE
DIC> @@
DIC> save_workspaces exa1 Y
DIC>
DIC> set_interactive
 --OK---
DIC>
```

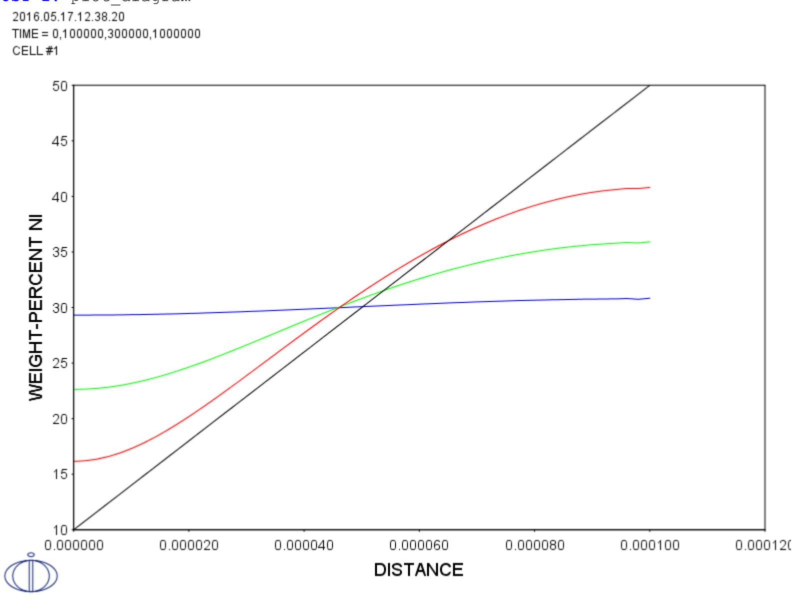
exa1-run

```
DIC>About NO SUCH COMMAND, USE HELP
DIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa1\run.DCM"DIC>
DIC>
DIC> @@ exa1_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE a1
DIC> @@
DIC> goto_module
MODULE NAME: dictra_monitor
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> read_workspaces exa1
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> simulate_reaction
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   1360.6038     DT =   1360.2037     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   4081.0112     DT =   2720.4074     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   9521.8259     DT =   5440.8148     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   20403.455     DT =   10881.630     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   42166.714     DT =   21763.259     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   85693.233     DT =   43526.518     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   172746.27     DT =   87053.036     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   272746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908541  NI = .290320144091459
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   372746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091459
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   472746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   572746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
```

```
 CPU time used in timestep                      0  seconds

TIME =   672746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091458
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                      0  seconds

TIME =   772746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .709679855908541  NI = .290320144091459
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                      0  seconds

TIME =   872746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .709679855908541  NI = .290320144091459
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                      0  seconds

TIME =   972746.27     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .709679855908542  NI = .290320144091459
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                      0  seconds

TIME =  1000000.00     DT =   27253.731     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .709679855908541  NI = .290320144091459
TOTAL SIZE OF SYSTEM:  1E-04 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.40010010
DELETING TIME-RECORD FOR TIME    1360.6038
DELETING TIME-RECORD FOR TIME    4081.0112
DELETING TIME-RECORD FOR TIME    9521.8259
DELETING TIME-RECORD FOR TIME    20403.455
DELETING TIME-RECORD FOR TIME    42166.714
DELETING TIME-RECORD FOR TIME    85693.233
DELETING TIME-RECORD FOR TIME    172746.27
DELETING TIME-RECORD FOR TIME    272746.27
DELETING TIME-RECORD FOR TIME    372746.27
DELETING TIME-RECORD FOR TIME    472746.27
DELETING TIME-RECORD FOR TIME    572746.27
DELETING TIME-RECORD FOR TIME    672746.27
DELETING TIME-RECORD FOR TIME    772746.27
DELETING TIME-RECORD FOR TIME    872746.27

KEEPING TIME-RECORD FOR TIME    972746.27
AND FOR TIME                   1000000.00
WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set_interactive
 --OK---
DIC>
```

exa1-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa1\plot.DCM"
DIC>
DIC> @@ exa1_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE a1
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> goto_module
MODULE NAME: dictra_monitor
 TIME STEP AT TIME  1.00000E+06
DIC> read_workspaces exa1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post_processor

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT SOME CONCENTRATION PROFILES
POST-1: @@
POST-1: set_diagram_axis
AXIS (X, Y OR Z) : x
VARIABLE : distance
  INFO: Distance is set as independent variable
DISTANCE : /GLOBAL/: global
POST-1:
POST-1: set_diagram_axis
AXIS (X, Y OR Z) : y
VARIABLE : weight-percent
FOR COMPONENT : ni
POST-1:
POST-1: set_plot_condition
CONDITION /TIME/: time
VALUE(S) /LAST/: 0 1e5 3e5 1e6
POST-1:
POST-1: plot_diagram
```

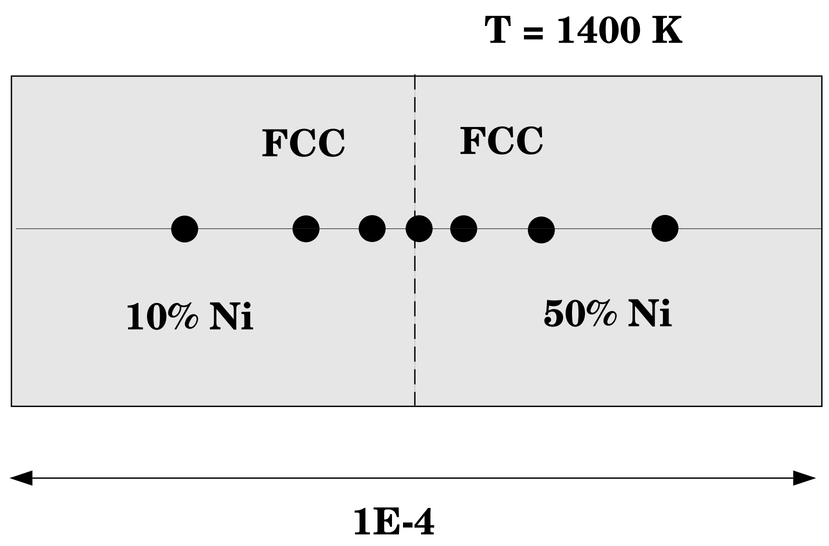



```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: set_interactive
 --OK---
POST-1:
```

Example a2

Homogenisation of a binary Fe-Ni alloy

(Initially we have a step-profile)



exa2a-setup

```
SYS:About Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa2a\setup.DCM"SYS: @@
SYS: @@ One-phase problem.
SYS: @@ Simple homogenization of a binary Fe-Ni alloy. We have put together a
SYS: @@ Ni rich and a Ni lean alloy.
SYS: @@
SYS:
SYS: @@ exa2a_setup.DCM
SYS:
SYS: @@
SYS: @@ IN exa1 WE WROTE OUT ALL THE COMMANDS IN FULL BUT FROM NOW ON WE WILL
SYS: @@ START USING ABBREVIATED COMMANDS.
SYS: @@
SYS:
SYS: @@
SYS: @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
SYS: @@
SYS: set_log_file setup
Heading:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
   ... the command in full is GOTO_MODULE
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
   ... the command in full is SWITCH_DATABASE
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: @@
TDB_FEDEMO: def-system fe ni
   ... the command in full is DEFINE_SYSTEM
 FE                       NI  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: @@
TDB_FEDEMO: rej ph * all
   ... the command in full is REJECT
 LIQUID:L                 BCC_A2                   FCC_A1
 HCP_A3                   LAVES_PHASE_C14  REJECTED
TDB_FEDEMO: res ph fcc
   ... the command in full is RESTORE
 FCC_A1  RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: @@
TDB_FEDEMO: get
   ... the command in full is GET_DATA
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
   ... the command in full is AMEND_PHASE_DESCRIPTION
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
   ... the command in full is APPEND_DATABASE
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
```

```
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe ni
   ... the command in full is DEFINE_SYSTEM
 FE                      NI  DEFINED
APP: rej ph * all
   ... the command in full is REJECT
 BCC_A2                  FCC_A1  REJECTED
APP: res ph fcc
   ... the command in full is RESTORE
 FCC_A1  RESTORED
APP: get
   ... the command in full is GET_DATA
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data
```

```
  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
   ... the command in full is GOTO_MODULE
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond
   ... the command in full is SET_CONDITION

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: glob
VARIABLE : T
LOW TIME LIMIT /0/: 0
T(TIME,X)= 1400;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC> @@
DIC> @@ WE START BY ENTERING A REGION
DIC> @@
DIC> enter-region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID INTO THE REGION.
DIC> @@ IN THIS CASE WE WANT SEVERAL POINTS IN THE MIDLE OF THE REGION,
DIC> @@ THUS WE CONSTRUCT A DOUBLE GEOMETRIC GRID.
DIC> @@
DIC> enter-grid
   ... the command in full is ENTER_GRID_COORDINATES
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 60
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.9
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.11
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
   ... the command in full is ENTER_PHASE_IN_REGION
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL NI COMPOSITION INTO THE PHASE. READ DATA FROM
DIC> @@ THE FILE ni.dat WHICH CONTAINS THE NI-PROFILE.
DIC> @@
DIC> enter-composition
   ... the command in full is ENTER_COMPOSITIONS
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /NI/: ni
TYPE /LINEAR/: read a2ani.dat
   ... the command in full is CREATE_NEW_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
```

```
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1E6
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON FILE
DIC> @@
DIC> save exa2a Y
   ... the command in full is SAVE_WORKSPACES
DIC>
DIC> set-inter
   ... the command in full is SET_INTERACTIVE
 --OK---
DIC>
```

exa2a-run

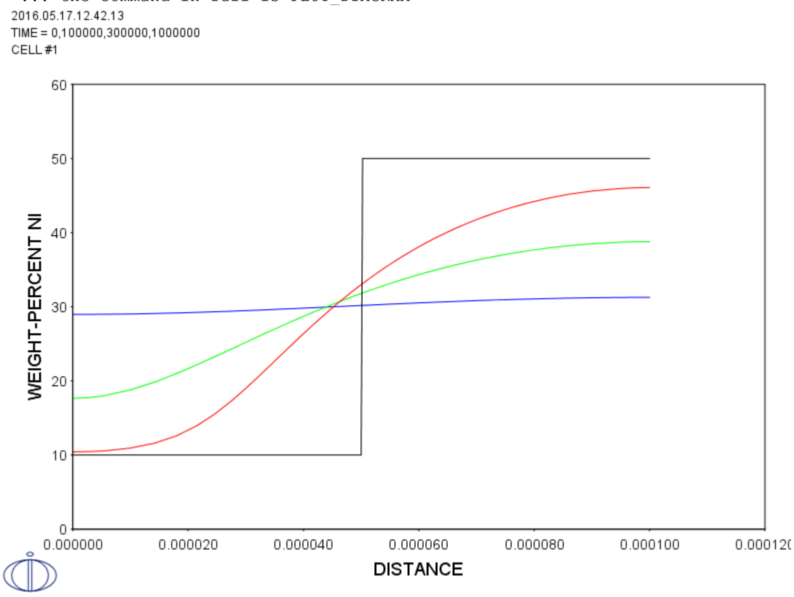
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa2a\run.DCM"
DIC>
DIC> @@ exa2a_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE a2a
DIC> @@
DIC>
DIC> @@
DIC> @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
DIC> @@
DIC> @@set-log-file run
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AN READ SETUP FROM FILE
DIC> @@
DIC> go d-m
   ... the command in full is GOTO_MODULE
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> read exa2a
   ... the command in full is READ_WORKSPACES
   ... the command in full is DEFINE_COMPONENTS
   ... the command in full is SELECT_EQUILIBRIUM
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> simulate
   ... the command in full is SIMULATE_REACTION
   ... the command in full is SET_NUMERICAL_LIMITS
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =   0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =   0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    10.408313     DT =   10.008213     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    30.424739     DT =   20.016426     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    70.457590     DT =   40.032851     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    150.52329     DT =   80.065703     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    310.65470     DT =   160.13141     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    630.91751     DT =   320.26281     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    1271.4431     DT =   640.52562     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    2552.4944     DT =   1281.0512     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =    5114.5969     DT =   2562.1025     SUM OF SQUARES =   0.0000000
```

```
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   10238.802     DT =   5124.2050     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   20487.212     DT =   10248.410     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .291111754593831
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   40984.032     DT =   20496.820     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   81977.671     DT =   40993.640     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .291111754593829
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   163964.95     DT =   81987.280     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406171  NI = .291111754593828
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   263964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406171  NI = .291111754593829
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   363964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   463964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406172  NI = .291111754593828
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     1  seconds

TIME =   563964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406174  NI = .291111754593825
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   663964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406174  NI = .291111754593826
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   763964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406176  NI = .291111754593824
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   863964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406174  NI = .291111754593825
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =   963964.95     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406173  NI = .291111754593827
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                     0  seconds

TIME =  1000000.00     DT =   36035.049     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406174  NI = .291111754593826
TOTAL SIZE OF SYSTEM:  1E-04 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.40010010
DELETING TIME-RECORD FOR TIME    10.408313
DELETING TIME-RECORD FOR TIME    30.424739
DELETING TIME-RECORD FOR TIME    70.457590
DELETING TIME-RECORD FOR TIME    150.52329
DELETING TIME-RECORD FOR TIME    310.65470
DELETING TIME-RECORD FOR TIME    630.91751
DELETING TIME-RECORD FOR TIME    1271.4431
DELETING TIME-RECORD FOR TIME    2552.4944
DELETING TIME-RECORD FOR TIME    5114.5969
DELETING TIME-RECORD FOR TIME    10238.802
DELETING TIME-RECORD FOR TIME    20487.212
DELETING TIME-RECORD FOR TIME    40984.032
DELETING TIME-RECORD FOR TIME    81977.671
DELETING TIME-RECORD FOR TIME    163964.95
DELETING TIME-RECORD FOR TIME    263964.95
DELETING TIME-RECORD FOR TIME    363964.95
DELETING TIME-RECORD FOR TIME    463964.95
DELETING TIME-RECORD FOR TIME    563964.95
DELETING TIME-RECORD FOR TIME    663964.95
DELETING TIME-RECORD FOR TIME    763964.95
DELETING TIME-RECORD FOR TIME    863964.95
```

```
 KEEPING TIME-RECORD FOR TIME    963964.95
 AND FOR TIME                   1000000.00
 WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
   ... the command in full is SET_INTERACTIVE
 --OK---
DIC>
```

exa2a-plot

```
DIC>About NO SUCH COMMAND, USE HELP
DIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-
Examples\examples\exa2a\plot.DCM"   ... the command in full is MACRO_FILE_OPEN
DIC>
DIC>
DIC> @@ exa2a_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE a2a
DIC> @@
DIC>
DIC> @@
DIC> @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
DIC> @@
DIC> set-log-file plot
 AMBIGUOUS COMMAND, USE HELP
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
   ... the command in full is GOTO_MODULE
 TIME STEP AT TIME  1.00000E+06
DIC>
DIC> read exa2a
   ... the command in full is READ_WORKSPACES
   ... the command in full is DEFINE_COMPONENTS
   ... the command in full is SELECT_EQUILIBRIUM
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post
   ... the command in full is POST_PROCESSOR

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT SOME NI-CONCENTRATION PROFILES
POST-1: @@
POST-1: s-d-a
   ... the command in full is SET_DIAGRAM_AXIS
AXIS (X, Y OR Z) : x
VARIABLE : dist
  INFO: Distance is set as independent variable
   ... the command in full is SET_INDEPENDENT_VARIABLE
DISTANCE : /GLOBAL/: glo
POST-1:
POST-1: s-d-a
   ... the command in full is SET_DIAGRAM_AXIS
AXIS (X, Y OR Z) : y
VARIABLE : weight-percent
FOR COMPONENT : ni
POST-1:
POST-1: s-p-c
   ... the command in full is SET_PLOT_CONDITION
CONDITION /TIME/: time
VALUE(S) /LAST/: 0 1e5 3e5 1e6
POST-1:
POST-1: @@
POST-1: @@ SET SCALING ON Y-AXIS BEFORE PLOTTING
POST-1: @@
POST-1: s-s-s
   ... the command in full is SET_SCALING_STATUS
AXIS (X, Y OR Z) : y
AUTOMATIC SCALING  (Y OR N) /N/: n
MIN VALUE : 0
MAX VALUE : 60
POST-1:
POST-1: plot
   ... the command in full is PLOT_DIAGRAM
```




```
POST-1:
```

```
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: set-inter
   ... the command in full is SET_INTERACTIVE_MODE
 --OK---
POST-1:
```

exa2b-setup

About

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa2b\setup.DCM"SYS: @@
SYS: @@ One-phase problem.
SYS: @@ Simple homogenization of a binary Fe-Ni alloy. We have put together a
SYS: @@ Ni rich and a Ni lean alloy. This example is identical to a2a. However,
SYS: @@ in this example implicit time integration is used instead of the trapetzoidal
SYS: @@ method for solving the PDEs.
SYS: @@
SYS:
SYS: @@ exa2b_setup.DCM
SYS:
SYS: @@
SYS: @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
SYS: @@
SYS: set_log_file setup
Heading:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
   ... the command in full is GOTO_MODULE
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw tcfe7
   ... the command in full is SWITCH_DATABASE
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_TCFE7: @@
TDB_TCFE7: def-system fe ni
   ... the command in full is DEFINE_SYSTEM
 FE                       NI  DEFINED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_TCFE7: @@
TDB_TCFE7: rej ph * all
   ... the command in full is REJECT
 LIQUID:L                 BCC_A2                   FCC_A1
 HCP_A3                   A1_KAPPA                 KAPPA
 LAVES_PHASE_C14          NBNI3                    NI3TI
   REJECTED
TDB_TCFE7: res ph fcc
   ... the command in full is RESTORE
 FCC_A1  RESTORED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_TCFE7: @@
TDB_TCFE7: get
   ... the command in full is GET_DATA
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
   ... the command in full is AMEND_PHASE_DESCRIPTION
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_TCFE7: @@
TDB_TCFE7: app
   ... the command in full is APPEND_DATABASE
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
```

```
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /TCFE7/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni
   ... the command in full is DEFINE_SYSTEM
 FE                       NI  DEFINED
APP: rej ph * all
   ... the command in full is REJECT
 BCC_A2                   FCC_A1                   HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc
   ... the command in full is RESTORE
 FCC_A1  RESTORED
APP: get
   ... the command in full is GET_DATA
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
```

```
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
   ... the command in full is GOTO_MODULE
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond
   ... the command in full is SET_CONDITION

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: glob
VARIABLE : T
LOW TIME LIMIT /0/: 0
T(TIME,X)= 1400;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC> @@
DIC> @@ WE START BY ENTERING A REGION
DIC> @@
DIC> enter-region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID INTO THE REGION.
DIC> @@ AS IN EXAMPLE a2a WE WANT SEVERAL POINTS IN THE MIDLE OF THE REGION,
DIC> @@ THUS WE CONSTRUCT A DOUBLE GEOMETRIC GRID ALSO NI THIS CASE.
DIC> @@
DIC> enter-grid
   ... the command in full is ENTER_GRID_COORDINATES
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 60
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.9
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.11
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
   ... the command in full is ENTER_PHASE_IN_REGION
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL NI COMPOSITION INTO THE PHASE. READ DATA FROM
DIC> @@ THE FILE ni.dat WHICH CONTAINS THE NI-PROFILE.
DIC> @@
DIC> enter-composition
   ... the command in full is ENTER_COMPOSITIONS
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /NI/: ni
TYPE /LINEAR/: read a2bni.dat
   ... the command in full is CREATE_NEW_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
```

```
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
   ... the command in full is COMPUTE_EQUILIBRIUM
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ AFTER THE SIMULATION IN exa2a THE PROFILES HAD SOME FLUCTUATIONS WE
DIC> @@ NOW TRY TO GET RID OF THEM USING IMPLICIT (1) TIME INTEGRATION
DIC> @@ INSTEAD OF THE MORE ACCURATE BUT LESS STABLE TRAPETZOIDAL METHOD
DIC> @@ WHICH IS THE DEFAULT METHOD.
DIC> @@
DIC> s-s-c
   ... the command in full is SET_SIMULATION_CONDITION
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1.0
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/: @@
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/: @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1E6
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON FILE
DIC> @@
DIC> save exa2b Y
   ... the command in full is SAVE_WORKSPACES
DIC>
DIC> set-inter
```

```
   ... the command in full is SET_INTERACTIVE
 --OK---
DIC>
```

exa2b-run

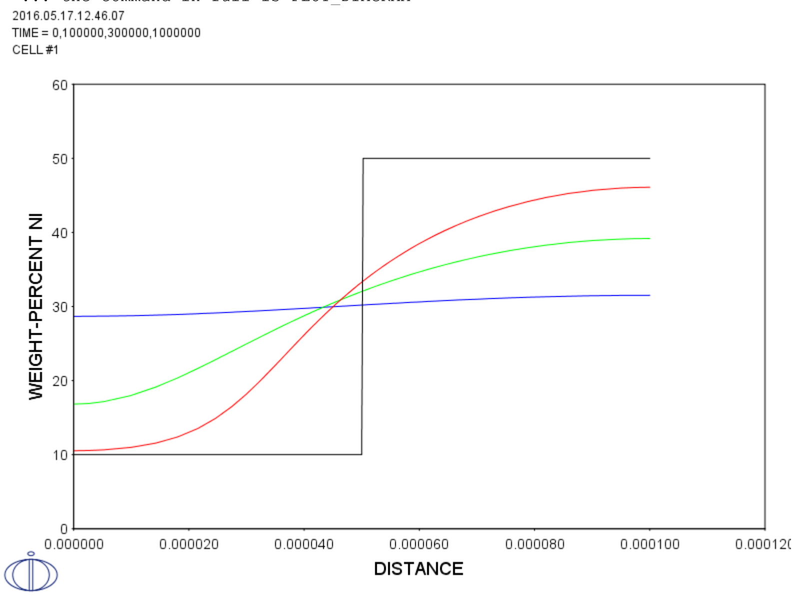
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa2b\run.DCM"
DIC>
DIC> @@ exa2b_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE a2b
DIC> @@
DIC>
DIC> @@
DIC> @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
DIC> @@
DIC> set-log-file run
 AMBIGUOUS COMMAND, USE HELP
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AN READ SETUP FROM FILE
DIC> @@
DIC> go d-m
   ... the command in full is GOTO_MODULE
 TIME STEP AT TIME  0.00000E+00
DIC> read exa2b
   ... the command in full is READ_WORKSPACES
   ... the command in full is DEFINE_COMPONENTS
   ... the command in full is SELECT_EQUILIBRIUM
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> simulate
   ... the command in full is SIMULATE_REACTION
   ... the command in full is SET_NUMERICAL_LIMITS
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   10.403455     DT =   10.003355     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   30.410165     DT =   20.006710     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   70.423585     DT =   40.013420     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   150.45043     DT =   80.026840     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   310.50411     DT =   160.05368     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   630.61147     DT =   320.10736     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   1270.8262     DT =   640.21472     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   2551.2556     DT =   1280.4294     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                      0  seconds

 TIME =   5112.1145     DT =   2560.8589     SUM OF SQUARES =   0.0000000
```

```
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   10233.832     DT =   5121.7178     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .291111754593831
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   20477.268     DT =   10243.436     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   40964.139     DT =   20486.871     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .29111175459383
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   81937.881     DT =   40973.742     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .70888824540617  NI = .291111754593831
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   163885.37     DT =   81947.484     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406169  NI = .291111754593831
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   263885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406172  NI = .291111754593828
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   363885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406167  NI = .291111754593833
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   463885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406166  NI = .291111754593834
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   563885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406164  NI = .291111754593836
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   663885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406166  NI = .291111754593834
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   763885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406165  NI = .291111754593835
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   863885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406166  NI = .291111754593834
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =   963885.37     DT =  100000.000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406166  NI = .291111754593834
TOTAL SIZE OF SYSTEM:  1E-04 [m]

 CPU time used in timestep                    0  seconds

TIME =  1000000.00     DT =   36114.635     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  FE = .708888245406167  NI = .291111754593833
TOTAL SIZE OF SYSTEM:  1E-04 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.40010010
DELETING TIME-RECORD FOR TIME    10.403455
DELETING TIME-RECORD FOR TIME    30.410165
DELETING TIME-RECORD FOR TIME    70.423585
DELETING TIME-RECORD FOR TIME    150.45043
DELETING TIME-RECORD FOR TIME    310.50411
DELETING TIME-RECORD FOR TIME    630.61147
DELETING TIME-RECORD FOR TIME    1270.8262
DELETING TIME-RECORD FOR TIME    2551.2556
DELETING TIME-RECORD FOR TIME    5112.1145
DELETING TIME-RECORD FOR TIME    10233.832
DELETING TIME-RECORD FOR TIME    20477.268
DELETING TIME-RECORD FOR TIME    40964.139
DELETING TIME-RECORD FOR TIME    81937.881
DELETING TIME-RECORD FOR TIME    163885.37
DELETING TIME-RECORD FOR TIME    263885.37
DELETING TIME-RECORD FOR TIME    363885.37
DELETING TIME-RECORD FOR TIME    463885.37
DELETING TIME-RECORD FOR TIME    563885.37
DELETING TIME-RECORD FOR TIME    663885.37
DELETING TIME-RECORD FOR TIME    763885.37
DELETING TIME-RECORD FOR TIME    863885.37
```

```
 KEEPING TIME-RECORD FOR TIME    963885.37
 AND FOR TIME                   1000000.00
 WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
   ... the command in full is SET_INTERACTIVE
 --OK---
DIC>
```

exa2b-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa2b\plot.DCM"DIC>
DIC>
DIC> @@ exa2b_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE a2b
DIC> @@
DIC>
DIC> @@
DIC> @@ LET US DEFINE A LOG-FILE FOR THIS EXAMPLE
DIC> @@
DIC> set-log-file plot
 AMBIGUOUS COMMAND, USE HELP
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
   ... the command in full is GOTO_MODULE
 TIME STEP AT TIME  1.00000E+06
DIC> read exa2b
   ... the command in full is READ_WORKSPACES
   ... the command in full is DEFINE_COMPONENTS
   ... the command in full is SELECT_EQUILIBRIUM
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post
   ... the command in full is POST_PROCESSOR

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT SOME CONCENTRATION PROFILES
POST-1: @@
POST-1: s-d-a
   ... the command in full is SET_DIAGRAM_AXIS
AXIS (X, Y OR Z) : x
VARIABLE : dist
  INFO: Distance is set as independent variable
   ... the command in full is SET_INDEPENDENT_VARIABLE
DISTANCE : /GLOBAL/: glo
POST-1:
POST-1: s-d-a
   ... the command in full is SET_DIAGRAM_AXIS
AXIS (X, Y OR Z) : y
VARIABLE : w-p
FOR COMPONENT : ni
POST-1:
POST-1: s-p-c
   ... the command in full is SET_PLOT_CONDITION
CONDITION /TIME/: time
VALUE(S) /LAST/: 0 1e5 3e5 1e6
POST-1:
POST-1: @@
POST-1: @@ SET SCALING ON Y-AXIS BEFORE PLOTTING
POST-1: @@
POST-1: s-s-s
   ... the command in full is SET_SCALING_STATUS
AXIS (X, Y OR Z) : y
AUTOMATIC SCALING  (Y OR N) /N/: n
MIN VALUE : 0
MAX VALUE : 60
POST-1:
POST-1: plot
   ... the command in full is PLOT_DIAGRAM
```

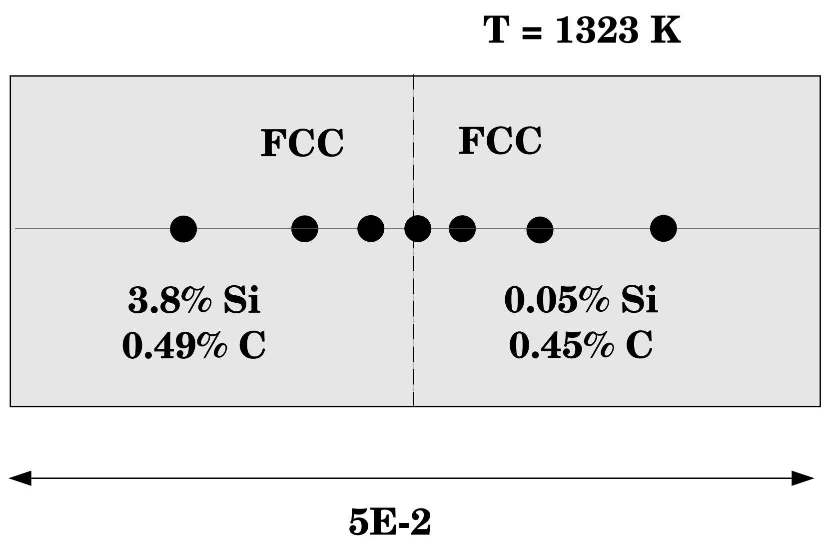



```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
```

```
POST-1: set-inter
   ... the command in full is SET_INTERACTIVE_MODE
 --OK---
POST-1:
```

Example a3

Uphill diffusion in an Fe-Si-C alloy



exa3-setup

```
About Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa3\setup.DCM"SYS: @@
SYS: @@ One-phase problem.
SYS: @@ Setup file for the simulation of uphill diffusion in a ternary single phase
SYS: @@ austenite matrix due to the classical darken experiment published by
SYS: @@ L.S. Darken: Trans. Aime, v.180 (1949), pp. 430-438.
SYS: @@ In this setup two pieces of austenite (3.80 wt%Si, 0.49 wt%C) and
SYS: @@ (0.05 wt%Si, 0.45 wt%C) are put together and are subsequently annealed
SYS: @@ at 1050 C for 13 days. As both pieces are austenite they must be entered
SYS: @@ into the same region. we can accomplish this by giving the compositions
SYS: @@ of Si and C in each gridpoint individually. For convenience we store
SYS: @@ these data on file.
SYS: @@
SYS:
SYS: @@ darken_setup.DCM
SYS:
SYS: @@
SYS: @@ FROM NOW ON WE STOP USING LOG-FILES AS WE USED IN EXAMPLES a2a and a2b
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE7: def-sys fe si c
 FE                       SI                       C
   DEFINED
TDB_TCFE7: rej ph * all
 GAS:G                    LIQUID:L                 BCC_A2
 FCC_A1                   HCP_A3                   DIAMOND_FCC_A4
 GRAPHITE                 CEMENTITE                M23C6
 M7C3                     M5C2                     KSI_CARBIDE
 A1_KAPPA                 KAPPA                    FE4N_LP1
 FECN_CHI                 LAVES_PHASE_C14          M3SI
 CR3SI                    FE2SI                    MSI
 M5SI3                    AL4C3                    FE8SI2C
 SIC  REJECTED
TDB_TCFE7: res ph fcc
 FCC_A1  RESTORED
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'J. Lacaze and B. Sundman, Metall. Mater. Trans. A, 22A (1991), 2211-2223;
      Fe-Si and Fe-Si-C'
  'J. Miettinen and B. Hallstedt, Calphad, 22 (1998), 231-256; Fe-Si and Fe
      -Si-C'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_TCFE7: @@
TDB_TCFE7: app mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys  fe si c
 FE                       SI                       C
   DEFINED
APP: rej ph * all
 BCC_A2                   CEMENTITE                FCC_A1
 FE4N_LP1                 HCP_A3                   LIQUID:L
   REJECTED
APP: res ph fcc
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
```

```
 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'D. Bergner et al., Defect and Diffusion Forum 66-69(1989)409. Impurity
      diffusion of Si in fcc Fe.'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1323; * N

DIC>
DIC> @@
DIC> @@ ENTER REGION austenite
DIC> @@
DIC> enter-region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID
DIC> @@ N.B. GRIDPOINT DISTANCES ARE SMALLEST AROUND THE MIDDLE
DIC> @@
DIC> enter-grid
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 50E-3
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.9
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.11
DIC>
DIC> @@
DIC> @@ ENTER PHASE INTO REGION (BOTH PIECES ARE AUSTENITIC)
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER COMPOSITIONS INTO THE PHASE
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /SI/: FE
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C  func 0.49-0.04*HS(X-25E-3);
PROFILE FOR /SI/: SI func 3.80-3.75*HS(X-25E-3);
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1e10
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /1E+09/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> save exa3 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa3-run

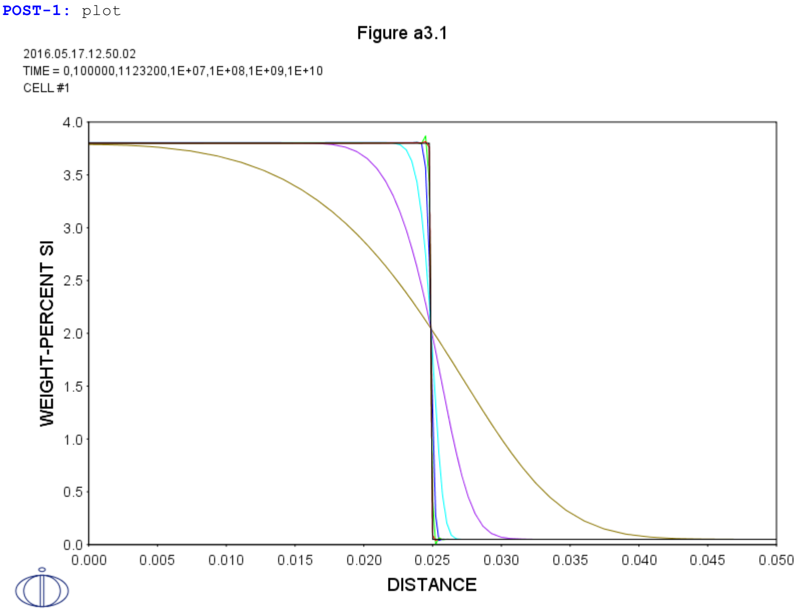
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa3\run.DCM"
DIC>
DIC> @@ darken_run.DCM
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> @@
DIC> @@ READ SETUP FROM FILE AND START SIMULATION
DIC> @@
DIC> read exa3
 OK
DIC>
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   456.31374     DT =   455.91364     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   1368.1410     DT =   911.82727     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   3191.7955     DT =   1823.6545     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970656  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   6839.1046     DT =   3647.3091     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970655  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   14133.723     DT =   7294.6182     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970661  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   28722.959     DT =   14589.236     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239970727  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   57901.432     DT =   29178.473     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239971033  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     1  seconds

 TIME =   116258.38     DT =   58356.945     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239972062  FE = .963089414810839
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]

  CPU time used in timestep                     0  seconds

 TIME =   232972.27     DT =   116713.89     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0215326239974973  FE = .963089414810838
                        SI = .0369105851891616
 TOTAL SIZE OF SYSTEM:  .05 [m]
```

```
 CPU time used in timestep                     0  seconds

TIME =   466400.05     DT =   233427.78     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239975114  FE = .963089414810838
                       SI = .0369105851891616
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =   933255.61     DT =   466855.56     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239974399  FE = .963087589566776
                       SI = .0369124104332238
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =   1866966.7     DT =   933711.13     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239967894  FE = .963085134899492
                       SI = .0369148651005078


output ignored...



... output resumed

 CPU time used in timestep                     0  seconds

TIME =   59757057.     DT =   29878756.     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239987765  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.11951457E+09 DT =   59757512.     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239989405  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.23902959E+09 DT =  0.11951502E+09 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239990903  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.47805964E+09 DT =  0.23903005E+09 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239992442  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.95611974E+09 DT =  0.47806010E+09 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239994041  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.19122399E+10 DT =  0.95612019E+09 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239995618  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.29122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326239996491  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.39122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240001231  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     1  seconds

TIME =  0.49122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240008828  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.59122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240014694  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.69122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240019108  FE = .963085134899492
                       SI = .0369148651005078
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.79122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240022539  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.89122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240025594  FE = .963085134899492
```

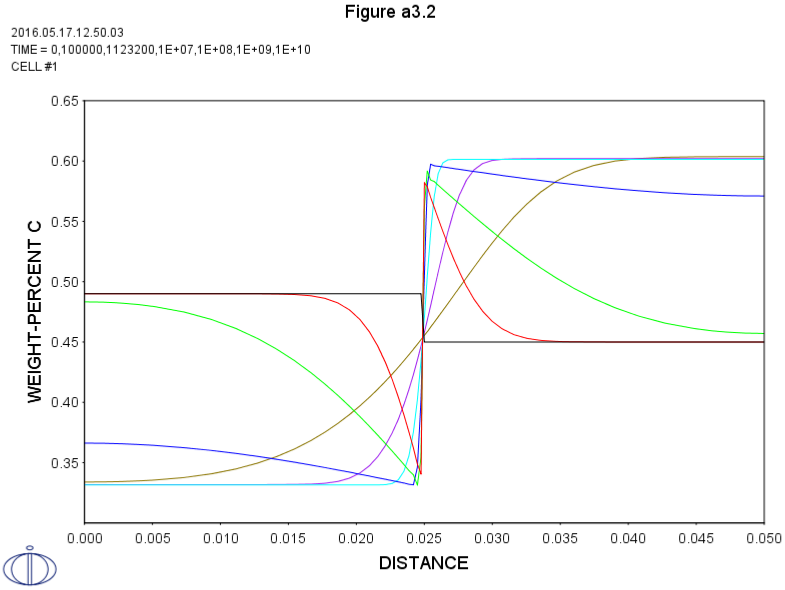
```
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.99122399E+10 DT =  0.10000000E+10 SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0215326240028067  FE = .963085134899492
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

 CPU time used in timestep                     0  seconds

TIME =  0.10000000E+11 DT =   87760071.     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .021532624002806  FE = .963085134899493
                       SI = .0369148651005077
TOTAL SIZE OF SYSTEM:  .05 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.40010010
DELETING TIME-RECORD FOR TIME    456.31374
DELETING TIME-RECORD FOR TIME    1368.1410
DELETING TIME-RECORD FOR TIME    3191.7955
DELETING TIME-RECORD FOR TIME    6839.1046
DELETING TIME-RECORD FOR TIME    14133.723
DELETING TIME-RECORD FOR TIME    28722.959
DELETING TIME-RECORD FOR TIME    57901.432
DELETING TIME-RECORD FOR TIME    116258.38
DELETING TIME-RECORD FOR TIME    232972.27
DELETING TIME-RECORD FOR TIME    466400.05
DELETING TIME-RECORD FOR TIME    933255.61
DELETING TIME-RECORD FOR TIME    1866966.7
DELETING TIME-RECORD FOR TIME    3734389.0
DELETING TIME-RECORD FOR TIME    7469233.5
DELETING TIME-RECORD FOR TIME    14938922.
DELETING TIME-RECORD FOR TIME    29878300.
DELETING TIME-RECORD FOR TIME    59757057.
DELETING TIME-RECORD FOR TIME   0.11951457E+09
DELETING TIME-RECORD FOR TIME   0.23902959E+09
DELETING TIME-RECORD FOR TIME   0.47805964E+09
DELETING TIME-RECORD FOR TIME   0.95611974E+09
DELETING TIME-RECORD FOR TIME   0.19122399E+10
DELETING TIME-RECORD FOR TIME   0.29122399E+10
DELETING TIME-RECORD FOR TIME   0.39122399E+10
DELETING TIME-RECORD FOR TIME   0.49122399E+10
DELETING TIME-RECORD FOR TIME   0.59122399E+10
DELETING TIME-RECORD FOR TIME   0.69122399E+10
DELETING TIME-RECORD FOR TIME   0.79122399E+10
DELETING TIME-RECORD FOR TIME   0.89122399E+10

KEEPING TIME-RECORD FOR TIME   0.99122399E+10
AND FOR TIME                   0.10000000E+11
WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa3-plot

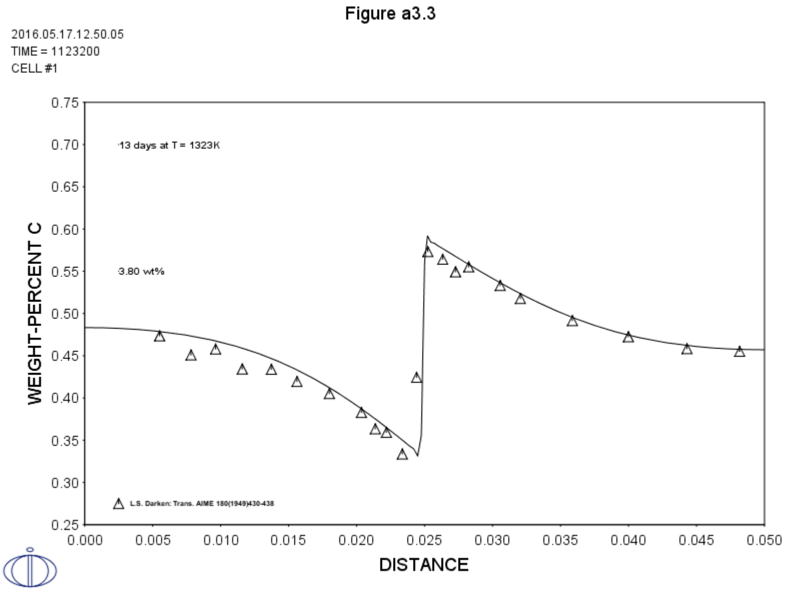
```
DIC>AboutDIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa3\plot.DCM"
DIC>
DIC> @@ darken_plot.DCM
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MODULE AND SPECIFY THE STORE-RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+10
DIC> read exa3
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ PLOT THE CONCENTRATION PROFILE FOR Si AT TIMES 0, 1E5, 1123200, 1E7,
POST-1: @@ 1E8, 1E9 AND 1E10 S
POST-1: @@
POST-1: @@ SET DISTANCE IN SYSTEM AS X-AXIS, WEIGHT-% SI ON Y-AXIS AND SPECIFY
POST-1: @@ FOR WHICH SIMULATION TIMES WE SHALL PLOT THE PROFILES.
POST-1: @@
POST-1: set-diagram-axis x distance global
  INFO: Distance is set as independent variable
POST-1: set-diagram-axis y weight-percent si
POST-1: set-plot-condition time 0 1E5 1123200 1e7 1E8 1E9 1E10
POST-1:
POST-1: @@
POST-1: @@ PLOT THE DIAGRAM
POST-1: @@
POST-1: set-title
TITLE : Figure a3.1
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1: @@
POST-1: @@ PLOT THE CONCENTRATION PROFILES FOR C
POST-1: @@
POST-1: @@ WE ONLY NEED TO CHANGE Y-AXIS
POST-1: @@
POST-1: set-diagram-axis y w-p c
POST-1: set-title Figure a3.2
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1: @@
POST-1: @@ COMPARE WITH DARKENS EXPERIMENTS
POST-1: @@
POST-1: append_experimental_data  yes exa3.exp 0; 1
POST-1:
POST-1: set-plot-condition time 1123200
POST-1:
POST-1: s-s-s
AXIS (X, Y OR Z) : y
AUTOMATIC SCALING  (Y OR N) /N/: n
MIN VALUE : 0.25
MAX VALUE : 0.75
POST-1:
POST-1: set-title Figure a3.3
POST-1: plot
```

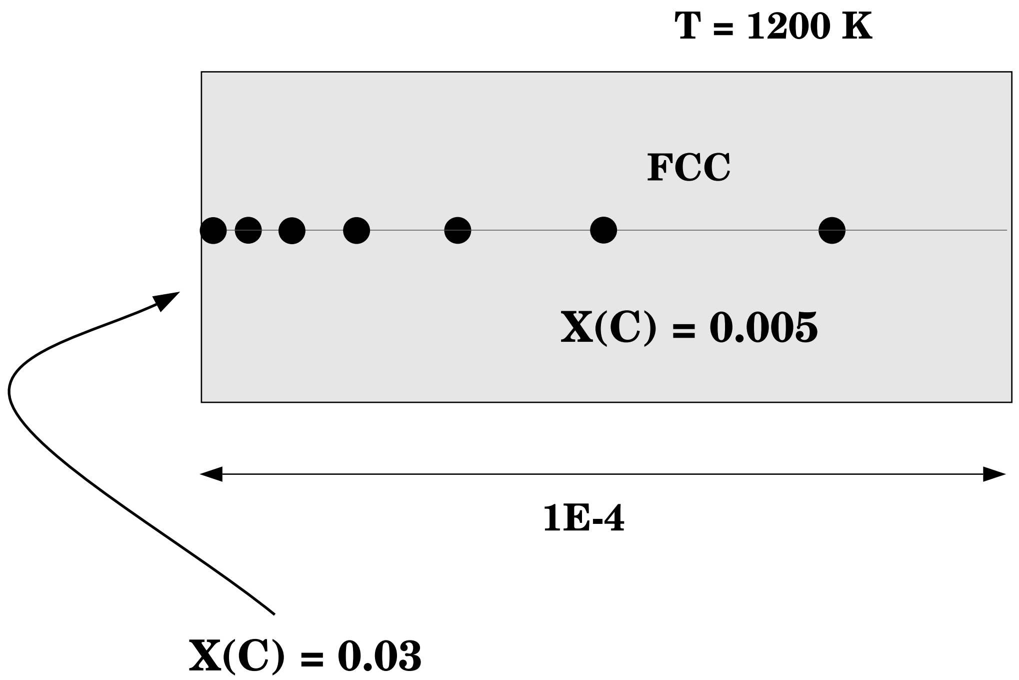



```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1: set-inter
 --OK---
POST-1:
```

Example a4

Carburization of a binary Fe-C alloy

(Comparison with an analytical erf-solution)



exa4-setup

```
SYS:About
 Thermo-Calc / DICTRA is software package for calculation of phase diagrams,
 simulation of phase transformation kinetics and much more.

 Copyright Foundation for Computational Thermodynamics,
 Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa4\setup.DCM" @@
SYS: @@ One-phase problem.
SYS: @@ This is a simple binary simulation with just one single phase region.
SYS: @@ We will compare our numerical simulation with an analytical erf-solution.
SYS: @@ For this reason a special database erf.tdb is created in which
SYS: @@ the diffusion coefficient is set to a concentration independent value.
SYS: @@
SYS:
SYS: @@ exa4_setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ READ DATA FROM DATABASES
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-system fe,c
 FE                      C  DEFINED
TDB_FEDEMO: rej-ph *
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: rest-ph fcc
 FCC_A1  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: append user exa4.TDB
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA  DEFINED
TDB_APP: def-system fe,c
 FE                      C  DEFINED
TDB_APP: rej-ph *
 FCC_A1  REJECTED
TDB_APP: rest-ph fcc
 FCC_A1  RESTORED
TDB_APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
 -OK-
TDB_APP:
TDB_APP: @@
TDB_APP: @@ GOTO DICTRA MODULE AND SETUP THE PROBLEM
TDB_APP: @@
TDB_APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1200; * N

DIC>
DIC> @@
DIC> @@ ENTER REGION steel
DIC> @@
DIC> enter-region
REGION NAME : steel
DIC>
DIC> @@
DIC> @@ ENTER GRID
DIC> @@ CARBON WILL ENTER THE SYSTEM FROM THE LOWER BOUNDARY AND CONSEQUENTLY
DIC> @@ WE REUIRE MORE POINTS AT THAT BOUNDARY. FOR THIS REASON WE USE A
DIC> @@ GEOMETRIC GRID IN THIS CASE.
DIC> @@
```

```
DIC> enter-grid
REGION NAME : /STEEL/: steel
WIDTH OF REGION /1/: 1E-4
TYPE /LINEAR/: geometric
NUMBER OF POINTS /50/: 32
VALUE OF R IN THE GEOMETRICAL SERIE : 1.11
DIC>
DIC> @@
DIC> @@ ENTER PHASE INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /STEEL/: steel
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC>
DIC> @@
DIC> @@ INITIAL COMPOSITION IN FCC-PHASE
DIC> @@
DIC> enter-composition
REGION NAME : /STEEL/: steel
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: mole-fraction
PROFILE FOR /C/: c
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.005
VALUE OF LAST POINT : /5E-3/: 0.005
DIC>
DIC>
DIC> @@
DIC> @@ SET A FIX COMPOSITION AS BOUNDARY VALUE
DIC> @@
DIC> set-condition

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: boundary
BOUNDARY /LOWER/: lower
CONDITION TYPE /CLOSED_SYSTEM/: state-variable-value
State variable expression #1 : /N=1/: n=1
State variable expression #2 : x(c)=0.03
DIC>
DIC>
DIC> @@
DIC> @@ SET SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 100
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /10/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC>
DIC> save exa4 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

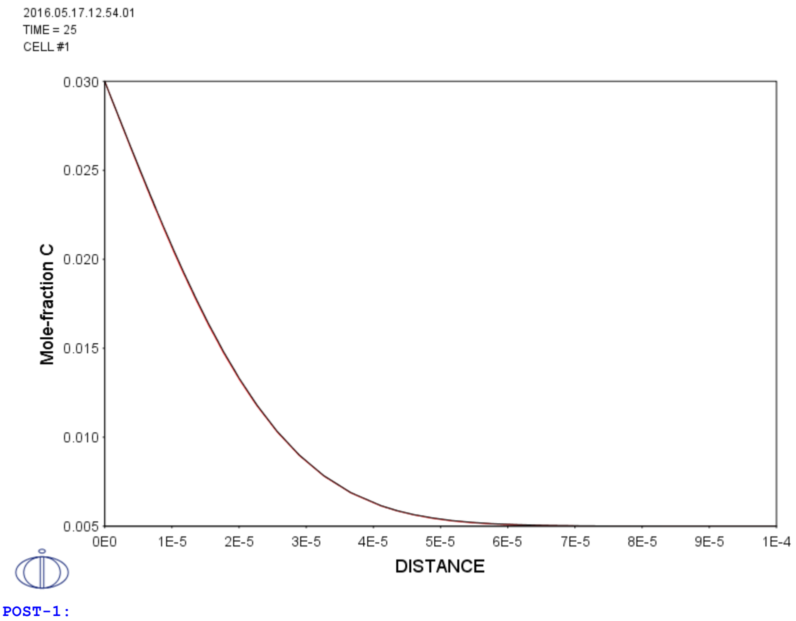
exa4-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa4\run.DCM"DIC>
DIC>
DIC> @@ exa4_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE a4
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exa4
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0050251256281407  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  C = .0050251256281407  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  9  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: STEEL

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00508348834679686  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds
  1  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: STEEL

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00508409543247046  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.12105107E-01 DT =  0.12005007E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0051362682962797  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.36115121E-01 DT =  0.24010014E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00519794167790873  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.84135150E-01 DT =  0.48020029E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00527960440299711  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.18017521     DT =  0.96040057E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00539181114486414  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.37225532     DT =  0.19208011     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0055483575543518  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.75641555     DT =  0.38416023     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00576829415910514  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   1.5247360     DT =  0.76832046     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00607832336005493  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3.0613769     DT =   1.5366409     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00651606477943324  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   6.1346588     DT =   3.0732818     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00713462772374614  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   12.281222     DT =   6.1465637     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00800905728756116  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   22.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00904285003239104  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   32.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
```

```
 U-FRACTION IN SYSTEM:  C = .00985903057950594  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   42.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .010556058900354  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   52.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0111746120989673  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   62.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0117364077404952  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   72.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .012254681140506  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   82.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0127382074556105  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   92.281222     DT =  10.0000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0131931583377927  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  100.000000     DT =   7.7187776     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0135276709358081  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.12105107E-01
 DELETING TIME-RECORD FOR TIME   0.36115121E-01
 DELETING TIME-RECORD FOR TIME   0.84135150E-01
 DELETING TIME-RECORD FOR TIME   0.18017521
 DELETING TIME-RECORD FOR TIME   0.37225532
 DELETING TIME-RECORD FOR TIME   0.75641555
 DELETING TIME-RECORD FOR TIME    1.5247360
 DELETING TIME-RECORD FOR TIME    3.0613769
 DELETING TIME-RECORD FOR TIME    6.1346588
 DELETING TIME-RECORD FOR TIME    12.281222
 DELETING TIME-RECORD FOR TIME    22.281222
 DELETING TIME-RECORD FOR TIME    32.281222
 DELETING TIME-RECORD FOR TIME    42.281222
 DELETING TIME-RECORD FOR TIME    52.281222
 DELETING TIME-RECORD FOR TIME    62.281222
 DELETING TIME-RECORD FOR TIME    72.281222
 DELETING TIME-RECORD FOR TIME    82.281222

 KEEPING TIME-RECORD FOR TIME    92.281222
 AND FOR TIME                   100.000000
 WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

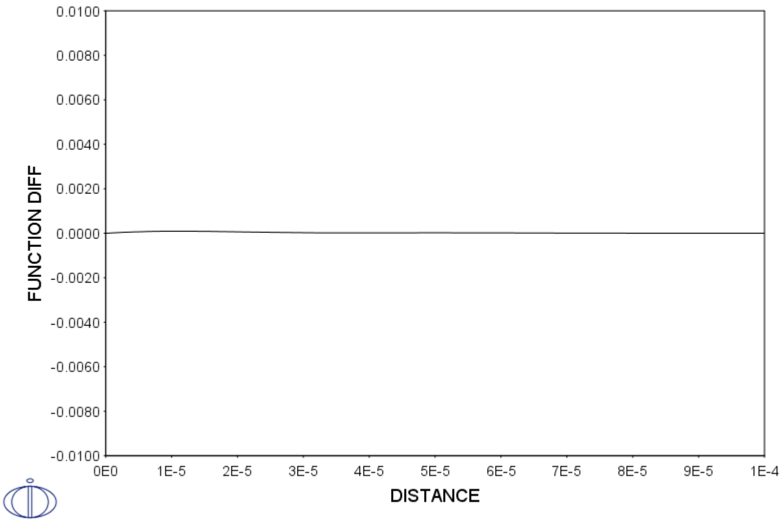
exa4-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa4\plot.DCM"
DIC>
DIC> @@ exa4_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE exa4
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+02
DIC> read exa4
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ PLOTT A COMPOSITION PROFILE
POST-1: @@
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-d-a y x(c)
POST-1: s-p-c time 25
POST-1:
POST-1: @@
POST-1: @@ ENTER THE ANALYTICAL SOLUTION, CALLED ERFSOL
POST-1: @@
POST-1: enter-symbol
Function or table /FUNCTION/: function
NAME: erfsol
FUNCTION: 0.03-0.025*erf(gd/sqrt(4*dc(fcc,c,c,fe)*25));
POST-1:
POST-1: @@
POST-1: @@ COMPARE THE ANALYTICAL AND NUMERICAL SOLUTIONS
POST-1: @@
POST-1: enter-symbol
Function or table /FUNCTION/: table
NAME: aaa
Variable(s) x(c) erfsol
POST-1:
POST-1: s-d-a y aaa
COLUMN NUMBER /*/: 1 2
POST-1:
POST-1: set-axis-text
AXIS (X, Y OR Z) : y
AUTOMATIC AXIS TEXT  (Y OR N) /N/: n
AXIS TEXT : Mole-fraction C
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE DIFFERENCE
POST-1: @@
POST-1: enter func diff=x(c)-erfsol;
POST-1: s-d-a y diff
POST-1: s-s-s y n -1e-2 1e-2
POST-1:
POST-1: plot
```

2016.05.17.12.54.01
TIME = 25
CELL #1

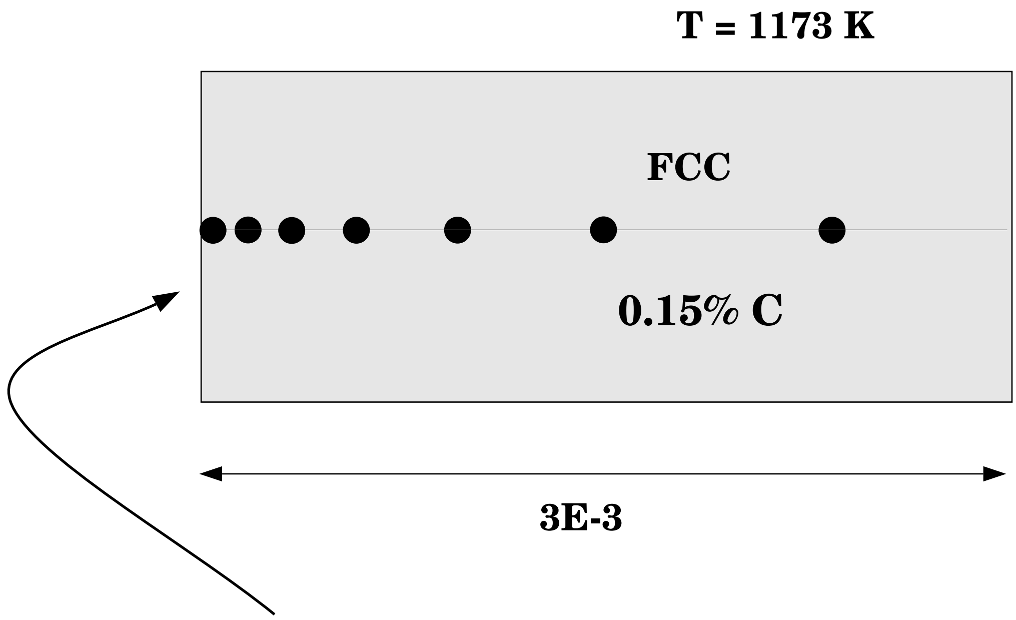



```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: set-interactive
 --OK---
POST-1:
```

Example a5

Carburization of a binary Fe-C alloy

(A surface reaction controls the flux of C at the surface)



$$J_C = -8.25E\text{-}9\left(a_C^{surf} - 0.64\right)$$

exa5-setup

```
About Copyright Foundation for Computational Thermodynamics,
 Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa5\setup.DCM"SYS: ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ One-phase problem.
SYS: @@ Simulation of carburization of a binary Fe-0.15 wt% C alloy.
SYS: @@ A mixture of 40% N2 and 60% cracked methanol was used as carrier gas.
SYS: @@ The carburizing "carbon potential" in the gas is 0.85 wt%.
SYS: @@ A surface reaction controles the flux of C at the surface.
SYS: @@
SYS:
SYS: @@ exa5_setup.DCM
SYS:
SYS: @@
SYS: @@ GOTO DATABASE AND READ THERMODYNAMIC AND KINETIC DATA.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-sys fe,c
 FE                      C  DEFINED
TDB_FEDEMO: rej-ph *
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: rest-ph fcc graphite
 FCC_A1                  GRAPHITE  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
      hard mater, Vol. 12, pp. 121-127; Molar volumes'
 -OK-
TDB_FEDEMO:@?
TDB_FEDEMO: append
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
```

```
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe,c
 FE                       C  DEFINED
APP: rej-ph *
 BCC_A2                   FCC_A1  REJECTED
APP: rest-ph fcc
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
 -OK-
APP:@?
APP: @@
APP: @@ GOTO THE DICTRA MONITOR TO SETUP THE INITIAL STATE OF THE SPECIMEN
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC> set-cond glob T 0 1173; * N

DIC>
DIC> @@
DIC> @@ SELECT REFERENCE STATE FOR THE C ACTIVITY
DIC> @@
DIC> set-ref-state
Component: c
Reference state: graph
Temperature /*/: *
Pressure /100000/: 1e5
DIC>
DIC> @@
DIC> @@ ENTER REGION ETC.
DIC> @@
DIC> enter-region
REGION NAME : steel
DIC>
```

```
DIC> enter-grid
REGION NAME : /STEEL/: steel
WIDTH OF REGION /1/: 3E-3
TYPE /LINEAR/: geo
NUMBER OF POINTS /50/: 60
VALUE OF R IN THE GEOMETRICAL SERIE : 1.11
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /STEEL/: steel
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> enter-composition
REGION NAME : /STEEL/: steel
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.15
VALUE OF LAST POINT : /0.15/: 0.15
DIC>
DIC> @@
DIC> @@ NOW SET BOUNDARY CONDITIONS. WE ARE ONLY INTERESTED IN THE
DIC> @@ SURFACE REGION IT IS THUS ENOUGH TO SET CONDITIONS AT I.E.
DIC> @@ THE LOWER BOUNDARY.
DIC> @@
DIC>
DIC> @@
DIC> @@ Specify the activity flux function which controls the uptake of C.
DIC> @@
DIC> @@ The functions f  and g  and the parameter N has to be specified.
DIC> @@                 k      k
DIC> @@
DIC> @@                                   N
DIC> @@  J  V = f (variables)*(ACTIVITY -g (variables))    (1)
DIC> @@   k  m   k                       k  k
DIC> @@
DIC> @@ f   and g   in equation 1 is the mass-transfer coefficient and
DIC> @@  k        k
DIC> @@ the activity of k in the gas respectively. ACTIVITY in eq. 1 means
DIC> @@ the actual activity of species k at the surface.
DIC> @@
DIC>
DIC> @@
DIC> @@ The main carburizing reaction for our atmosphere is:
DIC> @@
DIC> @@   CO  +  H    ->   C  +    H O       (I)
DIC> @@           2    <-   -        2
DIC> @@
DIC> @@ Following Sproge and Ågren (J. Heat Treating, Vol. 6, No1, 1988 pp. 9-19)
DIC> @@ we may callculate the mass-transfer coefficient for carbon, f in
DIC> @@ eq. 1 above by means of eq. 3,4 and 12 in Sproges and Ågrens paper.
DIC> @@
DIC> @@
DIC> @@          A *  K *  P  *  sqrt( P   )
DIC> @@                I    CO          H
DIC> @@                                  2
DIC> @@   f  =  -----------------------------  / gamma        (2)
DIC> @@          a  +  B * K * P  * sqrt( P  )
DIC> @@           C         I   CO         H
DIC> @@                                     2
DIC> @@
DIC> @@ K  is the equilibrium constant for reaction (I)
DIC> @@  I
DIC> @@
DIC> @@ A and B are constants defined in Sproges and Ågrens paper. gamma
DIC> @@ is the activity coefficient for carbon in the steel.
DIC> @@
DIC> @@ We will here assume a constant value for  P   *  sqrt( P   ) = 0.14
DIC> @@                                           CO          H
DIC> @@                                                        2
DIC> @@ The carbon activity in the gas will thus be controlled by the partial
DIC> @@ pressure of water as can be understood from reaction (I).
DIC> @@
DIC> @@ We will here assume that the carbon activity, a   of the gas is 0.64
DIC> @@                                                C
DIC> @@ which corresponds to a carburizing "carbon potential" of 0.85 wt%.
DIC> @@
DIC> @@ In this way we may calculate f to 8.25E-9 mol/s.
DIC> @@
DIC>
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: bound
BOUNDARY /LOWER/: lower
CONDITION TYPE /CLOSED_SYSTEM/: activity_flux_function
 ENTER THE EXPRESSION AS:
                                 N
   J  V = f (variables)*(ACTIVITY -g (variables))
    k  m   k                     k  k
 FLUX OF FCC_A1,C
LOW TIME LIMIT /0/: 0
f(T,P,TIME)= -8.25E-9;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
N /1/: 1
LOW TIME LIMIT /0/: 0
g(T,P,TIME)= 0.64;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC> @@
DIC> @@ Specify simulation time ...
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 18000
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /1800/:
INITIAL TIMESTEP : /1E-07/:
```

```
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> @@
DIC> @@ Save the setup on file
DIC> @@
DIC> Save exa5 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa5-run

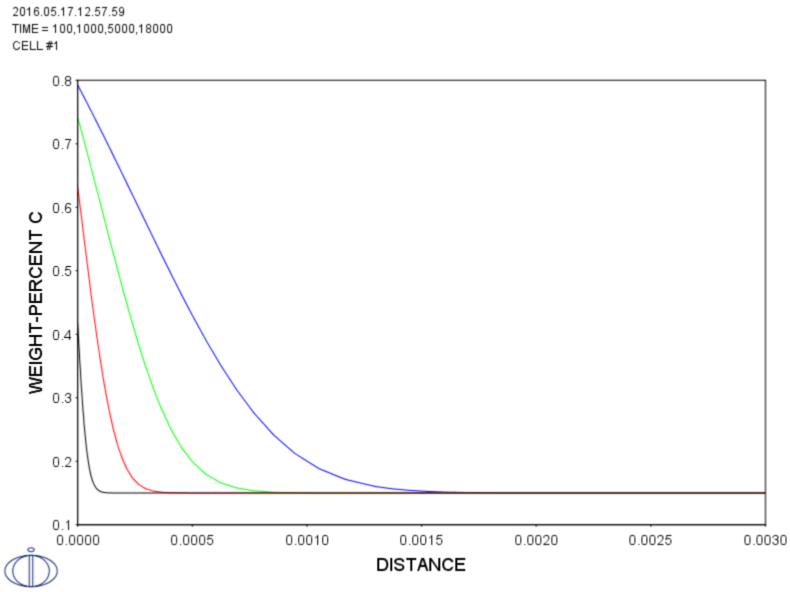
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa5\run.DCM"
DIC>
DIC> @@ exa5_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE a5
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC> read exa5
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]
  9  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: STEEL

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916398349  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds
  1  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: STEEL

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495931637932  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.33719637E-01 DT =  0.33619537E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698501030669505  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.10095871     DT =  0.67239073E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698511164116955  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.23543686     DT =  0.13447815     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698531294268508  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.50439315     DT =  0.26895629     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0069857120308775  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   1.0423057     DT =  0.53791259     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698650077061368  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   2.1181309     DT =   1.0758252     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698805257460461  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   4.2697813     DT =   2.1516503     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00699108630238  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   8.5730820     DT =   4.3033007     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00699696503140298  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   17.179683     DT =   8.6066014     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00700822016227208  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   34.392886     DT =   17.213203     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00702942229331613  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   68.819292     DT =   34.426406     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00706852220082184  FE = 1
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds
```

```
TIME =   137.67210     DT =   68.852811     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00713870843155665  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   275.37772     DT =   137.70562     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00726063934605858  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     1  seconds

TIME =   550.78897     DT =   275.41124     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00746470580143571  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   1101.6115     DT =   550.82249     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00779310153791826  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   2203.2564     DT =   1101.6450     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0083022445878165  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   4003.2564     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .008942151748702  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   5803.2564     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00946192476382277  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   7603.2564     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .00991191759849167  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   9403.2564     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0103145419290456  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   11203.256     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0106822470518482  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   13003.256     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0110228177770633  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   14803.256     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0113415095977369  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   16603.256     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0116420719686607  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   18000.000     DT =   1396.7436     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0118645957846413  FE = 1
TOTAL SIZE OF SYSTEM:  .003 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.33719637E-01
DELETING TIME-RECORD FOR TIME   0.10095871
DELETING TIME-RECORD FOR TIME   0.23543686
DELETING TIME-RECORD FOR TIME   0.50439315
DELETING TIME-RECORD FOR TIME    1.0423057
DELETING TIME-RECORD FOR TIME    2.1181309
DELETING TIME-RECORD FOR TIME    4.2697813
DELETING TIME-RECORD FOR TIME    8.5730820
DELETING TIME-RECORD FOR TIME    17.179683
DELETING TIME-RECORD FOR TIME    34.392886
DELETING TIME-RECORD FOR TIME    68.819292
DELETING TIME-RECORD FOR TIME    137.67210
DELETING TIME-RECORD FOR TIME    275.37772
DELETING TIME-RECORD FOR TIME    550.78897
DELETING TIME-RECORD FOR TIME    1101.6115
DELETING TIME-RECORD FOR TIME    2203.2564
DELETING TIME-RECORD FOR TIME    4003.2564
DELETING TIME-RECORD FOR TIME    5803.2564
DELETING TIME-RECORD FOR TIME    7603.2564
DELETING TIME-RECORD FOR TIME    9403.2564
DELETING TIME-RECORD FOR TIME    11203.256
DELETING TIME-RECORD FOR TIME    13003.256
DELETING TIME-RECORD FOR TIME    14803.256

KEEPING TIME-RECORD FOR TIME    16603.256
AND FOR TIME                    18000.000
WORKSPACE RECLAIMED
DIC>
```

```
DIC> set-inter
 --OK---
DIC>
```

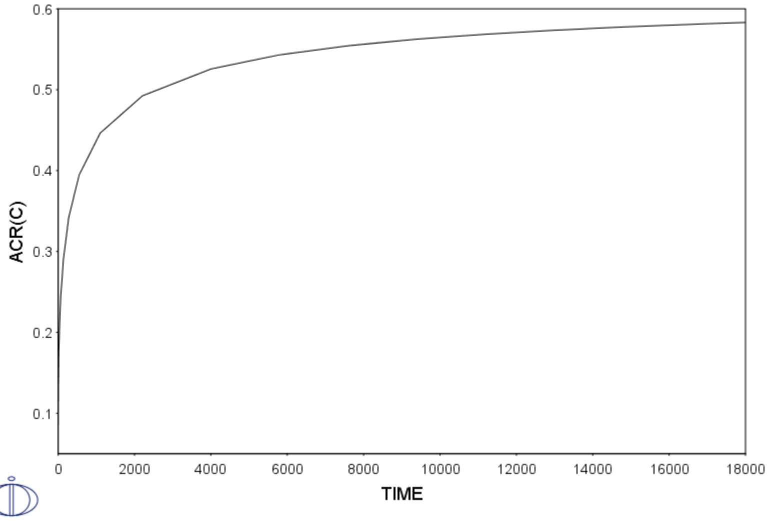
exa5-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa5\plot.DCM"
DIC>
DIC> @@ exa5_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE a5
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.80000E+04
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC> read exa5
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ PLOT SOME DIFFERENT CONCENTRATION PROFILES.
POST-1: @@
POST-1: s-d-a x dist glo
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p c
POST-1: s-p-c time 100 1000 5000 18000
POST-1:
POST-1: plot
```



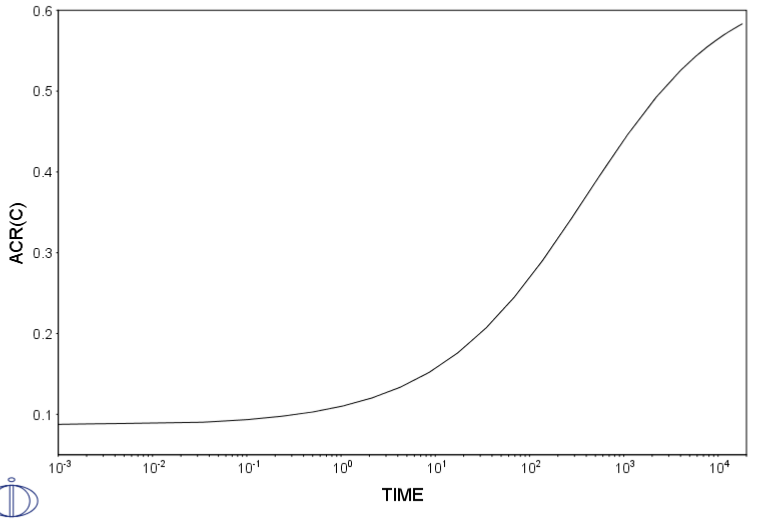

```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE VARIATION OF THE C ACTIVITY AT THE SURFACE
POST-1: @@
POST-1: s-d-a y acr(c)
POST-1:
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1:
POST-1: s-p-c
CONDITION /TIME/: interface
INTERFACE : first
POST-1:
POST-1: plot
```

2016.05.17.12.58.00
"FIRST" INTERFACE OF SYSTEM
CELL #1



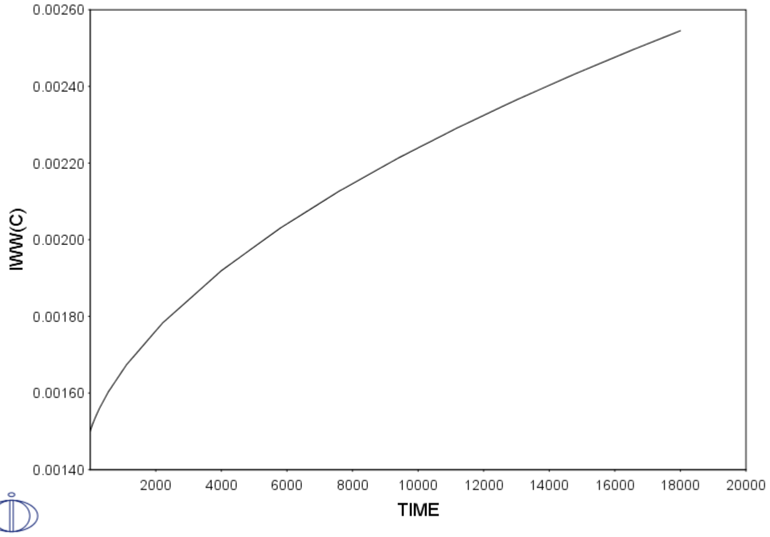

```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ LETS HAVE A LOGARITHMIC SCALE ON THE X-AXIS
POST-1: @@
POST-1: set-axis-type
AXIS (X, Y OR Z) : x
AXIS TYPE /LINEAR/: logarithmic
POST-1:
POST-1: s-s-s x n 0.001 2e4
POST-1:
POST-1: plot
```

2016.05.17.12.58.03
"FIRST" INTERFACE OF SYSTEM
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE AVERAGE WEIGHT FRACTION OF C IN THE SPECIMEN
POST-1: @@
POST-1: s-d-a y iww(c)
POST-1:
POST-1: set-ax-ty
AXIS (X, Y OR Z) : x
AXIS TYPE /LINEAR/: linear
POST-1:
POST-1: plot
```

2016.05.17.12.58.06
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

exa6-setup

About Stockholm, Sweden

```
 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016
```

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa6\setup.DCM"SYS: @@
SYS: @@ One-phase problem.
SYS: @@ A simple, perhaps not realistic, example on diffusion through a
SYS: @@ tube wall. The tube-material is an Fe-0.6%Mn-0.7%Si-0.05%C alloy. On
SYS: @@ the inside wall a carbon activity of 0.9 is maintained whereas on
SYS: @@ the outside the C-activity is very low. This example demonstrates
SYS: @@ the use of the command SET-FIRST-INTERFACE as well as the use of
SYS: @@ MIXED boundary conditions.
SYS: @@
SYS:
SYS: @@ setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ GO TO THE DATABASE MODULE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET'S USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE7: def-sys fe si mn c
 FE                      SI                      MN
 C  DEFINED
TDB_TCFE7: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 FCC_A1                  HCP_A3                  DIAMOND_FCC_A4
 GRAPHITE                CEMENTITE               M23C6
 M7C3                    M5C2                    KSI_CARBIDE
 A1_KAPPA                KAPPA                   FE4N_LP1
 FECN_CHI                LAVES_PHASE_C14         M3SI
 G_PHASE                 CR3SI                   FE2SI
 MSI                     M5SI3                   AL4C3
 FE8SI2C                 SIC  REJECTED
TDB_TCFE7: res ph fcc,grap
 FCC_A1                  GRAPHITE  RESTORED
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'W. Huang, Metall. Trans. A, 21A (1990), 2115-2123; TRITA-MAC 411 (Rev
      1989); C-FE-MN'
  'J. Lacaze and B. Sundman, Metall. Mater. Trans. A, 22A (1991), 2211-2223;
      Fe-Si and Fe-Si-C'
  'W. Huang, Calphad, 13 (1989), 243-252; TRITA-MAC 388 (rev 1989); FE-MN'
  'J.E. Tibballs, SI Norway (1991) Rep. 890221-5; Mn-Si'
  'J. Miettinen and B. Hallstedt, Calphad, 22 (1998), 231-256; Fe-Si and Fe
      -Si-C'
  'A. Forsberg and J. Agren, J. Phase Equil., 14 (1993), 354-363; Fe-Mn-Si'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
      hard mater, Vol. 12, pp. 121-127; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE TO RETRIVE KINETIC DATA
TDB_TCFE7: @@
TDB_TCFE7: app
 Use one of these databases

 TCFE8    =  Steels/Fe-Alloys  v8.0
 TCFE9    =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1   =  FROST database v1.0
 TCFE7    =  Steels/Fe-Alloys  v7.0
 TCFE6    =  Steels/Fe-Alloys  v6.2
 TCFE5    =  Steels/Fe-Alloys  v5.0
 TCFE4    =  Steels/Fe-Alloys  v4.1
 TCFE3    =  Steels/Fe-Alloys  v3.1
 TCFE2    =  Steels/Fe-Alloys  v2.1
 TCFE1    =  Steels/Fe-Alloys  v1.0
 FEDAT    =  TCS/TT Steels Database v1.0
 TCNI9    =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8    =  Ni-Alloys  v8.0
 TCNI7    =  Ni-Alloys  v7.1
 TCNI6    =  Ni-Alloys  v6.0
 TCNI5    =  Ni-Alloys  v5.1
 TCNI4    =  Ni-Alloys  v4.0
 TCNI1    =  Ni-Alloys  v1.3
 TCAL4    =  Al-Alloys  v4.0
```

```
 TCAL3    =  Al-Alloys  v3.0
 TCAL2    =  Al-Alloys  v2.0
 TCAL1    =  Al-Alloys  v1.2
 TCMG5    =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4    =  Mg-Alloys  v4.0
 TCMG3    =  Mg-Alloys  v3.0
 TCMG2    =  Mg-Alloys  v2.0
 TCMG1    =  Mg-Alloys  v1.1
 TCTI1    =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1    =  Copper  v1.0 SNAPSHOT
 TCCC1    =  Cemented carbide  v1.0
 TCHEA1   =  High Entropy Alloy  v1.0
 SSOL5    =  SGTE Alloy Solutions Database v5.0
 SSOL4    =  SGTE Alloy Solutions Database v4.9f
 SSOL2    =  SGTE Alloy Solutions Database v2.1
 SSUB5    =  SGTE Substances Database v5.1
 SSUB4    =  SGTE Substances Database v4.1
 SSUB3    =  SGTE Substances Database v3.3
 SSUB2    =  SGTE Substances Database v2.2
 SNOB3    =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2    =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1    =  SGTE Nobel Metal Alloys Database v1.2
 STBC2    =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1    =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1    =  SGTE Molten Salts Database v1.2
 SNUX6    =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2    =  TC Semi-Conductors  v2.1
 SLAG4    =  Fe-containing Slag  v4.0 snapshot
 SLAG3    =  Fe-containing Slag  v3.2
 SLAG2    =  Fe-containing Slag  v2.2
 SLAG1    =  Fe-containing Slag  v1.2
 TCOX7    =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6    =  Metal Oxide Solutions  v6.1
 TCOX5    =  Metal Oxide Solutions  v5.1
 TCOX4    =  Metal Oxide Solutions  v4.1
 ION3     =  Ionic Solutions  v3.0
 ION2     =  Ionic Solutions  v2.6
 ION1     =  Ionic Solutions  v1.5
 NOX2     =  NPL Oxide Solutions Database v2.1
 TCSLD3   =  Solder Alloys  v3.1
 TCSLD2   =  Solder Alloys  v2.0
 TCSLD1   =  Solder Alloys  v1.0
 TCSI1    =  Ultrapure Silicon  v1.1
 TCMP2    =  Materials Processing  v2.5
 TCES1    =  Combustion/Sintering  v1.1
 TCSC1    =  Super Conductor  v1.0
 TCFC1    =  SOFC Database v1.0
 TCNF2    =  Nuclear Fuels  v2.1b
 NUMT2    =  Nuclear Materials  v2.1
 NUOX4    =  Nuclear Oxides  v4.2
 NUTO1    =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1    =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10   =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11   =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2    =  Aqueous Solution  v2.5
 AQS2     =  TGG Aqueous Solution Database v2.5
 GCE2     =  TGG Geochemical/Environmental TDB v2.3
 PURE5    =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO   =  Aluminum Demo Database
 FEDEMO   =  Iron Demo Database
 NIDEMO   =  Nickel Demo Database
 SLDEMO   =  Solder Demo Database
 OXDEMO   =  Oxide Demo Database
 SUBDEMO  =  Substance Demo Database
 PTERN    =  Public Ternary Alloys TDB v1.3
 PAQ2     =  Public Aqueous Soln (SIT) TDB v2.4
 PG35     =  G35 Binary Semi-Conductors TDB v1.2
 MOB2     =  Alloys Mobility  v2.4
 MOB1     =  Alloys Mobility  v1.3
 MOBFE1   =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2   =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3   =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4   =  Ni-Alloys Mobility  v4.0
 MOBNI3   =  Ni-Alloys Mobility  v3.1
 MOBNI2   =  Ni-Alloys Mobility  v2.4
 MOBNI1   =  Ni-Alloys Mobility  v1.0
 MOBAL3   =  Al-Alloys Mobility  v3.0
 MOBAL2   =  Al-Alloys Mobility  v2.0
 MOBAL1   =  Al-Alloys Mobility  v1.0
 MOBCU1   =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1   =  Mg-Alloys Mobility  v1.0
 MOBSI1   =  Si-Alloys Mobility  v1.0
 MOBTI1   =  Ti-Alloys Mobility  v1.0
 MALDEMO  =  Al-Alloys Mobility demo database
 MFEDEMO  =  Fe-Alloys Mobility demo database
 MNIDEMO  =  Ni-Alloys Mobility demo database
 USER     =  User defined Database

DATABASE NAME /TCFE7/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe si mn c
 FE                     SI                      MN
 C  DEFINED
APP: rej ph * all
 BCC_A2                 CEMENTITE               FCC_A1
 FE4N_LP1               HCP_A3                  LIQUID:L
   REJECTED
APP: res ph fcc
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
```

```
  'Bae et al.: Z. Metallkunde 91(2000)672-674; fcc Fe-Mn
       Mn-Ni'
  'D. Bergner et al., Defect and Diffusion Forum 66-69(1989)409. Impurity
      diffusion of Si in fcc Fe.'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR PROBLEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1273; * N

DIC>
DIC> @@
DIC> @@ SET REFERENCE STATE FOR CARBON
DIC> @@
DIC> set-ref C grap * 101325
DIC>
DIC> @@
DIC> @@ ENTER A REGION
DIC> @@
DIC> enter-region aus
DIC>
DIC> @@
DIC> @@ ENTER DOUBLE GEOMETRIC GRID INTO THE REGION
DIC> @@
DIC> enter-grid
REGION NAME : /AUS/: aus
WIDTH OF REGION /1/: 1e-2
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 80
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 1.02
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 0.98
DIC>
DIC> @@
DIC> @@ SET GEOMETRY (1 = CYLINDER)
DIC> @@
DIC> enter-geo
GEOMETRICAL EXPONENT /0/: 1
DIC>
DIC> @@
DIC> @@ SET FIRST INTERFACE => TUBE
DIC> @@
DIC> set-first-interface
COORDINATE FOR FIRST INTERFACE /0/: 2e-2
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASE IN REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc_a1#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS INTO THE PHASE
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /SI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: si lin 0.7 0.7
PROFILE FOR /MN/: mn lin 0.6 0.6
PROFILE FOR /SI/: c lin 5e-2 5e-2
DIC>
DIC> @@
DIC> @@ SET BOUNDARY CONDITIONS ON BOTH LOWER AND UPPER PART OF THE REGION
DIC> @@
DIC> @@ USE MIXED CONDITIONS, AN ACTIVITY CONDITION FOR C AND CLOSED
DIC> @@ SYSTEMS FOR MN AND SI.
DIC> @@
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: boundary
BOUNDARY /LOWER/: lower
CONDITION TYPE /CLOSED_SYSTEM/: mixed
Dependent substitutional element:FE
Dependent interstitial element:VA
TYPE OF CONDITION FOR COMPONENT C /ZERO_FLUX/: activity
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 0.9;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
TYPE OF CONDITION FOR COMPONENT MN /ZERO_FLUX/: zero_flux
TYPE OF CONDITION FOR COMPONENT SI /ZERO_FLUX/: zero_flux
DIC>
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: boundary
BOUNDARY /UPPER/: upper
CONDITION TYPE /CLOSED_SYSTEM/: mixed
Dependent substitutional element:FE
Dependent interstitial element:VA
TYPE OF CONDITION FOR COMPONENT C /ZERO_FLUX/: activity
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 1e-5;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
TYPE OF CONDITION FOR COMPONENT MN /ZERO_FLUX/: zero_flux
TYPE OF CONDITION FOR COMPONENT SI /ZERO_FLUX/: zero_flux
DIC>
DIC> @@
```

```
DIC> @@ SET SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1e9
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ SAVE ON FILE
DIC> @@
DIC> save exa6 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa6-run

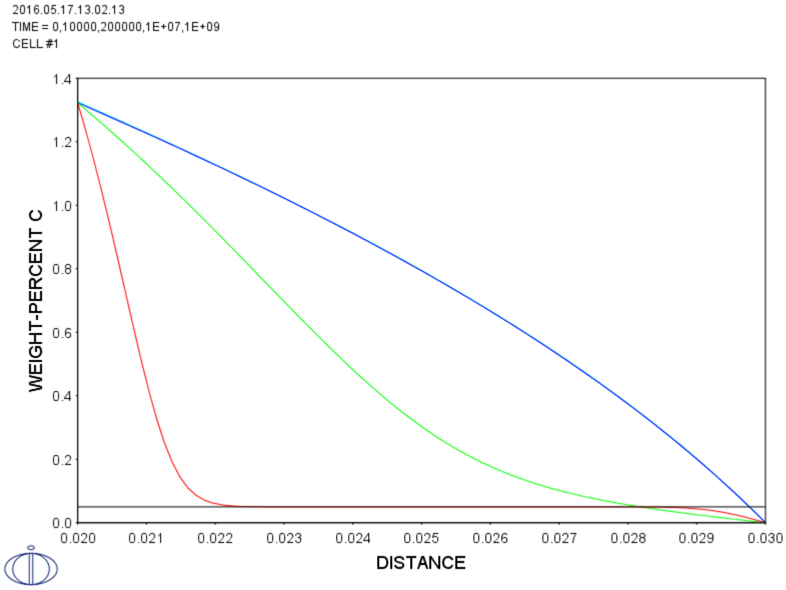
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa6\run.DCM"
DIC>
DIC> @@ run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING exa6
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC> read exa6
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> simulate
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00115488575879621  FE = .490055682684517
                        MN = .00302988813183617  SI = .00691442924890044
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]
 U-FRACTION IN SYSTEM:  C = .00115488575879621  FE = .490055682684517
                        MN = .00302988813183617  SI = .00691442924890044
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00125366195459931  FE = .490055682684517
                        MN = .00302988813183617  SI = .00691442924890044
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0012541227432527  FE = .490055682684517
                        MN = .00302988813183617  SI = .00691442924890044
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00125473479150122  FE = .490055682684518
                        MN = .0030298881318363  SI = .00691442924889893
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   160.87030     DT =   160.47020     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00140660867390162  FE = .490055682684802
                        MN = .00302988813186406  SI = .00691442924858674
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   481.81071     DT =   320.94041     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0015631121798786  FE = .490055682685047
                        MN = .00302988813188851  SI = .00691442924831815
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   1123.6915     DT =   641.88082     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00176551489262203  FE = .490055682685355
                        MN = .00302988813191944  SI = .00691442924797909
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   2407.4532     DT =   1283.7616     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00204188977019142  FE = .490055682685768
                        MN = .00302988813196098  SI = .00691442924752467
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     1   seconds

 TIME =   4974.9764     DT =   2567.5233     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00242787670585282  FE = .490055682686336
                        MN = .00302988813201827  SI = .00691442924689948
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   10110.023     DT =   5135.0465     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00297316977084057  FE = .490055682687126
                        MN = .00302988813209826  SI = .00691442924602937
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   20380.116     DT =   10270.093     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0037491613580386  FE = .490055682688229
                        MN = .00302988813221057  SI = .00691442924481427
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   40920.302     DT =   20540.186     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00486019233189309  FE = .490055682689761
                        MN = .0030298881323684  SI = .00691442924312399
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0   seconds

 TIME =   82000.674     DT =   41080.372     SUM OF SQUARES =   0.0000000
```

```
 U-FRACTION IN SYSTEM:  C = .00646111106297823  FE = .490055682691865
                        MN = .0030298881325897  SI = .00691442924079908
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     1  seconds

 TIME =   164161.42     DT =   82160.744     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0087824743281453  FE = .490055682694715
                        MN = .00302988813290127  SI = .00691442923763758
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =   328482.91     DT =   164321.49     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0120233349524823  FE = .490055682698822
                        MN = .00302988813337276  SI = .00691442923305911
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]


output ignored...



... output resumed

  CPU time used in timestep                     0  seconds

 TIME =   15119280.     DT =   6738922.5     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .016530430864208  FE = .490055682867987
                        MN = .00302988815669835  SI = .00691442904056798
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =   28597125.     DT =   13477845.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165300519884718  FE = .490055682940724
                        MN = .00302988816674853  SI = .0069144289577808l
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     1  seconds

 TIME =   55552815.     DT =   26955690.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .016529737105878  FE = .490055683047884
                        MN = .00302988818132928  SI = .00691442883604053
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.10946419E+09 DT =   53911380.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .016529699237533  FE = .490055683206572
                        MN = .00302988820257483  SI = .00691442865610666
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.20883343E+09 DT =   99369240.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165306811403286  FE = .49005568342848
                        MN = .00302988823170669  SI = .00691442840506641
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.30830636E+09 DT =   99472929.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165348571162441  FE = .490055683611437
                        MN = .00302988825525438  SI = .0069144281985626
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     1  seconds

 TIME =  0.40784707E+09 DT =   99540711.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .016538250858666  FE = .490055683774507
                        MN = .00302988827589775  SI = .00691442801484847
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.50744329E+09 DT =   99596213.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165412919615189  FE = .490055683925116
                        MN = .00302988829469354  SI = .00691442784544345
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.60708721E+09 DT =   99643921.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165441256829561  FE = .490055684067096
                        MN = .00302988831219105  SI = .00691442768596585
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.70677300E+09 DT =   99685789.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .016546817769877  FE = .490055684202729
                        MN = .00302988832871668  SI = .00691442753380813
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     1  seconds

 TIME =  0.80649607E+09 DT =   99723077.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165494037752657  FE = .490055684333495
                        MN = .00302988834448347  SI = .00691442738727498
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.90625273E+09 DT =   99756653.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165519052639314  FE = .490055684460418
                        MN = .00302988835963936  SI = .0069144272451963
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

  CPU time used in timestep                     0  seconds

 TIME =  0.99706021E+09 DT =   90807478.     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0165543980726468  FE = .490055684573206
                        MN = .00302988837299422  SI = .0069144271190536
 TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]
```

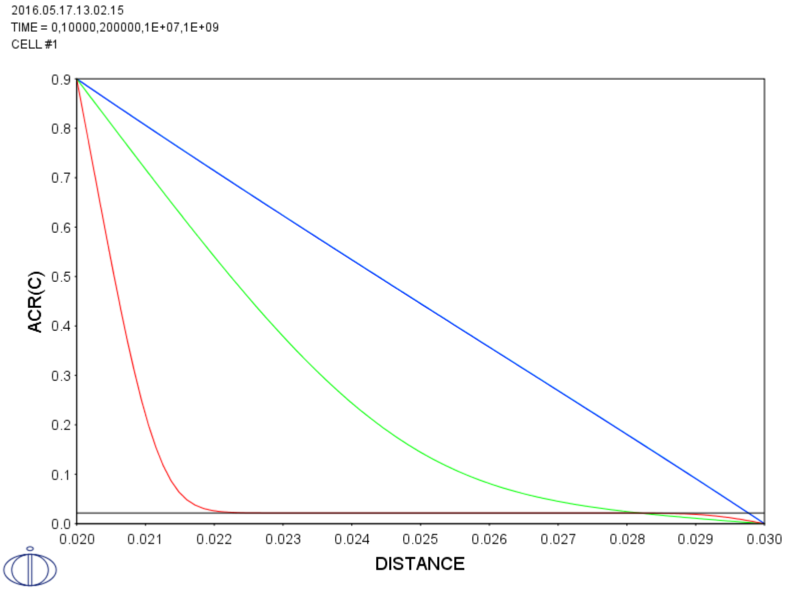
```
 CPU time used in timestep                    0  seconds

TIME =  0.10000000E+10 DT =   2939794.5     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0165570300715913  FE = .490055684576818
                       MN = .00302988837342096  SI = .00691442711501466
TOTAL SIZE OF SYSTEM:  .00314159265359 [m^2]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME   0.40010010
DELETING TIME-RECORD FOR TIME    160.87030
DELETING TIME-RECORD FOR TIME    481.81071
DELETING TIME-RECORD FOR TIME    1123.6915
DELETING TIME-RECORD FOR TIME    2407.4532
DELETING TIME-RECORD FOR TIME    4974.9764
DELETING TIME-RECORD FOR TIME    10110.023
DELETING TIME-RECORD FOR TIME    20380.116
DELETING TIME-RECORD FOR TIME    40920.302
DELETING TIME-RECORD FOR TIME    82000.674
DELETING TIME-RECORD FOR TIME    164161.42
DELETING TIME-RECORD FOR TIME    328482.91
DELETING TIME-RECORD FOR TIME    657125.89
DELETING TIME-RECORD FOR TIME    1314411.8
DELETING TIME-RECORD FOR TIME    2593116.0
DELETING TIME-RECORD FOR TIME    4748316.3
DELETING TIME-RECORD FOR TIME    8380357.3
DELETING TIME-RECORD FOR TIME    15119280.
DELETING TIME-RECORD FOR TIME    28597125.
DELETING TIME-RECORD FOR TIME    55552815.
DELETING TIME-RECORD FOR TIME   0.10946419E+09
DELETING TIME-RECORD FOR TIME   0.20883343E+09
DELETING TIME-RECORD FOR TIME   0.30830636E+09
DELETING TIME-RECORD FOR TIME   0.40784707E+09
DELETING TIME-RECORD FOR TIME   0.50744329E+09
DELETING TIME-RECORD FOR TIME   0.60708721E+09
DELETING TIME-RECORD FOR TIME   0.70677300E+09
DELETING TIME-RECORD FOR TIME   0.80649607E+09
DELETING TIME-RECORD FOR TIME   0.90625273E+09

KEEPING TIME-RECORD FOR TIME   0.99706021E+09
AND FOR TIME                   0.10000000E+10
WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa6-plot

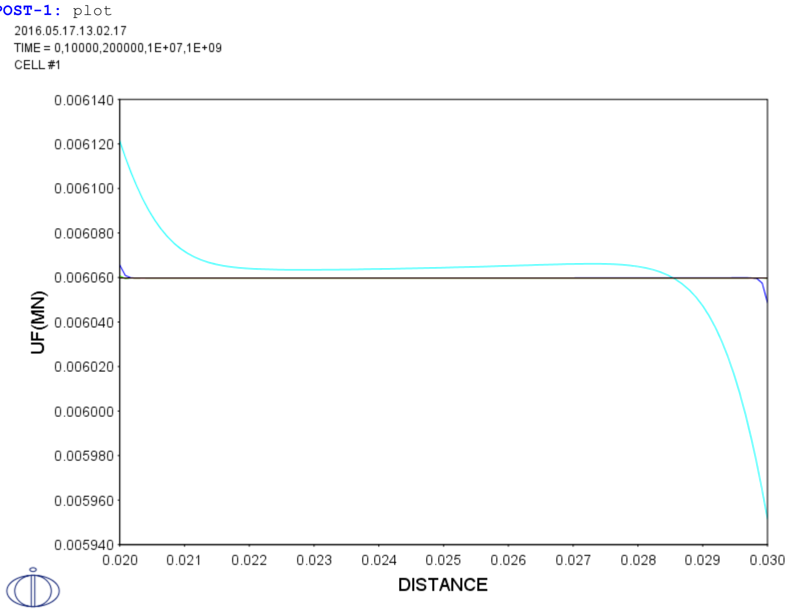
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa6\plot.DCM"
DIC>
DIC> @@ exa6_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+09
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC> read exa6
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT CONCENTRATION OF C AT DIFFERENT TIMES
POST-1: @@
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p c
POST-1: s-p-c time 0,1e4,2e5,1e7,1e9
POST-1:
POST-1: plot
```



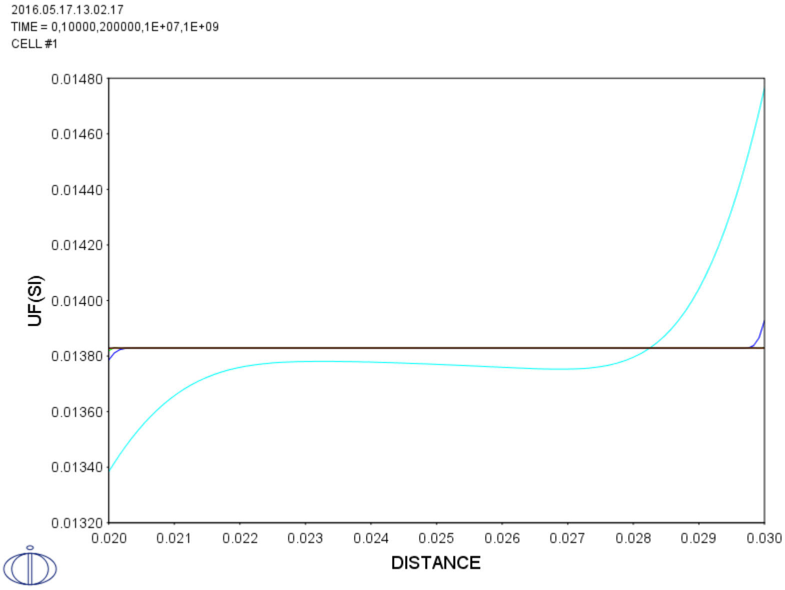

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ NOW, LET'S PLOT THE ACTIVITY OF C
POST-1: @@
POST-1:
POST-1: s-d-a y acr(c)
POST-1:
POST-1: plot
```



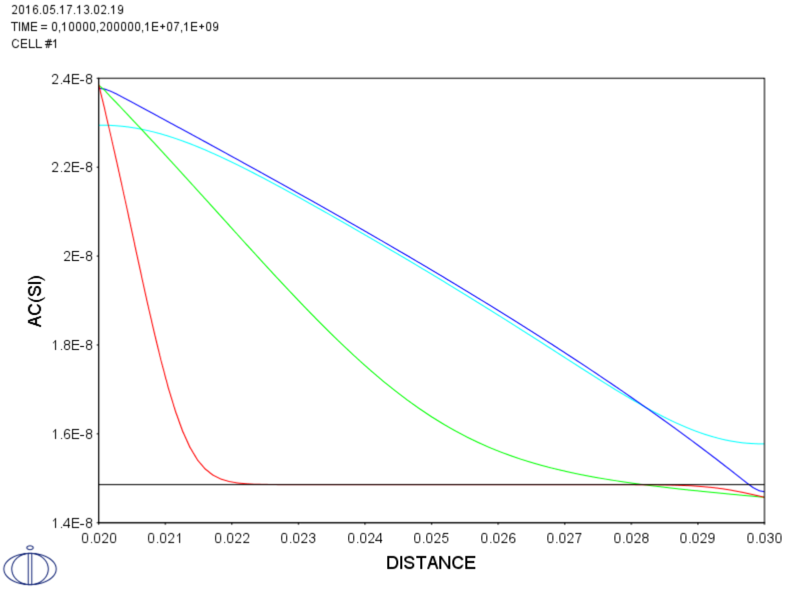

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ LET US LOOK AT THE MN AND SI PROFILES
POST-1: @@
POST-1: @@ WE PLOT THE U-FRACTION OF MN AND SI WHICH WILL BE INDEPENDENT
POST-1: @@ OF THE C-CONCENTRATION.
POST-1: @@
POST-1: s-d-a y u-f mn
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-d-a y u-f si
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ FINALLY, LET'S HAVE A LOOK ON THE ACTIVITY PROFILES OF SI
POST-1: @@
POST-1: s-d-a y ac(si)
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

exa7-setup

About Linked: Tue May 17 11:39:18 2016

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-
Examples\examples\exa7\setup.DCM" @@ DICTRA simulation of a homogenization heat treatment
SYS: @@
SYS: @@ The initial segregation profile is created from a
SYS: @@ Scheil calculation (see macro create_initial_profile.TCM).
SYS: @@ The command INPUT_SCHEIL_PROFILE in the DICTRA monitor
SYS: @@ performs most of the setup. Only time and temperature must
SYS: @@ be entered after the INPUT_SCHEIL_PROFILE command is
SYS: @@ executed.
SYS:
SYS:
SYS: @@ In this case only a single phase, ferrite, is entered in the simulation
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1   REJECTED
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-sys fe cr ni mn
 FE                      CR                      NI
 MN  DEFINED
TDB_FEDEMO: rej ph *
 LIQUID:L                BCC_A2                  CHI_A12
 FCC_A1                  HCP_A3                  LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: rest ph bcc
 BCC_A2  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'B.-J. Lee, Metall. Trans. A, 24A (1993), 1919-1933; Cr-Mn, Fe-Cr -Mn'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'W. Huang, Calphad, 13 (1989), 243-252; TRITA-MAC 388 (rev 1989); FE-MN'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'NPL, unpublished work (1989); Mn-Ni'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: app MFEDEMO
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe cr ni mn
 FE                      CR                      NI
 MN  DEFINED
APP: rej ph *
 BCC_A2                  FCC_A1   REJECTED
APP: rest ph bcc
 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
  'Assessed from data presented in Landholt-Bornstein, Vol. 26, ed. H.
      Mehrer, springer (1990); Impurity diff of Mn in bcc Fe.'
 -OK-
APP:
APP: go dict-mon
 NO TIME STEP DEFINED
DIC>
DIC>
DIC> @@ This command perform most of the setup
DIC> input_scheil_profile
 INFO: SCHEIL_REGION CREATED
FILE NAME /XF.TXT/: segregation_profile.TXT
ENTER WIDTH OF REGION /1/: 100e-6
 INFO: LINEAR GRID IN SCHEIL_REGION ENTERED WITH 100 GRID POINTS
 ENTER MAIN SOLID SOLUTION PHASE
PHASE NAME:: bcc#1
 INFO: COMPOSITION PROFILE ENTERED IN REGION
SHOULD MORE PHASES BE ENTERED IN THE REGION /NO/: n
 INFO: TO COMPLETE SETUP, ENTER TEMPERATURE AND
 SIMULATION TIME
DIC>
DIC>
DIC> @@ Enter heat treatment temperature
DIC> s-cond
```

```
GLOBAL OR BOUNDARY CONDITION /GLOBAL/: glob
VARIABLE : t
LOW TIME LIMIT /0/: 0 1473; * n
DIC>
DIC>
DIC> @@ Enter simulation time
DIC> se-si-ti
END TIME FOR INTEGRATION /.1/: 3600
AUTOMATIC TIMESTEP CONTROL /YES/: y
MAX TIMESTEP DURING INTEGRATION /360/: 360
INITIAL TIMESTEP : /1E-07/: 1e-7
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/: 1e-9
DIC>
DIC>
DIC> save exa7 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

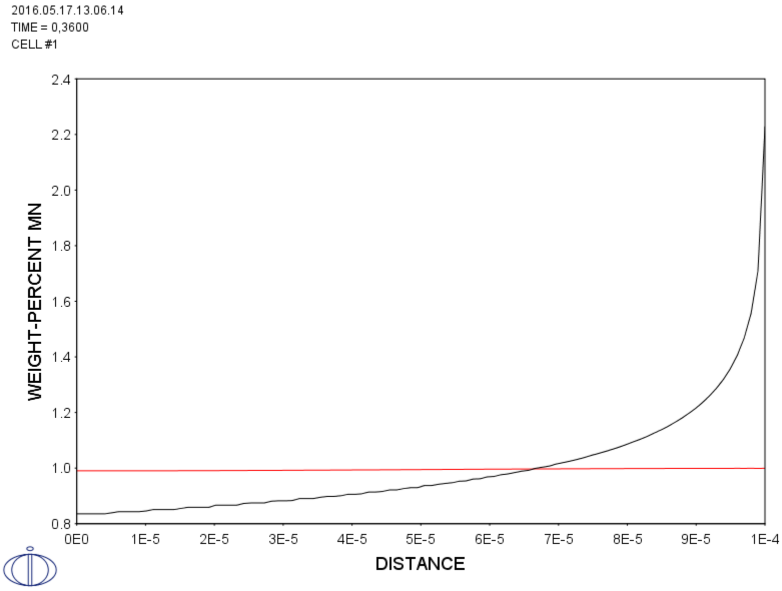
exa7-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa7\run.DCM" go dict-mon
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> read exa7
 OK
DIC>
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282829  NI = .0130905459141414
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282829  NI = .0130905459141414
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282829  NI = .0130905459141414
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282829  NI = .0130905459141414
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.41672379     DT =  0.41662369     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282773  NI = .0130905459141418
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   1.2499712     DT =  0.83324738     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424242  FE = .796498787878788
                        MN = .00999016878282775  NI = .0130905459141418
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    1  seconds

 TIME =   2.9164659     DT =   1.6664948     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424243  FE = .796498787878789
                        MN = .00999016878282916  NI = .0130905459141384
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   6.2494554     DT =   3.3329895     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424268  FE = .796498787878825
                        MN = .00999016878285739  NI = .0130905459140504
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   12.915434     DT =   6.6659790     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424346  FE = .796498787878952
                        MN = .00999016878291415  NI = .0130905459137879
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   26.247393     DT =   13.331958     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424366  FE = .796498787878903
                        MN = .00999016878296266  NI = .0130905459137678
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   52.911309     DT =   26.663916     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424359  FE = .796498787878652
                        MN = .00999016878321381  NI = .0130905459137756
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   106.23914     DT =   53.327832     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497424258  FE = .796498787878506
                        MN = .00999016878319633  NI = .0130905459140406
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    1  seconds

 TIME =   212.89481     DT =   106.65566     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497423953  FE = .796498787878027
                        MN = .00999016878329594  NI = .0130905459147245
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   426.20613     DT =   213.31133     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497423878  FE = .796498787878057
                        MN = .00999016878319387  NI = .0130905459148719
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   786.20613     DT =   360.00000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497423545  FE = .796498787877699
                        MN = .00999016878302824  NI = .0130905459157281
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =   1146.2061     DT =   360.00000     SUM OF SQUARES =   0.0000000
```

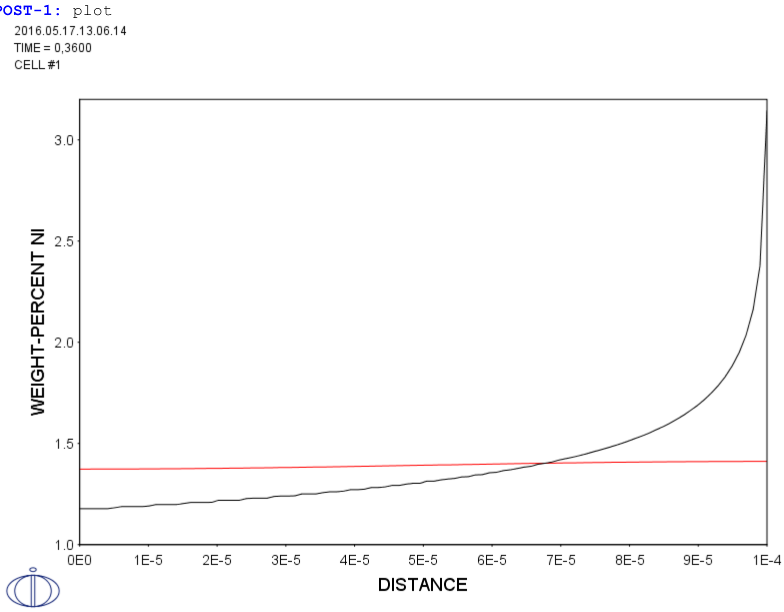
```
 U-FRACTION IN SYSTEM:  CR = .180420497423556  FE = .796498787877961
                        MN = .00999016878294436  NI = .0130905459155389
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    1506.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497423591  FE = .796498787878199
                        MN = .00999016878289028  NI = .0130905459153202
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    1866.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497423079  FE = .796498787877629
                        MN = .00999016878258566  NI = .0130905459167064
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    2226.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497422547  FE = .796498787875121
                        MN = .00999016878299424  NI = .0130905459193375
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    1  seconds

 TIME =    2586.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497421995  FE = .796498787871571
                        MN = .00999016878382109  NI = .0130905459226124
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    2946.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .18042049742169  FE = .796498787870128
                        MN = .00999016878407491  NI = .013090545924107
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    3306.2061     DT =    360.00000     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497421387  FE = .796498787868367
                        MN = .00999016878448678  NI = .0130905459257586
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds

 TIME =    3600.0000     DT =    293.79387     SUM OF SQUARES =    0.0000000
 U-FRACTION IN SYSTEM:  CR = .180420497421212  FE = .796498787867468
                        MN = .00999016878468089  NI = .0130905459266391
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.41672379
 DELETING TIME-RECORD FOR TIME    1.2499712
 DELETING TIME-RECORD FOR TIME    2.9164659
 DELETING TIME-RECORD FOR TIME    6.2494554
 DELETING TIME-RECORD FOR TIME    12.915434
 DELETING TIME-RECORD FOR TIME    26.247393
 DELETING TIME-RECORD FOR TIME    52.911309
 DELETING TIME-RECORD FOR TIME    106.23914
 DELETING TIME-RECORD FOR TIME    212.89481
 DELETING TIME-RECORD FOR TIME    426.20613
 DELETING TIME-RECORD FOR TIME    786.20613
 DELETING TIME-RECORD FOR TIME    1146.2061
 DELETING TIME-RECORD FOR TIME    1506.2061
 DELETING TIME-RECORD FOR TIME    1866.2061
 DELETING TIME-RECORD FOR TIME    2226.2061
 DELETING TIME-RECORD FOR TIME    2586.2061
 DELETING TIME-RECORD FOR TIME    2946.2061

 KEEPING TIME-RECORD FOR TIME    3306.2061
 AND FOR TIME                    3600.0000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exa7-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exa7\plot.DCM" go dict-mon
 TIME STEP AT TIME  3.60000E+03
DIC>
DIC> read exa7
 OK
DIC>
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: s-p-c time 0,3600
POST-1: s-d-a x d g
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p mn
POST-1: plot
```

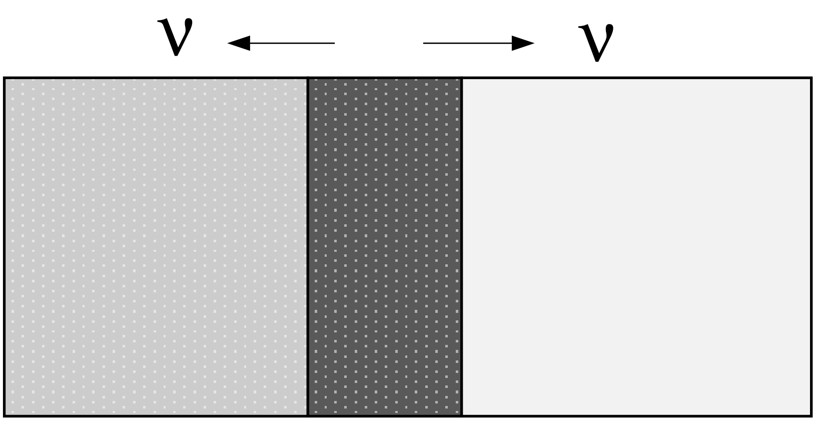



```
POST-1:
POST-1: @@ Hit enter for the next plot
POST-1:@?
POST-1:
POST-1: s-d-a y w-p ni
POST-1: plot
```



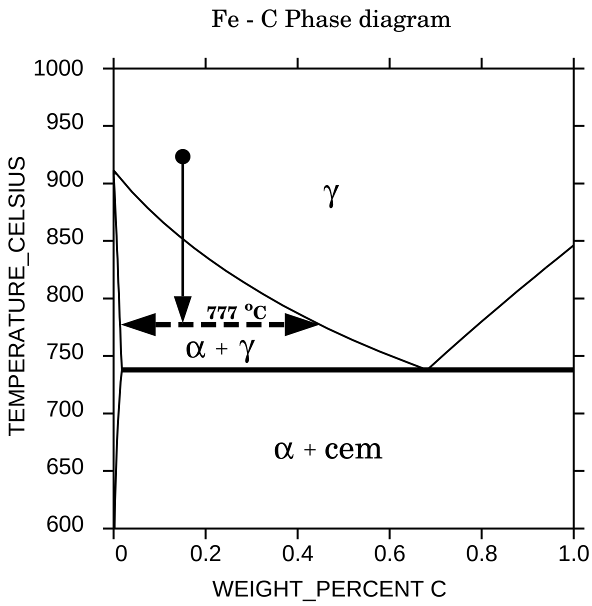

```
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

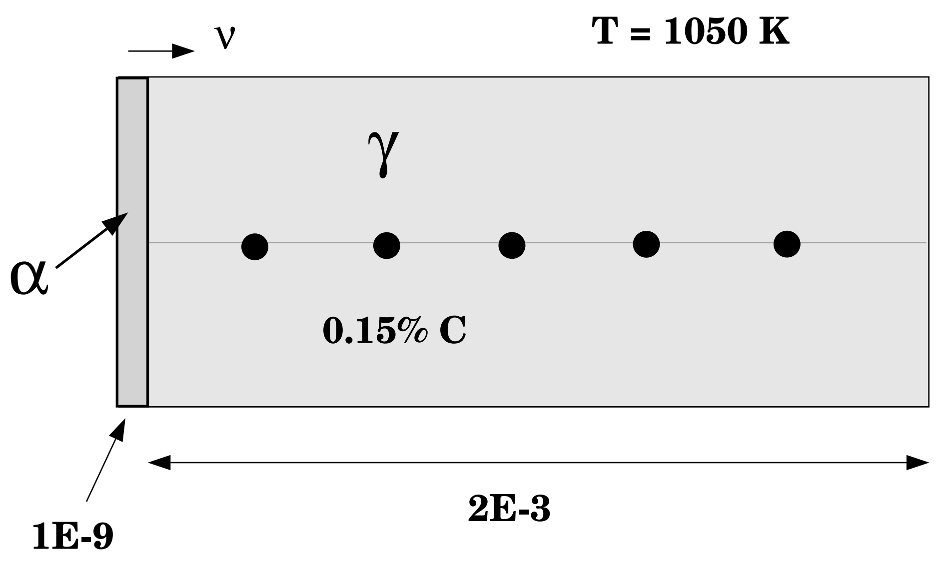
Moving boundary problems



Example b1a

γ to α transformation in a binary Fe - C alloy





exb1a-setup

```
About Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1a\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ Setup file for calculating a ferrite(BCC)/austenite(FCC)
SYS: @@ transformation in a binary Fe-C alloy. The initial state
SYS: @@ is an austenite of 2 mm thichness. The composition of the
SYS: @@ austenite is Fe-0.15wt%C. After austenitisation the
SYS: @@ specimen has been quenched down to 1050K. The system is
SYS: @@ assumed closed, so we do not set any boundary conditions
SYS: @@ (closed system is default). Ferrite is expected to grow
SYS: @@ into the austenite. For this reason we start with a thin
SYS: @@ region with ferrite adjacent to the austenite.
SYS: @@
SYS:
SYS: @@ exb1a_setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: @@
TDB_FEDEMO: def-sys fe c
 FE                      C  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: @@
TDB_FEDEMO: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph fcc bcc
 FCC_A1                  BCC_A2  RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: @@
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_FEDEMO: @@
TDB_FEDEMO: append
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
```

```
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: MFEDEMO
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe c
 FE                        C  DEFINED
APP: rej ph * all
 BCC_A2                    FCC_A1  REJECTED
APP: res ph fcc bcc
 FCC_A1                    BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
```

```
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> set-condition global T 0 1050; * N

DIC>
DIC> @@
DIC> @@ WE START BY ENTERING REGION ferrite AND austenite WHEREIN WE
DIC> @@ PUT THE BCC AND FCC PHASE RESPECTIVELY. THE ferrite REGION IS
DIC> @@ ASSUMED INITIALLY TO BE VERY THIN, 1E-9 METERS.
DIC> @@
DIC> enter-region
REGION NAME : ferrite
DIC>
DIC> enter-region
REGION NAME : austenite
ATTACH TO REGION NAMED /FERRITE/:
ATTACHED TO THE RIGHT OF FERRITE /YES/:
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO THE REGIONS.
DIC> @@
DIC> enter-grid
REGION NAME : /FERRITE/: ferrite
WIDTH OF REGION /1/: 1e-9
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 10
DIC>
DIC> enter-grid austenite
WIDTH OF REGION /1/: 20e-4
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /FERRITE/: ferrite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: bcc
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO BCC.
DIC> @@
DIC> enter-composition
REGION NAME : /FERRITE/: ferrite
PHASE NAME: /BCC_A2/: bcc
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.01
VALUE OF LAST POINT : /1E-2/: 0.01
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO FCC.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.15
VALUE OF LAST POINT : /0.15/: 0.15
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1e9
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /100000000/: 1e8
INITIAL TIMESTEP : /1E-07/: 1E-5
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/: 1E-5
DIC>
DIC>
DIC> @@
DIC> @@ WE USE IMPLICIT (1) TIME INTEGRATION IN THIS CASE INSTEAD OF THE
DIC> @@ MORE ACCURATE BUT LESS STABLE TRAPETZOIDAL METHOD WHICH IS THE
DIC> @@ DEFAULT METHOD.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1.0
```

```
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb1a Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

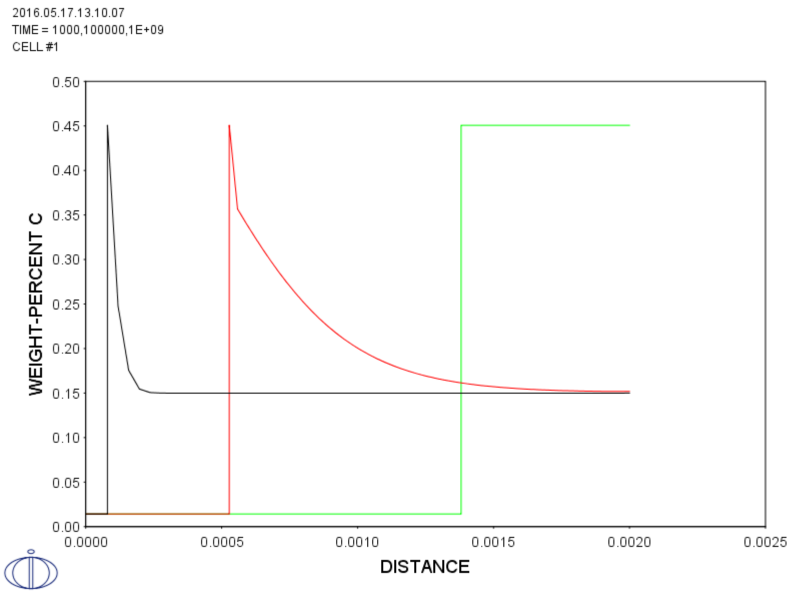
exb1a-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1a\run.DCM"
DIC>
DIC> @@ exb1a_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b1a
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb1a
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> simulate
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                   3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00698495590385912  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]
 U-FRACTION IN SYSTEM:  C = .00698495590385912  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: FERRITE
 3.010538010512301E-002        3.010525750452698E-002         8.73429491590867        6.430152741574597E-003        1.088329898161419E-
002        1.051437227511068E-006        5.743440325266627E-009        1.178785138973841E-015        1.400666910645825E-024
 TIME =  0.10000000E-04 DT =  0.10000000E-04 SUM OF SQUARES =  0.14006669E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS    4.7526995     AND    4.7526995
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.47527995E-04
 U-FRACTION IN SYSTEM:  C = .00698495582617669  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  12  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    0  seconds
 9.690361202918207E-004        9.692265256368234E-004        2.896523288692903E-006        1.678632812408759E-006        2.028472842468342E-
007        1.088890464990771E-010        3.846704448062238E-011        5.540201175632864E-012        1.010430811493044E-020
 TIME =  0.30000000E-04 DT =  0.20000000E-04 SUM OF SQUARES =  0.10104308E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.68730713E-07 AND   0.68730713E-07
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.47527996E-04
 U-FRACTION IN SYSTEM:  C = .00698495582718189  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  CPU time used in timestep                    0  seconds
 1.245102015014386E-006        1.243879535636653E-006        1.211416272534333E-010        1.544931360448411E-018
 TIME =  0.70000000E-04 DT =  0.40000000E-04 SUM OF SQUARES =  0.15449314E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.81209893E-07 AND   0.81209893E-07
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.47527999E-04
 U-FRACTION IN SYSTEM:  C = .006984955827182  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    0  seconds
 1.714218581361133E-008        1.697466232155603E-008        3.766544033508556E-015        1.942026823767841E-017
 TIME =  0.15000000E-03 DT =  0.80000000E-04 SUM OF SQUARES =  0.19420268E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.81211222E-07 AND   0.81211222E-07
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.47528006E-04
 U-FRACTION IN SYSTEM:  C = .00698495582718182  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  3  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    0  seconds
 1.710062946962381E-008        1.693310991721181E-008        4.639465320638915E-014        1.509473048315586E-020
 TIME =  0.31000000E-03 DT =  0.16000000E-03 SUM OF SQUARES =  0.15094730E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.81210553E-07 AND   0.81210553E-07
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.47528019E-04
 U-FRACTION IN SYSTEM:  C = .0069849558271818  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  5  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    0  seconds
 1.710977388578877E-008        1.694261933862132E-008        1.056019497312776E-014        8.897498076814288E-024

output ignored...



... output resumed

  CPU time used in timestep                    0  seconds
 4.865490156035480E-003        4.865454311493816E-003        6.582893196701601E-019
 TIME =  0.67592186E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.65828932E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.35461299E-14 AND  -0.35461299E-14
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.13804978E-02
 U-FRACTION IN SYSTEM:  C = .00697524927878742  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  7  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                    0  seconds
 2.130434936053536E-004        2.130641842460826E-004        9.682936824721111E-024
 TIME =  0.77592186E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.96829368E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.36815384E-14 AND   0.36815384E-14
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.13808660E-02
 U-FRACTION IN SYSTEM:  C = .00697149507645592  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]
```

```
  32  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                     0  seconds
 5.381921314889760E-005        5.383000993300233E-005        1.834891933744480E-021
 TIME =  0.87592186E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.18348919E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.10827503E-16 AND   0.10827503E-16
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.13808671E-02
 U-FRACTION IN SYSTEM:  C = .00697148403524442  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                     0  seconds
 6.682263988756765E-004        6.682375872831970E-004        1.053377883157667E-014        1.401144379975400E-027
 TIME =  0.97592186E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.14011444E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.11613767E-15 AND  -0.11613767E-15
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.13808555E-02
 U-FRACTION IN SYSTEM:  C = .00697160246512696  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                     0  seconds
 9.516159402577894E-005        9.515706532736865E-005        6.145392752784110E-019
 TIME =  0.10000000E+10 DT =   24078140.     SUM OF SQUARES =  0.61453928E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.59212809E-15 AND  -0.59212809E-15
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.13808412E-02
 U-FRACTION IN SYSTEM:  C = .00697174785269064  FE = 1
 TOTAL SIZE OF SYSTEM:  .002000001 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-04
 DELETING TIME-RECORD FOR TIME   0.30000000E-04
 DELETING TIME-RECORD FOR TIME   0.70000000E-04
 DELETING TIME-RECORD FOR TIME   0.15000000E-03
 DELETING TIME-RECORD FOR TIME   0.31000000E-03
 DELETING TIME-RECORD FOR TIME   0.63000000E-03
 DELETING TIME-RECORD FOR TIME   0.12700000E-02
 DELETING TIME-RECORD FOR TIME   0.25500000E-02
 DELETING TIME-RECORD FOR TIME   0.51100000E-02
 DELETING TIME-RECORD FOR TIME   0.10230000E-01
 DELETING TIME-RECORD FOR TIME   0.20470000E-01
 DELETING TIME-RECORD FOR TIME   0.40950000E-01
 DELETING TIME-RECORD FOR TIME   0.81910000E-01
 DELETING TIME-RECORD FOR TIME   0.16383000
 DELETING TIME-RECORD FOR TIME   0.32767000
 DELETING TIME-RECORD FOR TIME   0.65535000
 DELETING TIME-RECORD FOR TIME    1.3107100
 DELETING TIME-RECORD FOR TIME    2.6214300
 DELETING TIME-RECORD FOR TIME    5.2428700
 DELETING TIME-RECORD FOR TIME    10.485750
 DELETING TIME-RECORD FOR TIME    20.971510
 DELETING TIME-RECORD FOR TIME    41.943030
 DELETING TIME-RECORD FOR TIME    83.886070
 DELETING TIME-RECORD FOR TIME    167.77215
 DELETING TIME-RECORD FOR TIME    335.54431
 DELETING TIME-RECORD FOR TIME    671.08863
 DELETING TIME-RECORD FOR TIME    1342.1773
 DELETING TIME-RECORD FOR TIME    2684.3546
 DELETING TIME-RECORD FOR TIME    5368.7091
 DELETING TIME-RECORD FOR TIME    10737.418
 DELETING TIME-RECORD FOR TIME    21474.836
 DELETING TIME-RECORD FOR TIME    42949.673
 DELETING TIME-RECORD FOR TIME    85899.346
 DELETING TIME-RECORD FOR TIME    171798.69
 DELETING TIME-RECORD FOR TIME    343597.38
 DELETING TIME-RECORD FOR TIME    687194.77
 DELETING TIME-RECORD FOR TIME    1374389.5
 DELETING TIME-RECORD FOR TIME    2748779.1
 DELETING TIME-RECORD FOR TIME    5497558.1
 DELETING TIME-RECORD FOR TIME    10995116.
 DELETING TIME-RECORD FOR TIME    21990233.
 DELETING TIME-RECORD FOR TIME    43980465.
 DELETING TIME-RECORD FOR TIME    87960930.
 DELETING TIME-RECORD FOR TIME   0.17592186E+09
 DELETING TIME-RECORD FOR TIME   0.27592186E+09
 DELETING TIME-RECORD FOR TIME   0.37592186E+09
 DELETING TIME-RECORD FOR TIME   0.47592186E+09
 DELETING TIME-RECORD FOR TIME   0.57592186E+09
 DELETING TIME-RECORD FOR TIME   0.67592186E+09
 DELETING TIME-RECORD FOR TIME   0.77592186E+09
 DELETING TIME-RECORD FOR TIME   0.87592186E+09

 KEEPING TIME-RECORD FOR TIME   0.97592186E+09
 AND FOR TIME                  0.10000000E+10
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

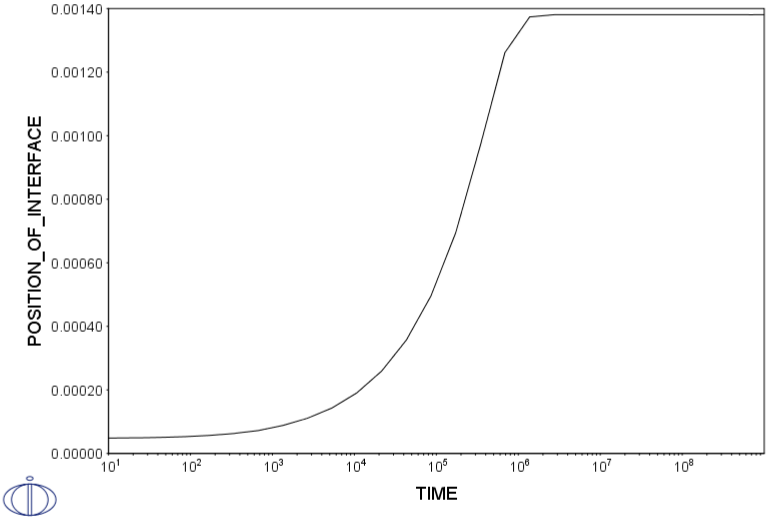
exb1a-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1a\plot.DCM"
DIC>
DIC> @@ exb1a_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b1a
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+09
DIC> read exb1a
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT CARBON CONCENTRATIONS AT DIFFERENT TIMES
POST-1: @@
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p c
POST-1: s-p-c time 1e3,1e5,1e9
POST-1:
POST-1: plot
```



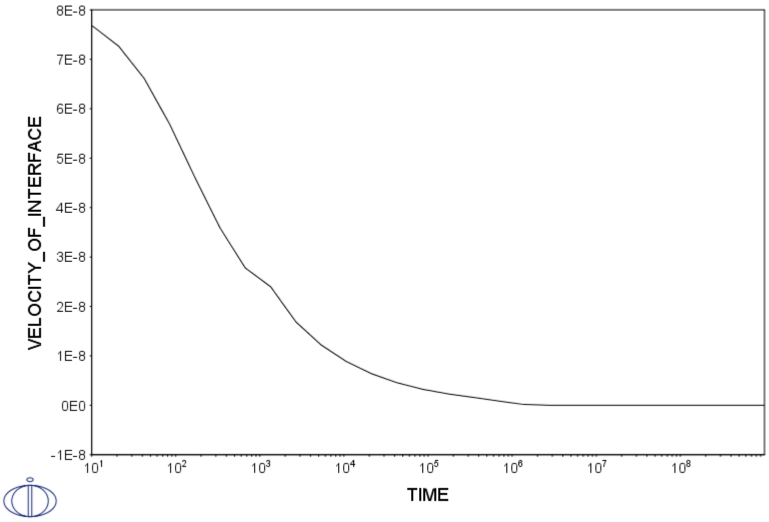

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ PLOT THE POSITION OF THE BCC/FCC INTERPHASE
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y
VARIABLE : pos
INTERFACE : aus
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1:
POST-1: set_axis_type
AXIS (X, Y OR Z) : x
AXIS TYPE /LINEAR/: log
POST-1:
POST-1: s-s-s
AXIS (X, Y OR Z) : x
AUTOMATIC SCALING  (Y OR N) /N/: n
MIN VALUE : 10
MAX VALUE : 1e9
POST-1:
POST-1: plot
```

2016.05.17.13.10.07
LOWER INTERFACE OF REGION "AUSTENITE#1"
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ PLOT THE VELOCITY OF THE BCC/FCC INTERPHASE
POST-1: @@
POST-1: s-d-a
AXIS (X, Y OR Z) : y
VARIABLE : velocity
INTERFACE : aus
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1:
POST-1: plot
```

2016.05.17.13.10.08
LOWER INTERFACE OF REGION "AUSTENITE#1"
CELL #1

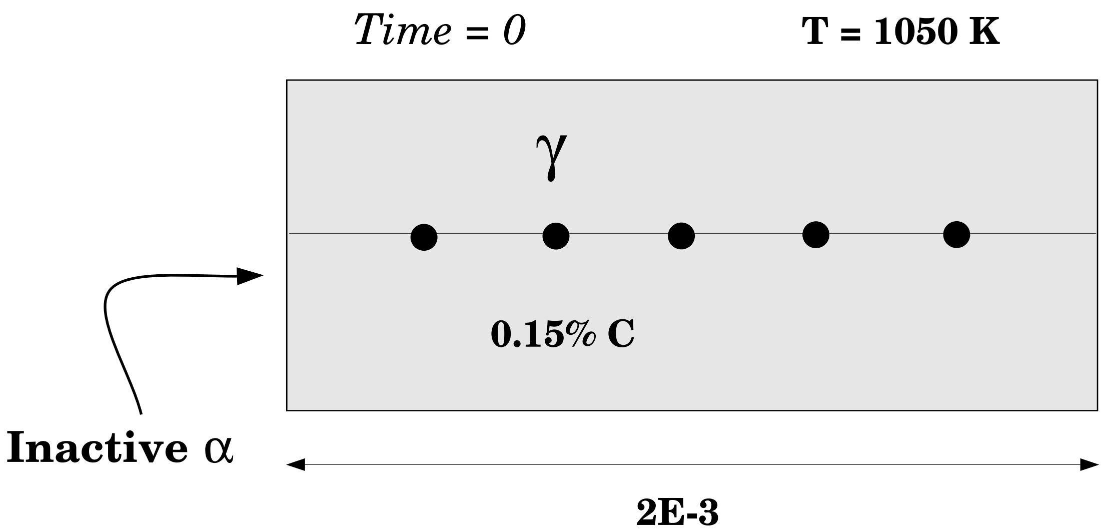


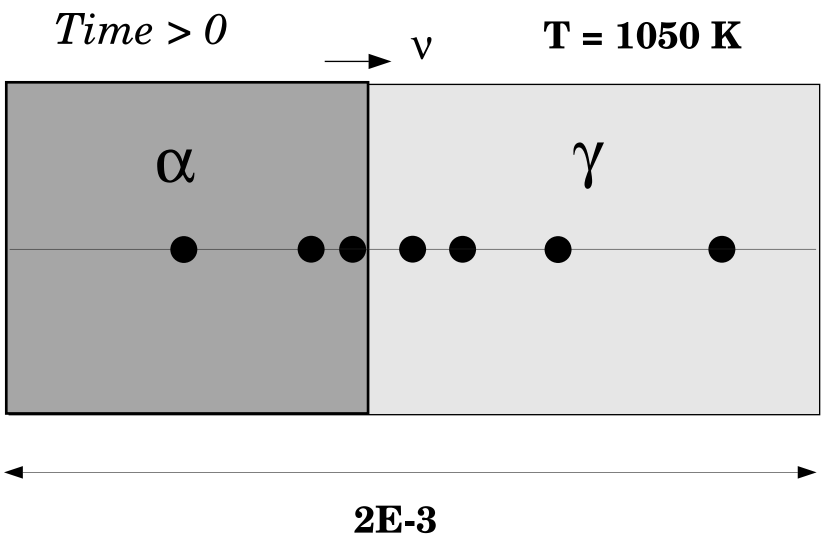

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b1b

γ to α transformation in a binary Fe - C alloy

(Inactive α)





exb1b-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1b\setup.DCM" @@
SYS: @@ Moving boundary problem.
SYS: @@ Same problem as in exb1a but now we set up the problem with
SYS: @@ ferrite as an inactive phase adjacent to the initial austenite.
SYS: @@
SYS:
SYS: @@ exb1b_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: @@
TDB_FEDEMO: def-sys fe c
 FE                      C  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: @@
TDB_FEDEMO: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph fcc bcc
 FCC_A1                  BCC_A2  RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: @@
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
```

```
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: MFEDEMO
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe c
 FE                     C  DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph fcc bcc
 FCC_A1                 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
```

```
DIC> set-cond glob T 0 1050; * N

DIC>
DIC> @@
DIC> @@ WE START BY ENTERING REGION austenite WHEREIN WE PUT THE fcc PHASE
DIC> @@
DIC> enter-region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 20e-4
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inactive
ATTACH TO REGION NAMED /AUSTENITE/: austenite
ATTACHED TO THE RIGHT OF AUSTENITE /YES/: no
PHASE NAME: /NONE/: bcc
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO FCC.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.15
VALUE OF LAST POINT : /0.15/: 0.15
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 1e9
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ WE USE IMPLICIT (1) TIME INTEGRATION IN THIS CASE INSTEAD OF THE
DIC> @@ MORE ACCURATE BUT LESS STABLE TRAPETZOIDAL METHOD WHICH IS THE
DIC> @@ DEFAULT METHOD.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1.0
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb1b Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

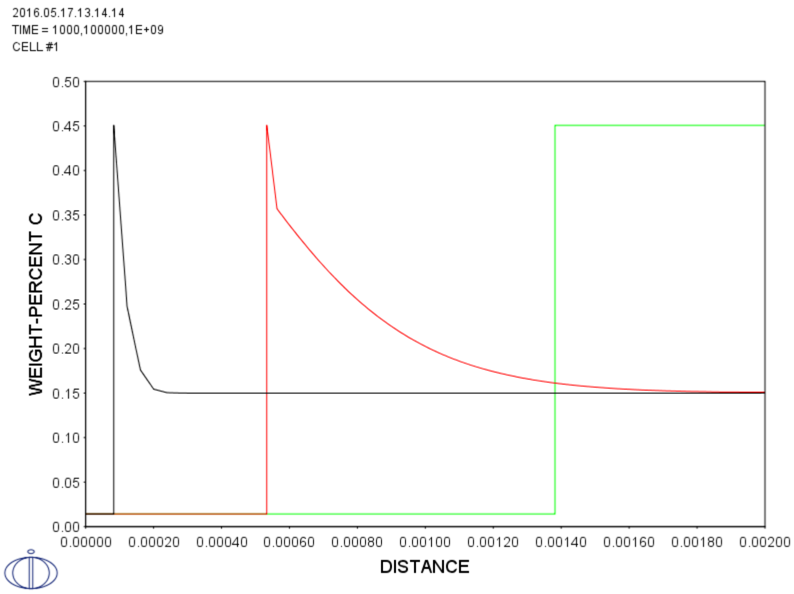
exb1b-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1b\run.DCM"
DIC>
DIC> @@ exb1b_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b1b
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb1b
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
 INFO: PHASE BCC_A2 IS SCHEDULED TO APPEAR
   REGION STATUS CHANGE, ITERATING: TIME=  0.50000000E-07
   REGION STATUS CHANGE, ITERATING: TIME=  0.25000000E-07
 TIME =  0.25000000E-07 DT =  0.25000000E-07
 U-FRACTION IN SYSTEM:  C = .0069849591638311  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE

 KEEPING TIME-RECORD FOR TIME    0.0000000
 AND FOR TIME                   0.25000000E-07
 WORKSPACE RECLAIMED
WIDTH OF NEW REGION R_BCC_A2 /1E-06/:
  Trying old scheme                        3
 START VALUE(S) FOR INTERFACE #2 R_BCC_A2/AUSTENITE, CELL #1
 -----------------------------------------------------------
VELOCITY /1/:
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                        3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00698156310125388  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
 U-FRACTION IN SYSTEM:  C = .00698156310125388  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
  1  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: R_BCC_A2
  2.75176053686623         2.75176116101698         393.018524826940         3.01758588658463         2.94562519077589         4.2088497499798

     ERROR RETURN FROM NS01A BECAUSE    5 CALLS OF CALFUN FAILED TO IMPROVE THE RESIDUALS

output ignored...


... output resumed

  31  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                     0  seconds
 2.339797817510186E-006        2.340226176090236E-006        2.188952073508997E-023
 TIME =  0.60995116E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.21889521E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.63501385E-15 AND  -0.63501385E-15
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13821621E-02
 U-FRACTION IN SYSTEM:  C = .00695827135120989  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                     0  seconds
 1.529107018043806E-006        1.529420923631504E-006        4.782898597443389E-024
 TIME =  0.70995116E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.47828986E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.15995671E-16 AND  -0.15995671E-16
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13821605E-02
 U-FRACTION IN SYSTEM:  C = .00695828766260135  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                     0  seconds
 5.244660515058744E-003        5.244706821005868E-003        2.591482083715306E-017
 TIME =  0.80995116E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.25914821E-16
```

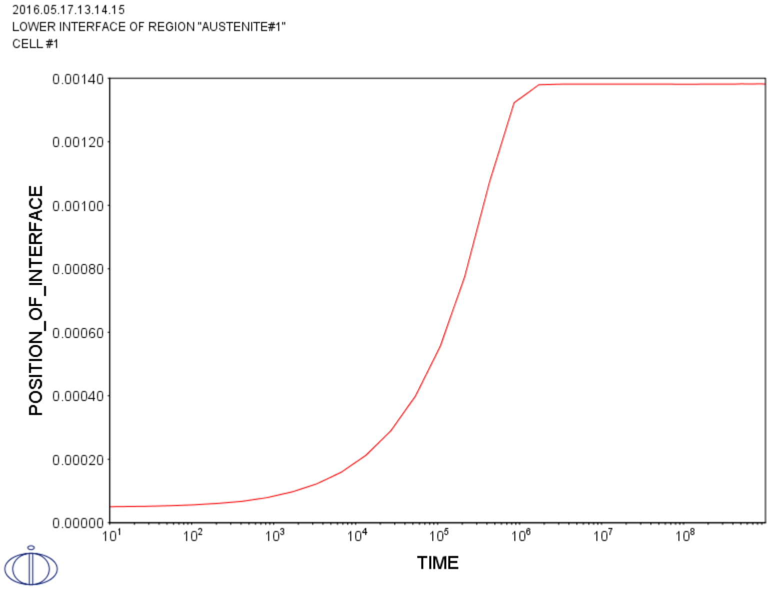
```
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.33941561E-15 AND   0.33941561E-15
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13821944E-02
 U-FRACTION IN SYSTEM:  C = .00695794154766557  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                      0  seconds
 4.582523979827140E-005        4.583442482791265E-005        2.451679519512417E-027
 TIME =  0.90995116E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.24516795E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.70698746E-18 AND   0.70698746E-18
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13821945E-02
 U-FRACTION IN SYSTEM:  C = .00695794082672409  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                      0  seconds
  1.08831955356856         1.08831958304909        4.940774416706345E-009        6.074722008267554E-028
 TIME =  0.10000000E+10 DT =    90048837.     SUM OF SQUARES =  0.60747220E-27
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.51144724E-14 AND  -0.51144724E-14
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13817339E-02
 U-FRACTION IN SYSTEM:  C = .00696263725290502  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.25000000E-07
 DELETING TIME-RECORD FOR TIME   0.12500000E-06
 DELETING TIME-RECORD FOR TIME   0.10012500E-03
 DELETING TIME-RECORD FOR TIME   0.30012500E-03
 DELETING TIME-RECORD FOR TIME   0.70012500E-03
 DELETING TIME-RECORD FOR TIME   0.15001250E-02
 DELETING TIME-RECORD FOR TIME   0.31001250E-02
 DELETING TIME-RECORD FOR TIME   0.63001250E-02
 DELETING TIME-RECORD FOR TIME   0.12700125E-01
 DELETING TIME-RECORD FOR TIME   0.25500125E-01
 DELETING TIME-RECORD FOR TIME   0.51100125E-01
 DELETING TIME-RECORD FOR TIME   0.10230012
 DELETING TIME-RECORD FOR TIME   0.20470012
 DELETING TIME-RECORD FOR TIME   0.40950012
 DELETING TIME-RECORD FOR TIME   0.81910012
 DELETING TIME-RECORD FOR TIME    1.6383001
 DELETING TIME-RECORD FOR TIME    3.2767001
 DELETING TIME-RECORD FOR TIME    6.5535001
 DELETING TIME-RECORD FOR TIME    13.107100
 DELETING TIME-RECORD FOR TIME    26.214300
 DELETING TIME-RECORD FOR TIME    52.428700
 DELETING TIME-RECORD FOR TIME    104.85750
 DELETING TIME-RECORD FOR TIME    209.71510
 DELETING TIME-RECORD FOR TIME    419.43030
 DELETING TIME-RECORD FOR TIME    838.86070
 DELETING TIME-RECORD FOR TIME    1677.7215
 DELETING TIME-RECORD FOR TIME    3355.4431
 DELETING TIME-RECORD FOR TIME    6710.8863
 DELETING TIME-RECORD FOR TIME    13421.773
 DELETING TIME-RECORD FOR TIME    26843.546
 DELETING TIME-RECORD FOR TIME    53687.091
 DELETING TIME-RECORD FOR TIME    107374.18
 DELETING TIME-RECORD FOR TIME    214748.36
 DELETING TIME-RECORD FOR TIME    429496.73
 DELETING TIME-RECORD FOR TIME    858993.46
 DELETING TIME-RECORD FOR TIME    1717986.9
 DELETING TIME-RECORD FOR TIME    3435973.8
 DELETING TIME-RECORD FOR TIME    6871947.7
 DELETING TIME-RECORD FOR TIME    13743895.
 DELETING TIME-RECORD FOR TIME    27487791.
 DELETING TIME-RECORD FOR TIME    54975581.
 DELETING TIME-RECORD FOR TIME   0.10995116E+09
 DELETING TIME-RECORD FOR TIME   0.20995116E+09
 DELETING TIME-RECORD FOR TIME   0.30995116E+09
 DELETING TIME-RECORD FOR TIME   0.40995116E+09
 DELETING TIME-RECORD FOR TIME   0.50995116E+09
 DELETING TIME-RECORD FOR TIME   0.60995116E+09
 DELETING TIME-RECORD FOR TIME   0.70995116E+09
 DELETING TIME-RECORD FOR TIME   0.80995116E+09

 KEEPING TIME-RECORD FOR TIME   0.90995116E+09
 AND FOR TIME                  0.10000000E+10
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb1b-plot

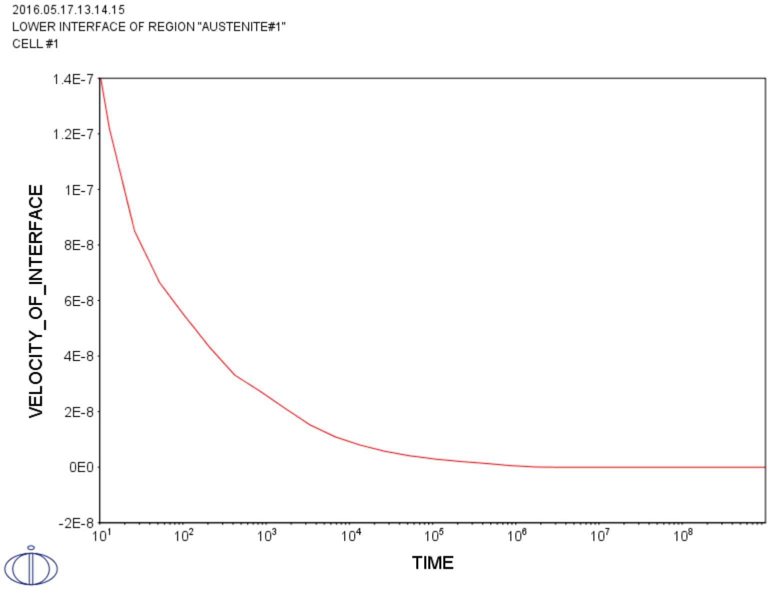
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1b\plot.DCM"
DIC>
DIC> @@ exb1b_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b1b
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+09
DIC> read exb1b
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT CARBON CONCENTRATIONS AT DIFFERENT TIMES
POST-1: @@
POST-1: s-d-a y w-p c
POST-1: s-d-a x dist glob
  INFO: Distance is set as independent variable
POST-1: s-p-c time 1e3,1e5,1e9
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ PLOT THE POSITION OF THE BCC/FCC INTERPHASE
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y
VARIABLE : pos
INTERFACE : aus
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1:
POST-1: set_axis_type
AXIS (X, Y OR Z) : x
AXIS TYPE /LINEAR/: log
POST-1:
POST-1: s-s-s
AXIS (X, Y OR Z) : x
AUTOMATIC SCALING  (Y OR N) /N/: n
MIN VALUE : 10
MAX VALUE : 1e9
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ PLOT THE VELOCITY OF THE BCC/FCC INTERPHASE
POST-1: @@
POST-1: s-d-a
AXIS (X, Y OR Z) : y
VARIABLE : velocity
INTERFACE : aus
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1:
POST-1: plot
```

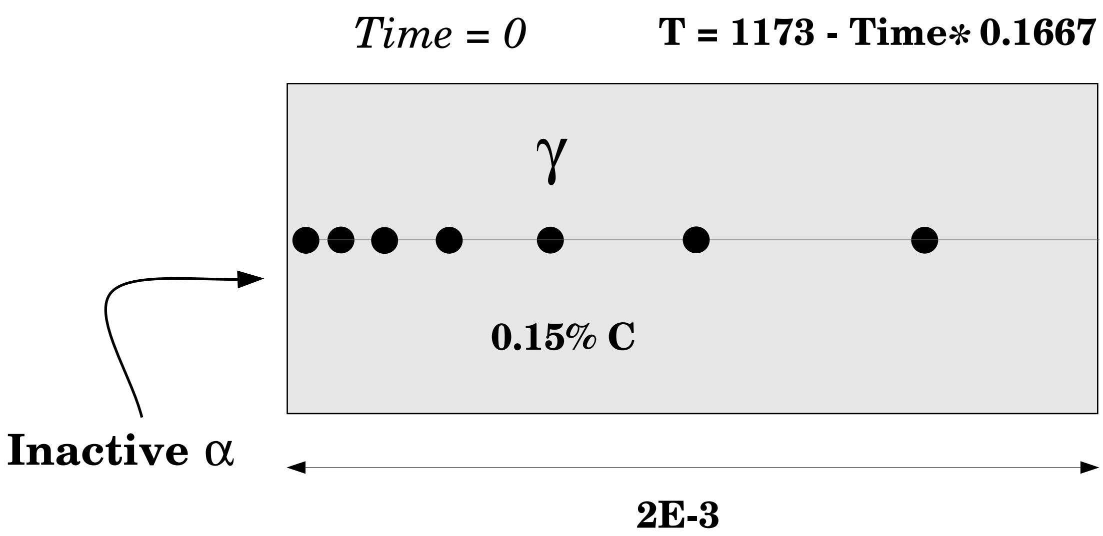


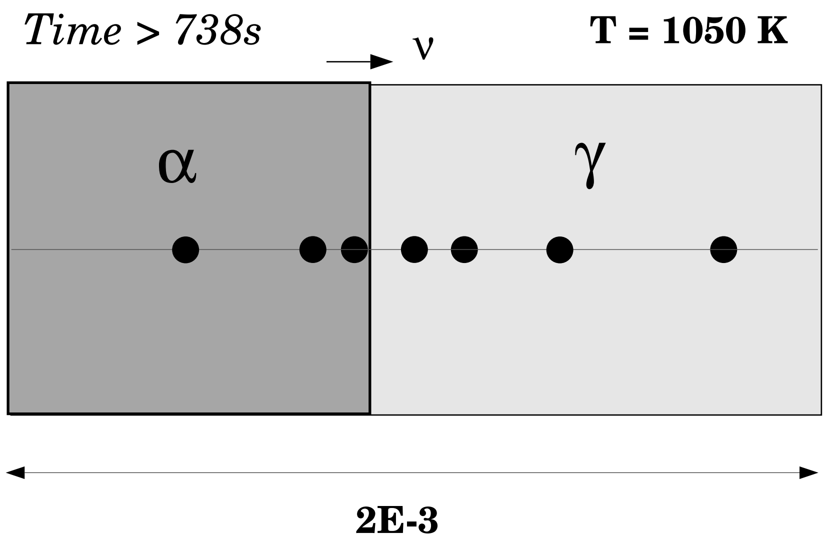

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b1c

γ to α transformation in a binary Fe - C alloy

(Gradual cooling down to 1050 K)





exb1c-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1c\setup.DCM"SYS: @@
SYS: @@ Moving boundary problems.
SYS: @@ Same problem as in exb1a and exb1b but now we start the simulation at
SYS: @@ a higher temperature and we assume a gradual cooling down to 1050 K.
SYS: @@ When we reach 1050 K we keep the temperature constant and thus have an
SYS: @@ isothermal transformation. As in exb1b we have ferrite as an inactive
SYS: @@ phase adjacent to the initial austenite.
SYS: @@
SYS:
SYS: @@ exb1c_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE THE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: @@
TDB_FEDEMO: def-sys fe c
 FE                      C  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: @@
TDB_FEDEMO: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph fcc bcc
 FCC_A1                  BCC_A2  RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: @@
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
```

```
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: MFEDEMO
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe c
 FE                     C  DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph fcc bcc
 FCC_A1                 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
```

```
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> @@ WE ASSUME THAT THE COOLING RATE IS 10K/MINUTE DOWN TO 1050K.
DIC> @@
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: glob
VARIABLE : T
LOW TIME LIMIT /0/: 0
T(TIME,X)= 1173-time*0.1667;
HIGH TIME LIMIT /*/: 738
ANY MORE RANGES /N/: y
T(TIME,X)= 1050;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC> @@
DIC> @@ WE START BY ENTERING REGION austenite WHEREIN WE PUT THE fcc PHASE
DIC> @@
DIC> enter-region
REGION NAME : austenite
DIC>
DIC> @@
DIC> @@ ENTER GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /AUSTENITE/: austenite
WIDTH OF REGION /1/: 20e-4
TYPE /LINEAR/: geometric
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE : 1.05
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /AUSTENITE/: austenite
ATTACHED TO THE RIGHT OF AUSTENITE /YES/: no
PHASE NAME: /NONE/: bcc
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO FCC.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.15
VALUE OF LAST POINT : /0.15/: 0.15
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 738
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /73.8/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ WE USE IMPLICIT (1) TIME INTEGRATION IN THIS CASE INSTEAD OF THE
DIC> @@ MORE ACCURATE BUT LESS STABLE TRAPETZOIDAL METHOD WHICH IS THE
DIC> @@ DEFAULT METHOD.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1.0
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb1c Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb1c-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1c\run.DCM"
DIC>
DIC> @@ exb1c_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE exb1c
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb1c
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   7.3675051     DT =   6.9674050     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   21.302315     DT =   13.934810     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   49.171935     DT =   27.869620     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   86.071935     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   122.97193     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   159.87193     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   196.77193     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   233.67193     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds

 TIME =   270.57193     DT =   36.900000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                    0  seconds
 INFO: PHASE BCC_A2 IS SCHEDULED TO APPEAR
   REGION STATUS CHANGE, ITERATING: TIME=   289.02193
   REGION STATUS CHANGE, ITERATING: TIME=   279.79693
   REGION STATUS CHANGE, ITERATING: TIME=   284.40943
   REGION STATUS CHANGE, ITERATING: TIME=   286.71568
   REGION STATUS CHANGE, ITERATING: TIME=   287.86881
   REGION STATUS CHANGE, ITERATING: TIME=   288.44537
   REGION STATUS CHANGE, ITERATING: TIME=   288.73365
 TIME =   288.73365     DT =   18.161719
 U-FRACTION IN SYSTEM:  C = .00698495916383109  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

 MUST SAVE WORKSPACE ON FILE
```

```
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.40010010
 DELETING TIME-RECORD FOR TIME    7.3675051
 DELETING TIME-RECORD FOR TIME    21.302315
 DELETING TIME-RECORD FOR TIME    49.171935
 DELETING TIME-RECORD FOR TIME    86.071935
 DELETING TIME-RECORD FOR TIME    122.97193
 DELETING TIME-RECORD FOR TIME    159.87193
 DELETING TIME-RECORD FOR TIME    196.77193
 DELETING TIME-RECORD FOR TIME    233.67193

 KEEPING TIME-RECORD FOR TIME    270.57193
 AND FOR TIME                    288.73365
 WORKSPACE RECLAIMED
```

WIDTH OF NEW REGION R_BCC_A2 /1E-06/:

```
  Trying old scheme                        3
```

output ignored...

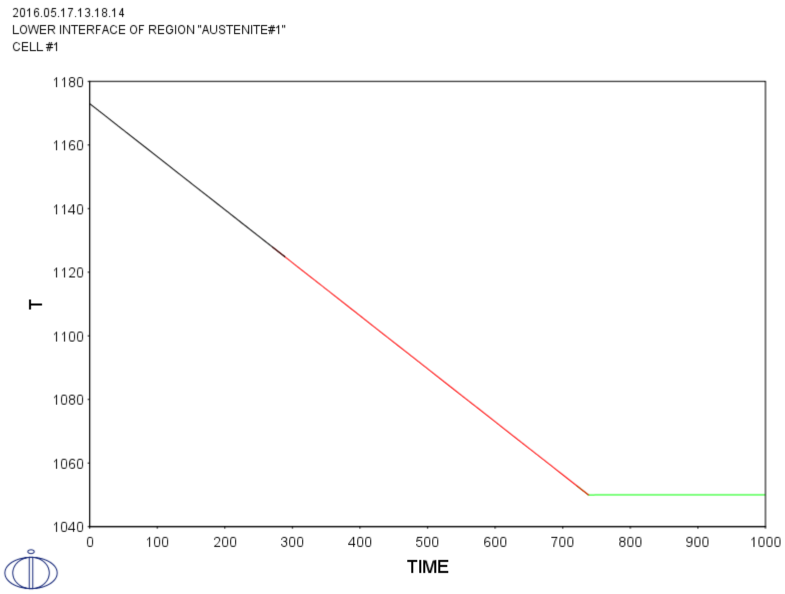
... output resumed

```
 7.287111442784800E-005        7.287778681727961E-005        5.022869839615039E-028
 TIME =  0.54091645E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.50228698E-27
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.22693946E-16 AND  -0.22693946E-16
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13828446E-02
 U-FRACTION IN SYSTEM:  C = .00695131116014133  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                    0  seconds
 3.489476905019513E-005        3.490012802192909E-005        9.530915970715003E-021
 TIME =  0.64091645E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.95309160E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.67241386E-17 AND   0.67241386E-17
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13828453E-02
 U-FRACTION IN SYSTEM:  C = .00695130430328151  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                    0  seconds
 2.265391429959814E-002        2.265395475561298E-002        3.764045187030191E-024
 TIME =  0.74091645E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.37640452E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.73140476E-15 AND  -0.73140476E-15
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13827722E-02
 U-FRACTION IN SYSTEM:  C = .00695205014442546  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                    0  seconds
 8.812948448948515E-006        8.813816416944759E-006        2.344562395885499E-024
 TIME =  0.84091645E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.23445624E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.73882339E-15 AND   0.73882339E-15
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13828460E-02
 U-FRACTION IN SYSTEM:  C = .0069512967382202  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  32  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                    0  seconds
 3.662262447934445E-006        3.662827653946400E-006        1.232224287092140E-020
 TIME =  0.94091645E+09 DT =  0.10000000E+09 SUM OF SQUARES =  0.12322243E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.21428427E-15 AND  -0.21428427E-15
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13828246E-02
 U-FRACTION IN SYSTEM:  C = .00695151525201083  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                    0  seconds
 2.788425520166119E-005        2.788877857103501E-005        2.450130199238436E-022
 TIME =  0.10000000E+10 DT =   59083546.     SUM OF SQUARES =  0.24501302E-21
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.48918657E-16 AND   0.48918657E-16
 POSITION OF INTERFACE R_BCC_A2 / AUSTENITE IS   0.13828275E-02
 U-FRACTION IN SYSTEM:  C = .00695148577865229  FE = 1
 TOTAL SIZE OF SYSTEM:  .002 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    721.20154
 DELETING TIME-RECORD FOR TIME    738.00000
 DELETING TIME-RECORD FOR TIME    754.79846
 DELETING TIME-RECORD FOR TIME    788.39539
 DELETING TIME-RECORD FOR TIME    855.58925
 DELETING TIME-RECORD FOR TIME    989.97697
 DELETING TIME-RECORD FOR TIME    1258.7524
 DELETING TIME-RECORD FOR TIME    1796.3033
 DELETING TIME-RECORD FOR TIME    2871.4050
 DELETING TIME-RECORD FOR TIME    5021.6084
 DELETING TIME-RECORD FOR TIME    9322.0153
 DELETING TIME-RECORD FOR TIME    17922.829
 DELETING TIME-RECORD FOR TIME    35124.457
 DELETING TIME-RECORD FOR TIME    69527.712
 DELETING TIME-RECORD FOR TIME    138334.22
 DELETING TIME-RECORD FOR TIME    275947.24
 DELETING TIME-RECORD FOR TIME    551173.28
 DELETING TIME-RECORD FOR TIME    1101625.4
 DELETING TIME-RECORD FOR TIME    2202529.5
 DELETING TIME-RECORD FOR TIME    4404337.9
 DELETING TIME-RECORD FOR TIME    8807954.5
 DELETING TIME-RECORD FOR TIME    17615188.
 DELETING TIME-RECORD FOR TIME    35229654.
 DELETING TIME-RECORD FOR TIME    70458588.
 DELETING TIME-RECORD FOR TIME   0.14091645E+09
 DELETING TIME-RECORD FOR TIME   0.24091645E+09
 DELETING TIME-RECORD FOR TIME   0.34091645E+09
 DELETING TIME-RECORD FOR TIME   0.44091645E+09
```

```
 DELETING TIME-RECORD FOR TIME   0.54091645E+09
 DELETING TIME-RECORD FOR TIME   0.64091645E+09
 DELETING TIME-RECORD FOR TIME   0.74091645E+09
 DELETING TIME-RECORD FOR TIME   0.84091645E+09

 KEEPING TIME-RECORD FOR TIME   0.94091645E+09
 AND FOR TIME                   0.10000000E+10
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

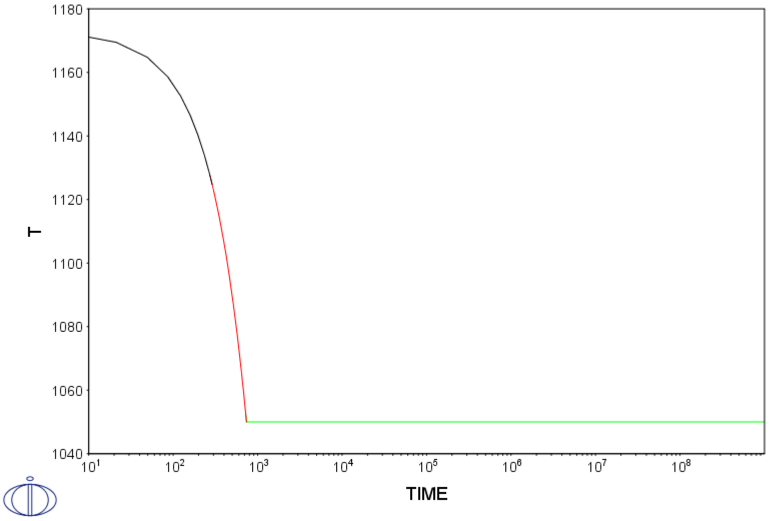
exb1c-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb1c\plot.DCM"
DIC>
DIC> @@ exb1c_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b1c
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+09
DIC> read exb1c
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ PLOT TEMPERATURE VS. TIME
POST-1: @@
POST-1: s-d-a y t
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-p-c
CONDITION /TIME/: interface
INTERFACE : austenite
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1: s-s-s x n 0 1000
POST-1:
POST-1: plot
```



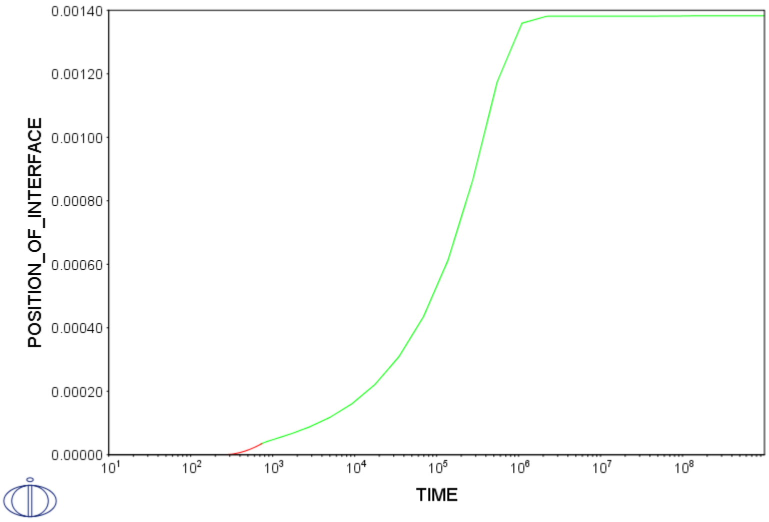

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT VS. LOG TIME
POST-1: @@
POST-1: set-axis-type x log
POST-1: s-s-s x n 10 1e9
POST-1:
POST-1: plot
```

2016.05.17.13.18.14
LOWER INTERFACE OF REGION "AUSTENITE#1"
CELL #1



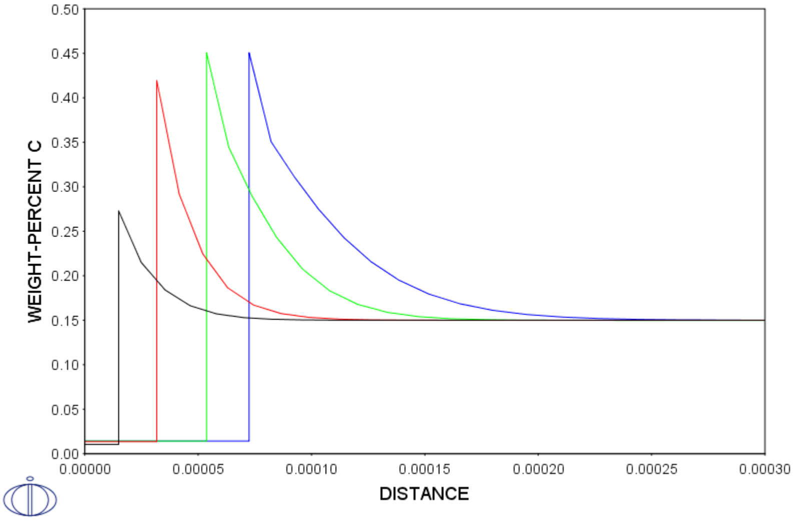

```
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE POSITION OF THE BCC/FCC INTERPHASE
POST-1: @@
POST-1: s-d-a
AXIS (X, Y OR Z) : y
VARIABLE : position
INTERFACE : austenite
UPPER OR LOWER INTERFACE OF REGION AUSTENITE#1 /LOWER/: lower
POST-1:
POST-1: plot
```

2016.05.17.13.18.14
LOWER INTERFACE OF REGION "AUSTENITE#1"
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE CARBON CONCENTRATION VS. DISTANCE
POST-1: @@
POST-1: s-d-a y w-p c
POST-1: s-d-a x dis glob
  INFO: Distance is set as independent variable
POST-1: s-p-c time 500,700,1200,2000
POST-1: set-axis-type x lin
POST-1: s-s-s x n 0 3e-4
POST-1:
POST-1: plot
```

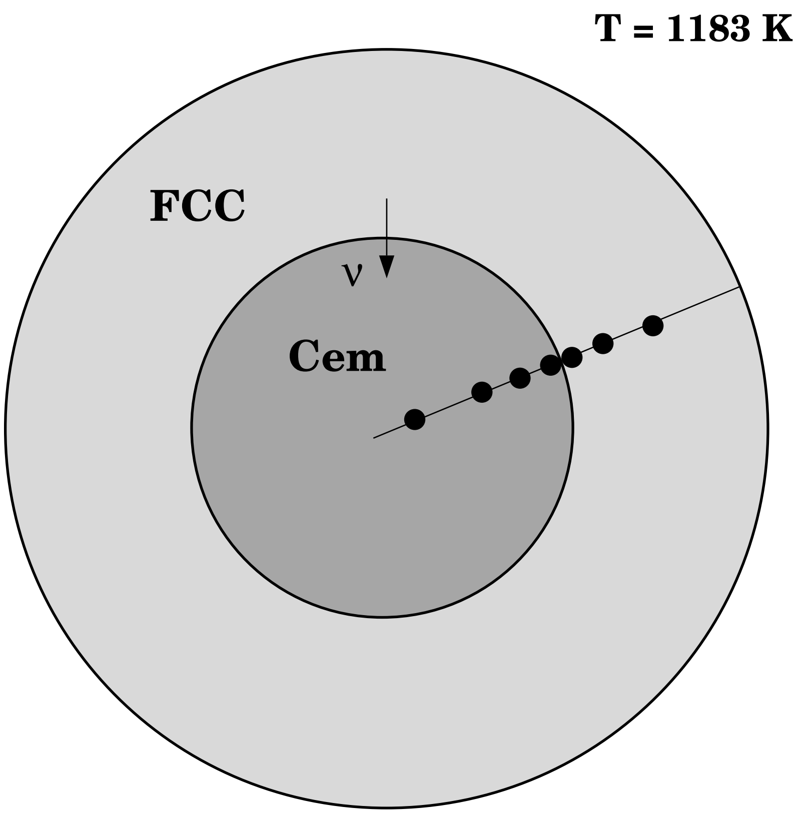
2016.05.17.13.18.15
TIME = 500,700,1200,2000
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1: set-inter
 --OK---
POST-1:
```

Example b2

Cementite dissolution in an Fe-Cr-C alloy



exb2-setup

```
About Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb2\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Moving boundary problems.
SYS: @@ Setup file for calculating the dissolution of a spherical cementite
SYS: @@ particle in an austenite matrix.
SYS: @@ This case is from Z.-K. Liu, L. Höglund, B. Jönsson and J. Ågren:
SYS: @@ Metall. Trans.A, v.22A (1991), pp. 1745-1752.
SYS:
SYS: @@
SYS: @@ In order to achieve the correct average composition in the calculation
SYS: @@ it is necessary to take into account the fact that the calculation in
SYS: @@ DICTRA is setup using the volume fraction of the phases. To calculate
SYS: @@ the initial state at the heat treatment temperature we need first to
SYS: @@ determine the state at the normalizing temperature. To calculate the
SYS: @@ volume fraction of the phases we need to enter a number of functions
SYS: @@ that calculate these quantities. NOTE: The volume fractions are
SYS: @@ determined by assuming that only the substitutional components
SYS: @@ contribute to the volume of system, whereas the interstitial components
SYS: @@ don't.
SYS: @@
SYS: @@ The total radius of the system can be calculated from the relation:
SYS: @@
SYS: @@    3
SYS: @@   R       V
SYS: @@    cem     cem    f
SYS: @@   ----  = ---- = V
SYS: @@    3              cem
SYS: @@   R       V
SYS: @@    tot     tot
SYS: @@
SYS: @@
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-sys fe cr c
 FE                      CR                      C
   DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               CHI_A12                 DIAMOND_FCC_A4
 FCC_A1                  GRAPHITE                HCP_A3
 KSI_CARBIDE             LAVES_PHASE_C14         M23C6
 M3C2                    M5C2                    M7C3
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc bcc cem
 FCC_A1                  BCC_A2                  CEMENTITE
   RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'J-O. Andersson, Calphad, 11 (1987), 271-276; TRITA 0314; C-CR'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson, Metall. Trans. A, 19A (1988), 627-636 TRITA 0207 (1986);
      C-CR-FE'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'P. Villars and L.D. Calvert (1985). Pearsons handbook of crystallographic
      data for intermetallic phases. Metals park, Ohio. American Society for
      Metals; Molar volumes'
  'J. Bratberg, Z. Metallkd., Vol 96 (2005), 335-344; Fe-Cr-Mo-C'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sp fe cr c
 FE                      CR                      C
   DEFINED
APP: rej ph * all
 BCC_A2                  CEMENTITE               FCC_A1
```

```
 FE4N_LP1                HCP_A3                  LIQUID:L
   REJECTED
APP: res ph fcc cementite
 FCC_A1                  CEMENTITE  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Z. Metallkunde 85(1994)502-509; C diffusion in fcc Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
  'This parameter has been estimated'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE POLY-3 MONITOR
APP: @@
APP: go p-3

 POLY version  3.32
POLY_3:
POLY_3: @@
POLY_3: @@  SET THE CONDITIONS AT THE NORMALIZING TEMPERATURE
POLY_3: @@
POLY_3: set-cond T=1008,P=101325,N=1
POLY_3: set-cond X(CR)=0.0206,X(C)=0.0391
POLY_3:
POLY_3:
POLY_3: @@
POLY_3: @@  ENTER FUNCTIONS IN ORDER TO DETERMINE THE VOLUME-FRACTIONS
POLY_3: @@
POLY_3:
POLY_3: @@ Radius of the cementite particle
POLY_3: ent-symb var rcem=0.5255e-6;
POLY_3:
POLY_3: @@ total number of moles of substitutional components
POLY_3: ent-symb func nstot=n(fe)+n(cr);
POLY_3:
POLY_3: @@ number of moles of substitutional components in cementite
POLY_3: ent-symb func nscem=n(cem,fe)+n(cem,cr);
POLY_3:
POLY_3: @@ volume fraction (U-fraction) of cementite
POLY_3: ent-symb func vfcem=nscem/nstot;
POLY_3:
POLY_3: @@ total radius of the system
POLY_3: ent-symb func rtot=rcem/vfcem**(1/3);
POLY_3:
POLY_3: @@ radius of the surrounding austenite matrix
POLY_3: ent-symb func rmat=rtot-rcem;
POLY_3:
POLY_3:
POLY_3: @@
POLY_3: @@  COMPUTE THE EQUILIBRIUM
POLY_3: @@
POLY_3: compute-eq
 Using global minimization procedure
Calculated          3775 grid points in             0 s
Found the set of lowest grid points in              0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3:
POLY_3:
POLY_3: @@
POLY_3: @@  SHOW THE COMPUTED VALUES THAT ARE TO BE USED IN THE DICTRA CALCULATION
POLY_3: @@
POLY_3: show rmat
 RMAT=5.3896724E-7
POLY_3: show w(cem,cr),w(bcc,cr),w(bcc,c)
 W(CEMENTITE,CR)=0.12907307
 W(BCC_A2,CR)=3.950573E-3
 W(BCC_A2,C)=1.4326065E-4
POLY_3:
POLY_3: ent var wmatcr=w(bcc,cr);
POLY_3: ent var wmatc=w(bcc,c);
POLY_3: ent var wcemcr=w(cem,cr);
POLY_3:
POLY_3:
POLY_3: @@
POLY_3: @@ ENTER THE DICTRA MONITOR
POLY_3: @@
POLY_3: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob t 0 1183; * n

DIC>
DIC> @@
DIC> @@ ENTER REGIONS carb AND aus
DIC> @@
DIC> enter-region
REGION NAME : carb
DIC>
DIC> enter-region
REGION NAME : aus
ATTACH TO REGION NAMED /CARB/:
ATTACHED TO THE RIGHT OF CARB /YES/:
```

```
DIC> @@
DIC> @@ ENTER GEOMTRICAL GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ THE INITIAL SIZE OF THE CEMENTITE PARTICLE IS ASSUMED TO BE KNOWN
DIC> @@ (IN THIS CASE WE TAKE OUR VALUE FROM LIU ET AL. WHO ESTIMATED THE
DIC> @@ AVERAGE INITIAL DIAMETER OF THE PARTICLES TO 1.051E-6 METERS).
DIC> @@
DIC> enter-grid
REGION NAME : /CARB/: carb
WIDTH OF REGION /1/: rcem
TYPE /LINEAR/: lin
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ THE SIZE OF THE FCC REGION WE MAY CALCULATE FROM A MASSBALANCE
DIC> @@ AFTER ESTIMATING THE INITIAL COMPOSITIONS IN THE TWO PHASES.
DIC> @@
DIC> enter-grid
REGION NAME : /AUS/: aus
WIDTH OF REGION /1/: rmat
TYPE /LINEAR/: lin
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> enter-phase act carb  matrix   cementite
DIC> enter-phase act aus   matrix   fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /CARB/: carb
PHASE NAME: /CEMENTITE/: cementite
COMPOSITION TYPE /MOLE_FRACTION/: weig-fraction
PROFILE FOR /CR/: cr lin wcemcr wcemcr
DIC>
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /FE/: fe
COMPOSITION TYPE /MOLE_FRACTION/: weig-fraction
PROFILE FOR /C/: CR lin wmatcr wmatcr
PROFILE FOR /CR/: C  lin wmatc wmatc
DIC>
DIC> @@
DIC> @@ SET SPHERICAL GEOMETRY
DIC> @@
DIC> enter-geo
GEOMETRICAL EXPONENT /0/: 2
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 10000
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /1000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exb2 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

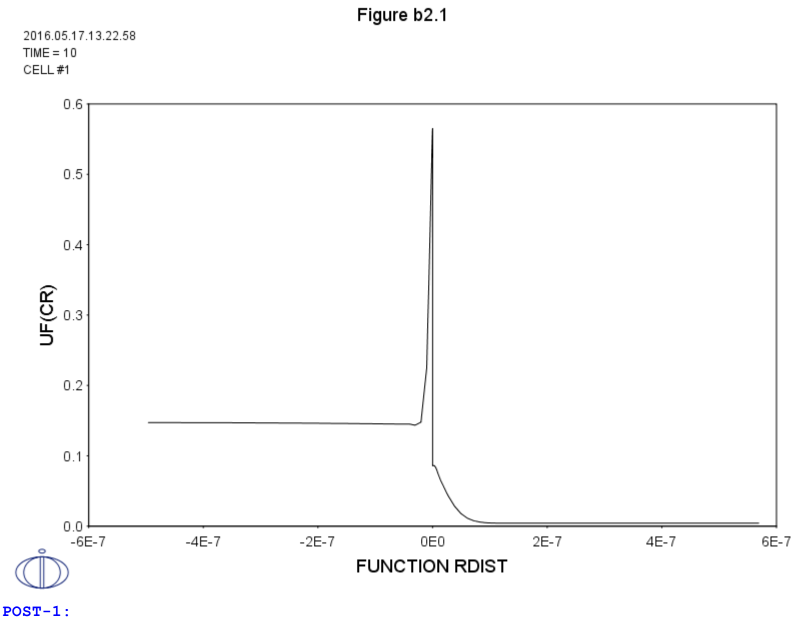
exb2-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb2\run.DCM"
DIC>
DIC> @@ exb2_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb2
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                  4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS              DONE     6 OUT OF     9
04

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0406910188418179  CR = .0214382349908299
                        FE = .978561765139677
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]
 U-FRACTION IN SYSTEM:  C = .0406910188418179  CR = .0214382349908299
                        FE = .978561765139677
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]
 0.177405339441760        0.177446992241502        0.177411181537361        3.728859775978641E-003        9.747104308361495E-
005        6.108946600947069E-006        6.594159458509722E-006        5.654438841505681E-006        5.472839040724207E-
006        5.461182564978494E-006        5.201967365105623E-006        4.951486821672496E-006        4.355333803486320E-
006        4.356816749815634E-006        3.408678579753769E-006        1.634102927019444E-006        1.598898150648279E-
007        2.720033434861970E-011        1.003404048300697E-013        1.277068672324857E-017
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.12770687E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.60095913E-02 AND  -0.60095913E-02
 POSITION OF INTERFACE CARB / AUS IS   0.52489904E-06
 U-FRACTION IN SYSTEM:  C = .0407506164218476  CR = .0214652636765068
                        FE = .978534736454
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                   0  seconds
 3.353176481612989E-006        3.353853628616580E-006        3.352643385712106E-006        1.925236297953492E-006        7.704238636388806E-
007        4.235645481009666E-011        1.085289201866041E-014        2.921113046235949E-017
 TIME =  0.30000000E-06 DT =  0.20000000E-06 SUM OF SQUARES =  0.29211130E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.80367980E-04 AND  -0.80367980E-04
 POSITION OF INTERFACE CARB / AUS IS   0.52488297E-06
 U-FRACTION IN SYSTEM:  C = .0407505694331617  CR = .0214659591412647
                        FE = .978534040989243
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                   0  seconds
 5.131104587759542E-008        5.130082785774904E-008        5.116732326505437E-008        1.611582416595455E-008        1.458738451782520E-
008        1.113400761916804E-008        6.044600920898712E-009        6.036388667388495E-009        4.118032913272870E-
010        1.990447101181755E-014        2.225309051224044E-019
 TIME =  0.70000000E-06 DT =  0.40000000E-06 SUM OF SQUARES =  0.22253091E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14841637E-03 AND  -0.14841637E-03
 POSITION OF INTERFACE CARB / AUS IS   0.52482360E-06
 U-FRACTION IN SYSTEM:  C = .0407521305510963  CR = .0214685193189163
                        FE = .978531480811591
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  2  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUS

  CPU time used in timestep                   0  seconds
 8.032525615956603E-010        8.015593632285115E-010        7.905738617770756E-010        3.740936529184463E-011        1.276331845186441E-
011        7.181336670591613E-017
 TIME =  0.15000000E-05 DT =  0.80000000E-06 SUM OF SQUARES =  0.71813367E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14174631E-03 AND  -0.14174631E-03
 POSITION OF INTERFACE CARB / AUS IS   0.52471020E-06
 U-FRACTION IN SYSTEM:  C = .0407545057430926  CR = .021473397851458
                        FE = .978526602279049
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                   1  seconds
 3.284975251768852E-008        3.285800806642525E-008        3.264444330708045E-008        3.510694670413916E-009        2.740014082637157E-
009

output ignored...



... output resumed

 8.808077002062640E-010        8.804077552846445E-010        8.965796258470293E-010        3.073991758195986E-010        2.687771730609543E-
010        1.994986521803806E-010        9.045580759823016E-011        9.036259558630597E-011        1.346249652009970E-
013        1.749959434571939E-018
 TIME =   6533.2124     DT =  1000.00000     SUM OF SQUARES =  0.17499594E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.10665664E-10 AND  -0.10665664E-10
 POSITION OF INTERFACE CARB / AUS IS   0.29578371E-06
 U-FRACTION IN SYSTEM:  C = .0407750830809891  CR = .0220489851205614
                        FE = .977951015009946
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                   1  seconds
 7.146463796556961E-010        7.142189568839657E-010        7.304896403991926E-010        1.602006850987146E-010        1.361561371440873E-
010        9.394334564078098E-011        3.256453938532368E-011        8.119117657769606E-016        3.950718719315297E-020
 TIME =   7533.2124     DT =  1000.00000     SUM OF SQUARES =  0.39507187E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.95173469E-11 AND  -0.95173469E-11
 POSITION OF INTERFACE CARB / AUS IS   0.28626636E-06
 U-FRACTION IN SYSTEM:  C = .0407751576552599  CR = .022034727955513
                        FE = .977965272174994
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                   1  seconds
 6.184668762287782E-010        6.180358709152639E-010        6.343864030204405E-010        8.122680363197340E-011        6.625097243886866E-
```

```
011        4.063995868859581E-011        8.145581231401476E-012        6.307994737106059E-017
 TIME =   8533.2124     DT =  1000.00000     SUM OF SQUARES =  0.63079947E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.86805916E-11 AND  -0.86805916E-11
 POSITION OF INTERFACE CARB / AUS IS   0.27758577E-06
 U-FRACTION IN SYSTEM:  C = .0407752467113364  CR = .0220207575465465
                        FE = .977979242583961
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                     1  seconds
 5.566556873994342E-010        5.562368646568403E-010        5.725799031151081E-010        4.025326071213033E-011        3.120723661886126E-
011        1.575149285770442E-011        6.491510460663237E-013        1.737224261592603E-018
 TIME =   9533.2124     DT =  1000.00000     SUM OF SQUARES =  0.17372243E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.80637042E-11 AND  -0.80637042E-11
 POSITION OF INTERFACE CARB / AUS IS   0.26952206E-06
 U-FRACTION IN SYSTEM:  C = .0407752784489151  CR = .0220081392204459
                        FE = .977991860910061
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

  CPU time used in timestep                     1  seconds
 3.986669497885249E-010        3.984494805070038E-010        4.234523409893835E-010        2.329083760124043E-011        2.071909103677504E-
011        1.283048267897848E-011        3.847675642650201E-012        3.878939814085631E-018
 TIME =  10000.0000     DT =   466.78757     SUM OF SQUARES =  0.38789398E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.76198392E-11 AND  -0.76198392E-11
 POSITION OF INTERFACE CARB / AUS IS   0.26596522E-06
 U-FRACTION IN SYSTEM:  C = .0407752825518751  CR = .0220009695654082
                        FE = .977999030565099
 TOTAL SIZE OF SYSTEM:  5.05225784732E-18 [m^3]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.30000000E-06
 DELETING TIME-RECORD FOR TIME   0.70000000E-06
 DELETING TIME-RECORD FOR TIME   0.15000000E-05
 DELETING TIME-RECORD FOR TIME   0.31000000E-05
 DELETING TIME-RECORD FOR TIME   0.63000000E-05
 DELETING TIME-RECORD FOR TIME   0.12700000E-04
 DELETING TIME-RECORD FOR TIME   0.25500000E-04
 DELETING TIME-RECORD FOR TIME   0.51100000E-04
 DELETING TIME-RECORD FOR TIME   0.10230000E-03
 DELETING TIME-RECORD FOR TIME   0.20470000E-03
 DELETING TIME-RECORD FOR TIME   0.40950000E-03
 DELETING TIME-RECORD FOR TIME   0.81910000E-03
 DELETING TIME-RECORD FOR TIME   0.16383000E-02
 DELETING TIME-RECORD FOR TIME   0.32767000E-02
 DELETING TIME-RECORD FOR TIME   0.65535000E-02
 DELETING TIME-RECORD FOR TIME   0.13039806E-01
 DELETING TIME-RECORD FOR TIME   0.25442231E-01
 DELETING TIME-RECORD FOR TIME   0.50247079E-01
 DELETING TIME-RECORD FOR TIME   0.99856777E-01
 DELETING TIME-RECORD FOR TIME   0.19907617
 DELETING TIME-RECORD FOR TIME   0.39751496
 DELETING TIME-RECORD FOR TIME   0.79439254
 DELETING TIME-RECORD FOR TIME    1.5881477
 DELETING TIME-RECORD FOR TIME    3.1756580
 DELETING TIME-RECORD FOR TIME    6.3506787
 DELETING TIME-RECORD FOR TIME    12.700720
 DELETING TIME-RECORD FOR TIME    25.400803
 DELETING TIME-RECORD FOR TIME    50.800968
 DELETING TIME-RECORD FOR TIME    101.60130
 DELETING TIME-RECORD FOR TIME    203.20196
 DELETING TIME-RECORD FOR TIME    406.40328
 DELETING TIME-RECORD FOR TIME    812.80592
 DELETING TIME-RECORD FOR TIME    1625.6112
 DELETING TIME-RECORD FOR TIME    2533.2124
 DELETING TIME-RECORD FOR TIME    3533.2124
 DELETING TIME-RECORD FOR TIME    4533.2124
 DELETING TIME-RECORD FOR TIME    5533.2124
 DELETING TIME-RECORD FOR TIME    6533.2124
 DELETING TIME-RECORD FOR TIME    7533.2124
 DELETING TIME-RECORD FOR TIME    8533.2124

 KEEPING TIME-RECORD FOR TIME    9533.2124
 AND FOR TIME                   10000.0000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

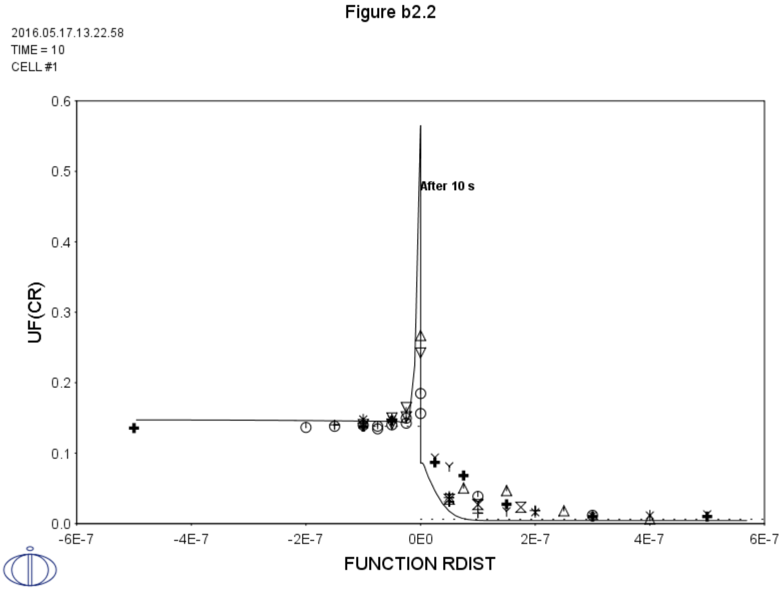
exb2-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb2\plot.DCM"DIC>
DIC>
DIC> @@ exb2_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b2
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+04
DIC> read exb2
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US PLOT CHROMIUM CONCENTRATION PROFILES
POST-1: @@ WE THEN SET DISTANCE AS X-AXIS (NOTE THAT DISTANCE IS SET AS
POST-1: @@ INDEPENDENT VARIABLE AUTOMATICALLY) AND U-FRACTION CARBON AS Y-AXIS
POST-1: @@ REMEMBER THAT ONE ALSO HAS TO SET PLOT CONDITION
POST-1: @@
POST-1: @@ NOTICE THAT ALL DISTANCES IN THE DATA FILE ARE GIVEN RELATIVE TO THE
POST-1: @@ CEM/FCC INTERFACE. FOR THIS REASON ONE HAS TO GIVE AN OFFSET TO THE
POST-1: @@ DATA ACCORDING TO THE ACTUAL PARTICLE RADIUS AT THE SPECIFIED TIME.
POST-1: @@
POST-1: enter-symb
Function or table /FUNCTION/: func
NAME: rdist
FUNCTION: gd-poi(carb,u);
POST-1:
POST-1: s-d-a x rdist
POST-1:
POST-1: s-i-v
VARIABLE /TIME/: dist
DISTANCE : /GLOBAL/: glo
POST-1:
POST-1: s-d-a y uf(cr)
POST-1:
POST-1: s-p-c time 10
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-title Figure b2.1
POST-1:
POST-1: plo
```

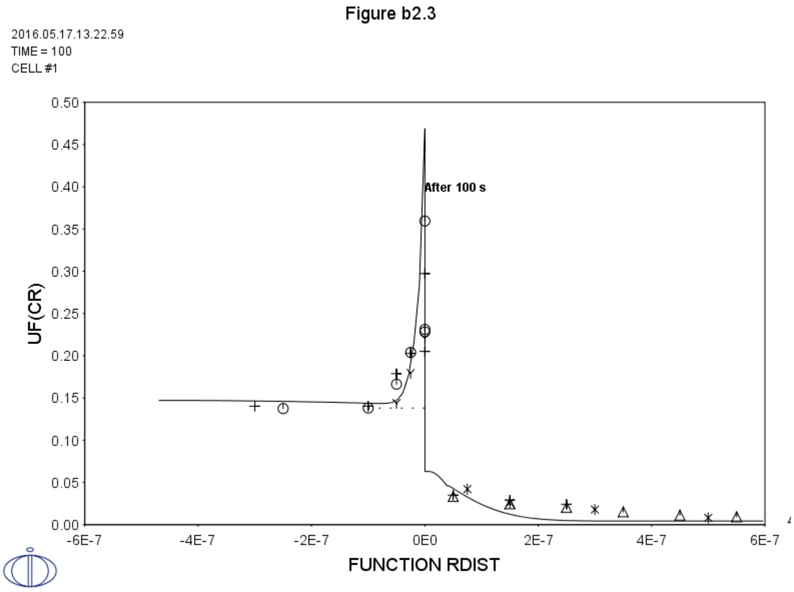



```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ INCLUDE EXPERIMENTAL DATAPOINTS ON THE FIGURE FOR COMPARISION
POST-1: @@
POST-1: @@ FIRST LIST DATASETS
POST-1: @@
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: -1
 DATASET 1 CONCENTRATION PROFILE T=10S
 DATASET 2 CONCENTRATION PROFILE T=100S
 DATASET 3 CONCENTRATION PROFILE T=1000S
 DATASET 4 CONCENTRATION PROFILE T=10000S
 DATASET 5 VOLUME FRACTION CEMENTITE VS. TIME
 DATASET 6 MEAN PARTICLE DIAMETER VS. TIME
```

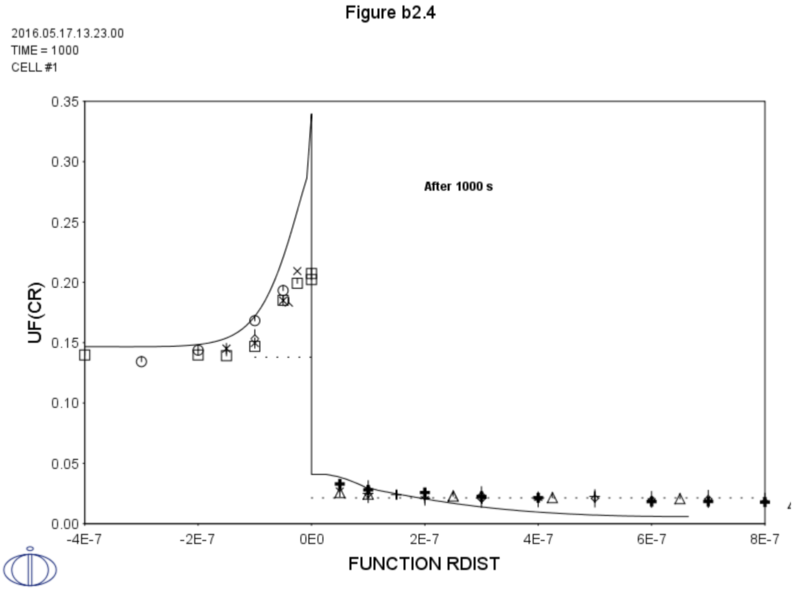
```
POST-1:
POST-1: @@
POST-1: @@ SELECT THE PROPER DATASET
POST-1: @@
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: set-title Figure b2.2
POST-1: plo
```



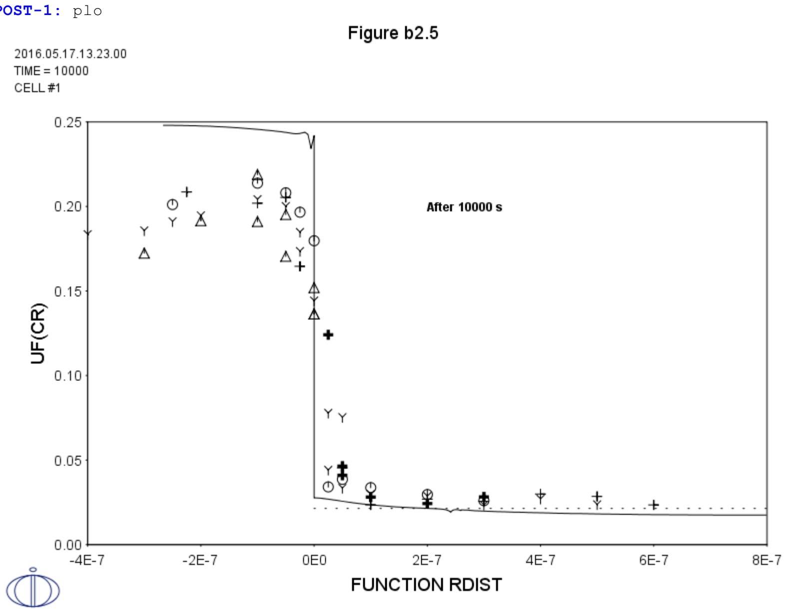

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ PLOT ALSO FOR 100, 1000 AND 10000S
POST-1: @@
POST-1:
POST-1: s-p-c time 100
POST-1:
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 2
POST-1:
POST-1: set-title Figure b2.3
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1:
POST-1: s-p-c time 1000
POST-1:
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 3
POST-1:
POST-1: set-title Figure b2.4
POST-1: plo
```



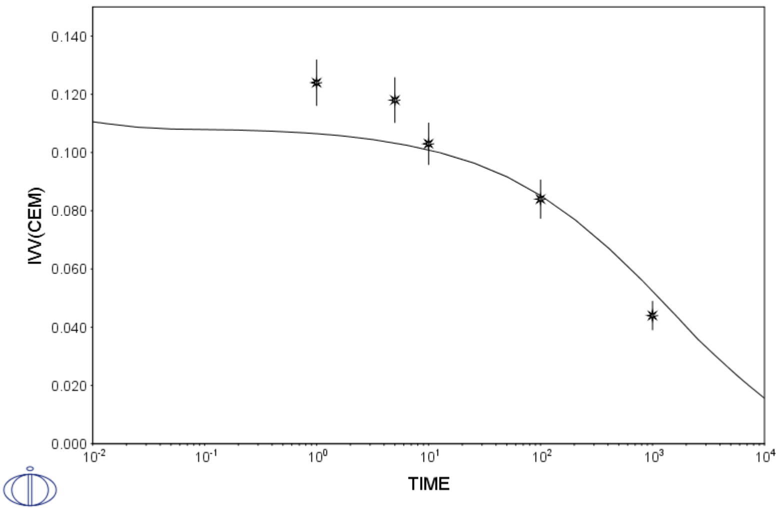

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: s-p-c time 10000
POST-1:
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 4
POST-1:
POST-1: set-title Figure b2.5
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ LET US ALSO PLOT HOW THE VOLUME FRACTION OF CEMENTITE VARIES
POST-1: @@ WITH TIME
POST-1: @@
POST-1: s-d-a y ivv(cem)
POST-1: s-s-s y n 0 .15
POST-1:
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: set-axis-type x log
POST-1: s-s-s x n .01 10000
POST-1:
POST-1: s-p-c integral
POST-1:
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 5
POST-1:
POST-1: set-title Figure b2.6
POST-1: plo
```

Figure b2.6

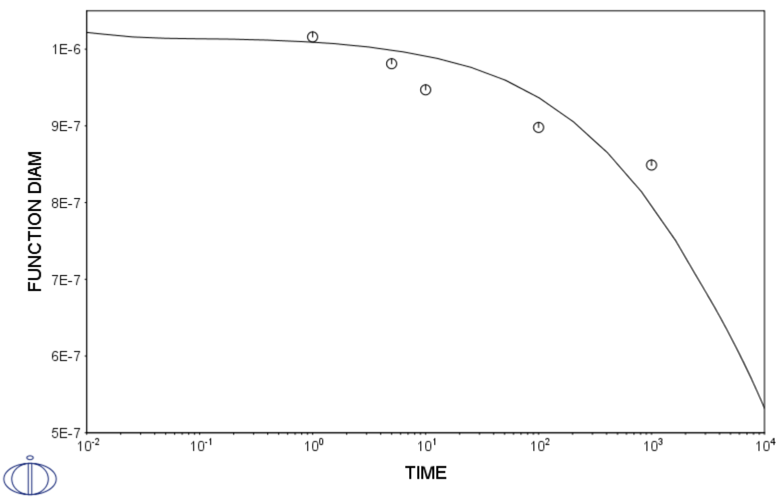
2016.05.17.13.23.00
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ LET US ALSO PLOT HOW THE DIAMETER OF CEMENTITE VARIES WITH TIME
POST-1: @@
POST-1: enter func diam=2*poi(carb,u);
POST-1: s-d-a y diam
POST-1:
POST-1: s-p-c interface carb upper
POST-1:
POST-1: app y exb2.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 6
POST-1:
POST-1: set-title Figure b2.7
POST-1: plo
```

Figure b2.7

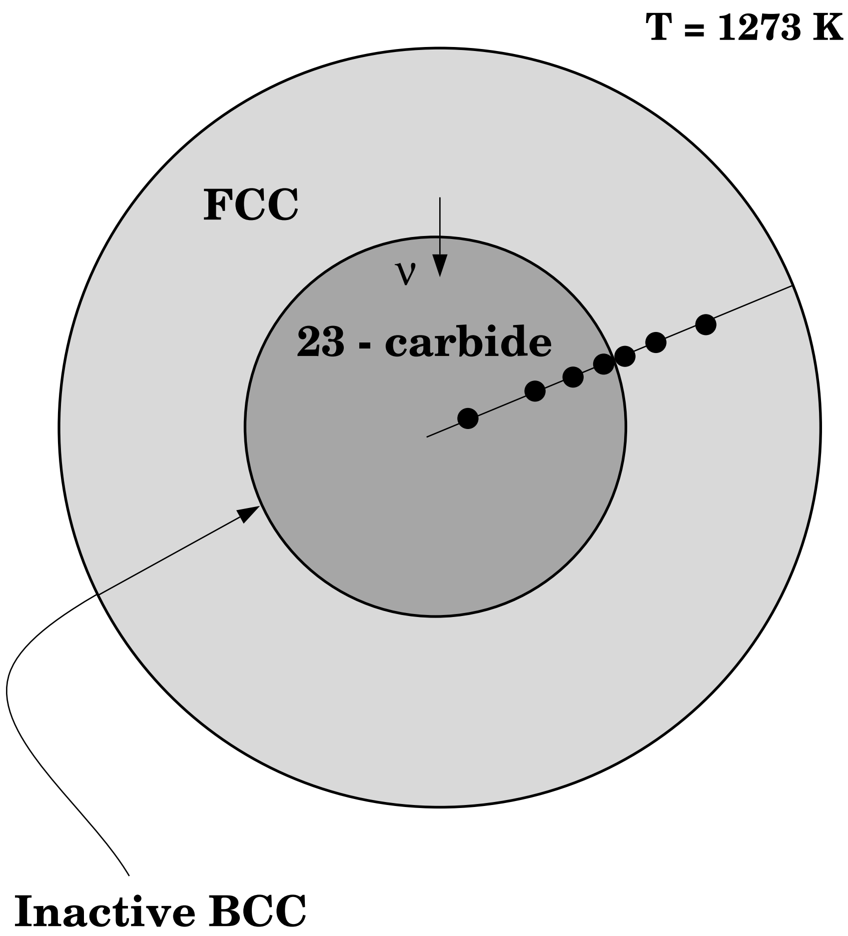
2016.05.17.13.23.01
UPPER INTERFACE OF REGION "CARB#1"
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b3

Dissolution of 23 - carbide in an austenitic matrix



exb3-setup

```
AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb3\setup.DCM"SYS:
SYS:
SYS: @@ exb3_setup.DCM
SYS:
SYS: @@-----------------------------------------------------------------
SYS: @@ SETUP FILE FOR CALCULATING THE DISSOLUTION OF A M23C6 PARTICLE IN
SYS: @@ AN AUSTENITE MATRIX. A FILM OF FERRITE IS ALLOWED TO NUCLEATE AROUND
SYS: @@ THE CARBIDE DURING THE PERCIPITATION.
SYS: @@-----------------------------------------------------------------
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE A PUBLIC DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO: def-sys fe c cr
 FE                       C                        CR
   DEFINED
TDB_FEDEMO: rej ph *
 GAS:G                    LIQUID:L                 BCC_A2
 CEMENTITE                CHI_A12                  DIAMOND_FCC_A4
 FCC_A1                   GRAPHITE                 HCP_A3
 KSI_CARBIDE              LAVES_PHASE_C14          M23C6
 M3C2                     M5C2                     M7C3
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc bcc m23
 FCC_A1                   BCC_A2                   M23C6
   RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'J-O. Andersson, Calphad, 11 (1987), 271-276; TRITA 0314; C-CR'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson, Metall. Trans. A, 19A (1988), 627-636 TRITA 0207 (1986);
      C-CR-FE'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'J. Bratberg, Z. Metallkd., Vol 96 (2005), 335-344; Fe-Cr-Mo-C'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys c cr fe
 C                        CR                       FE
   DEFINED
APP: rej ph *
 BCC_A2                   FCC_A1  REJECTED
APP: res ph fcc bcc
 FCC_A1                   BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Z. Metallkunde 85(1994)502-509; C diffusion in fcc Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
```

```
 *** ENTERING M23C6 AS A DIFFUSION NONE PHASE
DIC>
DIC>
DIC> @@ The mobility database lacks kinetic data for the M23-carbide
DIC> @@ so lets enter an estimate for the mobilities in this phase
DIC> ent-mob-est M23 c
M[M23,C](T)= 0;
DIC>
DIC> ent-mob-est M23 cr
M[M23,CR](T)= 3e-11*exp(-278000/8.3145/T);
DIC>
DIC> ent-mob-est M23 fe
M[M23,FE](T)= 1e-11*exp(-275000/8.3145/T);
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1273;   * N

DIC>
DIC> @@
DIC> @@ ENTER REGIONS carbide AND matrix
DIC> @@
DIC> enter-region  carbide
DIC> enter-region  matrix
ATTACH TO REGION NAMED /CARBIDE/:
ATTACHED TO THE RIGHT OF CARBIDE /YES/:
DIC> @@
DIC> @@ WE ASSUME SOME REASONABLE SIZE OF THE CARBIDE PARTICLE
DIC> @@
DIC> enter-grid  carbide  5.00000000E-7 geo 32 0.9
DIC> @@
DIC> @@ THE SIZE OF THE FCC REGION WE MAY CALCULATE FROM A MASSBALANCE
DIC> @@ AFTER ESTIMATING THE INITIAL COMPOSITIONS IN THE TWO PHASES.
DIC> @@
DIC> enter-grid  matrix  5.55859755E-7 geo 32 1.11
DIC> @@
DIC> @@ ENTER PHASES INTO REGION MATRIX, BOUNDARY CONDITIONS ARE GIVEN
DIC> @@ IF THE INACTIVE PHASE bcc IS NUCLEATED
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /CARBIDE/: matrix
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /MATRIX/: matrix
ATTACHED TO THE RIGHT OF MATRIX /YES/: no
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /FE/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/:
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER PHASE INTO REGION carbide
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /CARBIDE/: carbide
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: m23c6
DIC>
DIC> @@
DIC> @@ ENTER COMPOSITIONS INTO THE PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /CARBIDE/: carbide
PHASE NAME: /M23C6/: m23c6
DEPENDENT COMPONENT ? /FE/: fe
COMPOSITION TYPE /MOLE_FRACTION/: mole-fraction
PROFILE FOR /CR/: cr lin 0.55079807 0.55079807
DIC>
DIC>
DIC> enter-composition
REGION NAME : /MATRIX/: matrix
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /FE/: fe
COMPOSITION TYPE /MOLE_FRACTION/: mole-fraction
PROFILE FOR /C/: cr lin 8.5203899E-2 8.5203899E-2
PROFILE FOR /CR/: c lin 1.8072433E-4  1.8072433E-4
DIC>
DIC> @@
DIC> @@ SET SPHERICAL GEOMETRY
DIC> @@
DIC> enter-geo
GEOMETRICAL EXPONENT /0/: 2
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 8000
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /800/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE
DIC> @@
DIC> save exb3 Y
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

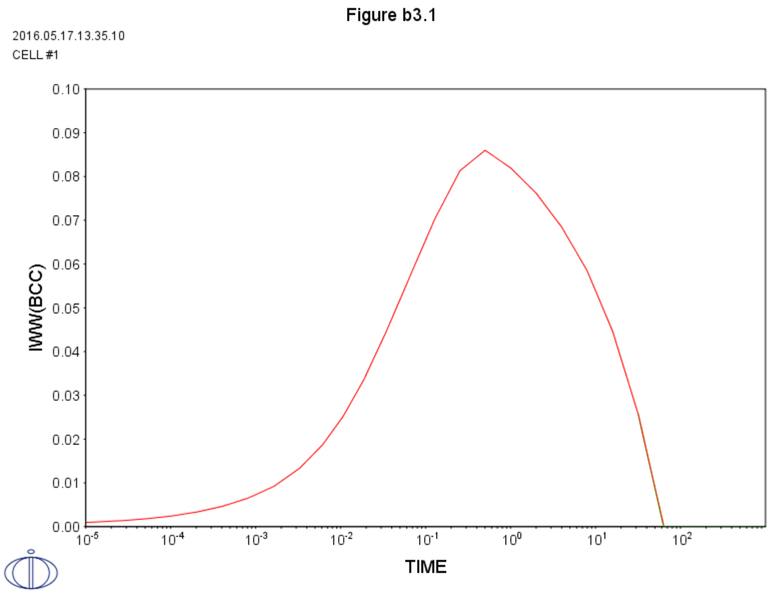
exb3-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb3\run.DCM"DIC>
DIC>
DIC> @@ exb3_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b3
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb3
 OK
DIC>
DIC> @@
DIC> @@ WHEN THE FERRITE NUCLEATES WE USE DEFAULT VALUES
DIC> @@ AS STARTING VALUES FOR THE WIDTH OF THE NEW REGION
DIC> @@ AND THE VELOCITY OF THE INTERFACES
DIC> @@
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim yes
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                    4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9
  try                 2  failed
 DETERMINED ACTIVITIES  ACR(CR) .00193522400114

  UNABLE TO OBTAIN GOOD STARTING VALUE USING THE OLD SCHEME

  Trying new scheme
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:        17   EQUILIBRIUM CALCULATIONS             DONE       1 OUT OF      17

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0278637912207471  CR = .149918318671311
                        FE = .850081681459196
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]
 U-FRACTION IN SYSTEM:  C = .0278637912207471  CR = .149918318671311
                        FE = .850081681459196
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]
  8  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: CARBIDE
  9  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: MATRIX
 0.406078562245216        0.406175066861045        0.406043028810051        1.652667091774450E-002        4.697577908754805E-
004        1.286399300751961E-006        4.498797723970991E-006        1.252366040764282E-006        1.225452384047102E-
006        1.222630342750657E-006        1.214611598374580E-006        1.207610954582430E-006        1.188367721482164E-
006        1.190896700840336E-006        1.153820652523919E-006        1.086439126100473E-006        9.579061299181118E-
007        9.613000668098307E-007        7.261311611131232E-007        3.618981185082453E-007        9.219869946093398E-
009        2.189818043604957E-010        2.006722802368765E-011        3.194152862077887E-013        3.975031200149744E-019
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.39750312E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.52442410E-03 AND  -0.52442410E-03
 POSITION OF INTERFACE CARBIDE / MATRIX IS   0.49994756E-06
 U-FRACTION IN SYSTEM:  C = .027866754688263  CR = .149927563894208
                        FE = .850072436236299
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]

  CPU time used in timestep                    1  seconds
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: CARBIDE
  1  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: MATRIX
 9.068230655054822E-005        9.070674041231456E-005        9.064866629157792E-005        5.908681383685962E-007        3.574299987459274E-
007        1.306729163631280E-007        4.922760464272274E-010        1.003128144658863E-013        4.335779304415484E-
018        SWITCHING ACTIVITIES FOR INTERFACE #2, CELL #1

output ignored...


... output resumed


  CPU time used in timestep                    6  seconds
 2.058525535508130E-012        2.058993137540773E-012        1.654461451504403E-012        1.059836968798536E-012        1.058321706230002E-
012        1.055338390452737E-012        1.049366810350756E-012        1.101582656217185E-012        1.037504957850469E-
012        1.013962756206022E-012        9.676924372782848E-013        1.018662032103261E-012        8.783603939739550E-
013        7.126785455731343E-013        4.331498648805274E-013        4.853326369099904E-013        8.175114836117634E-
014        1.149216704103610E-020
 TIME =    7005.2739      DT =    800.00000      SUM OF SQUARES =  0.11492167E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.12678066E-12 AND  -0.12678066E-12
 POSITION OF INTERFACE CARBIDE / MATRIX IS   0.39469525E-06
 U-FRACTION IN SYSTEM:  C = .0278361281174105  CR = .149892344306816
                        FE = .850107655823691
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: MATRIX

  CPU time used in timestep                    5  seconds
 1.310252565261409E-012        1.310370011185246E-012        9.063691477708980E-013        3.144506315400206E-013        3.141493247232567E-
013        3.135449247356077E-013        3.123253031758205E-013        3.644153076143249E-013        3.099036451116273E-
013        3.050937425067928E-013        2.956137723268057E-013        3.478746180426384E-013        2.770662929479743E-
013        2.417836813830149E-013        1.784192608527495E-013        2.305798699380520E-013        8.051425766307870E-
014        4.072423483513588E-017
 TIME =    7805.2739      DT =    800.00000      SUM OF SQUARES =  0.40724235E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.13266522E-13 AND  -0.13266522E-13
 POSITION OF INTERFACE CARBIDE / MATRIX IS   0.39468464E-06
 U-FRACTION IN SYSTEM:  C = .0278359760172941  CR = .149892293340065
```

```
                        FE = .850107706790442
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: MATRIX

  CPU time used in timestep                       6  seconds
 2.001868562192491E-012        2.001881440406044E-012        1.237120718352979E-012        1.211505292935635E-012        1.211491488609945E-
012        1.211457437341978E-012        1.211425520917648E-012        1.742368821997458E-012        1.211362508380167E-
012        1.211253677058556E-012        1.211009250044404E-012        1.741887702542412E-012        1.210673484095971E-
012        1.209977603890829E-012        1.208694052765932E-012        1.739698546019853E-012        1.206144255249798E-
012        1.201042426915995E-012        1.190880711222524E-012        1.721834504607874E-012        1.170670652231333E-
012        1.130773728113850E-012        1.053181465515072E-012        1.584079416111576E-012        9.076268311960808E-
013        6.489829972836532E-013        2.615111522296557E-013        1.466998147716722E-021
 TIME =   8000.0000     DT =   194.72609     SUM OF SQUARES =  0.14669981E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.19563070E-12 AND   0.19563070E-12
 POSITION OF INTERFACE CARBIDE / MATRIX IS   0.39472273E-06
 U-FRACTION IN SYSTEM:  C = .0278359754520787  CR = .149892308041719
                        FE = .850107692088788
 TOTAL SIZE OF SYSTEM:  4.93068569406E-18 [m^3]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    31.799534
 DELETING TIME-RECORD FOR TIME    63.096637
 DELETING TIME-RECORD FOR TIME    63.096647
 DELETING TIME-RECORD FOR TIME    63.096667
 DELETING TIME-RECORD FOR TIME    63.096707
 DELETING TIME-RECORD FOR TIME    63.096787
 DELETING TIME-RECORD FOR TIME    63.096947
 DELETING TIME-RECORD FOR TIME    63.097267
 DELETING TIME-RECORD FOR TIME    63.097907
 DELETING TIME-RECORD FOR TIME    63.099187
 DELETING TIME-RECORD FOR TIME    63.101747
 DELETING TIME-RECORD FOR TIME    63.106867
 DELETING TIME-RECORD FOR TIME    63.117107
 DELETING TIME-RECORD FOR TIME    63.137587
 DELETING TIME-RECORD FOR TIME    63.178547
 DELETING TIME-RECORD FOR TIME    63.260467
 DELETING TIME-RECORD FOR TIME    63.424307
 DELETING TIME-RECORD FOR TIME    63.751987
 DELETING TIME-RECORD FOR TIME    64.407347
 DELETING TIME-RECORD FOR TIME    65.718067
 DELETING TIME-RECORD FOR TIME    68.339507
 DELETING TIME-RECORD FOR TIME    73.582387
 DELETING TIME-RECORD FOR TIME    84.068147
 DELETING TIME-RECORD FOR TIME    105.03967
 DELETING TIME-RECORD FOR TIME    146.98271
 DELETING TIME-RECORD FOR TIME    230.86879
 DELETING TIME-RECORD FOR TIME    398.64095
 DELETING TIME-RECORD FOR TIME    734.18527
 DELETING TIME-RECORD FOR TIME    1405.2739
 DELETING TIME-RECORD FOR TIME    2205.2739
 DELETING TIME-RECORD FOR TIME    3005.2739
 DELETING TIME-RECORD FOR TIME    3805.2739
 DELETING TIME-RECORD FOR TIME    4605.2739
 DELETING TIME-RECORD FOR TIME    5405.2739
 DELETING TIME-RECORD FOR TIME    6205.2739
 DELETING TIME-RECORD FOR TIME    7005.2739

 KEEPING TIME-RECORD FOR TIME    7805.2739
 AND FOR TIME                    8000.0000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb3-plot

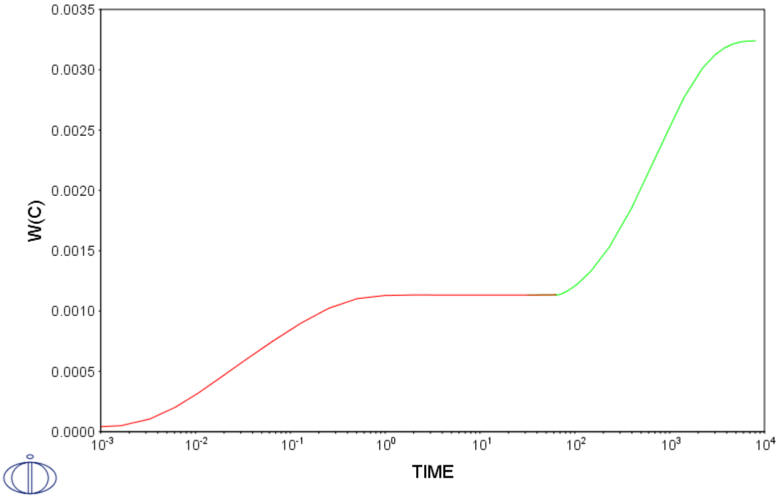
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb3\plot.DCM"
DIC>
DIC> @@ exb3_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b3
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MODULE AND SPECIFY THE STORE-RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  8.00000E+03
DIC> read exb3
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US FIRST SEE HOW THE AMOUNT OF FERRITE VARIED DURING THE
POST-1: @@ SIMULATION
POST-1: @@
POST-1: s-d-a y iww(bcc)
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-ax-typ x log
POST-1: s-s-s x n 1E-5 1E3
POST-1: s-s-s y n 0 0.1
POST-1:
POST-1: set-tit Figure b3.1
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ NOW LOOK AT THE ALLOYING ELEMENTS AT THE UPPER BOUND OF THE SYSTEM
POST-1: @@
POST-1: s-d-a y w(c)
POST-1: s-s-s x n 1E-3 1E4
POST-1: s-p-c interface last
POST-1:
POST-1: set-tit Figure b3.2
POST-1:
POST-1: plot
```

Figure b3.2

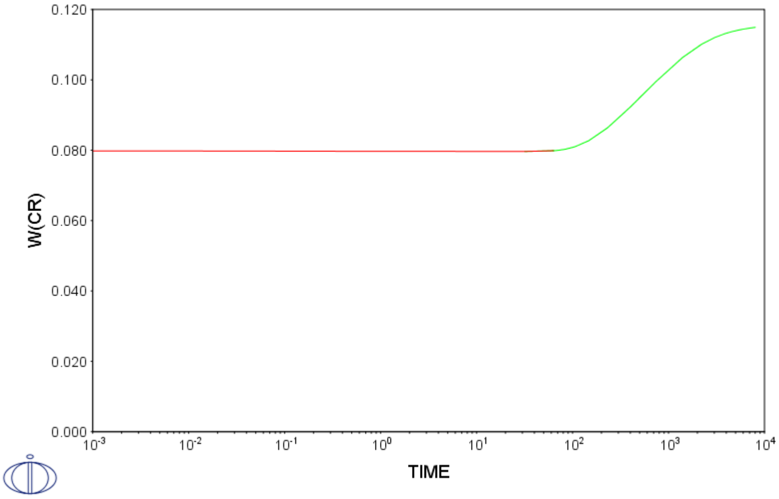
2016.05.17.13.35.10
UPPER INTERFACE OF REGION "LAST"
CELL #1




```
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: s-d-a y w(cr)
POST-1: s-s-s y n 0 0.12
POST-1:
POST-1: set-tit Figure b3.3
POST-1:
POST-1: plot
```

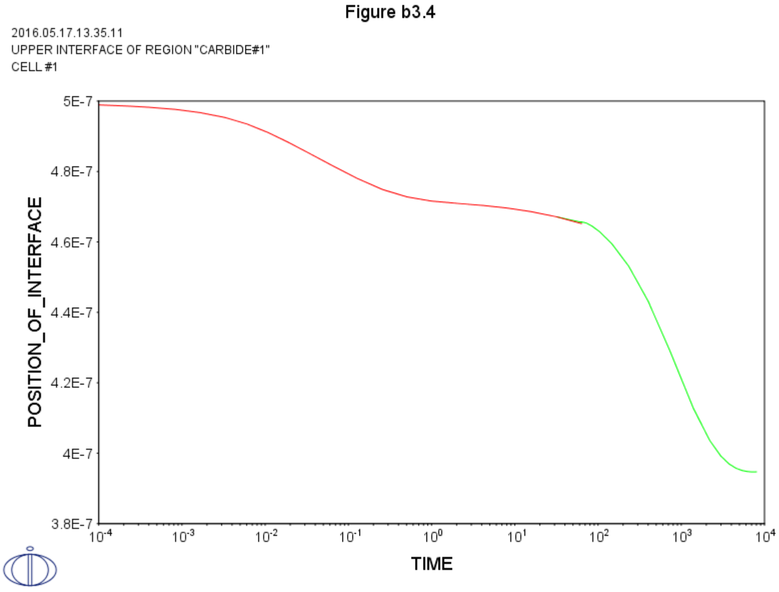
Figure b3.3

2016.05.17.13.35.11
UPPER INTERFACE OF REGION "LAST"
CELL #1




```
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ AND FINALLY LOOK AT THE CHANGE OF RADIUS OF THE M23-CARBIDE
POST-1: @@
POST-1: s-d-a y position carbide upper
POST-1: s-s-s x n 1E-4 1E4
POST-1:
POST-1: set-tit Figure b3.4
POST-1:
POST-1: plot
```

Figure b3.4

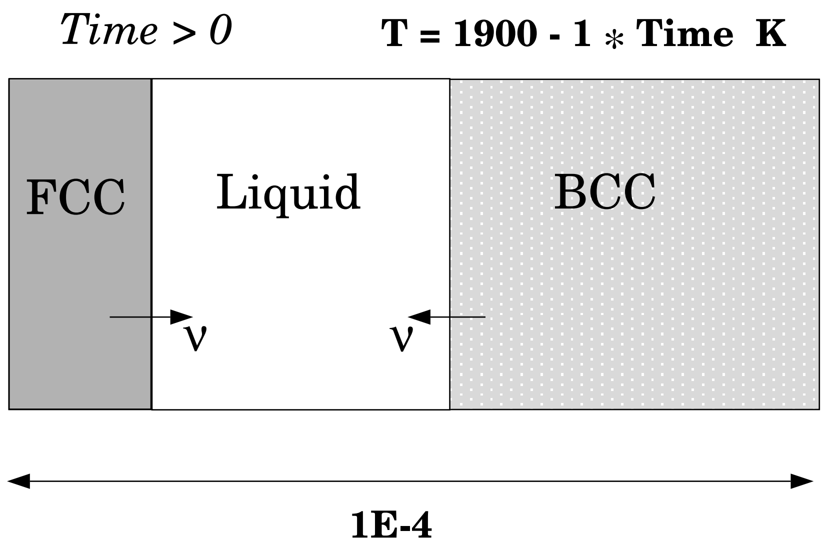



```
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b4a

Solidification path of a Fe - 18% Cr - 8% Ni alloy

(Eutectic reaction)



exb4a-setup

```
About Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4a\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ This examples demonstrates the solidification path of an Fe-18%Cr-8%Ni
SYS: @@ alloy. A eutectic reaction is assumed, LIQUID -> BCC + FCC. Hence the
SYS: @@ BCC and FCC regions should be on separate sides of the liquid region.
SYS: @@ Comparison is made with both a Scheil-Gulliver simulation and equilibrium
SYS: @@ solidification conditions, both made with Thermo-Calc.
SYS: @@
SYS:
SYS: @@ exb4a_setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@ LET US USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: def-sys fe ni cr
 FE                      NI                      CR
   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: rej ph /all
 LIQUID:L                BCC_A2                  CHI_A12
 FCC_A1                  HCP_A3                  LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc liq bcc
 FCC_A1                  LIQUID:L                BCC_A2
   RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'J. Brillo and I. Egry, Int. J. Thermophysics, 24, pp. 1155-1170'
  'B.-J. Lee, Calphad (1993); revison of Fe-Cr and Fe-Ni liquid'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE AND APPEND DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
```

```
 TCCC1    =  Cemented carbide  v1.0
 TCHEA1   =  High Entropy Alloy  v1.0
 SSOL5    =  SGTE Alloy Solutions Database v5.0
 SSOL4    =  SGTE Alloy Solutions Database v4.9f
 SSOL2    =  SGTE Alloy Solutions Database v2.1
 SSUB5    =  SGTE Substances Database v5.1
 SSUB4    =  SGTE Substances Database v4.1
 SSUB3    =  SGTE Substances Database v3.3
 SSUB2    =  SGTE Substances Database v2.2
 SNOB3    =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2    =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1    =  SGTE Nobel Metal Alloys Database v1.2
 STBC2    =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1    =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1    =  SGTE Molten Salts Database v1.2
 SNUX6    =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2    =  TC Semi-Conductors  v2.1
 SLAG4    =  Fe-containing Slag  v4.0 snapshot
 SLAG3    =  Fe-containing Slag  v3.2
 SLAG2    =  Fe-containing Slag  v2.2
 SLAG1    =  Fe-containing Slag  v1.2
 TCOX7    =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6    =  Metal Oxide Solutions  v6.1
 TCOX5    =  Metal Oxide Solutions  v5.1
 TCOX4    =  Metal Oxide Solutions  v4.1
 ION3     =  Ionic Solutions  v3.0
 ION2     =  Ionic Solutions  v2.6
 ION1     =  Ionic Solutions  v1.5
 NOX2     =  NPL Oxide Solutions Database v2.1
 TCSLD3   =  Solder Alloys  v3.1
 TCSLD2   =  Solder Alloys  v2.0
 TCSLD1   =  Solder Alloys  v1.0
 TCSI1    =  Ultrapure Silicon  v1.1
 TCMP2    =  Materials Processing  v2.5
 TCES1    =  Combustion/Sintering  v1.1
 TCSC1    =  Super Conductor  v1.0
 TCFC1    =  SOFC Database v1.0
 TCNF2    =  Nuclear Fuels  v2.1b
 NUMT2    =  Nuclear Materials  v2.1
 NUOX4    =  Nuclear Oxides  v4.2
 NUTO1    =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1    =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10   =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11   =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2    =  Aqueous Solution  v2.5
 AQS2     =  TGG Aqueous Solution Database v2.5
 GCE2     =  TGG Geochemical/Environmental TDB v2.3
 PURE5    =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO   =  Aluminum Demo Database
 FEDEMO   =  Iron Demo Database
 NIDEMO   =  Nickel Demo Database
 SLDEMO   =  Solder Demo Database
 OXDEMO   =  Oxide Demo Database
 SUBDEMO  =  Substance Demo Database
 PTERN    =  Public Ternary Alloys TDB v1.3
 PAQ2     =  Public Aqueous Soln (SIT) TDB v2.4
 PG35     =  G35 Binary Semi-Conductors TDB v1.2
 MOB2     =  Alloys Mobility  v2.4
 MOB1     =  Alloys Mobility  v1.3
 MOBFE1   =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2   =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3   =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4   =  Ni-Alloys Mobility  v4.0
 MOBNI3   =  Ni-Alloys Mobility  v3.1
 MOBNI2   =  Ni-Alloys Mobility  v2.4
 MOBNI1   =  Ni-Alloys Mobility  v1.0
 MOBAL3   =  Al-Alloys Mobility  v3.0
 MOBAL2   =  Al-Alloys Mobility  v2.0
 MOBAL1   =  Al-Alloys Mobility  v1.0
 MOBCU1   =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1   =  Mg-Alloys Mobility  v1.0
 MOBSI1   =  Si-Alloys Mobility  v1.0
 MOBTI1   =  Ti-Alloys Mobility  v1.0
 MALDEMO  =  Al-Alloys Mobility demo database
 MFEDEMO  =  Fe-Alloys Mobility demo database
 MNIDEMO  =  Ni-Alloys Mobility demo database
 USER     =  User defined Database

DATABASE NAME /FEDEMO/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni cr
 FE                       NI                       CR
   DEFINED
APP: rej ph /all
 BCC_A2                   FCC_A1                   HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
     Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
     -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
     diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
     in bcc Fe'
 -OK-
APP:
```

```
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> @@ LET US LOWER THE TEMPERATURE WITH A RATE OF 1 K/S
DIC> @@
DIC> set-cond glob T 0 1900-1*TIME; * N

DIC>
DIC> @@
DIC> @@ ENTER A REGION CALLED smalta
DIC> @@
DIC> enter-region smalta
DIC>
DIC> @@
DIC> @@ ENTER A DOUBLE GEOMETRIC GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /SMALTA/: smalta
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 1.11
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 0.9
DIC>
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /SMALTA/: smalta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: liq
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGION, ONE PHASE ON EACH SIDE OF THE LIQUID
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: no
PHASE NAME: /NONE/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR THE LIQUID
DIC> @@
DIC> enter-composition
REGION NAME : /SMALTA/: smalta
PHASE NAME: /LIQUID/: liq
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr lin 18 18
PROFILE FOR /NI/: ni lin  8  8
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM (DEFAULT) AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 200
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /20/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ CHECK INTERFACE POSSITION
DIC> @@
DIC> @@ THIS IN ORDER TO MAKE SURE THAT THE LIQUID REGION DOESN'T SHRINK
DIC> @@ TO MUCH DURING A TIMESTEP. THE TIMESTEP WILL NOW IN ADDITION BE
DIC> @@ CONTROLLED BY THE PHASE INTERFACE DISPLACEMENT DURING THE SIMULATION.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/: yes
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
```

```
DIC> @@
DIC> save exb4a Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb4a-run

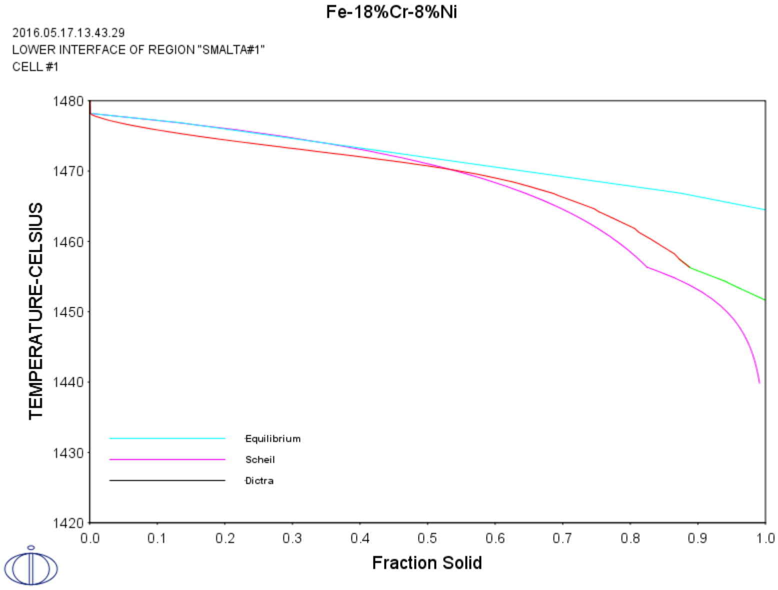
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4a\run.DCM"
DIC>
DIC> @@ exb4a_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b4a
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb4a
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim yes
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.92411922E-04 DT =  0.92311922E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.27703577E-03 DT =  0.18462384E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.64628345E-03 DT =  0.36924769E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.13847788E-02 DT =  0.73849537E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.28617696E-02 DT =  0.14769907E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.58157511E-02 DT =  0.29539815E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.11723714E-01 DT =  0.59079630E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.23539640E-01 DT =  0.11815926E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.47171492E-01 DT =  0.23631852E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.94435196E-01 DT =  0.47263704E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.18896260     DT =  0.94527408E-01 SUM OF SQUARES =   0.0000000
```

```
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.37801742     DT =  0.18905482     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.75612705     DT =  0.37810963     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992482  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]


output ignored...



... output resumed

 2.263863451431409E-011        7.398990027269817E-012        1.617606786018789E-017
 TIME =   180.45759     DT =   2.6214400     SUM OF SQUARES =  0.16176068E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.45473798E-07 AND  -0.45473798E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.63659770E-05
 U-FRACTION IN SYSTEM:  CR = .19146925279442  FE = .733119591795222
                        NI = .0754111554103576
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  2  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                     3  seconds
 1.447616147316402E-009        1.447687101801121E-009        1.495697158570165E-009        1.147147077545110E-010        1.125300446335715E-
010        1.083518947246778E-010        1.000784751630252E-010        1.004848114593226E-010        8.462278506043914E-
011        5.759643494014583E-011        1.912558802264325E-011        9.824543382582526E-017
 TIME =   185.70047     DT =   5.2428800     SUM OF SQUARES =  0.98245434E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.18033393E-07 AND  -0.18033393E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.62714301E-05
 U-FRACTION IN SYSTEM:  CR = .19146903525505З  FE = .733120215042374
                        NI = .0754107497025731
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  2  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                     2  seconds
 8.518966765611837E-010        8.519288384078409E-010        8.727832818965714E-010        2.584389757860910E-010        2.571153596940861E-
010        2.545420028977011E-010        2.493686611906302E-010        2.495285847618298E-010        2.392253609159385E-
010        2.195687151888975E-010        1.828256613563686E-010        1.830506532733437E-010        1.194068844717239E-
010        3.322397642817186E-011        1.002720167175312E-015        8.065334681107193E-020
 TIME =   196.18623     DT =   10.485760     SUM OF SQUARES =  0.80653347E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.22587482E-07 AND   0.22587482E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.65082770E-05
 U-FRACTION IN SYSTEM:  CR = .191469026922302  FE = .733120559587824
                        NI = .0754104134898746
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     3  seconds
 2.120740562054539E-010        2.120691353256732E-010        2.319052733591750E-010        1.296667920004023E-012        1.252936851721359E-
012        1.169008640767449E-012        1.009016487485973E-012        1.452477065831466E-012        7.111745749847487E-
013        2.893376773571913E-013        1.355544488313157E-019
 TIME =   200.00000     DT =   3.8137707     SUM OF SQUARES =  0.13555445E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.19287766E-07 AND   0.19287766E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.65818361E-05
 U-FRACTION IN SYSTEM:  CR = .191469015302911  FE = .733120717014991
                        NI = .0754102676820986
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    175.09029
 DELETING TIME-RECORD FOR TIME    175.21472
 DELETING TIME-RECORD FOR TIME    175.21473
 DELETING TIME-RECORD FOR TIME    175.21475
 DELETING TIME-RECORD FOR TIME    175.21479
 DELETING TIME-RECORD FOR TIME    175.21487
 DELETING TIME-RECORD FOR TIME    175.21503
 DELETING TIME-RECORD FOR TIME    175.21535
 DELETING TIME-RECORD FOR TIME    175.21599
 DELETING TIME-RECORD FOR TIME    175.21727
 DELETING TIME-RECORD FOR TIME    175.21983
 DELETING TIME-RECORD FOR TIME    175.22495
 DELETING TIME-RECORD FOR TIME    175.23519
 DELETING TIME-RECORD FOR TIME    175.25567
 DELETING TIME-RECORD FOR TIME    175.29663
 DELETING TIME-RECORD FOR TIME    175.37855
 DELETING TIME-RECORD FOR TIME    175.54239
 DELETING TIME-RECORD FOR TIME    175.87007
 DELETING TIME-RECORD FOR TIME    176.52543
 DELETING TIME-RECORD FOR TIME    177.83615
 DELETING TIME-RECORD FOR TIME    180.45759
 DELETING TIME-RECORD FOR TIME    185.70047

 KEEPING TIME-RECORD FOR TIME    196.18623
 AND FOR TIME                    200.00000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
```

```
DIC> set-inter
 --OK---
DIC>
```

exb4a-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4a\plot.DCM"
DIC>
DIC> @@ exb4a_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b4a
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  2.00000E+02
DIC> read exb4a
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Fe-18%Cr-8%Ni
POST-1:
POST-1: @@
POST-1: @@ PLOT FRACTION SOLID AND COMPARE WITH SCHEIL-GULLIVER SIMULATION
POST-1: @@ AND EQUILIBRIUM SOLIDIFICATION (DATA ON FILE exb4.exp)
POST-1: @@
POST-1: enter function fs=1-ivv(liquid);
POST-1: s-d-a x fs
POST-1: s-s-s x n 0 1
POST-1: set-axis-text
AXIS (X, Y OR Z) : x
AUTOMATIC AXIS TEXT  (Y OR N) /N/: n
AXIS TEXT : Fraction Solid
POST-1:
POST-1: s-d-a y t-c
POST-1: s-s-s y n 1420 1480
POST-1:
POST-1: s-p-c interface smalta lower
POST-1:
POST-1: app y exb4a.exp 0; 1
POST-1: plo
```

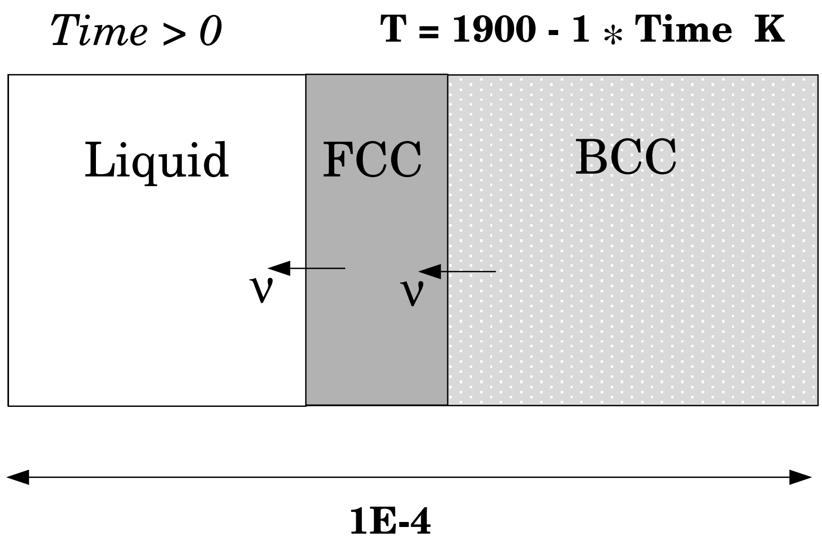



```
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b4b

Solidification path of a Fe - 18% Cr - 8% Ni alloy

(Peritectic reaction)



exb4b-setup

About

```
 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4b\setup.DCM"SYS: @@
SYS: @@ Moving boundary problems.
SYS: @@ The same as exb4a but now a peritectic reaction is assumed, LIQUID + BCC ->
SYS: @@ FCC. Hence the FCC region should appear inbetween the LIQUID and the BCC.
SYS: @@ Comparison is made with both a Scheil-Gulliver simulation and equilibrium
SYS: @@ solidification conditions, both made with Thermo-Calc.
SYS: @@
SYS:
SYS: @@ exb4b_setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@ LET US USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: def-sys fe ni cr
 FE                      NI                      CR
   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: rej ph /all
 LIQUID:L                BCC_A2                  CHI_A12
 FCC_A1                  HCP_A3                  LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc liq bcc
 FCC_A1                  LIQUID:L                BCC_A2
   RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'J. Brillo and I. Egry, Int. J. Thermophysics, 24, pp. 1155-1170'
  'B.-J. Lee, Calphad (1993); revison of Fe-Cr and Fe-Ni liquid'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE AND APPEND DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
```

```
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni cr
 FE                       NI                       CR
   DEFINED
APP: rej ph /all
 BCC_A2                   FCC_A1                   HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
     Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
     -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
     diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
     in bcc Fe'
 -OK-
APP:
```

```
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> @@ LET US LOWER THE TEMPERATURE WITH A RATE OF 1 K/S
DIC> @@
DIC> set-cond glob T 0 1900-1*TIME; * N

DIC>
DIC> @@
DIC> @@ ENTER A REGION CALLED smalta
DIC> @@
DIC> enter-region smalta
DIC>
DIC> @@
DIC> @@ ENTER A GEOMETRIC GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /SMALTA/: smalta
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: geo
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE : 0.9
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /SMALTA/: smalta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: liq
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGION, BOTH PHASES ON THE SAME SIDE OF
DIC> @@ THE LIQUID REGION IN ORDER TO GET A PERITECTIC REACTION.
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR THE LIQUID
DIC> @@
DIC> enter-composition
REGION NAME : /SMALTA/: smalta
PHASE NAME: /LIQUID/: liq
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr lin 18 18
PROFILE FOR /NI/: ni lin  8  8
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM (DEFAULT) AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 200
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /20/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ CHECK INTERFACE POSSITION
DIC> @@
DIC> @@ THIS IN ORDER TO MAKE SURE THAT THE LIQUID REGION DOESN'T SHRINK
DIC> @@ TO MUCH DURING A TIMESTEP. THE TIMESTEP WILL NOW IN ADDITION BE
DIC> @@ CONTROLLED BY THE PHASE INTERFACE DISPLACEMENT DURING THE SIMULATION.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/: yes
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/: @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
```

```
DIC> save exb4b Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

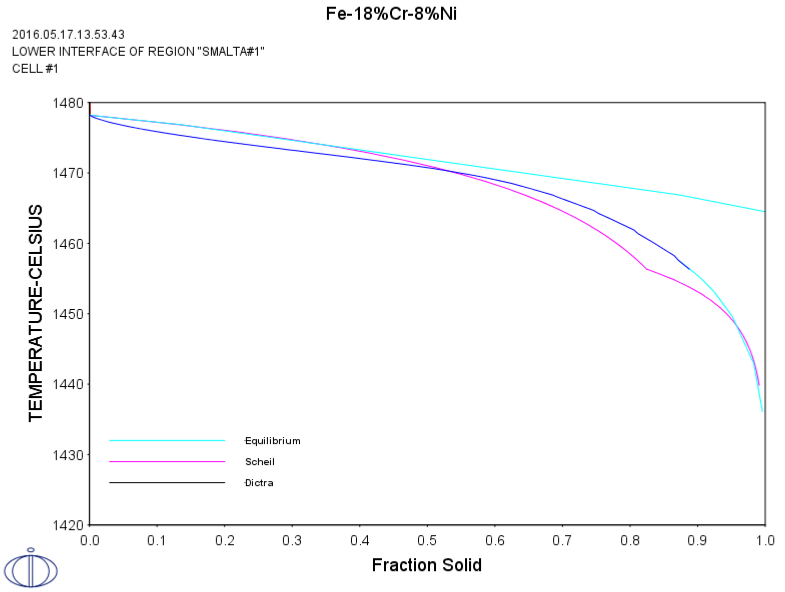
exb4b-run

```
DIC>AboutDIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4b\run.DCM"
DIC> @@ exb4b_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b4b
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb4b
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim yes
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  8  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.21475717E-05 DT =  0.20475717E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.62427150E-05 DT =  0.40951433E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.14433002E-04 DT =  0.81902866E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.30813575E-04 DT =  0.16380573E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.63574721E-04 DT =  0.32761146E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.12909701E-03 DT =  0.65522293E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.26014160E-03 DT =  0.13104459E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.52223077E-03 DT =  0.26208917E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10464091E-02 DT =  0.52417834E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.20947658E-02 DT =  0.10483567E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.41914792E-02 DT =  0.20967134E-02 SUM OF SQUARES =   0.0000000
```

```
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.83849059E-02 DT =  0.41934268E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.16771759E-01 DT =  0.83868535E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219291
                        NI = .0754116207882255


output ignored...



... output resumed


 TIME =   193.36105     DT =   1.3107200     SUM OF SQUARES =  0.24408939E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.20261503E-07 AND  -0.20261503E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.95075315E-05
 U-FRACTION IN SYSTEM:  CR = .191022392191326  FE = .733594444985149
                        NI = .0753831628235253
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     3  seconds
 3.605942371803940E-011        3.606880609775161E-011        3.840640016529831E-011        3.494827612818608E-011        3.493722522132748E-
011        3.491307708555611E-011        3.486684808236957E-011        3.547239650204272E-011        3.477243599318372E-
011        3.458602064878590E-011        3.421269830130384E-011        3.486275688435508E-011        3.347396655361643E-
011        3.201905703793979E-011        2.920759193701749E-011        2.982207622170759E-011        2.397039843176390E-
011        1.504624098662694E-011        3.398768609132232E-012        5.986484640599880E-018
 TIME =   195.98249     DT =   2.6214400     SUM OF SQUARES =  0.59864846E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.43697919E-08 AND  -0.43697919E-08
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.94960763E-05
 U-FRACTION IN SYSTEM:  CR = .191022404588176  FE = .733594446693907
                        NI = .0753831487179175
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_BCC_A2

  CPU time used in timestep                     4  seconds
 4.167272823190759E-011        4.167491474159786E-011        4.340994024711214E-011        4.055286310319276E-011        4.054937973507171E-
011        4.054236849852689E-011        4.052839267144280E-011        4.093269770176441E-011        4.050040281221730E-
011        4.044449845784315E-011        4.033275845455592E-011        4.073144870218160E-011        4.010978956025057E-
011        3.966564613407791E-011        3.878480801359579E-011        3.918977041631408E-011        3.705251684063334E-
011        3.370631995097483E-011        2.748710791286106E-011        2.788904886680020E-011        1.695195438581070E-
011        3.482865031823045E-012        1.785970025558311E-018
 TIME =   200.00000     DT =   4.0175055     SUM OF SQUARES =  0.17859700E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.12320061E-07 AND   0.12320061E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.95455722E-05
 U-FRACTION IN SYSTEM:  CR = .191022446422264  FE = .733594437149161
                        NI = .0753831164285753
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    183.92388
 DELETING TIME-RECORD FOR TIME    190.73962
 DELETING TIME-RECORD FOR TIME    190.73963
 DELETING TIME-RECORD FOR TIME    190.73965
 DELETING TIME-RECORD FOR TIME    190.73969
 DELETING TIME-RECORD FOR TIME    190.73977
 DELETING TIME-RECORD FOR TIME    190.73993
 DELETING TIME-RECORD FOR TIME    190.74025
 DELETING TIME-RECORD FOR TIME    190.74089
 DELETING TIME-RECORD FOR TIME    190.74217
 DELETING TIME-RECORD FOR TIME    190.74473
 DELETING TIME-RECORD FOR TIME    190.74985
 DELETING TIME-RECORD FOR TIME    190.76009
 DELETING TIME-RECORD FOR TIME    190.78057
 DELETING TIME-RECORD FOR TIME    190.82153
 DELETING TIME-RECORD FOR TIME    190.90345
 DELETING TIME-RECORD FOR TIME    191.06729
 DELETING TIME-RECORD FOR TIME    191.39497
 DELETING TIME-RECORD FOR TIME    192.05033
 DELETING TIME-RECORD FOR TIME    193.36105

 KEEPING TIME-RECORD FOR TIME    195.98249
 AND FOR TIME                    200.00000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb4b-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4b\plot.DCM"
DIC>
DIC> @@ exb4b_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b4b
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  2.00000E+02
DIC> read exb4b
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Fe-18%Cr-8%Ni
POST-1:
POST-1: @@
POST-1: @@ PLOT FRACTION SOLID AND COMPARE WITH SCHEIL-GULLIVER SIMULATION AND
POST-1: @@ EQUILIBRIUM SOLIDIFICATION (DATA ON FILE exb4.exp)
POST-1: @@
POST-1: enter func fs=1-ivv(liquid);
POST-1: s-d-a x fs
POST-1: s-s-s x n 0 1
POST-1: s-ax-te x n Fraction Solid
POST-1:
POST-1: s-d-a y t-c
POST-1: s-s-s y n 1420 1480
POST-1:
POST-1: s-p-c interf smalta lower
POST-1:
POST-1: app y exb4b.exp 0; 1
POST-1: plo
```

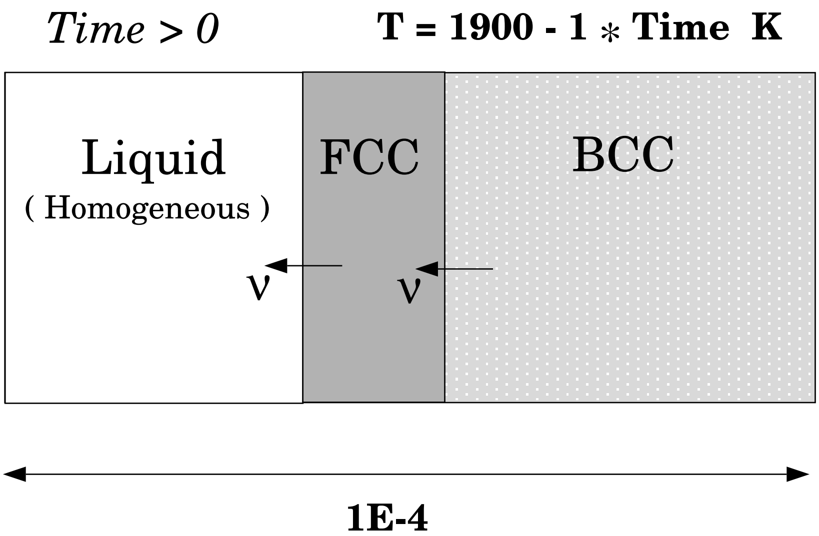



```
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b4c

Solidification path of a Fe - 18% Cr - 8% Ni alloy

(Peritectic reaction , Homogeneous liquid)



exb4c-setup

About Linked: Tue May 17 11:39:18 2016

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4c\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ Same as exb4b but, now the diffusivity data is amended for the LIQUID
SYS: @@ and a very high value for the diffusivity is used in order to simulate a
SYS: @@ case where we assume that the composition in the LIQUID is always
SYS: @@ homogeneous. This case should be considered less realistic than exb4b.
SYS: @@ Comparison is made with both a Scheil-Gulliver simulation and equilibrium
SYS: @@ solidification conditions, both made with Thermo-Calc.
SYS: @@
SYS:
SYS: @@ exb4c_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@ LET US USE THE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: def-sys fe ni cr
 FE                       NI                       CR
   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: rej ph /all
 LIQUID:L                 BCC_A2                   CHI_A12
 FCC_A1                   HCP_A3                   LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'J. Brillo and I. Egry, Int. J. Thermophysics, 24, pp. 1155-1170'
  'B.-J. Lee, Calphad (1993); revison of Fe-Cr and Fe-Ni liquid'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE AND APPEND DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8    =  Steels/Fe-Alloys  v8.0
 TCFE9    =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1   =  FROST database v1.0
 TCFE7    =  Steels/Fe-Alloys  v7.0
 TCFE6    =  Steels/Fe-Alloys  v6.2
 TCFE5    =  Steels/Fe-Alloys  v5.0
 TCFE4    =  Steels/Fe-Alloys  v4.1
 TCFE3    =  Steels/Fe-Alloys  v3.1
 TCFE2    =  Steels/Fe-Alloys  v2.1
 TCFE1    =  Steels/Fe-Alloys  v1.0
 FEDAT    =  TCS/TT Steels Database v1.0
 TCNI9    =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8    =  Ni-Alloys  v8.0
 TCNI7    =  Ni-Alloys  v7.1
 TCNI6    =  Ni-Alloys  v6.0
 TCNI5    =  Ni-Alloys  v5.1
 TCNI4    =  Ni-Alloys  v4.0
 TCNI1    =  Ni-Alloys  v1.3
 TCAL4    =  Al-Alloys  v4.0
 TCAL3    =  Al-Alloys  v3.0
 TCAL2    =  Al-Alloys  v2.0
 TCAL1    =  Al-Alloys  v1.2
 TCMG5    =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4    =  Mg-Alloys  v4.0
 TCMG3    =  Mg-Alloys  v3.0
 TCMG2    =  Mg-Alloys  v2.0
 TCMG1    =  Mg-Alloys  v1.1
 TCTI1    =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1    =  Copper  v1.0 SNAPSHOT
 TCCC1    =  Cemented carbide  v1.0
 TCHEA1   =  High Entropy Alloy  v1.0
 SSOL5    =  SGTE Alloy Solutions Database v5.0
```

```
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni cr
 FE                       NI                       CR
   DEFINED
APP: rej ph /all
 BCC_A2                   FCC_A1                   HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
     Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
     -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
     diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
     in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
```

```
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ LIST MOBILITIES IN THE LIQUID
DIC> @@
DIC> list-mobility-data
 Sorry, LIST-DATA disabled for this database
DIC>
DIC>
DIC> liquid
 NO SUCH COMMAND, USE HELP
DIC>
DIC>
DIC> @@
DIC> @@ AMEND THE DIFFUSIVITY DATA IN THE LIQUID
DIC> @@
DIC> @@ LET'S CHANGE TO A DIFFUSIVITY WHICH IS 1000 TIMES HIGHER THAN THE
DIC> @@ VALUE IN THE MOB-DATABASE. THIS SHOULD BE ENOUGH IN ORDER FOR US TO
DIC> @@ ASSUME THAT THE COMPOSITION IN THE LIQUID IS ATT ALL TIMES HOMOGENEOUS.
DIC> @@
DIC> amend_mobility_data
PARAMETER:
 *** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY
IDENTIFIER: dq
PHASE NAME: liquid&cr
CONSTITUENT: cr
INTERACTING CONSTITUENT:
  DQ(LIQUID&CR#1,CR;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: yes
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&CR#1,CR;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15
FUNCTION: +R*T*LN(1E-06);
HIGH TEMPERATURE LIMIT /6000/: 6000
ANY MORE RANGES /N/: no
DIC>
DIC> amend_mobility_data
PARAMETER: dq(liquid&cr,fe;0)
  DQ(LIQUID&CR#1,FE;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&CR#1,FE;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&cr,ni;0)
  DQ(LIQUID&CR#1,NI;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&CR#1,NI;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&ni,cr;0)
  DQ(LIQUID&NI#1,CR;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&NI#1,CR;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&ni,fe;0)
  DQ(LIQUID&NI#1,FE;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&NI#1,FE;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&ni,ni;0)
  DQ(LIQUID&NI#1,NI;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&NI#1,NI;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&fe,cr;0)
  DQ(LIQUID&FE#1,CR;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&FE#1,CR;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&fe,fe;0)
  DQ(LIQUID&FE#1,FE;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&FE#1,FE;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> am-mob dq(liquid&fe,ni;0)
  DQ(LIQUID&FE#1,NI;0) =  Sorry, database encrypted
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: y
 I AM SORRY BUT YOU MUST THEN REENTER ALL RANGES
 DQ(LIQUID&FE#1,NI;0) =
LOW TEMPERATURE LIMIT /298.15/: 298.15 +R*T*LN(1E-06); 6000 n
DIC>
DIC> li-mob
 AMBIGUOUS COMMAND, USE HELP
DIC>
DIC>
DIC> liquid
 NO SUCH COMMAND, USE HELP
DIC>
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> @@ LET US LOWER THE TEMPERATURE WITH A RATE OF 1 K/S
DIC> set-cond glob T 0 1900-1*TIME; * N

DIC>
DIC> @@
DIC> @@ ENTER A REGION CALLED smalta
```

```
DIC> @@
DIC> enter-region smalta
DIC>
DIC> @@
DIC> @@ ENTER A GEOMETRIC GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /SMALTA/: smalta
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: geo
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE : 0.9
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /SMALTA/: smalta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: liq
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGION, BOTH PHASES ON THE SAME SIDE OF
DIC> @@ THE LIQUID REGION IN ORDER TO GET A PERITECTIC REACTION.
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR THE LIQUID
DIC> @@
DIC> enter-composition
REGION NAME : /SMALTA/: smalta
PHASE NAME: /LIQUID/: liq
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr lin 18 18
PROFILE FOR /NI/: ni lin  8  8
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM (DEFAULT) AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 200
AUTOMATIC TIMESTEP CONTROL /YES/: yes
MAX TIMESTEP DURING INTEGRATION /20/: 1
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ CHECK INTERFACE POSSITION
DIC> @@
DIC> @@ THIS IN ORDER TO MAKE SURE THAT THE LIQUID REGION DOESN'T SHRINK
DIC> @@ TO MUCH DURING A TIMESTEP. THE TIMESTEP WILL NOW IN ADDITION BE
DIC> @@ CONTROLLED BY THE PHASE INTERFACE DISPLACEMENT DURING THE SIMULATION.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/: yes
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/: @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb4c Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb4c-run

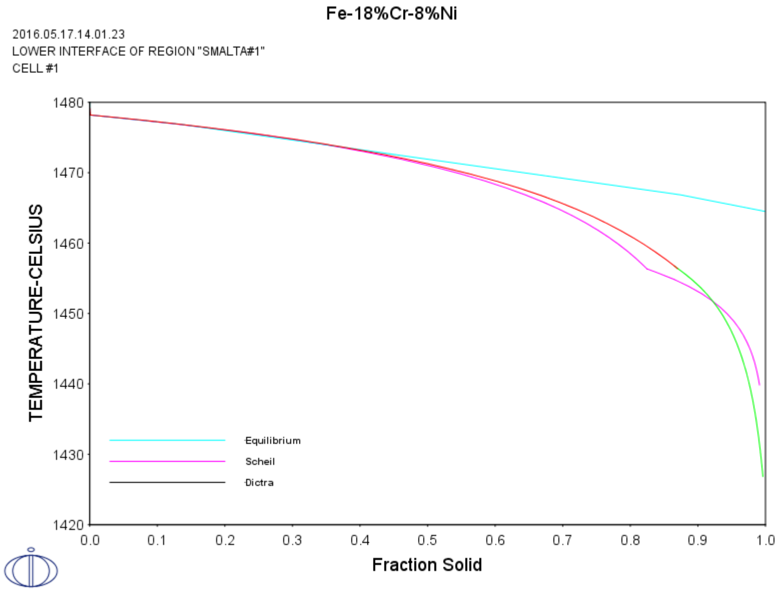
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4c\run.DCM"DIC>
DIC> @@ exb4c_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b4b
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb4c
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim yes
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  8  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.30000000E-06 DT =  0.20000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.70000000E-06 DT =  0.40000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.15000000E-05 DT =  0.80000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.31000000E-05 DT =  0.16000000E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.63000000E-05 DT =  0.32000000E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.12700000E-04 DT =  0.64000000E-05 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.25500000E-04 DT =  0.12800000E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.51100000E-04 DT =  0.25600000E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10230000E-03 DT =  0.51200000E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992482  FE = .733068011219292
                        NI = .0754116207882253
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.20470000E-03 DT =  0.10240000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40950000E-03 DT =  0.20480000E-03 SUM OF SQUARES =   0.0000000
```

```
 U-FRACTION IN SYSTEM:  CR = .191520367992481  FE = .733068011219293
                        NI = .0754116207882258
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.81910000E-03 DT =  0.40960000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .191520367992479  FE = .733068011219296
                        NI = .0754116207882255
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.16383000E-02 DT =  0.81920000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .19152036799248  FE = .733068011219296
                        NI = .0754116207882239
 TOTAL SIZE OF SYSTEM:  1E-04 [m]


output ignored...



... output resumed

  CPU time used in timestep                     4  seconds
 6.720654764218337E-010        6.735114491877951E-010        6.720755345628061E-010        6.580652352915213E-010        7.006432161125256E-
010        2.202952111999320E-010        1.460039567425147E-010        1.235174081568564E-010        1.099722657696368E-
010        1.100657420880965E-010        1.010710424872652E-010        7.380504412027919E-011        6.835454878493349E-
011        6.850154560143956E-011        6.791695480450173E-011        3.973059663999489E-011        5.490735533222957E-
011        2.233251411361525E-011        1.841123252680386E-011        1.065008042475111E-011        5.786467716564188E-
012        6.062173684559055E-012        6.068595112256801E-012        1.916201534426192E-013        8.136554140904482E-
016        1.620621635863000E-015        2.539716117159599E-016
 TIME =   200.00000     DT =  0.84586945     SUM OF SQUARES =  0.25397161E-15
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.94234191E-07 AND  -0.94234191E-07
 POSITION OF INTERFACE SMALTA / R_FCC_A1 IS   0.38455076E-06
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.13704506E-07 AND   0.13704506E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.11079231E-04
 U-FRACTION IN SYSTEM:  CR = .191193887753306  FE = .733399961136488
                        NI = .075406151110207
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    170.09360
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47641
 DELETING TIME-RECORD FOR TIME    170.47642
 DELETING TIME-RECORD FOR TIME    170.47642
 DELETING TIME-RECORD FOR TIME    170.47643
 DELETING TIME-RECORD FOR TIME    170.47646
 DELETING TIME-RECORD FOR TIME    170.47651
 DELETING TIME-RECORD FOR TIME    170.47661
 DELETING TIME-RECORD FOR TIME    170.47682
 DELETING TIME-RECORD FOR TIME    170.47723
 DELETING TIME-RECORD FOR TIME    170.47805
 DELETING TIME-RECORD FOR TIME    170.47969
 DELETING TIME-RECORD FOR TIME    170.48296
 DELETING TIME-RECORD FOR TIME    170.48952
 DELETING TIME-RECORD FOR TIME    170.50262
 DELETING TIME-RECORD FOR TIME    170.52884
 DELETING TIME-RECORD FOR TIME    170.58127
 DELETING TIME-RECORD FOR TIME    170.68612
 DELETING TIME-RECORD FOR TIME    170.89584
 DELETING TIME-RECORD FOR TIME    171.31527
 DELETING TIME-RECORD FOR TIME    172.15413
 DELETING TIME-RECORD FOR TIME    173.15413
 DELETING TIME-RECORD FOR TIME    174.15413
 DELETING TIME-RECORD FOR TIME    175.15413
 DELETING TIME-RECORD FOR TIME    176.15413
 DELETING TIME-RECORD FOR TIME    177.15413
 DELETING TIME-RECORD FOR TIME    178.15413
 DELETING TIME-RECORD FOR TIME    179.15413
 DELETING TIME-RECORD FOR TIME    180.15413
 DELETING TIME-RECORD FOR TIME    181.15413
 DELETING TIME-RECORD FOR TIME    182.15413
 DELETING TIME-RECORD FOR TIME    183.15413
 DELETING TIME-RECORD FOR TIME    184.15413
 DELETING TIME-RECORD FOR TIME    185.15413
 DELETING TIME-RECORD FOR TIME    186.15413
 DELETING TIME-RECORD FOR TIME    187.15413
 DELETING TIME-RECORD FOR TIME    188.15413
 DELETING TIME-RECORD FOR TIME    189.15413
 DELETING TIME-RECORD FOR TIME    190.15413
 DELETING TIME-RECORD FOR TIME    191.15413
 DELETING TIME-RECORD FOR TIME    192.15413
 DELETING TIME-RECORD FOR TIME    193.15413
 DELETING TIME-RECORD FOR TIME    194.15413
 DELETING TIME-RECORD FOR TIME    195.15413
 DELETING TIME-RECORD FOR TIME    196.15413
 DELETING TIME-RECORD FOR TIME    197.15413
 DELETING TIME-RECORD FOR TIME    198.15413

 KEEPING TIME-RECORD FOR TIME    199.15413
 AND FOR TIME                    200.00000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
```

```
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb4c-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4c\plot.DCM"
DIC>
DIC> @@ exb4c_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b4c
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  2.00000E+02
DIC> read exb4c
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Fe-18%Cr-8%Ni
POST-1:
POST-1: @@
POST-1: @@ PLOT FRACTION SOLID AND COMPARE WITH SCHEIL-GULLIVER SIMULATION AND
POST-1: @@ EQUILIBRIUM SOLIDIFICATION (DATA ON FILE exb4.exp). IN THIS CASE WE
POST-1: @@ MAY SEE THAT ALL THREE LINES INITIALLY FALL ON THE SAME LINE.
POST-1: @@
POST-1: enter func fs=1-ivv(liquid);
POST-1: s-d-a x fs
POST-1: s-s-s x n 0 1
POST-1: s-ax-te x n Fraction Solid
POST-1:
POST-1: s-d-a y t-c
POST-1: s-s-s y n 1420 1480
POST-1:
POST-1: s-p-c interf smalta lower
POST-1:
POST-1: app y exb4c.exp 0; 1
POST-1: plo
```

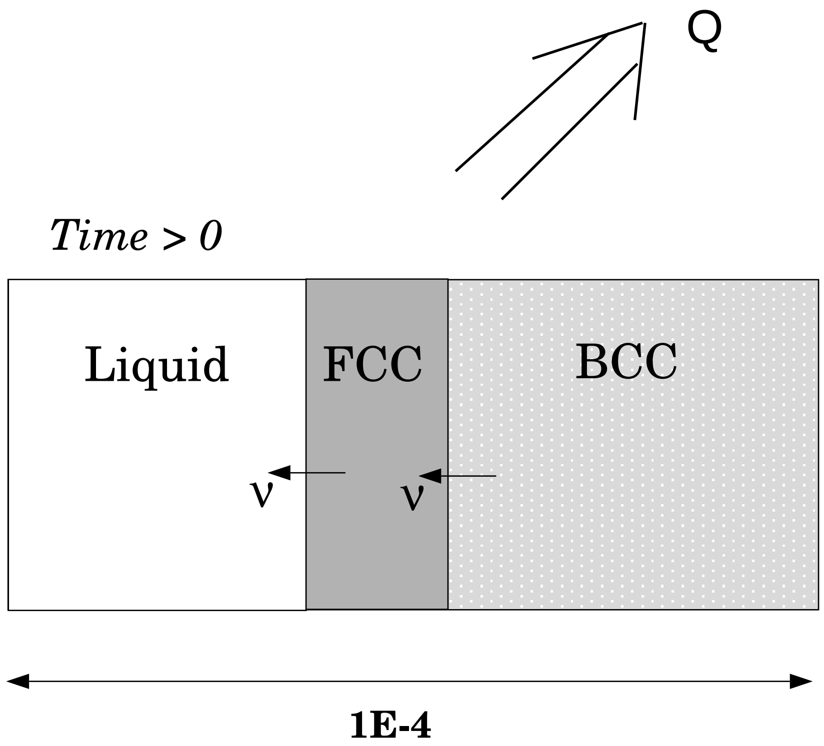



```
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b4d

Solidification path of a Fe - 18% Cr - 8% Ni alloy

(Peritectic reaction, Heat-flux controls the temperature)



exb4d-setup

About

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4d\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ Same as exb4b but instead of controlling the temperature the amount
SYS: @@ of heat extracted is given. Comparison is made with both a Scheil-Gulliver
SYS: @@ simulation and equilibrium solidification conditions, both made
SYS: @@ with Thermo-Calc.
SYS: @@
SYS:
SYS: @@ exb4d_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@ LET US USE THE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_FEDEMO: def-sys fe ni cr
 FE                       NI                       CR
   DEFINED
TDB_FEDEMO:
TDB_FEDEMO: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_FEDEMO: rej ph /all
 LIQUID:L                 BCC_A2                   CHI_A12
 FCC_A1                   HCP_A3                   LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
TDB_FEDEMO:
TDB_FEDEMO: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'J. Brillo and I. Egry, Int. J. Thermophysics, 24, pp. 1155-1170'
  'B.-J. Lee, Calphad (1993); revison of Fe-Cr and Fe-Ni liquid'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE AND APPEND DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app
 Use one of these databases

 TCFE8    =  Steels/Fe-Alloys  v8.0
 TCFE9    =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1   =  FROST database v1.0
 TCFE7    =  Steels/Fe-Alloys  v7.0
 TCFE6    =  Steels/Fe-Alloys  v6.2
 TCFE5    =  Steels/Fe-Alloys  v5.0
 TCFE4    =  Steels/Fe-Alloys  v4.1
 TCFE3    =  Steels/Fe-Alloys  v3.1
 TCFE2    =  Steels/Fe-Alloys  v2.1
 TCFE1    =  Steels/Fe-Alloys  v1.0
 FEDAT    =  TCS/TT Steels Database v1.0
 TCNI9    =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8    =  Ni-Alloys  v8.0
 TCNI7    =  Ni-Alloys  v7.1
 TCNI6    =  Ni-Alloys  v6.0
 TCNI5    =  Ni-Alloys  v5.1
 TCNI4    =  Ni-Alloys  v4.0
 TCNI1    =  Ni-Alloys  v1.3
 TCAL4    =  Al-Alloys  v4.0
 TCAL3    =  Al-Alloys  v3.0
 TCAL2    =  Al-Alloys  v2.0
 TCAL1    =  Al-Alloys  v1.2
 TCMG5    =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4    =  Mg-Alloys  v4.0
 TCMG3    =  Mg-Alloys  v3.0
 TCMG2    =  Mg-Alloys  v2.0
 TCMG1    =  Mg-Alloys  v1.1
 TCTI1    =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1    =  Copper  v1.0 SNAPSHOT
 TCCC1    =  Cemented carbide  v1.0
 TCHEA1   =  High Entropy Alloy  v1.0
 SSOL5    =  SGTE Alloy Solutions Database v5.0
 SSOL4    =  SGTE Alloy Solutions Database v4.9f
 SSOL2    =  SGTE Alloy Solutions Database v2.1
 SSUB5    =  SGTE Substances Database v5.1
```

```
 SSUB4    =  SGTE Substances Database v4.1
 SSUB3    =  SGTE Substances Database v3.3
 SSUB2    =  SGTE Substances Database v2.2
 SNOB3    =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2    =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1    =  SGTE Nobel Metal Alloys Database v1.2
 STBC2    =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1    =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1    =  SGTE Molten Salts Database v1.2
 SNUX6    =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2    =  TC Semi-Conductors  v2.1
 SLAG4    =  Fe-containing Slag  v4.0 snapshot
 SLAG3    =  Fe-containing Slag  v3.2
 SLAG2    =  Fe-containing Slag  v2.2
 SLAG1    =  Fe-containing Slag  v1.2
 TCOX7    =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6    =  Metal Oxide Solutions  v6.1
 TCOX5    =  Metal Oxide Solutions  v5.1
 TCOX4    =  Metal Oxide Solutions  v4.1
 ION3     =  Ionic Solutions  v3.0
 ION2     =  Ionic Solutions  v2.6
 ION1     =  Ionic Solutions  v1.5
 NOX2     =  NPL Oxide Solutions Database v2.1
 TCSLD3   =  Solder Alloys  v3.1
 TCSLD2   =  Solder Alloys  v2.0
 TCSLD1   =  Solder Alloys  v1.0
 TCSI1    =  Ultrapure Silicon  v1.1
 TCMP2    =  Materials Processing  v2.5
 TCES1    =  Combustion/Sintering  v1.1
 TCSC1    =  Super Conductor  v1.0
 TCFC1    =  SOFC Database v1.0
 TCNF2    =  Nuclear Fuels  v2.1b
 NUMT2    =  Nuclear Materials  v2.1
 NUOX4    =  Nuclear Oxides  v4.2
 NUTO1    =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1    =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10   =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11   =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2    =  Aqueous Solution  v2.5
 AQS2     =  TGG Aqueous Solution Database v2.5
 GCE2     =  TGG Geochemical/Environmental TDB v2.3
 PURE5    =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO   =  Aluminum Demo Database
 FEDEMO   =  Iron Demo Database
 NIDEMO   =  Nickel Demo Database
 SLDEMO   =  Solder Demo Database
 OXDEMO   =  Oxide Demo Database
 SUBDEMO  =  Substance Demo Database
 PTERN    =  Public Ternary Alloys TDB v1.3
 PAQ2     =  Public Aqueous Soln (SIT) TDB v2.4
 PG35     =  G35 Binary Semi-Conductors TDB v1.2
 MOB2     =  Alloys Mobility  v2.4
 MOB1     =  Alloys Mobility  v1.3
 MOBFE1   =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2   =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3   =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4   =  Ni-Alloys Mobility  v4.0
 MOBNI3   =  Ni-Alloys Mobility  v3.1
 MOBNI2   =  Ni-Alloys Mobility  v2.4
 MOBNI1   =  Ni-Alloys Mobility  v1.0
 MOBAL3   =  Al-Alloys Mobility  v3.0
 MOBAL2   =  Al-Alloys Mobility  v2.0
 MOBAL1   =  Al-Alloys Mobility  v1.0
 MOBCU1   =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1   =  Mg-Alloys Mobility  v1.0
 MOBSI1   =  Si-Alloys Mobility  v1.0
 MOBTI1   =  Ti-Alloys Mobility  v1.0
 MALDEMO  =  Al-Alloys Mobility demo database
 MFEDEMO  =  Fe-Alloys Mobility demo database
 MNIDEMO  =  Ni-Alloys Mobility demo database
 USER     =  User defined Database

DATABASE NAME /FEDEMO/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni cr
 FE                      NI                      CR
   DEFINED
APP: rej ph /all
 BCC_A2                  FCC_A1                  HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc liq bcc
 FCC_A1                  LIQUID:L                BCC_A2
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
      Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
```

```
DIC>
DIC> @@
DIC> @@ EXTRACT HEAT 91.19 J/mole/s
DIC> @@
DIC> set-cond glob Q 0 91.19; * N

DIC>
DIC> @@
DIC> @@ ENTER INITIAL TEMPERATURE
DIC> @@
DIC> set-initial-temp 1900
DIC>
DIC> @@
DIC> @@ ENTER A REGION CALLED smalta
DIC> @@
DIC> enter-region smalta
DIC>
DIC> @@
DIC> @@ ENTER A GEOMETRIC GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /SMALTA/: smalta
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: geo
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE : 0.9
DIC>
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /SMALTA/: smalta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: liq
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGION, BOTH PHASES ON THE SAME SIDE OF
DIC> @@ THE LIQUID REGION IN ORDER TO GET A PERITECTIC REACTION.
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-3
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: yes
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /NI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-3
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR THE LIQUID
DIC> @@
DIC> enter-composition
REGION NAME : /SMALTA/: smalta
PHASE NAME: /LIQUID/: liq
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr lin 18 18
PROFILE FOR /NI/: ni lin  8  8
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM (DEFAULT) AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 200
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /20/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ CHECK INTERFACE POSSITION
DIC> @@
DIC> @@ THIS IN ORDER TO MAKE SURE THAT THE LIQUID REGION DOESN'T SHRINK
DIC> @@ TO MUCH DURING A TIMESTEP. THE TIMESTEP WILL NOW IN ADDITION BE
DIC> @@ CONTROLLED BY THE PHASE INTERFACE DISPLACEMENT DURING THE SIMULATION.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/: yes
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/: no
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/: @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb4d Y
```

```
DIC>
DIC> set-inter
 --OK---
DIC>
```

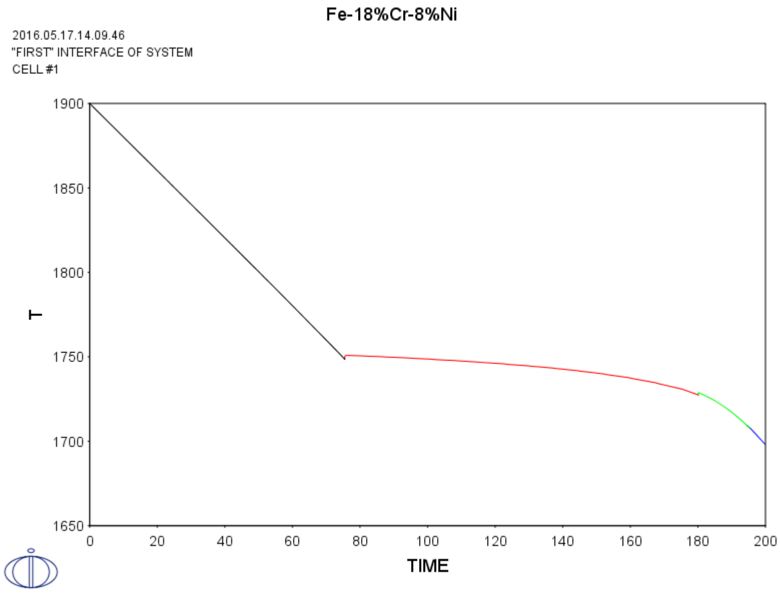
exb4d-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4d\run.DCM"
DIC> @@ exb4d_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b4b
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb4d
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  8  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA
 3.863918610058573E-007        3.864691456613560E-007        3.834124705096621E-016        3.828409457237955E-025
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.38284095E-24
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA
 6.376920891964565E-018
 TIME =  0.21475717E-05 DT =  0.20475717E-05 SUM OF SQUARES =  0.63769209E-17
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds
 2.550767959196425E-017
 TIME =  0.62427150E-05 DT =  0.40951433E-05 SUM OF SQUARES =  0.25507680E-16
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds
 1.020307183678570E-016        1.020307183678570E-016


     ERROR RETURN FROM NS01A BECAUSE A NEARBY STATIONARY POINT OF F(X) IS PREDICTED

 *** ERROR  1890 IN DCNS01
 *** ERROR RETURN FROM NS01A
 1.020307183678570E-016        1.020307183678570E-016        1.020307183678570E-016        6.442437456732094E-018        9.472916511688731E-
017        1.294091040261595E-017        1.294091040261595E-017        4.146316551541531E-020        4.146316551541531E-
020        4.146316551541531E-020        4.146316551541531E-020        1.017673253704363E-017        4.146276476914365E-
020        4.146276476914365E-020        4.146276476914365E-020        4.146276476914365E-020
 TIME =  0.14433002E-04 DT =  0.81902866E-05 SUM OF SQUARES =  0.41462765E-19
 TEMPERATURE:   1900.     ENTHALPY:  0.7394E+05
 U-FRACTION IN SYSTEM:  CR = .191520367992483  FE = .733068011219292
                        NI = .0754116207882254
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0  seconds
 1.113040265032366E-016        1.113040265032366E-016


     ERROR RETURN FROM NS01A BECAUSE A NEARBY STATIONARY POINT OF F(X) IS PREDICTED

 *** ERROR  1890 IN DCNS01
 *** ERROR RETURN FROM NS01A
 1.113040265032366E-016        1.113040265032366E-016        1.113040265032366E-016        9.113523509687586E-013        1.113040265032366E-
016        7.266876066376097E-032        7.266876066376097E-032        7.266876066376097E-032        7.266876066376097E-
032        1.113040265032366E-016        7.395841312681207E-031        7.395841312681207E-031

output ignored...



... output resumed

 3.134867858854075E-007        3.080213721813523E-007        2.941859684782598E-007        2.941867726247802E-007        2.717743200237868E-
007        2.193153739109705E-007        1.428732908781453E-007        1.429123033953288E-007        3.101974213296560E-
008        1.030961691570364E-013        7.185975794656296E-014        3.780734320397528E-020
 TIME =   196.73008     DT =  0.70100947     SUM OF SQUARES =  0.37807343E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.61099744E-07 AND   0.61099744E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.55141385E-05
 TEMPERATURE:   1705.     ENTHALPY:  0.5598E+05
 U-FRACTION IN SYSTEM:  CR = .190908171700819  FE = .733905325525142
                        NI = .0751865027740392
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  3  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_FCC_A1

  CPU time used in timestep                    3  seconds
 1.260148240944312E-006        1.260140990623653E-006        1.260232336611207E-006        1.259896592402558E-006        1.117680360227887E-
006        8.646283243390180E-007        5.706998606874766E-007        6.450195676987618E-008        4.012506745014100E-
011        1.030798469944521E-011        2.480499204462683E-017
 TIME =   198.13210     DT =   1.4020189     SUM OF SQUARES =  0.24804992E-16
```

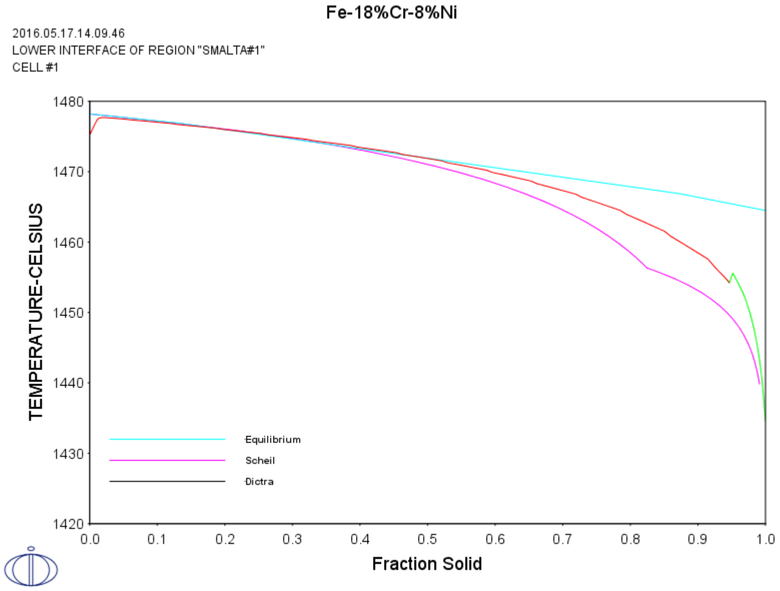
```
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.82926178E-07 AND   0.82926178E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.56304026E-05
 TEMPERATURE:   1702.     ENTHALPY:  0.5585E+05
 U-FRACTION IN SYSTEM:  CR = .190908108674085  FE = .733905359047497
                        NI = .0751865322784179
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  12  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: R_FCC_A1

  CPU time used in timestep                     2  seconds
 5.573888841648700E-007        5.573800772694394E-007        5.575027082405226E-007        5.570547482888640E-007        3.354392831785962E-
007        6.104059833865107E-008        1.519888678943710E-011        8.247735887165833E-013        5.251646673740660E-020
 TIME =   200.00000     DT =   1.8679016     SUM OF SQUARES =  0.52516467E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.99919714E-07 AND   0.99919714E-07
 POSITION OF INTERFACE R_FCC_A1 / R_BCC_A2 IS   0.58170428E-05
 TEMPERATURE:   1698.     ENTHALPY:  0.5568E+05
 U-FRACTION IN SYSTEM:  CR = .190908055440512  FE = .733905368903247
                        NI = .0751865756562413
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    194.48925
 DELETING TIME-RECORD FOR TIME    195.41113
 DELETING TIME-RECORD FOR TIME    195.50331
 DELETING TIME-RECORD FOR TIME    195.67857
 DELETING TIME-RECORD FOR TIME    196.02907
 DELETING TIME-RECORD FOR TIME    196.73008

 KEEPING TIME-RECORD FOR TIME    198.13210
 AND FOR TIME                    200.00000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb4d-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb4d\plot.DCM"DIC>
DIC>
DIC> @@ exb4d_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b4b
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  2.00000E+02
DIC> read exb4d
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Fe-18%Cr-8%Ni
POST-1:
POST-1: @@
POST-1: @@ PLOT FRACTION SOLID AND COMPARE WITH SCHEIL-GULLIVER SIMULATION AND
POST-1: @@ EQUILIBRIUM SOLIDIFICATION (DATA ON FILE exb4.exp)
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y T
POST-1: s-p-c inter first
POST-1: plot
```



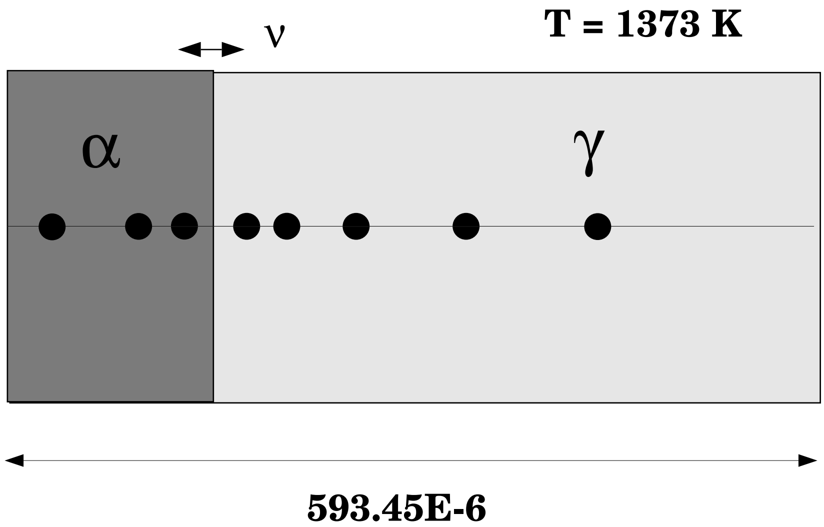

```
POST-1:
POST-1:Hit RETURN to continue
POST-1: enter func fs=1-ivv(liquid);
POST-1: s-d-a x fs
POST-1: s-s-s x n 0 1
POST-1: s-ax-te x n Fraction Solid
POST-1:
POST-1: s-d-a y t-c
POST-1: s-s-s y n 1420 1480
POST-1:
POST-1: s-p-c interf smalta lower
POST-1:
POST-1: app y exb4d.exp 0; 1
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example b5

$\gamma/\alpha/\gamma$ diffusion couple of Fe-Ni-Cr alloys



exb5-setup

```
About Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb5\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ This example demonstrates the evaluation of a ternary Fe-Cr-Ni diffusion
SYS: @@ couple. A thin slice of ALPHA phase (38%Cr, 0%Ni) is clamped between
SYS: @@ two thicker slices of GAMMA phase (27%Cr, 20%Ni). The assembly is
SYS: @@ subsequently heat treated at 1373K. This setup corresponds to diffusion
SYS: @@ couple A in M. Kajihara, C.-B. Lim and M. Kikuchi: ISIJ International
SYS: @@ 33 (1993), pp. 498-507. See also M. Kajihara and M. Kikichi: Acta Metall.Mater.
SYS: @@ 41 (1993), pp.2045-2059.
SYS: @@
SYS:
SYS: @@ exb5_setup.DCM
SYS:
SYS: @@
SYS: @@ GOTO DATABASE AND READ THERMODYNAMIC AND KINETIC DATA.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO: def-sys cr fe ni
 CR                       FE                       NI
   DEFINED
TDB_FEDEMO: rej-ph /all
 LIQUID:L                 BCC_A2                   CHI_A12
 FCC_A1                   HCP_A3                   LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res-ph bcc,fcc
 BCC_A2                   FCC_A1  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
```

```
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /FEDEMO/: mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys cr fe ni
 CR                      FE                      NI
   DEFINED
APP: rej-ph /all
 BCC_A2                  FCC_A1  REJECTED
APP: res-ph bcc,fcc
 BCC_A2                  FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
     Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
     -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
     diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
     in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ GOTO DICTRA MODULE TO SETUP THE SIMULATION
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ SET GLOBAL CONDITIONS
DIC> @@
DIC> set-cond glob T 0 1373; * N

DIC>
DIC> @@
DIC> @@ ENTER TWO REGIONS, ONE FOR EACH PHASE
DIC> @@
DIC> enter-region alfa
DIC> enter-region gamma
ATTACH TO REGION NAMED /ALFA/:
ATTACHED TO THE RIGHT OF ALFA /YES/:
DIC> @@
```

```
DIC> @@ ENTER GRID SIZE AND SPACINGS
DIC> @@
DIC> enter-grid alfa  93.45E-6 geo 50 0.8
DIC> enter-grid gamma 500.0E-6 geo 50 1.05
DIC>
DIC> @@
DIC> @@ SPECIFY WHICH PHASE GOES INTO WHICH REGION
DIC> @@
DIC> enter-phase act alfa  matrix bcc
DIC> enter-phase act gamma matrix fcc
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE PHASES
DIC> @@ N.B. DO NOT PUT 0%NI IN PHASE BCC, USE SOME SMALL VALUE INSTEAD
DIC> @@
DIC> enter-composition
REGION NAME : /ALFA/: alfa
PHASE NAME: /BCC_A2/: bcc
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-f
PROFILE FOR /CR/: cr lin  .38 .38
PROFILE FOR /NI/: ni lin  1e-5 1e-5
DIC>
DIC> enter-composition
REGION NAME : /GAMMA/: gamma
PHASE NAME: /FCC_A1/: fcc
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-f
PROFILE FOR /CR/: cr lin .27 .27
PROFILE FOR /NI/: ni lin .28 .28
DIC>
DIC> @@
DIC> @@ SPECIFY SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 36E5
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /360000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> SAVE exb5 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

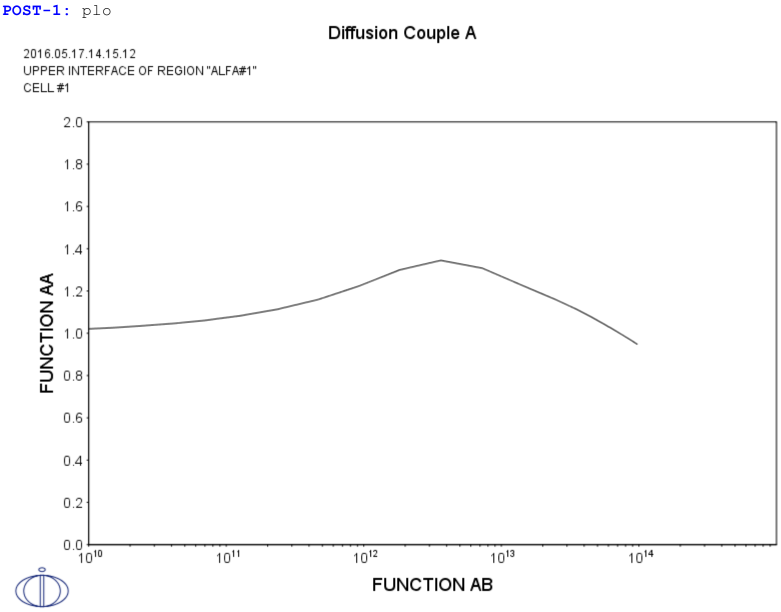
exb5-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb5\run.DCM"
DIC>
DIC> @@ exb5_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE b5
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER DICTRA MONITOR AND READ THE SETUP FROM FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb5
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> simulate
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                    4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS
 *** ERROR  1611 IN QTHISS
 *** TOO MANY ITERATIONS
 Give the command INFO TROUBLE for help

 *** ERROR  1611 IN QTHISS
 *** TOO MANY ITERATIONS
 Give the command INFO TROUBLE for help

 *** ERROR  1611 IN QTHISS
 *** TOO MANY ITERATIONS
 Give the command INFO TROUBLE for help
 DONE      6 OUT OF      9                                                 DONE     9 OUT OF     9
  try                       3  failed
 DETERMINED ACTIVITIES  ACR(CR) .00275331206673

  UNABLE TO OBTAIN GOOD STARTING VALUE USING THE OLD SCHEME
USE NEW SCHEME /YES/:

  Trying new scheme
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:        18   EQUILIBRIUM CALCULATIONS             DONE       1 OUT OF      18
04

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .305280432605602  FE = .471672082221692
                        NI = .223047485172706
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]
 U-FRACTION IN SYSTEM:  CR = .305280432605602  FE = .471672082221692
                        NI = .223047485172706
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]
  13  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: ALFA
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: GAMMA
 0.675892969673219        0.675921598415291        0.675324065177569        0.258956717329603        0.224804955198800        0.16450503770297
002        6.022431656019592E-003        1.019996827594893E-004        2.311253953313433E-006        1.352635011035715E-
009        3.583818060655041E-014        2.453979817164101E-007        1.577916194008969E-021
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.15779162E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   -1.2940192     AND   -1.2940192
 POSITION OF INTERFACE ALFA / GAMMA IS   0.93320598E-04
 U-FRACTION IN SYSTEM:  CR = .305290714836365  FE = .471686804196505
                        NI = .22302248096713
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]

  16  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: ALFA

  CPU time used in timestep                   1  seconds
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: ALFA
 9.889218077032400E-004        9.891196210603200E-004        9.892010187275732E-004        4.513086970466132E-004        1.699616033992955E-
004        5.936365506688998E-007        5.137284765972353E-007        5.258824716836466E-007        5.200776949064532E-
007        5.160997330447358E-007        4.500872904422601E-007        4.535434545363741E-007        4.497954131662141E-
007        4.472088473559315E-007        4.426188644160794E-007        4.368130797713725E-007

output ignored...



... output resumed

 1.106334372673309E-012        1.698721348301072E-017
 TIME =   2665230.0      DT =   360000.00     SUM OF SQUARES =  0.16987213E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.63558133E-11 AND  -0.63558133E-11
 POSITION OF INTERFACE ALFA / GAMMA IS   0.92616241E-04
 U-FRACTION IN SYSTEM:  CR = .305364010200166  FE = .471375759357622
                        NI = .223260230442212
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: GAMMA

  CPU time used in timestep                   2  seconds
 9.563146498314843E-010        9.559803869244067E-010        9.110554441444396E-010        3.412711525694628E-012        3.078309671680936E-
012        2.473548398268935E-012        1.461524613323578E-012        2.014175066382132E-012        2.308161620493316E-
013        3.059837715250873E-019
 TIME =   3025230.0      DT =   360000.00     SUM OF SQUARES =  0.30598377E-18
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.57034810E-11 AND  -0.57034810E-11
 POSITION OF INTERFACE ALFA / GAMMA IS   0.90562988E-04
 U-FRACTION IN SYSTEM:  CR = .305364180961055  FE = .471375583059625
                        NI = .223260235979319
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]
```

```
  7  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: GAMMA

  CPU time used in timestep                      1  seconds
 8.286004013450376E-010        8.283152627009559E-010        7.867060860016127E-010        1.381843128880088E-012        1.202523903029534E-
012        8.889485611422328E-013        4.034436062712664E-013        9.579659673970775E-013        4.761318699927787E-
016        3.213941240446396E-020
 TIME =   3385230.0      DT =    360000.00      SUM OF SQUARES =  0.32139412E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.52565619E-11 AND  -0.52565619E-11
 POSITION OF INTERFACE ALFA / GAMMA IS   0.88670626E-04
 U-FRACTION IN SYSTEM:  CR = .305364353202139  FE = .47137532625928
                        NI = .223260320538581
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]

  5  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: GAMMA

  CPU time used in timestep                      1  seconds
 5.948635803278480E-011        5.949453684102886E-011        5.508807804883499E-011        2.595887056595887E-013        2.400666603806427E-
013        2.034208662867204E-013        1.391376996512642E-013        2.241819982971998E-013        4.713425943220663E-
014        2.078054093834980E-021
 TIME =   3600000.0      DT =    214770.03      SUM OF SQUARES =  0.20780541E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.47956182E-11 AND  -0.47956182E-11
 POSITION OF INTERFACE ALFA / GAMMA IS   0.87640671E-04
 U-FRACTION IN SYSTEM:  CR = .305364466880408  FE = .471375115464078
                        NI = .223260417655514
 TOTAL SIZE OF SYSTEM:  5.9345E-04 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.94220141E-06
 DELETING TIME-RECORD FOR TIME   0.26266042E-05
 DELETING TIME-RECORD FOR TIME   0.59954099E-05
 DELETING TIME-RECORD FOR TIME   0.12733021E-04
 DELETING TIME-RECORD FOR TIME   0.26208244E-04
 DELETING TIME-RECORD FOR TIME   0.53158689E-04
 DELETING TIME-RECORD FOR TIME   0.10705958E-03
 DELETING TIME-RECORD FOR TIME   0.21486136E-03
 DELETING TIME-RECORD FOR TIME   0.43046492E-03
 DELETING TIME-RECORD FOR TIME   0.86167204E-03
 DELETING TIME-RECORD FOR TIME   0.17240863E-02
 DELETING TIME-RECORD FOR TIME   0.34489148E-02
 DELETING TIME-RECORD FOR TIME   0.68985718E-02
 DELETING TIME-RECORD FOR TIME   0.13797886E-01
 DELETING TIME-RECORD FOR TIME   0.23215301E-01
 DELETING TIME-RECORD FOR TIME   0.40621931E-01
 DELETING TIME-RECORD FOR TIME   0.50155935E-01
 DELETING TIME-RECORD FOR TIME   0.69223943E-01
 DELETING TIME-RECORD FOR TIME   0.10735996
 DELETING TIME-RECORD FOR TIME   0.18363199
 DELETING TIME-RECORD FOR TIME   0.22540790
 DELETING TIME-RECORD FOR TIME   0.30895973
 DELETING TIME-RECORD FOR TIME   0.47606338
 DELETING TIME-RECORD FOR TIME   0.81027068
 DELETING TIME-RECORD FOR TIME    1.0827341
 DELETING TIME-RECORD FOR TIME    1.6276609
 DELETING TIME-RECORD FOR TIME    2.7175146
 DELETING TIME-RECORD FOR TIME    4.8972220
 DELETING TIME-RECORD FOR TIME    7.1375798
 DELETING TIME-RECORD FOR TIME    11.618295
 DELETING TIME-RECORD FOR TIME    14.072484
 DELETING TIME-RECORD FOR TIME    18.980862
 DELETING TIME-RECORD FOR TIME    28.797619
 DELETING TIME-RECORD FOR TIME    48.431131
 DELETING TIME-RECORD FOR TIME    70.272705
 DELETING TIME-RECORD FOR TIME    113.95585
 DELETING TIME-RECORD FOR TIME    152.26676
 DELETING TIME-RECORD FOR TIME    228.88858
 DELETING TIME-RECORD FOR TIME    341.33861
 DELETING TIME-RECORD FOR TIME    514.90908
 DELETING TIME-RECORD FOR TIME    862.05001
 DELETING TIME-RECORD FOR TIME    1469.1405
 DELETING TIME-RECORD FOR TIME    2454.9738
 DELETING TIME-RECORD FOR TIME    4426.6405
 DELETING TIME-RECORD FOR TIME    8369.9737
 DELETING TIME-RECORD FOR TIME    16256.640
 DELETING TIME-RECORD FOR TIME    32029.973
 DELETING TIME-RECORD FOR TIME    63576.640
 DELETING TIME-RECORD FOR TIME    126669.97
 DELETING TIME-RECORD FOR TIME    252856.64
 DELETING TIME-RECORD FOR TIME    505229.97
 DELETING TIME-RECORD FOR TIME    865229.97
 DELETING TIME-RECORD FOR TIME    1225230.0
 DELETING TIME-RECORD FOR TIME    1585230.0
 DELETING TIME-RECORD FOR TIME    1945230.0
 DELETING TIME-RECORD FOR TIME    2305230.0
 DELETING TIME-RECORD FOR TIME    2665230.0
 DELETING TIME-RECORD FOR TIME    3025230.0

 KEEPING TIME-RECORD FOR TIME    3385230.0
 AND FOR TIME                    3600000.0
 WORKSPACE RECLAIMED
DIC> set-inter
 --OK---
DIC>
```

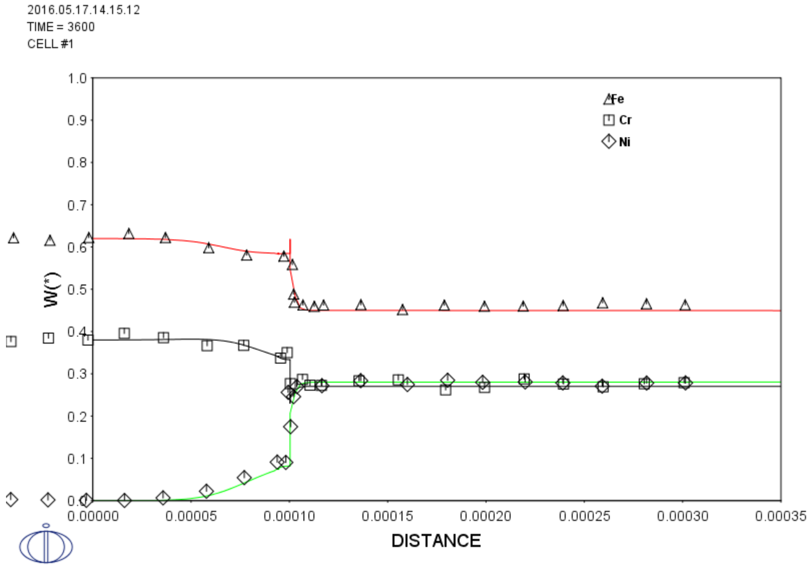
exb5-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb5\plot.DCM"DIC>
DIC>
DIC> @@ exb5_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE b5
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.60000E+06
DIC> read exb5
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE POST PROCESSOR, PLOT SOME QUANTITIES AND COMPARE WITH EXPERIMENTS
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Diffusion Couple A
POST-1:
POST-1: @@
POST-1: @@ WE ARE INTERESTED IN THE POSITION OF THE UPPER INTERFACE OF REGION ALFA
POST-1: @@
POST-1: s-p-c interf alfa upper
POST-1:
POST-1: @@
POST-1: @@ l0 IS THE INITIAL THICKNESS USED FOR NORMALIZATION
POST-1: @@
POST-1: enter func l0=186.9e-6;
POST-1: enter func aa=2*poi(alfa,u)/l0;
POST-1: enter func ab=time/l0**2;
POST-1: s-i-v time
POST-1:
POST-1: s-d-a x ab
POST-1: s-s-s x n 1e10 1e15
POST-1: s-ax-ty x log
POST-1:
POST-1: s-d-a y aa
POST-1: s-s-s y n 0 2
POST-1:
POST-1: app y exb5.exp
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 7
POST-1:
POST-1: plo
```



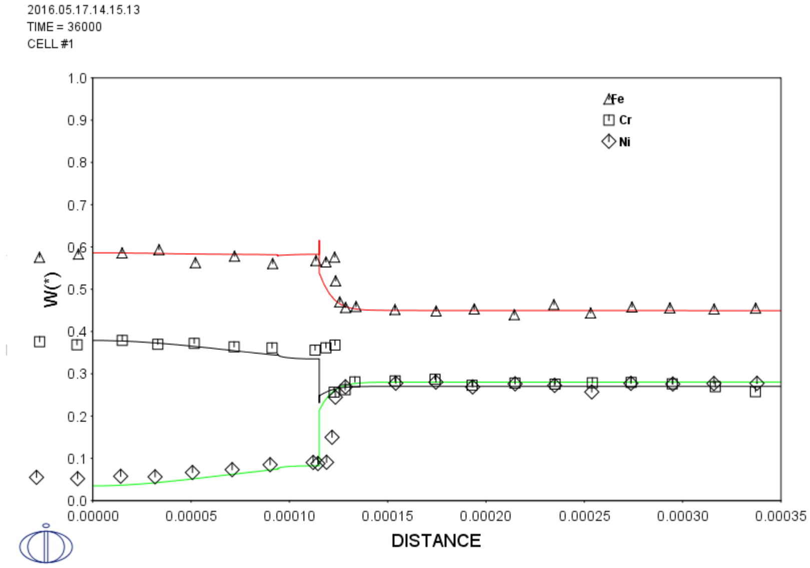

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ LET US PLOT CONCENTRATION PROFILES FOR SOME DIFFERENT ANNEALING TIMES
POST-1: @@
POST-1: s-d-a x dist glo
  INFO: Distance is set as independent variable
POST-1: s-ax-ty x lin
POST-1: s-s-s x n 0 350e-6
POST-1:
POST-1: s-d-a y w(*)
POST-1: s-s-s y n 0 1
POST-1:
POST-1: s-p-c time 3600
POST-1:
POST-1: app y exb5.exp 0; 1
POST-1: plo
```

Diffusion Couple A

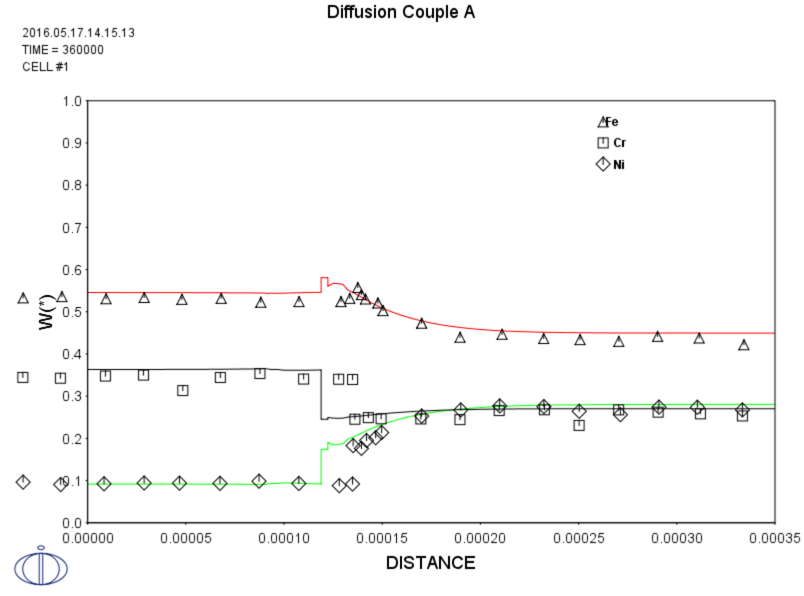



```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 36000
POST-1: app y exb5.exp 0; 2
POST-1: plo
```

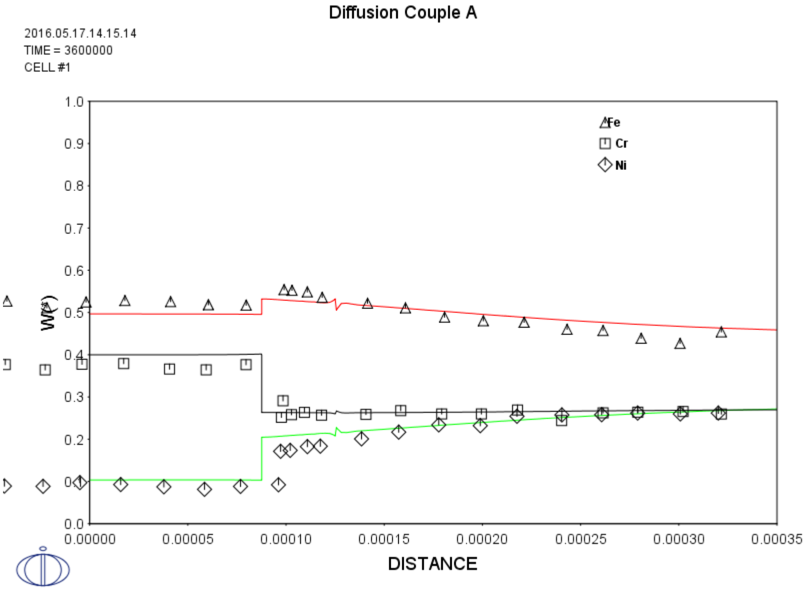
Diffusion Couple A



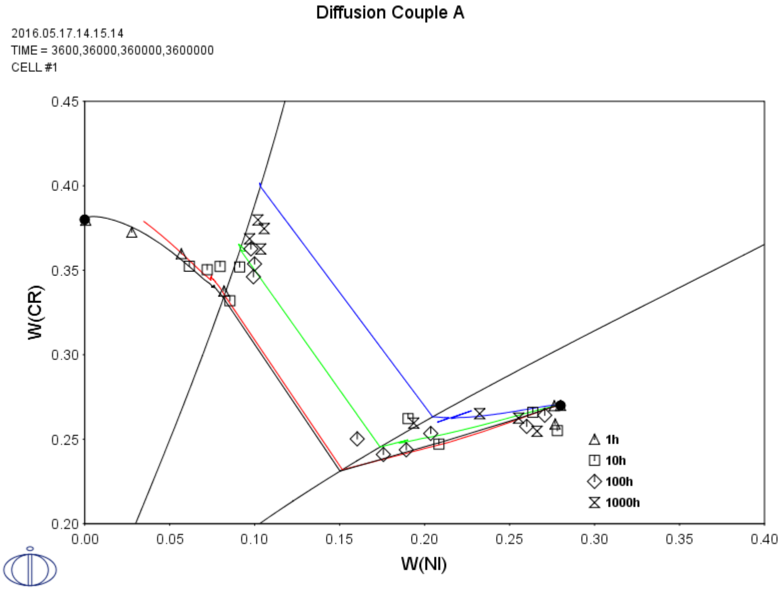

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 360000
POST-1: app y exb5.exp 0; 3
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 3600000
POST-1: app y exb5.exp 0; 4
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ FINALLY PLOT SOME DIFFERENT DIFFUSION PATHS.
POST-1: @@
POST-1: s-d-a x w(ni)
POST-1: s-s-s x n .00 .40
POST-1:
POST-1: s-d-a y w(cr)
POST-1: s-s-s y n .20 .45
POST-1:
POST-1: s-i-v dist glob
POST-1:
POST-1: s-p-c time 3600,36000,360000,3600000
POST-1:
POST-1: app y exb5.exp 0; 5 6
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

exb6-setup

About Linked: Tue May 17 11:39:18 2016

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb6\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ This example illustrates the effect of microsegregation
SYS: @@ of phosphorus during peritectic solidification in steel.
SYS: @@
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@ LET US USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE7:
TDB_TCFE7: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_TCFE7: def-sys fe c si mn p
 FE                      C                       SI
 MN                      P  DEFINED
TDB_TCFE7:
TDB_TCFE7: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_TCFE7: rej ph /all
 GAS:G                   LIQUID:L                BCC_A2
 FCC_A1                  HCP_A3                  DIAMOND_FCC_A4
 RED_P                   WHITE_P                 GRAPHITE
 CEMENTITE               M23C6                   M7C3
 M5C2                    KSI_CARBIDE             A1_KAPPA
 KAPPA                   FE4N_LP1                FECN_CHI
 LAVES_PHASE_C14         M3SI                    G_PHASE
 CR3SI                   FE2SI                   MSI
 M5SI3                   AL4C3                   FE8SI2C
 SIC                     FEP                     CU3P
 M2P                     M3P  REJECTED
TDB_TCFE7: res ph fcc liq bcc
 FCC_A1                  LIQUID:L                BCC_A2
   RESTORED
TDB_TCFE7:
TDB_TCFE7: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo-C, C-Mn'
  'J. Grobner, H.L. Lukas and F. Aldinger, Calphad, 20 (1996), 247-254; Si-C
      and Al-Si-C'
  'W. Huang, Calphad, 13 (1989), 243-252; TRITA-MAC 388 (rev 1989); FE-MN'
  'J.-H. Shim, C.-S. Oh and D.N. Lee, J. Korean Inst. Met. Mater., 34 (1996),
       1385-1393; Fe-P'
  'J. Lacaze and B. Sundman, Metall. Mater. Trans. A, 22A (1991), 2211-2223;
      Fe-Si and Fe-Si-C'
  'J. Miettinen and B. Hallstedt, Calphad, 22 (1998), 231-256; Fe-Si and Fe
      -Si-C'
  'J.E. Tibballs, SI Norway (1991) Rep. 890221-5; Mn-Si'
  'P. Franke, estimated parameter within SGTE, 2008; Fe-Mn-C'
  'J.-H. Shim, H.-J. Chung and D.N. Lee, Z. fur Metallkde., (1999); Fe-C-P'
  'B. Sundman, 1999, revision of the liquid Fe-Si-C description'
  'NPL, unpublished work (1989); C-Mn-Si'
  'A. Forsberg and J. Agren, J. Phase Equil., 14 (1993), 354-363; Fe-Mn-Si'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
      hard mater, Vol. 12, pp. 121-127; Molar volumes'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'W. Huang, Metall. Trans. A, 21A (1990), 2115-2123; TRITA-MAC 411 (Rev
      1989); C-FE-MN'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE AND APPEND DATA
TDB_TCFE7: @@
TDB_TCFE7: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
```

```
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /TCFE7/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe c si mn p
 FE                       C                        SI
 MN                       P  DEFINED
APP: rej ph /all
 BCC_A2                   CEMENTITE                FCC_A1
 FE4N_LP1                 HCP_A3                   LIQUID:L
   REJECTED
APP: res ph fcc liq bcc
 FCC_A1                   LIQUID:L                 BCC_A2
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
```

```
 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'Bae et al.: Z. Metallkunde 91(2000)672-674; fcc Fe-Mn
       Mn-Ni'
  'V. V. Mural and P. L. Gruzin, Phys. Met. Metallogr. (English Transl.) 17
      (5) (1964)154.; Impurity diff of P in fcc Fe.'
  'D. Bergner et al., Defect and Diffusion Forum 66-69(1989)409. Impurity
      diffusion of Si in fcc Fe.'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
  'B. Jonsson: Unpublished research bcc Fe-Si 1994'
  'Assessed from data presented in Landholt-Bornstein, Vol. 26, ed. H.
      Mehrer, springer (1990); Impurity diff of Mn in bcc Fe.'
  'Assessed from data presented in Landholt-Bornstein, Vol. 26, ed. H.
      Mehrer, springer (1990); Impurity diff of P in bcc Fe.'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> @@ LET US LOWER THE TEMPERATURE WITH A RATE OF 0.2 K/S
DIC> @@
DIC> set-cond glob T 0 1780-0.2*TIME; * N

DIC>
DIC> @@
DIC> @@ ENTER A REGION CALLED smalta
DIC> @@
DIC> enter-region smalta
DIC>
DIC> @@
DIC> @@ ENTER A DOUBLE GEOMETRIC GRID INTO THE REGION.
DIC> @@
DIC> enter-grid
REGION NAME : /SMALTA/: smalta
WIDTH OF REGION /1/: 1e-4
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 1.11
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 0.9
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /SMALTA/: smalta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: liq
DIC>
DIC> @@
DIC> @@ ENTER INACTIVE PHASES INTO REGION, BOTH PHASES ON THE SAME SIDE OF
DIC> @@ THE LIQUID REGION IN ORDER TO GET A PERITECTIC REACTION.
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: no
PHASE NAME: /NONE/: fcc#1
DEPENDENT COMPONENT ? /SI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: inact
ATTACH TO REGION NAMED /SMALTA/: smalta
ATTACHED TO THE RIGHT OF SMALTA /YES/: no
PHASE NAME: /NONE/: bcc#1
DEPENDENT COMPONENT ? /SI/: fe
 REQUIRED DRIVING FORCE FOR PRECIPITATION: /1E-05/: 1e-5
CONDITION TYPE /CLOSED_SYSTEM/: closed
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR THE LIQUID
DIC> @@
DIC> enter-composition
REGION NAME : /SMALTA/: smalta
PHASE NAME: /LIQUID/: liq
DEPENDENT COMPONENT ? /SI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c lin 0.4 0.4
PROFILE FOR /MN/: si lin 0.7 0.7
PROFILE FOR /P/: mn lin 0.8 0.8
PROFILE FOR /SI/: p lin 0.03 0.03
DIC>
DIC> @@
DIC> @@ BOUNDARY CONDITION WILL BE A CLOSED SYSTEM (DEFAULT) AS WE DO NOT SPECIFY
DIC> @@ ANYTHING ELSE.
DIC> @@
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 3000
AUTOMATIC TIMESTEP CONTROL /YES/: yes
MAX TIMESTEP DURING INTEGRATION /300/: 15
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
```

```
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ CHECK INTERFACE POSSITION
DIC> @@
DIC> @@ THIS IN ORDER TO MAKE SURE THAT THE LIQUID REGION DOESN'T SHRINK
DIC> @@ TO MUCH DURING A TIMESTEP. THE TIMESTEP WILL NOW IN ADDITION BE
DIC> @@ CONTROLLED BY THE PHASE INTERFACE DISPLACEMENT DURING THE SIMULATION.
DIC> @@
DIC> s-s-c
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/: yes
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb6 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb6-run

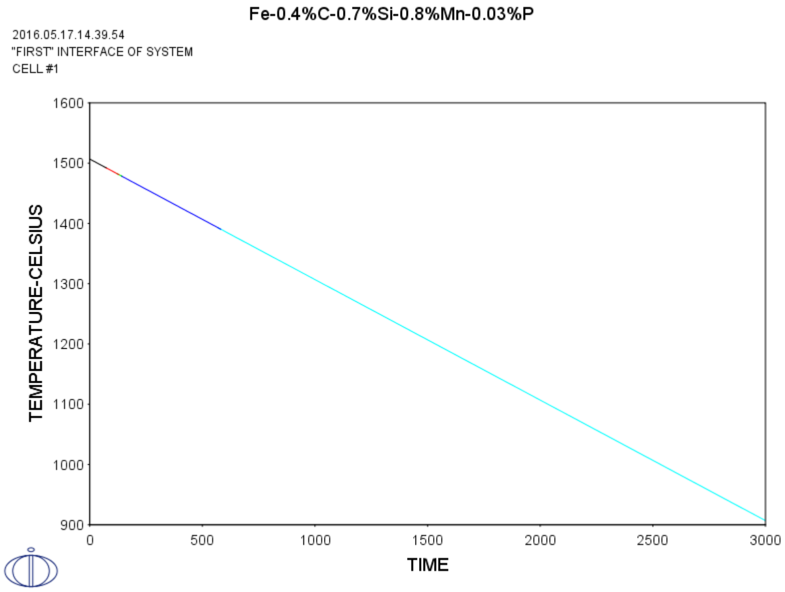
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb6\run.DCM"DIC>
DIC>
DIC> @@ exb6_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exb6
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: SMALTA

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.92411922E-04 DT =  0.92311922E-04 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.27703577E-03 DT =  0.18462384E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.64628345E-03 DT =  0.36924769E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.13847788E-02 DT =  0.73849537E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.28617696E-02 DT =  0.14769907E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.58157511E-02 DT =  0.29539815E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.2932119056092E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.11723714E-01 DT =  0.59079630E-02 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.2932119056092E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.23539640E-01 DT =  0.11815926E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.47171492E-01 DT =  0.23631852E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.2932119056092E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                    0   seconds

 TIME =  0.94435196E-01 DT =  0.47263704E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.2932119056092E-04
```

```
                         SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.18896260     DT =  0.94527408E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198536  P = 5.2932119056092E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.37801742     DT =  0.18905482     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054
                        MN = .00795815625198537  P = 5.29321190560921E-04
                        SI = .0136208267339645
 TOTAL SIZE OF SYSTEM:  1E-04 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.75612705     DT =  0.37810963     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0182001993244348  FE = .959691496499054

output ignored...

... output resumed

 DELETING TIME-RECORD FOR TIME    854.70672
 DELETING TIME-RECORD FOR TIME    869.70672
 DELETING TIME-RECORD FOR TIME    884.70672
 DELETING TIME-RECORD FOR TIME    899.70672
 DELETING TIME-RECORD FOR TIME    914.70672
 DELETING TIME-RECORD FOR TIME    929.70672
 DELETING TIME-RECORD FOR TIME    944.70672
 DELETING TIME-RECORD FOR TIME    959.70672
 DELETING TIME-RECORD FOR TIME    974.70672
 DELETING TIME-RECORD FOR TIME    989.70672
 DELETING TIME-RECORD FOR TIME    1004.7067
 DELETING TIME-RECORD FOR TIME    1019.7067
 DELETING TIME-RECORD FOR TIME    1034.7067
 DELETING TIME-RECORD FOR TIME    1049.7067
 DELETING TIME-RECORD FOR TIME    1064.7067
 DELETING TIME-RECORD FOR TIME    1079.7067
 DELETING TIME-RECORD FOR TIME    1094.7067
 DELETING TIME-RECORD FOR TIME    1109.7067
 DELETING TIME-RECORD FOR TIME    1124.7067
 DELETING TIME-RECORD FOR TIME    1139.7067
 DELETING TIME-RECORD FOR TIME    1154.7067
 DELETING TIME-RECORD FOR TIME    1169.7067
 DELETING TIME-RECORD FOR TIME    1184.7067
 DELETING TIME-RECORD FOR TIME    1199.7067
 DELETING TIME-RECORD FOR TIME    1214.7067
 DELETING TIME-RECORD FOR TIME    1229.7067
 DELETING TIME-RECORD FOR TIME    1244.7067
 DELETING TIME-RECORD FOR TIME    1259.7067
 DELETING TIME-RECORD FOR TIME    1274.7067
 DELETING TIME-RECORD FOR TIME    1289.7067
 DELETING TIME-RECORD FOR TIME    1304.7067
 DELETING TIME-RECORD FOR TIME    1319.7067
 DELETING TIME-RECORD FOR TIME    1334.7067
 DELETING TIME-RECORD FOR TIME    1349.7067
 DELETING TIME-RECORD FOR TIME    1364.7067
 DELETING TIME-RECORD FOR TIME    1379.7067
 DELETING TIME-RECORD FOR TIME    1394.7067
 DELETING TIME-RECORD FOR TIME    1409.7067
 DELETING TIME-RECORD FOR TIME    1424.7067
 DELETING TIME-RECORD FOR TIME    1439.7067
 DELETING TIME-RECORD FOR TIME    1454.7067
 DELETING TIME-RECORD FOR TIME    1469.7067
 DELETING TIME-RECORD FOR TIME    1484.7067
 DELETING TIME-RECORD FOR TIME    1499.7067
 DELETING TIME-RECORD FOR TIME    1514.7067
 DELETING TIME-RECORD FOR TIME    1529.7067
 DELETING TIME-RECORD FOR TIME    1544.7067
 DELETING TIME-RECORD FOR TIME    1559.7067
 DELETING TIME-RECORD FOR TIME    1574.7067
 DELETING TIME-RECORD FOR TIME    1589.7067
 DELETING TIME-RECORD FOR TIME    1604.7067
 DELETING TIME-RECORD FOR TIME    1619.7067
 DELETING TIME-RECORD FOR TIME    1634.7067
 DELETING TIME-RECORD FOR TIME    1649.7067
 DELETING TIME-RECORD FOR TIME    1664.7067
 DELETING TIME-RECORD FOR TIME    1679.7067
 DELETING TIME-RECORD FOR TIME    1694.7067
 DELETING TIME-RECORD FOR TIME    1709.7067
 DELETING TIME-RECORD FOR TIME    1724.7067
 DELETING TIME-RECORD FOR TIME    1739.7067
 DELETING TIME-RECORD FOR TIME    1754.7067
 DELETING TIME-RECORD FOR TIME    1769.7067
 DELETING TIME-RECORD FOR TIME    1784.7067
 DELETING TIME-RECORD FOR TIME    1799.7067
 DELETING TIME-RECORD FOR TIME    1814.7067
 DELETING TIME-RECORD FOR TIME    1829.7067
 DELETING TIME-RECORD FOR TIME    1844.7067
 DELETING TIME-RECORD FOR TIME    1859.7067
 DELETING TIME-RECORD FOR TIME    1874.7067
 DELETING TIME-RECORD FOR TIME    1889.7067
 DELETING TIME-RECORD FOR TIME    1904.7067
 DELETING TIME-RECORD FOR TIME    1919.7067
 DELETING TIME-RECORD FOR TIME    1934.7067
 DELETING TIME-RECORD FOR TIME    1949.7067
 DELETING TIME-RECORD FOR TIME    1964.7067
 DELETING TIME-RECORD FOR TIME    1979.7067
 DELETING TIME-RECORD FOR TIME    1994.7067
 DELETING TIME-RECORD FOR TIME    2009.7067
 DELETING TIME-RECORD FOR TIME    2024.7067
 DELETING TIME-RECORD FOR TIME    2039.7067
 DELETING TIME-RECORD FOR TIME    2054.7067
 DELETING TIME-RECORD FOR TIME    2069.7067
 DELETING TIME-RECORD FOR TIME    2084.7067
 DELETING TIME-RECORD FOR TIME    2099.7067
 DELETING TIME-RECORD FOR TIME    2114.7067
```

```
 DELETING TIME-RECORD FOR TIME    2129.7067
 DELETING TIME-RECORD FOR TIME    2144.7067
 DELETING TIME-RECORD FOR TIME    2159.7067
 DELETING TIME-RECORD FOR TIME    2174.7067
 DELETING TIME-RECORD FOR TIME    2189.7067
 DELETING TIME-RECORD FOR TIME    2204.7067
 DELETING TIME-RECORD FOR TIME    2219.7067
 DELETING TIME-RECORD FOR TIME    2234.7067
 DELETING TIME-RECORD FOR TIME    2249.7067
 DELETING TIME-RECORD FOR TIME    2264.7067
 DELETING TIME-RECORD FOR TIME    2279.7067
 DELETING TIME-RECORD FOR TIME    2294.7067
 DELETING TIME-RECORD FOR TIME    2309.7067
 DELETING TIME-RECORD FOR TIME    2324.7067
 DELETING TIME-RECORD FOR TIME    2339.7067
 DELETING TIME-RECORD FOR TIME    2354.7067
 DELETING TIME-RECORD FOR TIME    2369.7067
 DELETING TIME-RECORD FOR TIME    2384.7067
 DELETING TIME-RECORD FOR TIME    2399.7067
 DELETING TIME-RECORD FOR TIME    2414.7067
 DELETING TIME-RECORD FOR TIME    2429.7067
 DELETING TIME-RECORD FOR TIME    2444.7067
 DELETING TIME-RECORD FOR TIME    2459.7067
 DELETING TIME-RECORD FOR TIME    2474.7067
 DELETING TIME-RECORD FOR TIME    2489.7067
 DELETING TIME-RECORD FOR TIME    2504.7067
 DELETING TIME-RECORD FOR TIME    2519.7067
 DELETING TIME-RECORD FOR TIME    2534.7067
 DELETING TIME-RECORD FOR TIME    2549.7067
 DELETING TIME-RECORD FOR TIME    2564.7067
 DELETING TIME-RECORD FOR TIME    2579.7067
 DELETING TIME-RECORD FOR TIME    2594.7067
 DELETING TIME-RECORD FOR TIME    2609.7067
 DELETING TIME-RECORD FOR TIME    2624.7067
 DELETING TIME-RECORD FOR TIME    2639.7067
 DELETING TIME-RECORD FOR TIME    2654.7067
 DELETING TIME-RECORD FOR TIME    2669.7067
 DELETING TIME-RECORD FOR TIME    2684.7067
 DELETING TIME-RECORD FOR TIME    2699.7067
 DELETING TIME-RECORD FOR TIME    2714.7067
 DELETING TIME-RECORD FOR TIME    2729.7067
 DELETING TIME-RECORD FOR TIME    2744.7067
 DELETING TIME-RECORD FOR TIME    2759.7067
 DELETING TIME-RECORD FOR TIME    2774.7067
 DELETING TIME-RECORD FOR TIME    2789.7067
 DELETING TIME-RECORD FOR TIME    2804.7067
 DELETING TIME-RECORD FOR TIME    2819.7067
 DELETING TIME-RECORD FOR TIME    2834.7067
 DELETING TIME-RECORD FOR TIME    2849.7067
 DELETING TIME-RECORD FOR TIME    2864.7067
 DELETING TIME-RECORD FOR TIME    2879.7067
 DELETING TIME-RECORD FOR TIME    2894.7067
 DELETING TIME-RECORD FOR TIME    2909.7067
 DELETING TIME-RECORD FOR TIME    2924.7067
 DELETING TIME-RECORD FOR TIME    2939.7067
 DELETING TIME-RECORD FOR TIME    2954.7067
 DELETING TIME-RECORD FOR TIME    2969.7067
 DELETING TIME-RECORD FOR TIME    2984.7067

 KEEPING TIME-RECORD FOR TIME    2999.7067
 AND FOR TIME                    3000.0000
 WORKSPACE RECLAIMED
DIC> set-inter
 --OK---
DIC>
```

exb6-plot

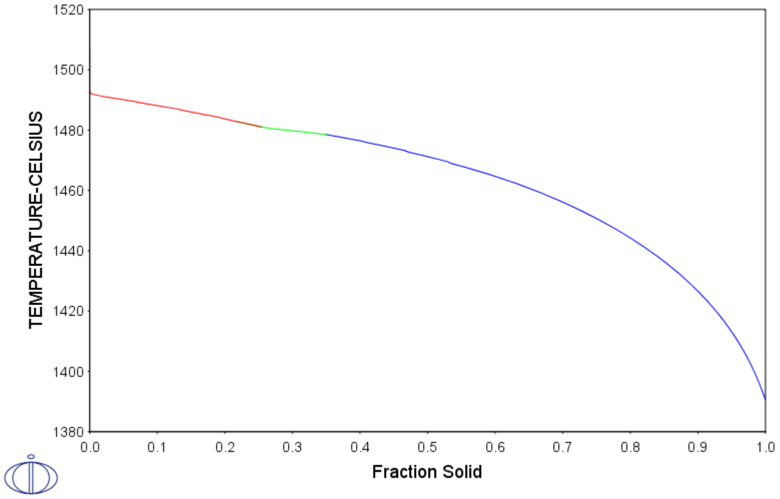
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb6\plot.DCM"
DIC>
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.00000E+03
DIC> read exb6
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: set-title Fe-0.4%C-0.7%Si-0.8%Mn-0.03%P
POST-1:
POST-1: s-d-a y t-c
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-p-c interf first
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ PLOT FRACTION SOLID
POST-1: @@
POST-1: enter func fs=1-ivv(liq);
POST-1: s-d-a x fs
POST-1: s-s-s x n 0 1
POST-1: s-ax-te x n Fraction Solid
POST-1:
POST-1: s-d-a y t-c
POST-1:
POST-1:
POST-1: s-p-c interf smalta lower
POST-1:
POST-1:
POST-1: plot
```

Fe-0.4%C-0.7%Si-0.8%Mn-0.03%P

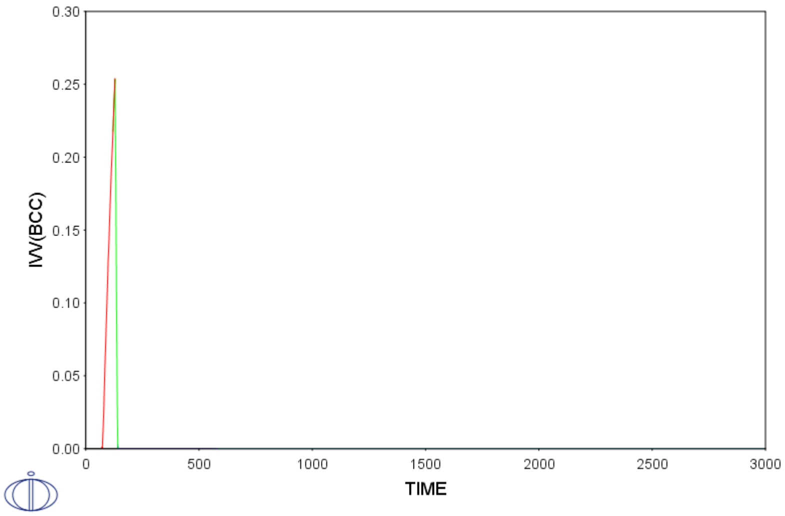
2016.05.17.14.39.55
LOWER INTERFACE OF REGION "SMALTA#1"
CELL #1




```
POST-1:
POST-1:Hit RETURN to continue
POST-1: s-d-a y ivv(bcc)
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1:
POST-1: plot
```

Fe-0.4%C-0.7%Si-0.8%Mn-0.03%P

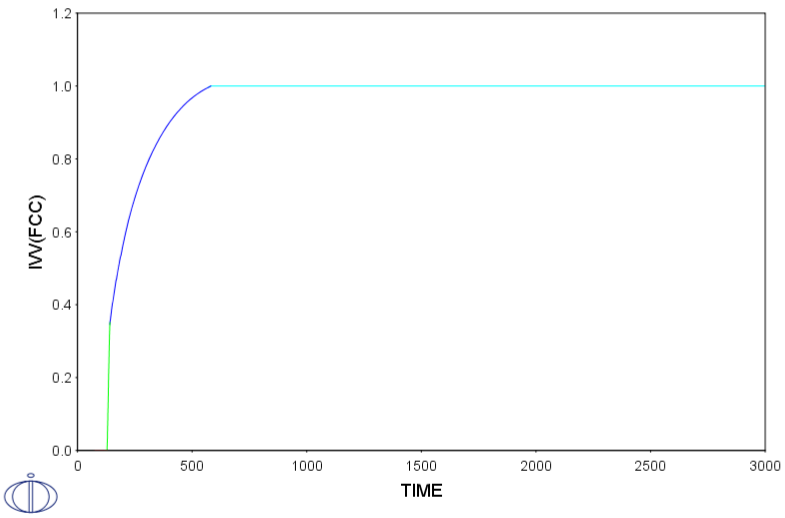
2016.05.17.14.39.58
CELL #1




```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-d-a y ivv(fcc)
POST-1: plot
```

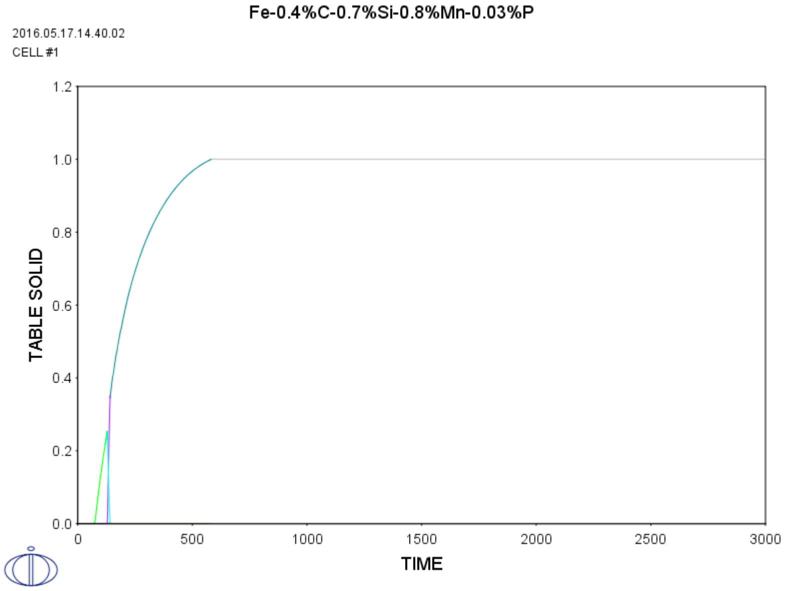
Fe-0.4%C-0.7%Si-0.8%Mn-0.03%P

2016.05.17.14.40.00
CELL #1

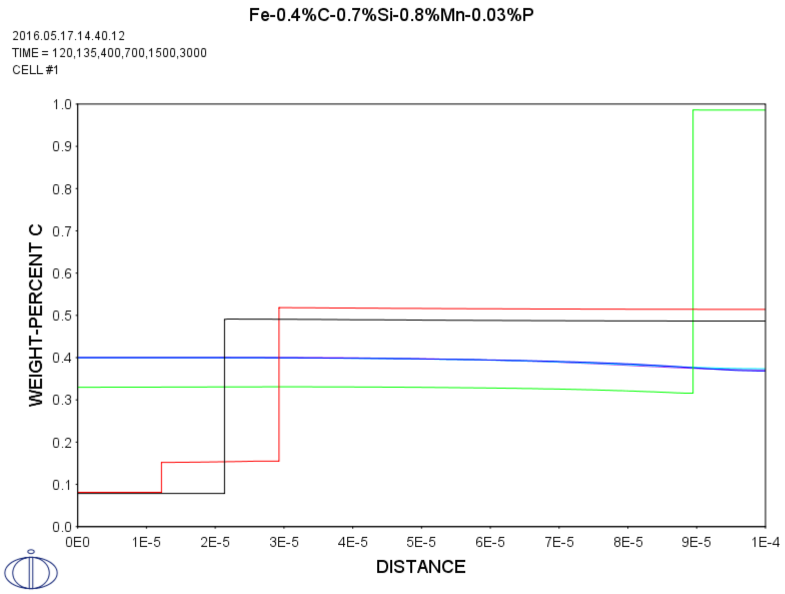



```
POST-1:
POST-1:Hit RETURN to continue
POST-1: ent table solid
```

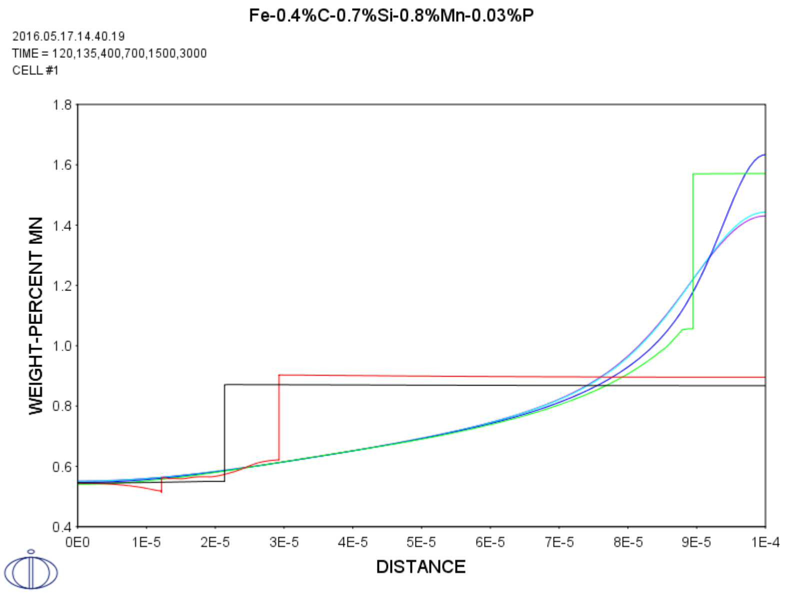
```
Variable(s) ivv(bcc) ivv(fcc)
POST-1:
POST-1: s-d-a y solid
COLUMN NUMBER /*/:
POST-1:
POST-1: plot
```



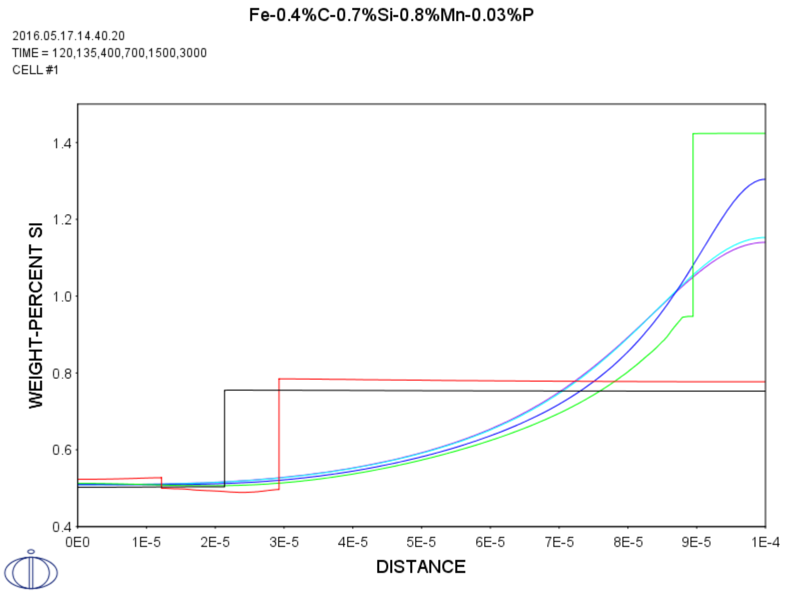

```
POST-1:
POST-1:Hit RETURN to continue
POST-1: s-d-a y w-p c
POST-1: s-d-a x dis gl
  INFO: Distance is set as independent variable
POST-1: s-p-c time 120,135,400,700,1500,3000
POST-1:
POST-1: plot
```



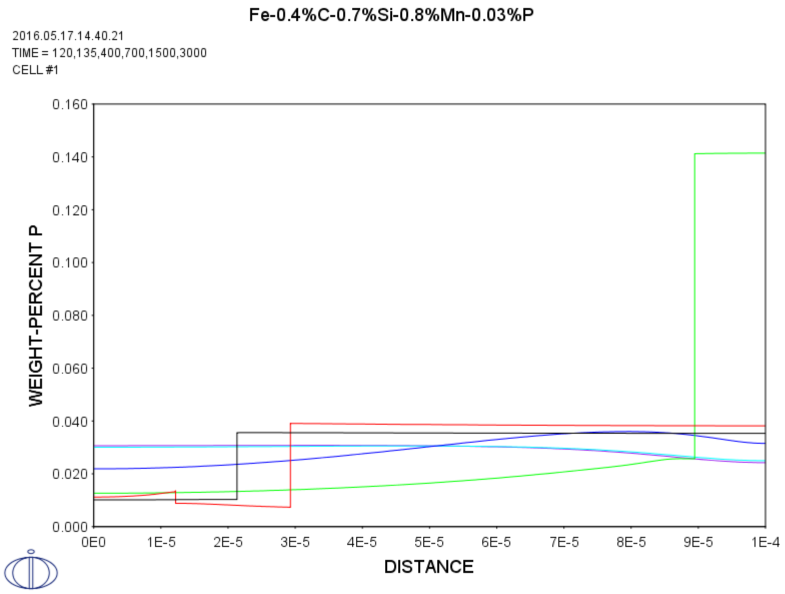

```
POST-1:
POST-1:Hit RETURN to continue
POST-1: s-d-a y w-p mn
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1: s-d-a y w-p si
POST-1: plot
```

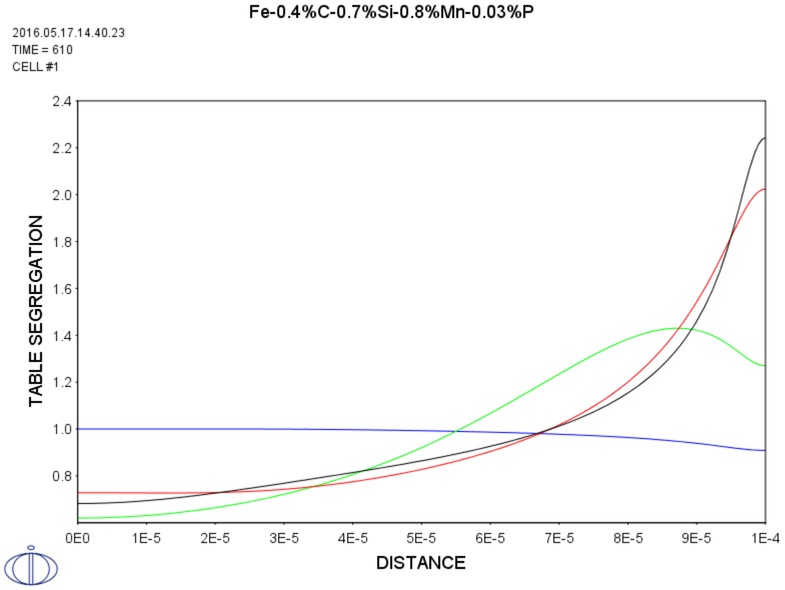



```
POST-1:
POST-1:Hit RETURN to continue
POST-1: s-d-a y w-p p
POST-1: plot
```

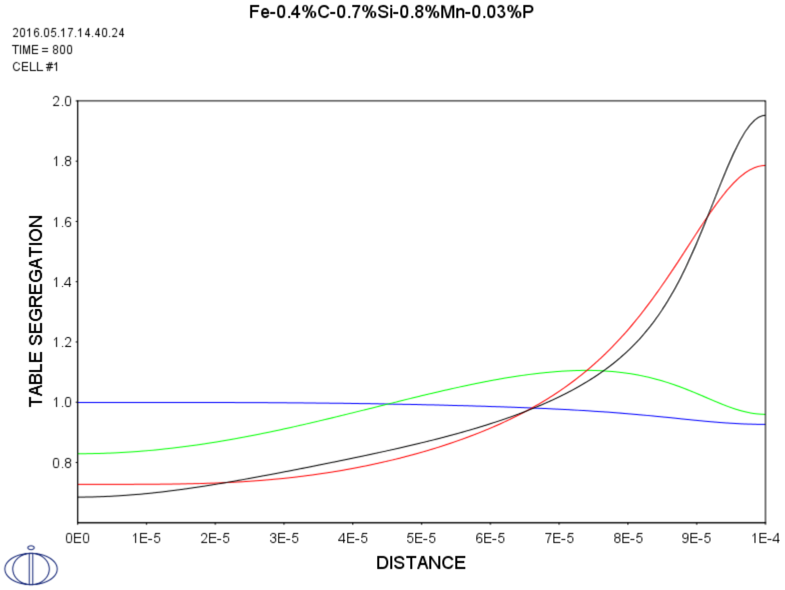



```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: ent function mnn
FUNCTION: w(mn)/0.008
&
POST-1: ent function sin
```

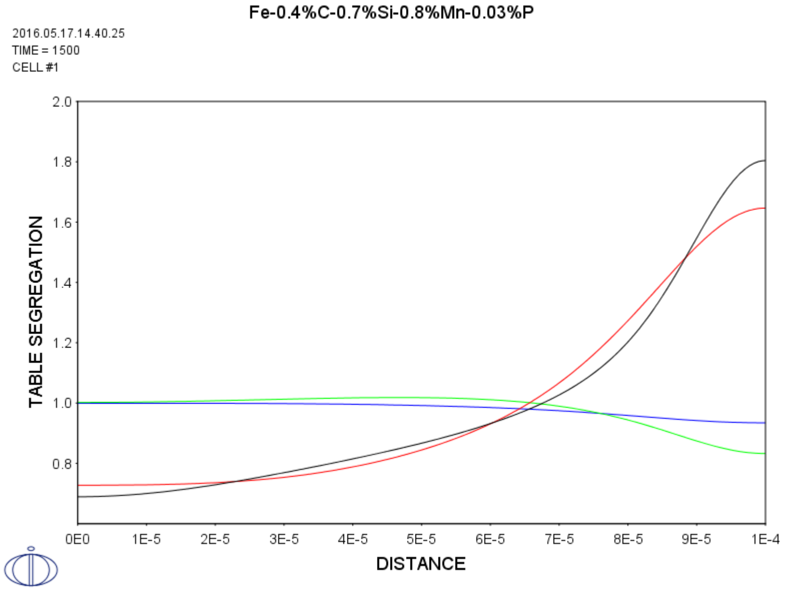
```
FUNCTION: w(si)/0.007
&
POST-1: ent function pn
FUNCTION: w(p)/0.0003
&
POST-1: ent function cn
FUNCTION: w(c)/0.004
&
POST-1: ent tabel segregation
Variable(s) mnn sin pn cn
POST-1:
POST-1:
POST-1: s-d-a y segregation
COLUMN NUMBER /*/:
POST-1:
POST-1:
POST-1: s-p-c time 610
POST-1: plot
```



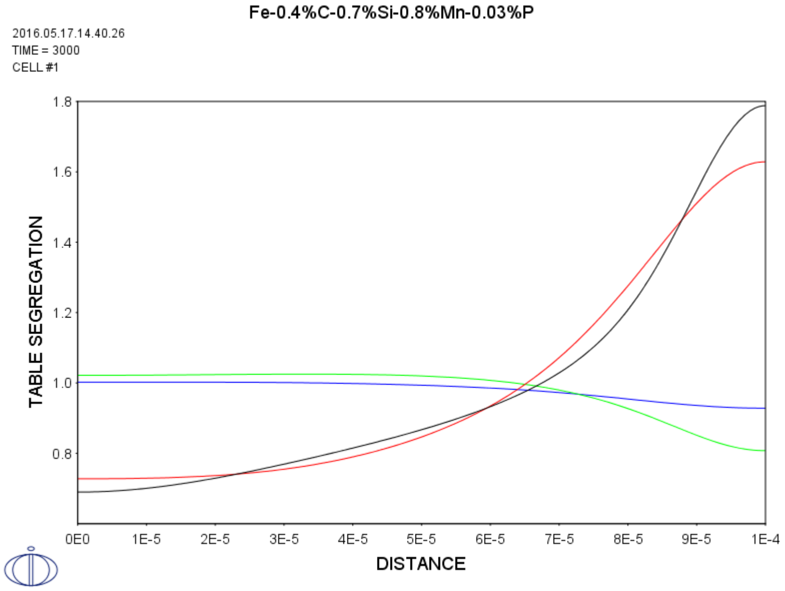

```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 800
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 1500
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: s-p-c time 3000
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1: set-inter
 --OK---
POST-1:
```

exb7-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb7\setup.DCM" @@
SYS: @@ Moving boundary problems.
SYS: @@ This example shows how to enter dispersed phases on either side
SYS: @@ of a phase interface. The particular case shows how
SYS: @@ the kinetics of a ferrite to austenite transformation is
SYS: @@ affected by simultaneous precipitation of niobium carbide.
SYS: @@ The transformation is caused by carburization.
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ LET US USE A TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_TCFE7: @@
TDB_TCFE7: def-species fe c nb
 FE                       C                        NB
   DEFINED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_TCFE7: @@
TDB_TCFE7: rej ph * all
 GAS:G                    LIQUID:L                 BCC_A2
 FCC_A1                   HCP_A3                   DIAMOND_FCC_A4
 GRAPHITE                 CEMENTITE                M23C6
 M7C3                     M6C                      M5C2
 KSI_CARBIDE              A1_KAPPA                 KAPPA
 Z_PHASE                  FE4N_LP1                 FECN_CHI
 SIGMA                    MU_PHASE                 LAVES_PHASE_C14
 G_PHASE                  CR3SI                    NBNI3
   REJECTED
TDB_TCFE7: res ph fcc bcc grap
 FCC_A1                   BCC_A2                   GRAPHITE
   RESTORED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_TCFE7: @@
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo-C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
     -FE'
  'B.-J. Lee, estimated parameter 1999'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
     Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'S. Canderyd, Report IM-2005-109, Stockholm, Sweden; Fe-Nb-C'
  'W. Huang, Zeitschrift fur Metallkunde, Vol. 6 (1990), pp. 397-404; Fe-Nb-C'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
     hard mater, Vol. 12, pp. 121-127; Molar volumes'
  'B.-J. Lee, Metall. Mater. Trans. A, 32A, 2423-39(2001); Fe-Nb'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ NOW APPEND A SSUB DATABASE FROM WHICH WE READ THE THERMODYNAMIC
TDB_TCFE7: @@ DESCRIPTION OF NIOBIUM CARBIDE
TDB_TCFE7: @@
TDB_TCFE7:
TDB_TCFE7: app SSUB5
 Current database: SGTE Substances Database v5.1

 VA  DEFINED
APP: def-sys  fe c nb
 FE                       C                        NB
   DEFINED
APP: rej ph *
 GAS:G                    GRAPHITE                 GRAPHITE_L
 DIAMOND                  C0_749NB1_S              C0_877NB1_S
 C0_98NB1_S               C1FE3_S                  C1NB1_S
 C1NB2_S                  C60_S                    FE_S
 FE_S2                    FE_S3                    FE_L
 FE2NB1_S                 NB_S                     NB_L
   REJECTED
APP: rest ph c1nb1_s
 C1NB1_S  RESTORED
APP: get
```

```
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

   C1NB1 I. BARIN 3rd. Edition
      C1NB1 NbC
      Data taken from BARIN 3rd. Ed. (1995)
 -OK-
APP:
APP:
APP: @@
APP: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
APP: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
APP: @@
APP: app mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe c nb
 FE                     C                       NB
   DEFINED
APP: rej ph * all
 BCC_A2                 CEMENTITE               FCC_A1
 FE4N_LP1               HCP_A3                  LIQUID:L
   REJECTED
APP: res ph fcc bcc
 FCC_A1                 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'J. Geise and Ch. Herzig, Z. Metallkd. 76(1985)622.; Impurity diffusion of
      Nb in fcc Fe.'
  'Assessed from data presented by B. B. Yu and R. F. Davis J. Phys. Chem.
      Solids 40(1979)997.; C Self-Diff in NbC.'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
  'R. F. Peart, Acta Metall. 10(1962)519.; Impurity diffusion of Fe in bcc
      Nb.'
  'Assessed from data presented in Landholt-Bornstein, Vol. 26, ed. H.
      Mehrer, springer (1990); Impurity diff of Nb in bcc Fe.'
  'R. E. Einziger et al., Phys. Rev. B 17(1978)440.; self-diffusion of Nb in
      bcc Nb.'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING C1NB1_S AS A DIFFUSION NONE PHASE
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> set-cond glob T 0 1073.15; * N

DIC>
DIC> @@
DIC> @@ ENTER REGIONS ferr AND aus
DIC> @@
DIC> enter-region
REGION NAME : ferr
DIC>
DIC> ent-reg
REGION NAME : aus
ATTACH TO REGION NAMED /FERR/: ferr
ATTACHED TO THE RIGHT OF FERR /YES/: n
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO REGIONS
DIC> @@
DIC> enter-grid
REGION NAME : /AUS/: ferr
WIDTH OF REGION /1/: 2.499999e-3
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC>
DIC> enter-grid
REGION NAME : /AUS/: aus
WIDTH OF REGION /1/: 1e-9
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 5
DIC>
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUS/: ferr
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: bcc
DIC>
```

```
DIC>
DIC> en-ph
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: ferr
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: c1nb1_s
DIC>
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC>
DIC> en-ph
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: c1nb1_s
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS OF THE PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: ferr
PHASE NAME: /BCC_A2/: bcc
DEPENDENT COMPONENT ? /NB/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 1e-3
VALUE OF LAST POINT : /1E-3/: 1e-3
PROFILE FOR /NB/: nb
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.28
VALUE OF LAST POINT : /0.28/: 0.28
DIC>
DIC> en-co
REGION NAME : /AUS/: ferr
PHASE NAME: /BCC_A2/: c1nb1_s
USE EQUILIBRIUM VALUE /Y/: y
DIC>
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NB/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.89
VALUE OF LAST POINT : /0.89/: 0.89
PROFILE FOR /NB/: nb
TYPE /LINEAR/: lin
VALUE OF FIRST POINT : 0.28
VALUE OF LAST POINT : /0.28/: 0.28
DIC>
DIC> en-co
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: c1nb1_s
USE EQUILIBRIUM VALUE /Y/: y
DIC>
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 32400
AUTOMATIC TIMESTEP CONTROL /YES/: y
MAX TIMESTEP DURING INTEGRATION /3240/: 3240
INITIAL TIMESTEP : /1E-07/: 1e-8
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/: 1e-15
DIC>
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ SET THE REFERENCE PHASE OF CARBON AS GRAPHITE
DIC> @@
DIC> s-ref
Component: c
Reference state: grap
Temperature /*/: *
Pressure /100000/: 1e5
DIC>
DIC> @@
DIC> @@ SET THE BOUNDARY CONDITION
DIC> @@ THE CARBON ACTIVITY IS ONE ON THE BOUNDARY
DIC> @@
DIC> s-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: bound
BOUNDARY /LOWER/: low
CONDITION TYPE /CLOSED_SYSTEM/: mix
Dependent substitutional element:FE
Dependent interstitial element:VA
TYPE OF CONDITION FOR COMPONENT C /ZERO_FLUX/: act
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 1.0;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
TYPE OF CONDITION FOR COMPONENT NB /ZERO_FLUX/: zero
DIC>
DIC> @@
DIC> @@ ENABLE THE HOMOGENIZATION MODEL
DIC> @@
DIC> ho y y
 INFO: HOMOGENIZATION MODEL ENABLED
DIC>
DIC>
```

```
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT DICTRA
DIC> @@
DIC> save exb7 Y
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb7-run

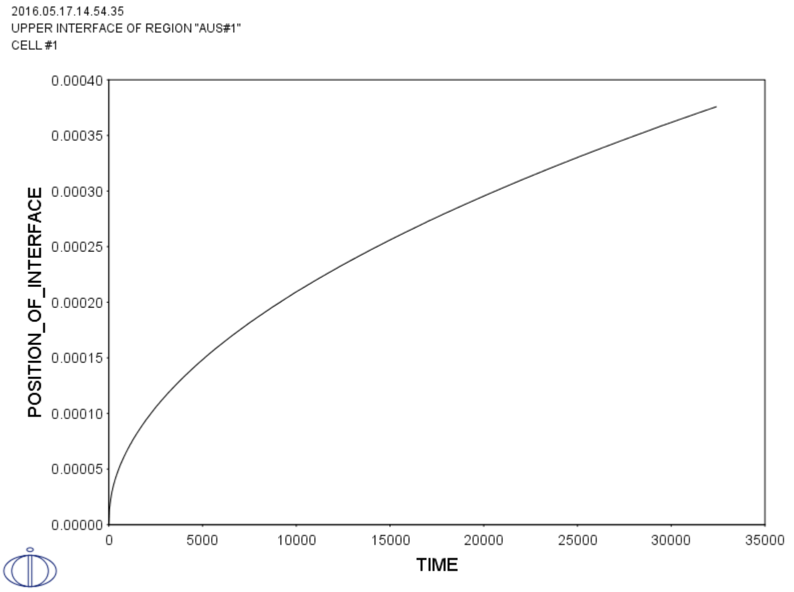
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb7\run.DCM"DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING C1NB1_S AS A DIFFUSION NONE PHASE
DIC> read exb7
 OK
DIC> sim yes
 STARTING SIMULATION USING HOMOGENIZATION MODEL
 ----------------------------------------------
 WARNING: ELEMENT C
 IS BOTH INTERSTITIAL AND SUBSTITUTIONAL
 AND RESULTS MUST BE INTERPRETED WITH CARE
 INFO: PHASE WITH LIMITED SOLUBILITY OF ELEMENT(S) EXIST
 A FALLBACK PHASE ZZDICTRA_GHOST WILL BE DEFINED
 ALONG WITH THE FOLLOWING PARAMETERS:
 G(ZZDICTRA_GHOST,C;0)-H298(GRAPHITE,C;0)
 G(ZZDICTRA_GHOST,FE;0)-H298(BCC_A2,FE;0)
 G(ZZDICTRA_GHOST,NB;0)-H298(BCC_A2,NB;0)
 L(ZZDICTRA_GHOST,C,FE;0)
 L(ZZDICTRA_GHOST,C,NB;0)
 L(ZZDICTRA_GHOST,FE,NB;0)
 Saving results
 ------------------------------------------------
  WARNING:C1NB1_S HAS NO VOLUME FRACTION, CREATING ONE
  WARNING:C1NB1_S HAS NO VOLUME FRACTION, CREATING ONE
 Starting time-step t0=  0.000000000000000E+000  dt=  1.000000000000000E-008
 Stage 1
 Evaluating Jacobian
 Iteration #                     1  sum residual **2:    761484143.600867
 Iteration #                     2  sum residual **2:    272521611.463422
 Error                    408
 Reducing dt to   5.000000000000000E-009
 Stage 1
 Evaluating Jacobian
 Iteration #                     1  sum residual **2:    86854830.3872002
 Iteration #                     2  sum residual **2:    15186837.0125629
 Iteration #                     3  sum residual **2:    2475650.44865618
 Iteration #                     4  sum residual **2:    367734.699643220
 Iteration #                     5  sum residual **2:    50581.0313455482
 Iteration #                     6  sum residual **2:    6556.96729626404
 Iteration #                     7  sum residual **2:    810.833950019362
 Iteration #                     8  sum residual **2:    96.2585474856122
 Iteration #                     9  sum residual **2:    10.6884134877324
 Iteration #                    10  sum residual **2:   0.856089223512080
 Stage 2
 Iteration #                     1  sum residual **2:    48773046.9957205
 Iteration #                     2  sum residual **2:    10783248.0718347
 Iteration #                     3  sum residual **2:    2305081.09591607
 Iteration #                     4  sum residual **2:    466667.450512141
 Iteration #                     5  sum residual **2:    89929.3498828119
 Iteration #                     6  sum residual **2:    16641.6780366640
 Iteration #                     7  sum residual **2:    2978.93480949521
 Iteration #                     8  sum residual **2:    518.427577464409
 Iteration #                     9  sum residual **2:    87.8381290592130
 Iteration #                    10  sum residual **2:    14.2503495043070
 Iteration #                    11  sum residual **2:    1.96914022533882
 Iteration #                    12  sum residual **2:   5.583992191911839E-002
 Stage 3
 Iteration #                     1  sum residual **2:    16731304.2175551
 Iteration #                     2  sum residual **2:    13288152.5208820
 Iteration #                     3  sum residual **2:    3235386.51283236
 Iteration #                     4  sum residual **2:    729331.639059452
 Iteration #                     5  sum residual **2:    154857.572119951
 Iteration #                     6  sum residual **2:    31164.4466297938
 Iteration #                     7  sum residual **2:    5987.76393588770
 Iteration #                     8  sum residual **2:    1104.77306030137
 Iteration #                     9  sum residual **2:    196.244929947586
 Iteration #                    10  sum residual **2:    33.3567652771319
 Iteration #                    11  sum residual **2:    5.15830679686436
 Iteration #                    12  sum residual **2:   0.502534226921674
 Reducing dt to   2.500000000000000E-009
 Stage 2
 Iteration #                     1  sum residual **2:    19124391.1735335
 Iteration #                     2  sum residual **2:    5418894.49468780
 Iteration #                     3  sum residual **2:    1462456.75364628
 Iteration #                     4  sum residual **2:    377052.931823163
 Iteration #                     5  sum residual **2:    93602.7851390764
 Iteration #                     6  sum residual **2:    22537.5494076056
 Iteration #                     7  sum residual **2:    5292.09730315031
 Iteration #                     8  sum residual **2:    1216.63377785119
 Iteration #                     9  sum residual **2:    274.444593354382
 Iteration #                    10  sum residual **2:    60.6514115442645
 Iteration #                    11  sum residual **2:    12.9463195383948
 Iteration #                    12  sum residual **2:    2.44996420228756
 Iteration #                    13  sum residual **2:   0.221490777949307
 Stage 3
 Iteration #                     1  sum residual **2:    11721041.2302707
 Iteration #                     2  sum residual **2:    3342677.72389746
 Iteration #                     3  sum residual **2:    914207.559987675
 Iteration #                     4  sum residual **2:    240832.841336049
 Iteration #                     5  sum residual **2:    61409.8256782736
 Iteration #                     6  sum residual **2:    15238.0397451104
 Iteration #                     7  sum residual **2:    3695.52784862717
 Iteration #                     8  sum residual **2:    878.355965709476
 Iteration #                     9  sum residual **2:    204.869790215520
 Iteration #                    10  sum residual **2:    46.7800782558305
 Iteration #                    11  sum residual **2:    10.2661960931187
 Iteration #                    12  sum residual **2:    1.96507907290198
 Iteration #                    13  sum residual **2:   0.145181242124560
 Reducing dt to   1.250000000000000E-009
 Stage 2
 Iteration #                     1  sum residual **2:    6060946.48730088
 Iteration #                     2  sum residual **2:    2051441.21616307
 Iteration #                     3  sum residual **2:    653666.750955425
 Iteration #                     4  sum residual **2:    198902.543204315
 Iteration #                     5  sum residual **2:    58431.5087447917
 Iteration #                     6  sum residual **2:    16698.6510204835
 Iteration #                     7  sum residual **2:    4667.66016351647
```

```
 Iteration #                      8  sum residual **2:    1280.99548020213
 Iteration #                      9  sum residual **2:    345.743251527409
 Iteration #                     10  sum residual **2:    91.6910846910978
 Iteration #                     11  sum residual **2:    23.8078314256144
 Iteration #                     12  sum residual **2:    5.86482228352070
 Iteration #                     13  sum residual **2:    1.22626136538328
 Stage 3
 Iteration #                      1  sum residual **2:    4487161.26802159
 Iteration #                      2  sum residual **2:    1503990.38762600
 Iteration #                      3  sum residual **2:    478848.625467493
 Iteration #                      4  sum residual **2:    146482.070971023
 Iteration #                      5  sum residual **2:    43427.9192383714
 Iteration #                      6  sum residual **2:    12558.4048158770
 Iteration #                      7  sum residual **2:    3559.00700255928
 Iteration #                      8  sum residual **2:    991.267386148905
 Iteration #                      9  sum residual **2:    271.711725154067
 Iteration #                     10  sum residual **2:    73.2816642442880
 Iteration #                     11  sum residual **2:    19.2985750377137
 Iteration #                     12  sum residual **2:    4.77538245395629
 Iteration #                     13  sum residual **2:   0.979041667562023
 Moles of elements / mole fractions in system:
 C    1.164142616651208E-007  /   4.656353640216400E-005
 FE   2.495787462274513E-003  /   0.998268499833665
 NB   4.212537725487070E-006  /   1.684936629933338E-003

 Time-step converged t0=  0.000000000000000E+000  dt=  1.250000000000000E-009
 ------------------------------------------------


output ignored...



... output resumed

 DELETING TIME-RECORD FOR TIME    26568.213
 DELETING TIME-RECORD FOR TIME    26605.794
 DELETING TIME-RECORD FOR TIME    26643.375
 DELETING TIME-RECORD FOR TIME    26680.956
 DELETING TIME-RECORD FOR TIME    26718.536
 DELETING TIME-RECORD FOR TIME    26772.224
 DELETING TIME-RECORD FOR TIME    26815.173
 DELETING TIME-RECORD FOR TIME    26858.123
 DELETING TIME-RECORD FOR TIME    26901.073
 DELETING TIME-RECORD FOR TIME    26944.022
 DELETING TIME-RECORD FOR TIME    26986.972
 DELETING TIME-RECORD FOR TIME    27029.922
 DELETING TIME-RECORD FOR TIME    27072.871
 DELETING TIME-RECORD FOR TIME    27115.821
 DELETING TIME-RECORD FOR TIME    27158.771
 DELETING TIME-RECORD FOR TIME    27201.720
 DELETING TIME-RECORD FOR TIME    27244.670
 DELETING TIME-RECORD FOR TIME    27282.251
 DELETING TIME-RECORD FOR TIME    27319.832
 DELETING TIME-RECORD FOR TIME    27357.413
 DELETING TIME-RECORD FOR TIME    27394.994
 DELETING TIME-RECORD FOR TIME    27432.575
 DELETING TIME-RECORD FOR TIME    27470.156
 DELETING TIME-RECORD FOR TIME    27507.737
 DELETING TIME-RECORD FOR TIME    27545.318
 DELETING TIME-RECORD FOR TIME    27582.899
 DELETING TIME-RECORD FOR TIME    27620.480
 DELETING TIME-RECORD FOR TIME    27658.061
 DELETING TIME-RECORD FOR TIME    27695.642
 DELETING TIME-RECORD FOR TIME    27733.223
 DELETING TIME-RECORD FOR TIME    27770.803
 DELETING TIME-RECORD FOR TIME    27808.384
 DELETING TIME-RECORD FOR TIME    27845.965
 DELETING TIME-RECORD FOR TIME    27883.546
 DELETING TIME-RECORD FOR TIME    27921.127
 DELETING TIME-RECORD FOR TIME    27958.708
 DELETING TIME-RECORD FOR TIME    27996.289
 DELETING TIME-RECORD FOR TIME    28033.870
 DELETING TIME-RECORD FOR TIME    28071.451
 DELETING TIME-RECORD FOR TIME    28109.032
 DELETING TIME-RECORD FOR TIME    28146.613
 DELETING TIME-RECORD FOR TIME    28184.194
 DELETING TIME-RECORD FOR TIME    28219.091
 DELETING TIME-RECORD FOR TIME    28253.987
 DELETING TIME-RECORD FOR TIME    28296.937
 DELETING TIME-RECORD FOR TIME    28339.887
 DELETING TIME-RECORD FOR TIME    28425.786
 DELETING TIME-RECORD FOR TIME    28597.585
 DELETING TIME-RECORD FOR TIME    28769.383
 DELETING TIME-RECORD FOR TIME    28941.182
 DELETING TIME-RECORD FOR TIME    29112.981
 DELETING TIME-RECORD FOR TIME    29155.930
 DELETING TIME-RECORD FOR TIME    29198.880
 DELETING TIME-RECORD FOR TIME    29241.830
 DELETING TIME-RECORD FOR TIME    29284.779
 DELETING TIME-RECORD FOR TIME    29327.729
 DELETING TIME-RECORD FOR TIME    29370.679
 DELETING TIME-RECORD FOR TIME    29413.628
 DELETING TIME-RECORD FOR TIME    29456.578
 DELETING TIME-RECORD FOR TIME    29499.528
 DELETING TIME-RECORD FOR TIME    29542.477
 DELETING TIME-RECORD FOR TIME    29585.427
 DELETING TIME-RECORD FOR TIME    29628.377
 DELETING TIME-RECORD FOR TIME    29671.327
 DELETING TIME-RECORD FOR TIME    29714.276
 DELETING TIME-RECORD FOR TIME    29757.226
 DELETING TIME-RECORD FOR TIME    29800.176
 DELETING TIME-RECORD FOR TIME    29843.125
 DELETING TIME-RECORD FOR TIME    29886.075
 DELETING TIME-RECORD FOR TIME    29929.025
 DELETING TIME-RECORD FOR TIME    29971.974
 DELETING TIME-RECORD FOR TIME    30014.924
 DELETING TIME-RECORD FOR TIME    30057.874
 DELETING TIME-RECORD FOR TIME    30100.823
 DELETING TIME-RECORD FOR TIME    30143.773
 DELETING TIME-RECORD FOR TIME    30186.723
 DELETING TIME-RECORD FOR TIME    30229.672
 DELETING TIME-RECORD FOR TIME    30272.622
 DELETING TIME-RECORD FOR TIME    30315.572
 DELETING TIME-RECORD FOR TIME    30358.521
 DELETING TIME-RECORD FOR TIME    30401.471
```

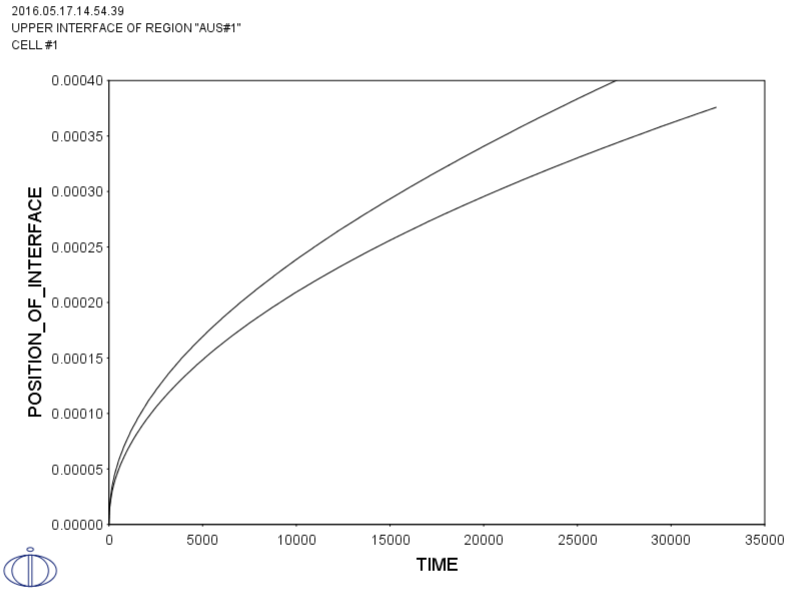
```
 DELETING TIME-RECORD FOR TIME    30439.052
 DELETING TIME-RECORD FOR TIME    30476.633
 DELETING TIME-RECORD FOR TIME    30514.214
 DELETING TIME-RECORD FOR TIME    30551.795
 DELETING TIME-RECORD FOR TIME    30589.376
 DELETING TIME-RECORD FOR TIME    30626.957
 DELETING TIME-RECORD FOR TIME    30664.538
 DELETING TIME-RECORD FOR TIME    30702.119
 DELETING TIME-RECORD FOR TIME    30739.700
 DELETING TIME-RECORD FOR TIME    30777.281
 DELETING TIME-RECORD FOR TIME    30814.862
 DELETING TIME-RECORD FOR TIME    30852.443
 DELETING TIME-RECORD FOR TIME    30890.023
 DELETING TIME-RECORD FOR TIME    30965.185
 DELETING TIME-RECORD FOR TIME    31008.135
 DELETING TIME-RECORD FOR TIME    31051.085
 DELETING TIME-RECORD FOR TIME    31094.034
 DELETING TIME-RECORD FOR TIME    31136.984
 DELETING TIME-RECORD FOR TIME    31179.934
 DELETING TIME-RECORD FOR TIME    31222.883
 DELETING TIME-RECORD FOR TIME    31265.833
 DELETING TIME-RECORD FOR TIME    31308.783
 DELETING TIME-RECORD FOR TIME    31351.732
 DELETING TIME-RECORD FOR TIME    31389.313
 DELETING TIME-RECORD FOR TIME    31426.894
 DELETING TIME-RECORD FOR TIME    31464.475
 DELETING TIME-RECORD FOR TIME    31502.056
 DELETING TIME-RECORD FOR TIME    31539.637
 DELETING TIME-RECORD FOR TIME    31577.218
 DELETING TIME-RECORD FOR TIME    31614.799
 DELETING TIME-RECORD FOR TIME    31652.380
 DELETING TIME-RECORD FOR TIME    31689.961
 DELETING TIME-RECORD FOR TIME    31727.542
 DELETING TIME-RECORD FOR TIME    31765.123
 DELETING TIME-RECORD FOR TIME    31802.704
 DELETING TIME-RECORD FOR TIME    31840.285
 DELETING TIME-RECORD FOR TIME    31877.866
 DELETING TIME-RECORD FOR TIME    31915.447
 DELETING TIME-RECORD FOR TIME    31953.028
 DELETING TIME-RECORD FOR TIME    31990.609
 DELETING TIME-RECORD FOR TIME    32028.190
 DELETING TIME-RECORD FOR TIME    32065.771
 DELETING TIME-RECORD FOR TIME    32103.352
 DELETING TIME-RECORD FOR TIME    32140.933
 DELETING TIME-RECORD FOR TIME    32178.514
 DELETING TIME-RECORD FOR TIME    32216.095
 DELETING TIME-RECORD FOR TIME    32253.676
 DELETING TIME-RECORD FOR TIME    32291.257
 DELETING TIME-RECORD FOR TIME    32328.838

 KEEPING TIME-RECORD FOR TIME    32366.418
 AND FOR TIME                    32400.000
 WORKSPACE RECLAIMED
 ------------------------------------------------
 INTERPOLATION SCHEME USED THIS FRACTION OF
 THE ALLOCATED MEMORY:   0.571338814013787
 EFFICIENCY FACTOR:    43.1739877155594
 MEMORY FRACTION USAGE PER BRANCH:
  0.670439308197541
  0.301787928482861
  0.625394877389150
 ------------------------------------------------
 DEALLOCATING
 ------------------------------------------------
DIC>
DIC> set-inter
 --OK---
DIC>
```

exb7-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exb7\plot.DCM"DIC>
DIC>
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.24000E+04
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING C1NB1_S AS A DIFFUSION NONE PHASE
DIC> read exb7
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ PLOT THE INTERFACE POSITION AS A FUNCTION OF TIME
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y po-o-in aus upp
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ APPEND DATA FROM A CORRESPONDING SIMULATION
POST-1: @@ WITHOUT NIOBIUM
POST-1: @@
POST-1:
POST-1: app y fec.exp 0
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: plot
```

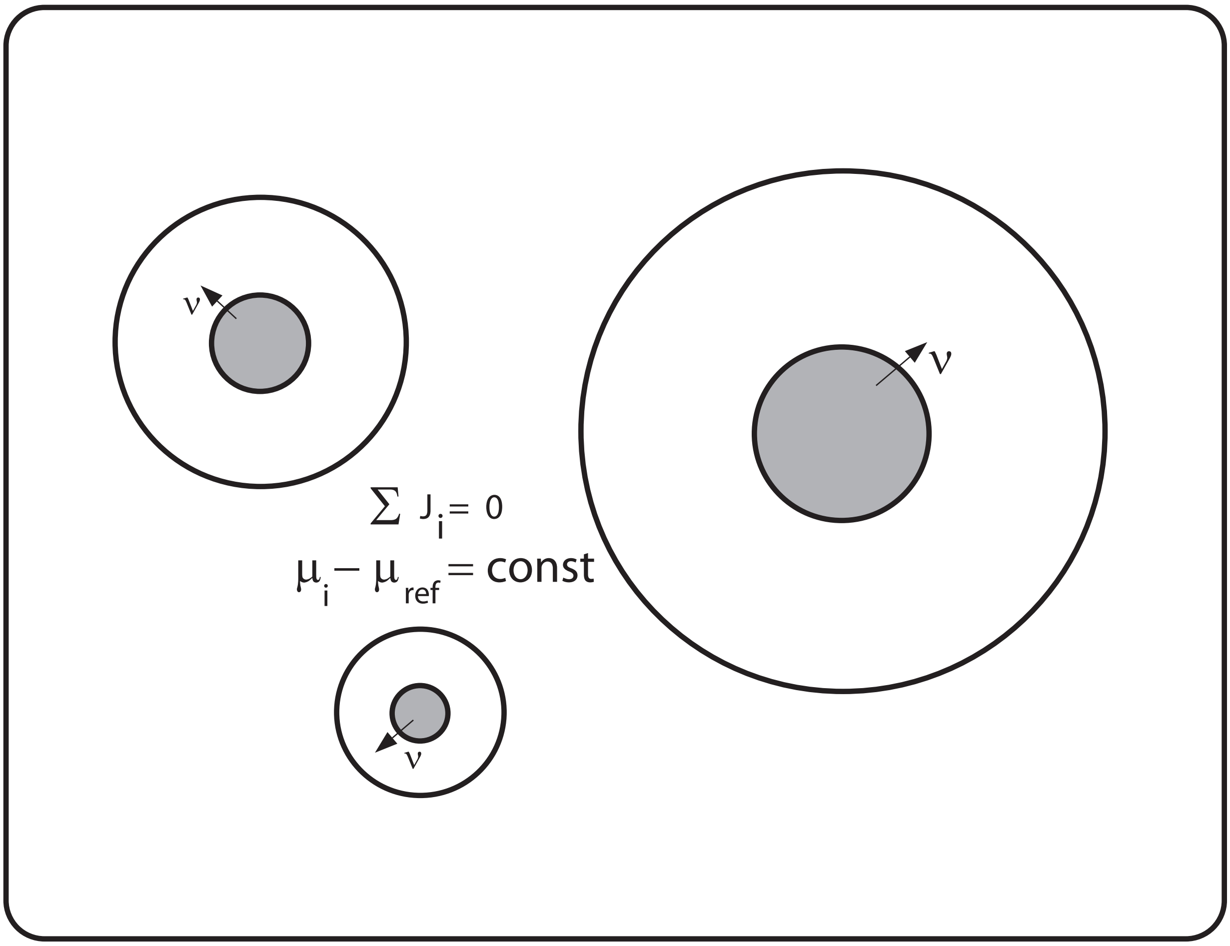



```
POST-1:
POST-1: set-inter
```

--OK---

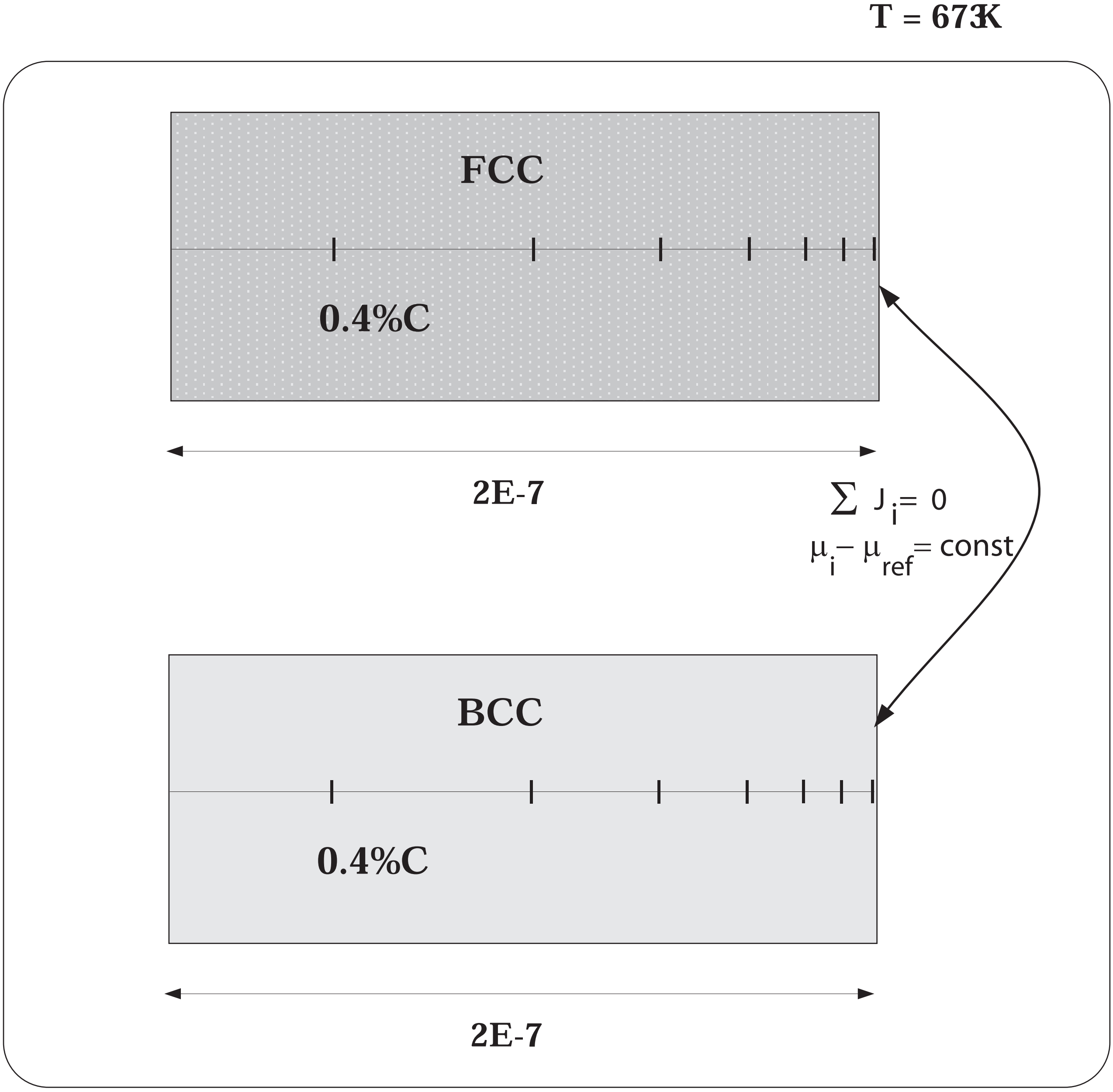
POST-1:

Cell calculations



Example c1

'Carbon cannon' in α/γ Fe-C system, two-cell calculation



exc1-setup

About

```
 Thermo-Calc / DICTRA is software package for calculation of phase diagrams,
 simulation of phase transformation kinetics and much more.

 Copyright Foundation for Computational Thermodynamics,
 Stockholm, Sweden

 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc1\setup.DCM"SYS: ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Cell calculations.
SYS: @@ This example simulates what happens to a FERRITE plate that has
SYS: @@ inherited the carbon content of its parent AUSTENITE. The FERRITE
SYS: @@ plate formed is embedded in an AUSTENITE matrix. This setup
SYS: @@ corresponds to a proposed mechanism for formation of WIDMANNSTÄTTEN
SYS: @@ FERRITE or for the FERRITE phase of the BAINITE structure. It is
SYS: @@ assumed that the phase boundary between FERRITE and AUSTENITE is
SYS: @@ immobile, this is achieved in the simulation by putting the FERRITE
SYS: @@ and the AUSTENITE in two different cells. See also M. Hillert,
SYS: @@ L. Höglund and J. Ågren: Acta Metall. Mater. 41 (1993), pp.1951-1957.
SYS: @@
SYS:
SYS: @@ exc1_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                     /-  DEFINED
TDB_FEDEMO: def-sys fe c
 FE                     C  DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                  LIQUID:L               BCC_A2
 CEMENTITE              DIAMOND_FCC_A4         FCC_A1
 GRAPHITE               HCP_A3                 KSI_CARBIDE
 LAVES_PHASE_C14        M23C6                  M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph fcc,bcc
 FCC_A1                 BCC_A2  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe c
 FE                     C  DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph fcc,bcc
 FCC_A1                 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
```

```
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 673; * N

DIC>
DIC> @@
DIC> @@ IN THE FIRST CELL
DIC> @@
DIC> @@ ENTER REGION aus CONTAINING AUSTENITE
DIC> @@ ENTER A GEOMETRICAL GRID INTO THAT REGION
DIC> @@ ENTER THE INITIAL COMPOSITION INTO THE AUSTENITE
DIC> @@
DIC> enter-region aus
DIC> enter-grid aus  0.2e-6 geo 50 0.9
DIC> enter-phase act aus  matrix fcc_a1#1
DIC>
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc_a1#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c
TYPE /LINEAR/: lin 0.4 0.4
DIC>
DIC> @@
DIC> @@ IN THE SECOND CELL
DIC> @@
DIC> create-new-cell
CELL DISTRIBUTION FACTOR /1/: 1
 CREATING NEW CELL, NUMBER:  2
 CELL   2 SELECTED
DIC-2>
DIC-2> @@
DIC-2> @@ ENTER REGION fer CONTAINING FERRITE
DIC-2> @@ ENTER A GEOMETRICAL GRID INTO THAT REGION
DIC-2> @@ ENTER THE INITIAL COMPOSITION INTO THE FERRITE
DIC-2> @@
DIC-2> enter-region fer
DIC-2>
DIC-2>
DIC-2>
DIC-2> enter-grid fer  0.2e-6 geo 50 0.9
DIC-2> enter-phase act fer  matrix bcc_a2#1
DIC-2>
DIC-2> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /BCC_A2/: bcc_a2#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c
TYPE /LINEAR/: lin 0.4 0.4
DIC-2>
DIC-2> @@
DIC-2> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC-2> @@
DIC-2> set-simulation-time
END TIME FOR INTEGRATION /.1/: 0.5
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /.05/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC-2>
DIC-2>
DIC-2>
DIC-2> @@
DIC-2> @@ USE IMPLICIT ( 1 ) TIME INTEGRATION
DIC-2> @@
DIC-2> set-simulation-cond
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /POTENTIAL/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/:
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1.0
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC-2> @@
DIC-2> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC-2> @@
DIC-2> save exc1 Y
DIC-2>
DIC-2> set-inter
 --OK---
DIC-2>
```

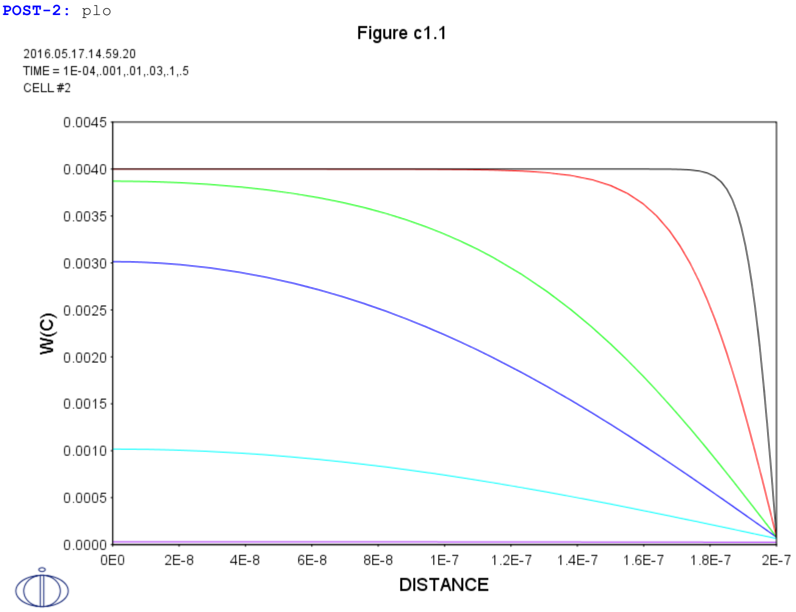
exc1-run

```
DIC-2>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc1\run.DCM"
DIC-2>
DIC-2> @@ exc1_run.DCM
DIC-2>
DIC-2> @@
DIC-2> @@ READ THE WORKSPACE AND START THE SIMULATION
DIC-2> @@
DIC-2> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC-2> read exc1
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .018673311178274  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]
 U-FRACTION IN SYSTEM:  C = .018673311178274  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]
  8  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUS
  8  GRIDPOINT(S) ADDED   TO   CELL #2   REGION: FER
  2.30679910526803        2.30682857573068        0.185365720751719        1.844616299439429E-002        1.799561461653859E-
004        1.820934192432645E-007        1.795395315298027E-012        1.794304743932742E-020
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.17943047E-19
 U-FRACTION IN SYSTEM:  C = .0186733111782747  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUS
  2  GRIDPOINT(S) ADDED   TO   CELL #2   REGION: FER
 0.696006682779119        0.696001554784437        6.364814783458658E-002        3.888451360235548E-003        1.834884547296997E-
005        5.776878217929879E-009        8.413943290240480E-015        3.454963151337941E-024
 TIME =  0.20000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.34549632E-23
 U-FRACTION IN SYSTEM:  C = .0186739419503572  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 7.898873853708528E-002        7.898665850846875E-002        2.385150335699038E-003        4.961482487111795E-005        2.363776520440259E-
008        2.458496928987162E-013        1.211943893529034E-021
 TIME =  0.40000000E-06 DT =  0.20000000E-06 SUM OF SQUARES =  0.12119439E-20
 U-FRACTION IN SYSTEM:  C = .0186739419503568  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    1  seconds
 1.535244980079278E-003        1.534996293935825E-003        1.300174937086121E-006        1.022668516664565E-009        6.348395939071756E-
016        2.828648081302397E-025
 TIME =  0.80000000E-06 DT =  0.40000000E-06 SUM OF SQUARES =  0.28286481E-24
 U-FRACTION IN SYSTEM:  C = .0186739419503568  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 2.302581892893035E-003        2.302404557382640E-003        8.455940911146676E-006        2.630278736409703E-008        2.576054822941231E-
013        7.955607505993671E-021
 TIME =  0.16000000E-05 DT =  0.80000000E-06 SUM OF SQUARES =  0.79556075E-20
 U-FRACTION IN SYSTEM:  C = .0186739419503604  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 2.749108558999770E-003        2.748983979423816E-003        3.647493011508664E-005        3.584870312934646E-007        3.689110696730408E-
011        4.282967352943916E-017
 TIME =  0.32000000E-05 DT =  0.16000000E-05 SUM OF SQUARES =  0.42829674E-16
 U-FRACTION IN SYSTEM:  C = .0186739419508839  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    1  seconds
 6.428059480216762E-002        6.427593640463225E-002        3.233542624058647E-004        1.343716287107163E-006        2.350089326068363E-
011        1.747798784006240E-018
 TIME =  0.64000000E-05 DT =  0.32000000E-05 SUM OF SQUARES =  0.17477988E-17
 U-FRACTION IN SYSTEM:  C = .0186739419509051  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 3.028153956200530E-002

output ignored...



... output resumed

 1.160395293904018E-009        3.497897785567765E-014        1.362631390335181E-023
 TIME =  0.22713049     DT =  0.28244549E-01 SUM OF SQUARES =  0.13626314E-22
 U-FRACTION IN SYSTEM:  C = .0186739419454411  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 9.557109657128673E-005        9.563628884839968E-005        3.054020187525157E-009        1.411119494399848E-013        1.740103577238751E-
022
 TIME =  0.25944000     DT =  0.32309516E-01 SUM OF SQUARES =  0.17401036E-21
 U-FRACTION IN SYSTEM:  C = .0186739419454624  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    1  seconds
 1.377536754045833E-004        1.378172454906032E-004        9.992552750892514E-009        8.649032415481732E-013        4.869648073270869E-
021
 TIME =  0.29843338     DT =  0.38993380E-01 SUM OF SQUARES =  0.48696481E-20
 U-FRACTION IN SYSTEM:  C = .0186739419455983  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                    0  seconds
 1.894884677150934E-004        1.895470655511840E-004        2.886418118663366E-008        5.311056993941265E-012        1.379195886557497E-
019
 TIME =  0.34843338     DT =  0.50000000E-01 SUM OF SQUARES =  0.13791959E-18
```

```
 U-FRACTION IN SYSTEM:  C = .0186739419465268  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                     1  seconds
 2.028183858863455E-005       2.029866073663051E-005       3.441277484583118E-010       9.677779311882027E-015       3.472678190936507E-
024
 TIME =  0.39843338     DT =  0.50000000E-01 SUM OF SQUARES =  0.34726782E-23
 U-FRACTION IN SYSTEM:  C = .0186739419465316  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                     1  seconds
 1.650236172353750E-006       1.645927405801784E-006       6.714047096669519E-012       4.818074921825390E-018
 TIME =  0.44843338     DT =  0.50000000E-01 SUM OF SQUARES =  0.48180749E-17
 U-FRACTION IN SYSTEM:  C = .0186739419410439  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                     0  seconds
 2.592437877522293E-005       2.590874907582452E-005       1.137861387582137E-009       3.372822838955120E-014       4.999424675279440E-
023
 TIME =  0.49843338     DT =  0.50000000E-01 SUM OF SQUARES =  0.49994247E-22
 U-FRACTION IN SYSTEM:  C = .0186739419410263  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

  CPU time used in timestep                     1  seconds
 2.675342272651581E-002       2.675083426091587E-002       3.917016022884895E-005       5.095592953091640E-008       2.577290229118809E-
013        1.922429928935196E-022
 TIME =  0.50000000     DT =  0.15666161E-02 SUM OF SQUARES =  0.19224299E-21
 U-FRACTION IN SYSTEM:  C = .0186739419410274  FE = 1
 TOTAL SIZE OF SYSTEM:  4E-07 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.20000000E-06
 DELETING TIME-RECORD FOR TIME   0.40000000E-06
 DELETING TIME-RECORD FOR TIME   0.80000000E-06
 DELETING TIME-RECORD FOR TIME   0.16000000E-05
 DELETING TIME-RECORD FOR TIME   0.32000000E-05
 DELETING TIME-RECORD FOR TIME   0.64000000E-05
 DELETING TIME-RECORD FOR TIME   0.12800000E-04
 DELETING TIME-RECORD FOR TIME   0.25600000E-04
 DELETING TIME-RECORD FOR TIME   0.51200000E-04
 DELETING TIME-RECORD FOR TIME   0.10240000E-03
 DELETING TIME-RECORD FOR TIME   0.20480000E-03
 DELETING TIME-RECORD FOR TIME   0.40960000E-03
 DELETING TIME-RECORD FOR TIME   0.81920000E-03
 DELETING TIME-RECORD FOR TIME   0.16384000E-02
 DELETING TIME-RECORD FOR TIME   0.32768000E-02
 DELETING TIME-RECORD FOR TIME   0.65536000E-02
 DELETING TIME-RECORD FOR TIME   0.13107200E-01
 DELETING TIME-RECORD FOR TIME   0.26214400E-01
 DELETING TIME-RECORD FOR TIME   0.45260211E-01
 DELETING TIME-RECORD FOR TIME   0.64861961E-01
 DELETING TIME-RECORD FOR TIME   0.84941825E-01
 DELETING TIME-RECORD FOR TIME   0.10563413
 DELETING TIME-RECORD FOR TIME   0.12709187
 DELETING TIME-RECORD FOR TIME   0.14953541
 DELETING TIME-RECORD FOR TIME   0.17329599
 DELETING TIME-RECORD FOR TIME   0.19888594
 DELETING TIME-RECORD FOR TIME   0.22713049
 DELETING TIME-RECORD FOR TIME   0.25944000
 DELETING TIME-RECORD FOR TIME   0.29843338
 DELETING TIME-RECORD FOR TIME   0.34843338
 DELETING TIME-RECORD FOR TIME   0.39843338
 DELETING TIME-RECORD FOR TIME   0.44843338

 KEEPING TIME-RECORD FOR TIME   0.49843338
 AND FOR TIME                   0.50000000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

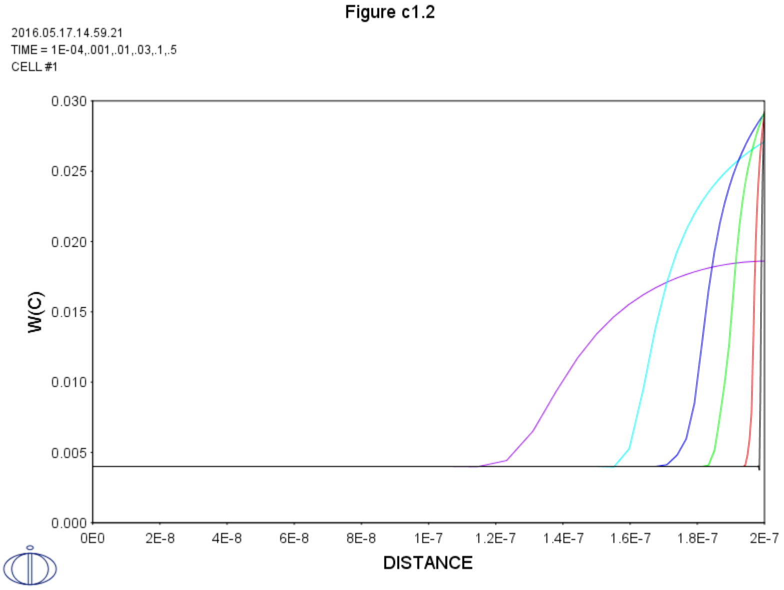
exc1-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc1\plot.DCM"DIC>
DIC>
DIC> @@ exc1_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE c1
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  5.00000E-01
DIC> read exc1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US PLOT CARBON CONCENTRATION PROFILES IN FERRITE (CELL-2)
POST-1: @@ WE THEN SET DISTANCE AS X-AXIS (NOT THAT DISTANCE IS SET INDEPENDENT
POST-1: @@ VARIABLE AUTOMATICALLY) AND W-FRACTION CARBON AS Y-AXIS
POST-1: @@ REMEMBER THAT ONE ALSO HAS TO SET PLOT CONDITION
POST-1: @@
POST-1: select-cell
Number /NEXT/: 2
 CELL   2 SELECTED
POST-2:
POST-2: @@
POST-2: @@ NOTICE THAT THE PROMPT INCLUDES THE CURRENT CELL NUMBER
POST-2: @@
POST-2: s-d-a x dist glo
  INFO: Distance is set as independent variable
POST-2: s-d-a y w(c)
POST-2: s-p-c time .0001 .001 .01 .03 .1 .5
POST-2:
POST-2: @@
POST-2: @@ SET TITLE ON DIAGRAMS
POST-2: @@
POST-2: set-title Figure c1.1
POST-2: plo
```



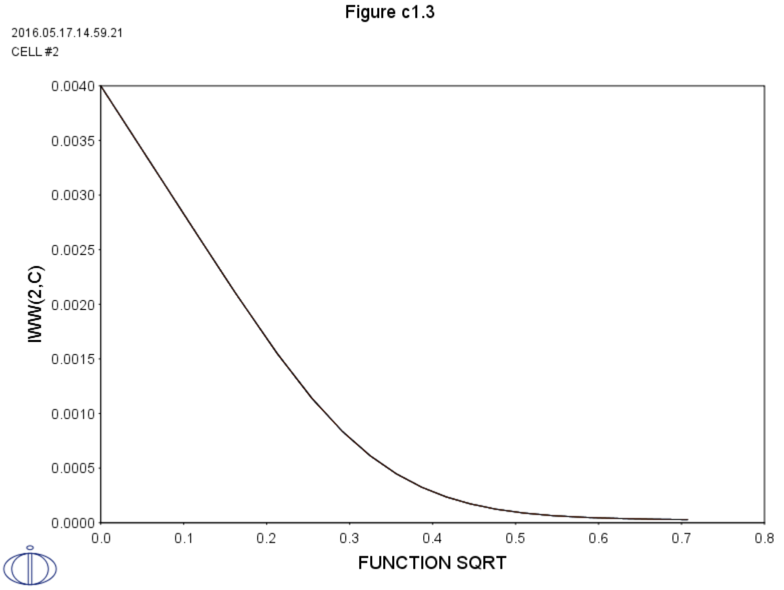

```
POST-2:
POST-2:
POST-2:
POST-2:@?<_hit_return_to_continue_>
POST-2:
POST-2: @@
POST-2: @@ DO THE SAME THING FOR THE AUSTENITE (CELL-1)
POST-2: @@
POST-2: select-cell
Number /NEXT/: 1
 CELL   1 SELECTED
POST-1: set-title Figure c1.2
POST-1: plo
```

Figure c1.2



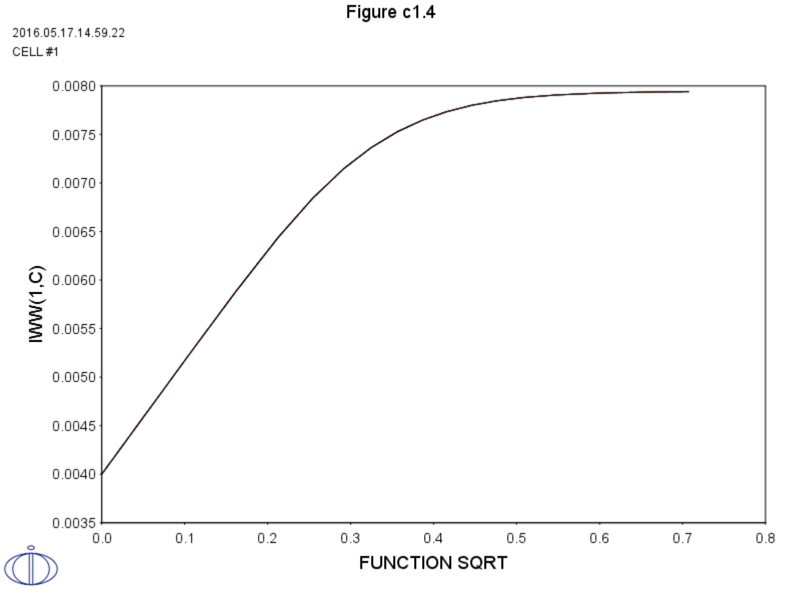

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ PLOT THE AVARAGE WEIGHT FRACTION OF CARBON IN FERRITE VS. SQUARE ROOT
POST-1: @@ OF TIME. START BY DEFINING A "SQUARE-ROOT-OF-TIME" FUNCTION.
POST-1: @@
POST-1: sel-cell 2
 CELL   2 SELECTED
POST-2: enter func sqrt=sqrt(time);
POST-2: s-d-a x sqrt
POST-2: s-d-a y iww(2,c)
POST-2: s-i-v time
POST-2: set-title Figure c1.3
POST-2: plo
```

Figure c1.3



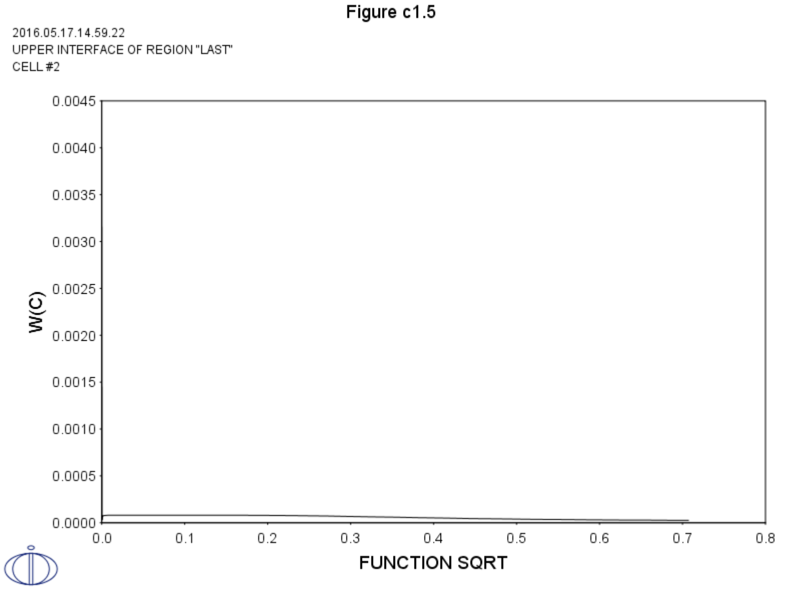

```
POST-2:
POST-2:
POST-2:
POST-2:@?<_hit_return_to_continue_>
POST-2:
POST-2: @@
POST-2: @@ DO THE SAME THING FOR THE AUSTENITE
POST-2: @@
POST-2: sel-cell 1
 CELL   1 SELECTED
POST-1: s-d-a y iww(1,c)
POST-1: set-title Figure c1.4
POST-1: plo
```

Figure c1.4



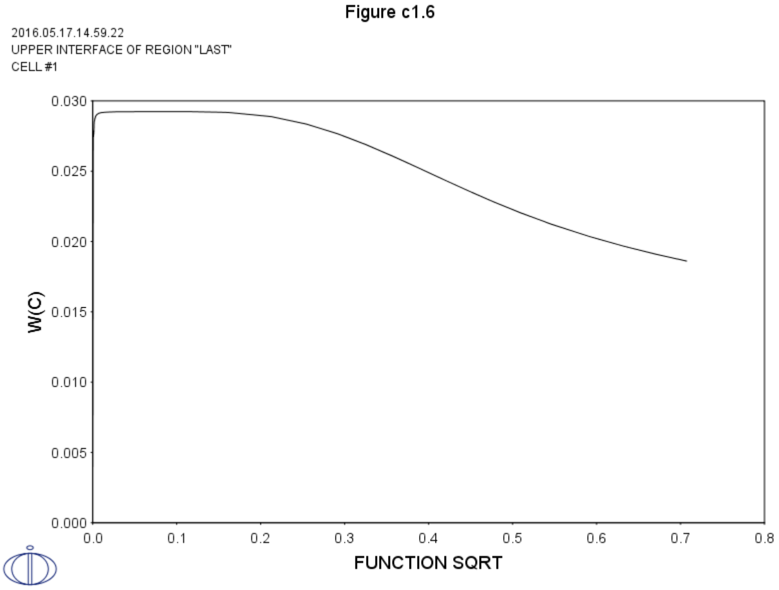

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ PLOT HOW THE CONCENTRATION IN FERRITE AT THE FERRITE/AUSTENITE BOUNDARY
POST-1: @@ V.S SQUARE ROOT OF TIME. THE  FERRITE/AUSTENITE BOUNDARY IS REPRESENTED
POST-1: @@ BY THE CELL BOUNDARY I.E. THE "LAST" INTERFACE.
POST-1: @@
POST-1: sel-cell 2
 CELL   2 SELECTED
POST-2: s-d-a y w(c)
POST-2: s-p-c interface last
POST-2: set-title Figure c1.5
POST-2: plo
```

Figure c1.5




```
POST-2:
POST-2:
POST-2:
POST-2:@?<_hit_return_to_continue_>
POST-2:
POST-2: @@
POST-2: @@ DO THE SAME THING FOR THE AUSTENITE
POST-2: @@
POST-2: sel-cell 1
 CELL   1 SELECTED
POST-1: set-title Figure c1.6
POST-1: plo
```

Figure c1.6



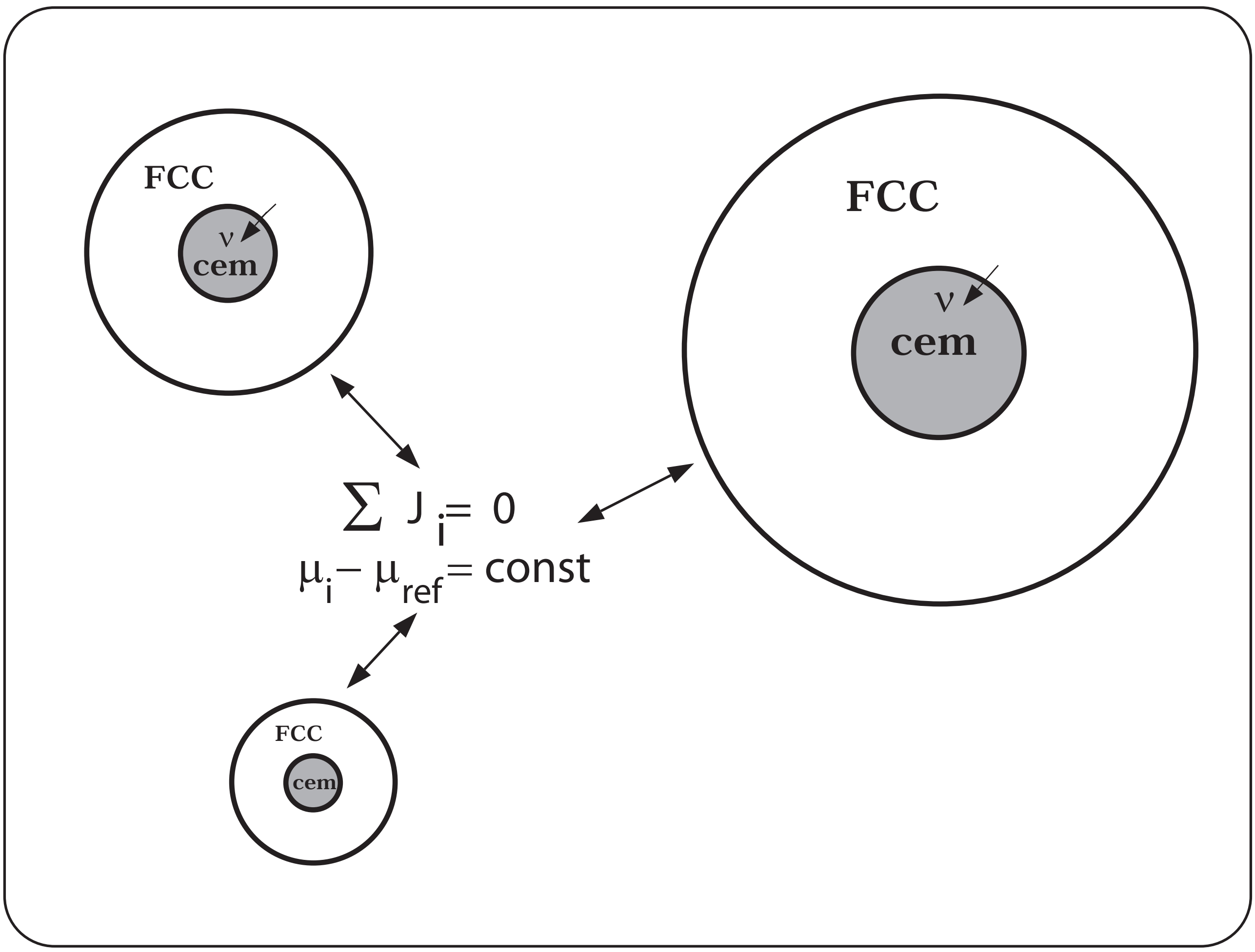

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example c2

Cementite disolution in an Fe-Cr-C alloy

(Three different particle sizes, Three cell problem)

T = 1183K



exc2-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc2\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Cell calculations.
SYS: @@ Setup file for calculating the dissolution of CEMENTITE particles
SYS: @@ in an AUSTENITE matrix. This case is identical to exb2 except
SYS: @@ that we here have three different particle sizes. Altogether six
SYS: @@ particles are considered using three different cells. This is in
SYS: @@ order to represent some size distribution among the CEMENTITE
SYS: @@ particles. See also Z.-K. Liu, L. Höglund, B. Jönsson and J. Ågren:
SYS: @@ Metall.Trans.A, v. 22A (1991), pp. 1745-1752.
SYS: @@
SYS:
SYS: @@ exc2_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE TCFE DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: switch fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO: def-sys fe cr c
 FE                       CR                       C
   DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                    LIQUID:L                 BCC_A2
 CEMENTITE                CHI_A12                  DIAMOND_FCC_A4
 FCC_A1                   GRAPHITE                 HCP_A3
 KSI_CARBIDE              LAVES_PHASE_C14          M23C6
 M3C2                     M5C2                     M7C3
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc cementite
 FCC_A1                   CEMENTITE  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'P. Villars and L.D. Calvert (1985). Pearsons handbook of crystallographic
      data for intermetallic phases. Metals park, Ohio. American Society for
      Metals; Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'J. Bratberg, Z. Metallkd., Vol 96 (2005), 335-344; Fe-Cr-Mo-C'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe cr c
 FE                       CR                       C
   DEFINED
APP: rej ph * all
 BCC_A2                   CEMENTITE                FCC_A1
 FE4N_LP1                 HCP_A3                   LIQUID:L
   REJECTED
APP: res ph fcc cementite
 FCC_A1                   CEMENTITE  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Z. Metallkunde 85(1994)502-509; C diffusion in fcc Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'This parameter has been estimated'
 -OK-
```

```
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob t 0 1183; * n

DIC>
DIC> @@---------------------------------------------------------------------
DIC> @@ CELL NUMBER ONE
DIC> @@---------------------------------------------------------------------
DIC>
DIC> @@
DIC> @@ ENTER REGIONS carb AND aus
DIC> @@
DIC> enter-region carb
DIC> enter-region aus
ATTACH TO REGION NAMED /CARB/:
ATTACHED TO THE RIGHT OF CARB /YES/:
DIC> @@
DIC> @@ ENTER GEOMTRICAL GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ THE SIZE OF THE CEMENTITE PARTICLES WE KNOW SINCE WE ASSUME
DIC> @@ IT HAS BEEN MEASSURED.
DIC> @@
DIC> enter-grid
REGION NAME : /CARB/: carb
WIDTH OF REGION /1/: 0.700000e-6
TYPE /LINEAR/: lin
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ THE SIZE OF THE FCC REGION WE MAY CALCULATE FROM A MASSBALANCE
DIC> @@ AFTER ESTIMATING THE INITIAL COMPOSITIONS IN THE TWO PHASES.
DIC> @@
DIC> enter-grid
REGION NAME : /AUS/: aus
WIDTH OF REGION /1/: 7.1832993E-7
TYPE /LINEAR/: lin
NUMBER OF POINTS /50/: 50
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /CARB/: carb
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: cementite
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL VALUES FOR THE COMPOSITIONS IN THE PHASES
DIC> @@
DIC> enter-composition carb  cementite w-f
PROFILE FOR /CR/: cr lin 0.12423326 0.12423326
DIC>
DIC> enter-composition aus  fcc#1 fe w-f
PROFILE FOR /C/: cr lin 4.6615447E-3 4.6615447E-3
PROFILE FOR /CR/: c lin 1.5135207E-4 1.5135207E-4
DIC>
DIC> @@--------------------------------------------------------------
DIC> @@ CELL NUMBER TWO
DIC> @@--------------------------------------------------------------
DIC> create-new-cell
CELL DISTRIBUTION FACTOR /1/: 2
 CREATING NEW CELL, NUMBER:  2
 CELL   2 SELECTED
DIC-2>
DIC-2> @@
DIC-2> @@ ENTER REGIONS carb AND aus
DIC-2> @@
DIC-2> enter-region carb
DIC-2> enter-region aus
ATTACH TO REGION NAMED /CARB/:
ATTACHED TO THE RIGHT OF CARB /YES/:
DIC-2> @@
DIC-2> @@ ENTER GRIDS GEOMTRICAL GRIDS INTO THE REGIONS
DIC-2> @@
DIC-2> enter-grid carb  0.300000e-6 lin 50
DIC-2> enter-grid aus  3.0785568E-7 lin 50
DIC-2>
DIC-2> @@
DIC-2> @@ ENTER PHASES INTO REGIONS
DIC-2> @@
DIC-2> enter-phase act carb  matrix   cementite
DIC-2> enter-phase act aus   matrix   fcc#1
DIC-2>
DIC-2> @@
DIC-2> @@ ENTER INITIAL VALUES FOR THE COMPOSITIONS IN THE PHASES
DIC-2> @@
DIC-2> enter-composition carb  cementite w-f
PROFILE FOR /CR/: cr lin 0.12423326 0.12423326
DIC-2>
DIC-2> enter-composition aus  fcc#1 fe w-f
PROFILE FOR /C/: cr lin 4.6615447E-3 4.6615447E-3
PROFILE FOR /CR/: c lin 1.5135207E-4 1.5135207E-4
DIC-2>
```

```
DIC-2> @@---------------------------------------------------------------
DIC-2> @@ CELL NUMBER THREE
DIC-2> @@---------------------------------------------------------------
DIC-2> create-new-cell
CELL DISTRIBUTION FACTOR /1/: 3
 CREATING NEW CELL, NUMBER:  3
 CELL   3 SELECTED
DIC-3>
DIC-3> @@
DIC-3> @@ ENTER REGIONS carb AND aus
DIC-3> @@
DIC-3> enter-region carb
DIC-3> enter-region aus
ATTACH TO REGION NAMED /CARB/:
ATTACHED TO THE RIGHT OF CARB /YES/:
DIC-3> @@
DIC-3> @@ ENTER GRIDS GEOMTRICAL GRIDS INTO THE REGIONS
DIC-3> @@
DIC-3> enter-grid carb  0.525500e-6 ,,,,,,,,
DIC-3> enter-grid aus   5.3926054E-7 ,,,,,,,
DIC-3>
DIC-3> @@
DIC-3> @@ ENTER PHASES INTO REGIONS
DIC-3> @@
DIC-3> enter-phase act carb  matrix   cementite
DIC-3> enter-phase act aus   matrix   fcc#1
DIC-3>
DIC-3> @@
DIC-3> @@ ENTER INITIAL VALUES FOR THE COMPOSITIONS IN THE PHASES
DIC-3> @@
DIC-3> enter-composition carb  cementite w-f
PROFILE FOR /CR/: cr lin 0.12423326 0.12423326
DIC-3>
DIC-3> enter-composition aus  fcc#1 fe w-f
PROFILE FOR /C/: cr lin 4.6615447E-3 4.6615447E-3
PROFILE FOR /CR/: c lin 1.5135207E-4 1.5135207E-4
DIC-3>
DIC-3> @@---------------------------------------------------------------
DIC-3> @@ GLOBAL CONDITIONS
DIC-3> @@---------------------------------------------------------------
DIC-3>
DIC-3> @@
DIC-3> @@ SET SPHERICAL GEOMETRY
DIC-3> @@
DIC-3> enter-geo 2
DIC-3>
DIC-3> @s-n-l
s-n-l ?r inte ett internt kommando, externt kommando,
program eller kommandofil.
DIC-3> @
DIC-3> @
DIC-3> @
DIC-3> @
DIC-3> @
DIC-3> @
DIC-3> @
DIC-3> @1E-3
1E-3 ?r inte ett internt kommando, externt kommando,
program eller kommandofil.
DIC-3>
DIC-3>
DIC-3>
DIC-3>
DIC-3> @@
DIC-3> @@ SET THE SIMULATION TIME
DIC-3> @@
DIC-3> set-simulation-time
END TIME FOR INTEGRATION /.1/: 10000
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /1000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC-3>
DIC-3> @@
DIC-3> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC-3> @@
DIC-3> save exc2 Y
DIC-3>
DIC-3>
DIC-3> set-inter
 --OK---
DIC-3>
```

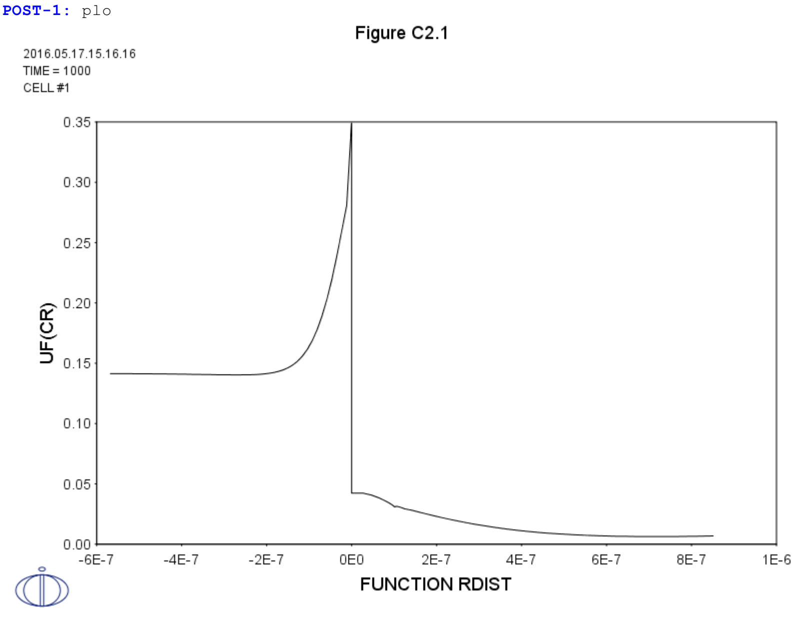
exc2-run

```
DIC-3>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc2\run.DCM"
DIC-3>
DIC-3> @@ exc2_run.DCM
DIC-3>
DIC-3> @@
DIC-3> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC-3> @@
DIC-3>
DIC-3> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC-3> read exc2
 OK
DIC> sim yes
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                  4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9
90445.3431211

  Trying old scheme                  4
 GENERATING STARTING VALUES FOR CELL #  2 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9
90445.3431211

  Trying old scheme                  4
 GENERATING STARTING VALUES FOR CELL #  3 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9
90445.3431211

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0406910187346776  CR = .0214382352304608
                        FE = .978561764900046
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]
 U-FRACTION IN SYSTEM:  C = .0406910187346776  CR = .0214382352304608
                        FE = .978561764900046
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]
 0.554229199924703        0.554264984244757        0.554287009345297        0.554269728044852        0.554229022173383        0.55422894889578(
002        3.619402770411673E-004        7.964124758061274E-005        2.187184519409445E-005        1.685079579871908E-
005        2.082114175152542E-005        1.240049578941501E-005        1.180448179452067E-005        1.180006359636886E-
005        1.208333394857920E-005        1.178105652465068E-005        1.193042101110187E-005        1.178076345426604E-
005        1.178414757360783E-005        1.178076596995851E-005        1.178080073750014E-005        1.180633641259092E-
005        1.179102683944175E-005        1.157290618485549E-005        1.159360739955918E-005        1.156918657317480E-
005        1.156430061138646E-005        1.154825648914751E-005        1.152472540597479E-005        1.152472806518411E-
005        1.158764704559660E-005        1.152519860873122E-005        1.150958690299894E-005        1.149542600235595E-
005        1.149531732504404E-005        1.149542851699429E-005        1.149546328425079E-005        1.149716460166777E-
005        1.147839789158560E-005        1.145916445200538E-005        1.142029946240986E-005        1.134289001838290E-
005        1.118833025241526E-005        1.088121169353384E-005        1.027391978240659E-005        1.027417657923200E-
005        1.027378891597178E-005        1.027473998117712E-005        9.089661016846937E-006        6.859554149148086E-
006        6.859616706370794E-006        6.859556663696659E-006

output ignored...


... output resumed

 1.167568379343949E-007        5.073075580865085E-011        8.520543436477685E-015        2.467276531993932E-018
 TIME =    7745.5356      DT =  1000.00000      SUM OF SQUARES =  0.24672765E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.90190637E-11 AND  -0.90190637E-11
 POSITION OF INTERFACE CARB / AUS IS   0.44659328E-06
 CELL # 3 VELOCITY AT INTERFACE # 2 IS  -0.96990890E-11 AND  -0.96990890E-11
 POSITION OF INTERFACE CARB / AUS IS   0.27462311E-06
 U-FRACTION IN SYSTEM:  C = .0407517241361818  CR = .0222252564372713
                        FE = .977774743693236
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]

  8  GRIDPOINT(S) REMOVED FROM CELL #3   REGION: AUS

  CPU time used in timestep                    6  seconds
 8.354352901459685E-003        8.354352853918412E-003        8.354352815249903E-003        8.354266499090863E-003        8.354205634797744E-
003        8.353996599950418E-003        8.353308511630452E-003        2.900884617965064E-004        1.622712539256617E-
007        9.136237749805124E-011        1.853768836151455E-014        5.865348934187477E-018
 TIME =    8745.5356      DT =  1000.00000      SUM OF SQUARES =  0.58653489E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.80706114E-11 AND  -0.80706114E-11
 POSITION OF INTERFACE CARB / AUS IS   0.43852266E-06
 CELL # 3 VELOCITY AT INTERFACE # 2 IS  -0.91026672E-11 AND  -0.91026672E-11
 POSITION OF INTERFACE CARB / AUS IS   0.26552044E-06
 U-FRACTION IN SYSTEM:  C = .0407517448349966  CR = .0222317572358824
                        FE = .977768242894624
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]
```

```
  20  GRIDPOINT(S) REMOVED FROM CELL #3   REGION: AUS

  CPU time used in timestep                    8  seconds
 1.031070573645433E-002        1.031070569566087E-002        1.031070565557092E-002        1.031062657045966E-002        1.031058097472161E-
002        1.031030925688489E-002        1.030950148420863E-002        2.914435703140629E-004        2.079871503797112E-
007        1.177624287058533E-010        2.818360173231995E-014        1.154409380737801E-017
 TIME =   9745.5356     DT =  1000.00000     SUM OF SQUARES =  0.11544094E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.72460993E-11 AND  -0.72460993E-11
 POSITION OF INTERFACE CARB / AUS IS   0.43127656E-06
 CELL # 3 VELOCITY AT INTERFACE # 2 IS  -0.86568277E-11 AND  -0.86568277E-11
 POSITION OF INTERFACE CARB / AUS IS   0.25686361E-06
 U-FRACTION IN SYSTEM:  C = .0407517499915482  CR = .0222347631980067
                       FE = .977765236932501
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]

  2  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUS
  24  GRIDPOINT(S) REMOVED FROM CELL #3   REGION: AUS

  CPU time used in timestep                    7  seconds
 1.711008577900788E-003        1.711008570204687E-003        1.711008577306692E-003        1.710918744444262E-003        1.710867203277678E-
003        1.710915483790776E-003        1.710926868494922E-003        5.745004846889339E-004        1.929628272327394E-
004        4.688614335316031E-007        1.597706081270243E-010        3.365657123692855E-014        2.411850712868458E-017
 TIME =  10000.0000     DT =   254.46437     SUM OF SQUARES =  0.24118507E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.61705257E-11 AND  -0.61705257E-11
 POSITION OF INTERFACE CARB / AUS IS   0.42970639E-06
 CELL # 3 VELOCITY AT INTERFACE # 2 IS  -0.73737459E-11 AND  -0.73737459E-11
 POSITION OF INTERFACE CARB / AUS IS   0.25498725E-06
 U-FRACTION IN SYSTEM:  C = .0407517561139694  CR = .0222352225402863
                       FE = .977764777590221
 TOTAL SIZE OF SYSTEM:  2.90023192349E-17 [m^3]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    3814.6587
 DELETING TIME-RECORD FOR TIME    4403.3584
 DELETING TIME-RECORD FOR TIME    4403.3584
 DELETING TIME-RECORD FOR TIME    4403.3584
 DELETING TIME-RECORD FOR TIME    4403.3584
 DELETING TIME-RECORD FOR TIME    4403.3585
 DELETING TIME-RECORD FOR TIME    4403.3587
 DELETING TIME-RECORD FOR TIME    4403.3590
 DELETING TIME-RECORD FOR TIME    4403.3596
 DELETING TIME-RECORD FOR TIME    4403.3609
 DELETING TIME-RECORD FOR TIME    4403.3635
 DELETING TIME-RECORD FOR TIME    4403.3686
 DELETING TIME-RECORD FOR TIME    4403.3788
 DELETING TIME-RECORD FOR TIME    4403.3993
 DELETING TIME-RECORD FOR TIME    4403.4403
 DELETING TIME-RECORD FOR TIME    4403.5222
 DELETING TIME-RECORD FOR TIME    4403.6860
 DELETING TIME-RECORD FOR TIME    4404.0137
 DELETING TIME-RECORD FOR TIME    4404.6691
 DELETING TIME-RECORD FOR TIME    4405.9798
 DELETING TIME-RECORD FOR TIME    4408.6012
 DELETING TIME-RECORD FOR TIME    4413.8441
 DELETING TIME-RECORD FOR TIME    4424.3299
 DELETING TIME-RECORD FOR TIME    4445.3014
 DELETING TIME-RECORD FOR TIME    4487.2444
 DELETING TIME-RECORD FOR TIME    4571.1305
 DELETING TIME-RECORD FOR TIME    4738.9027
 DELETING TIME-RECORD FOR TIME    5074.4470
 DELETING TIME-RECORD FOR TIME    5745.5356
 DELETING TIME-RECORD FOR TIME    6745.5356
 DELETING TIME-RECORD FOR TIME    7745.5356
 DELETING TIME-RECORD FOR TIME    8745.5356

 KEEPING TIME-RECORD FOR TIME    9745.5356
 AND FOR TIME                   10000.0000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

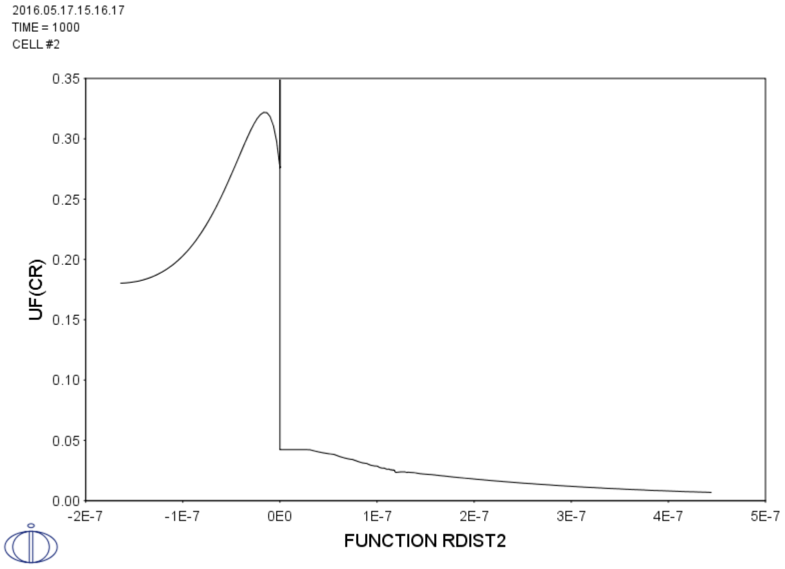
exc2-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exc2\plot.DCM"
DIC>
DIC> @@ exc2_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE c2
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+04
DIC> read exc2
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US PLOT CHROMIUM CONCENTRATION PROFILES IN THE SAME WAY AS IN exb2
POST-1: @@ BUT NOW FOR EACH PARTICLE. LET'S LOOK AT THE PROFILES AFTER 1000S.
POST-1: @@
POST-1:
POST-1: @@
POST-1: @@ FIRST CELL
POST-1: @@
POST-1: enter-symb
Function or table /FUNCTION/: func
NAME: rdist
FUNCTION: gd-poi(carb,u);
POST-1:
POST-1: s-d-a x rdist
POST-1:
POST-1: s-d-a y uf(cr)
POST-1:
POST-1: s-i-v
VARIABLE /TIME/: dist
DISTANCE : /GLOBAL/: glo
POST-1:
POST-1: s-p-c time 1000
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-title Figure C2.1
POST-1: plo
```



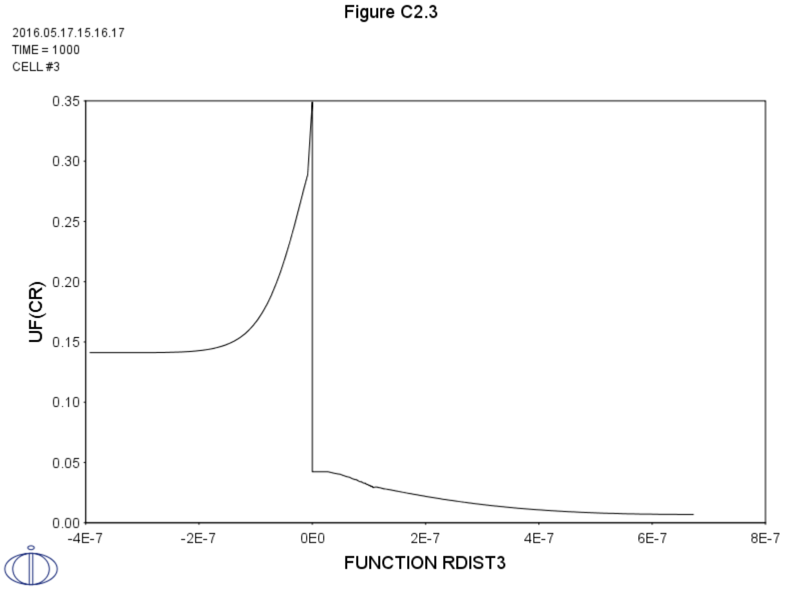

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ SELECT CELL 2
POST-1: @@
POST-1: select-cell
Number /NEXT/: 2
 CELL   2 SELECTED
POST-2:
POST-2: enter-symb
Function or table /FUNCTION/: func
NAME: rdist2
FUNCTION: gd-poi(carb,u);
POST-2:
POST-2: s-d-a x rdist2
POST-2:
POST-2: set-title Figure C2.2
POST-2: plo
```

Figure C2.2




```
POST-2:
POST-2:
POST-2:
POST-2:@?<_hit_return_to_continue_>
POST-2:
POST-2: @@
POST-2: @@ SELECT CELL 3
POST-2: @@
POST-2: select-cell 3
 CELL   3 SELECTED
POST-3:
POST-3: enter-symb
Function or table /FUNCTION/: func
NAME: rdist3
FUNCTION: gd-poi(carb,u);
POST-3:
POST-3: s-d-a x rdist3
POST-3:
POST-3: set-title Figure C2.3
POST-3: plo
```

Figure C2.3

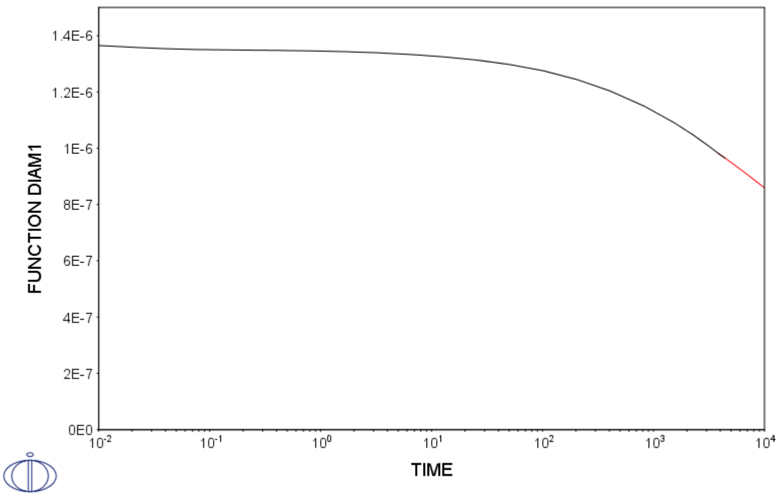



```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3:
POST-3: @@
POST-3: @@ LET US ALSO PLOT HOW THE DIAMETER OF THE CEMENTITE PARTICLE VARIES
POST-3: @@ WITH TIME IN THE THREE CELLS
POST-3: @@
POST-3:
POST-3: @@
POST-3: @@ SELECT FIRST CELL
POST-3: @@
POST-3: sel-cell 1
 CELL   1 SELECTED
POST-1:
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-s-s x n .01 10000
POST-1: set-axis-type x log
POST-1:
POST-1: enter func diam1=2*poi(carb,u);
POST-1: s-d-a y diam1
POST-1: s-s-s y n 0 1.5e-6
POST-1:
```

```
POST-1: s-p-c interf carb upp
POST-1:
POST-1: app n
POST-1: set-title Figure C2.4
POST-1: plo
```

Figure C2.4

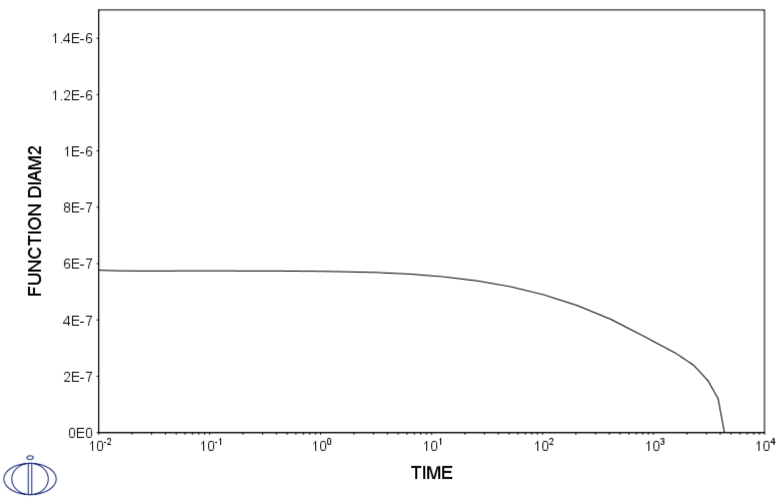
2016.05.17.15.16.18
UPPER INTERFACE OF REGION "CARB#1"
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1:
POST-1: @@
POST-1: @@ SELECT CELL 2
POST-1: @@
POST-1: sel-cell 2
 CELL    2 SELECTED
POST-2:
POST-2: enter func diam2=2*poi(carb,u);
POST-2: s-d-a y diam2
POST-2: s-s-s y n 0 1.5e-6
POST-2:
POST-2: s-p-c interf carb upp
POST-2:
POST-2: set-title Figure C2.5
POST-2: plo
```

Figure C2.5

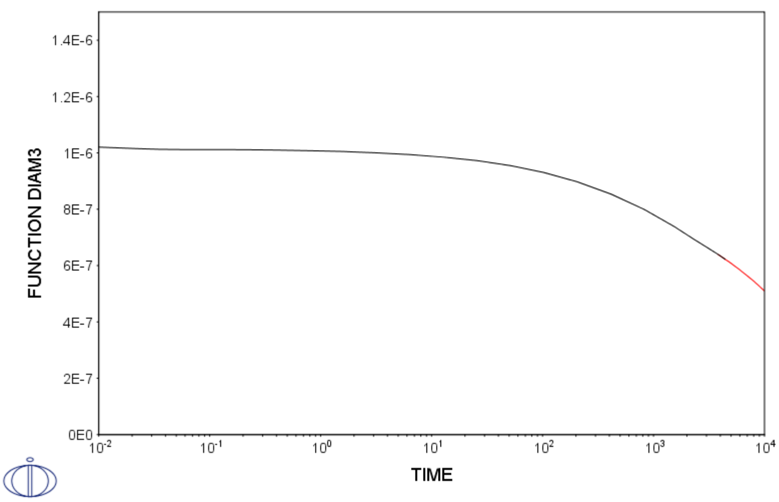
2016.05.17.15.16.19
UPPER INTERFACE OF REGION "CARB#2"
CELL #2




```
POST-2:
POST-2:
POST-2:
POST-2:
POST-2:@?<_hit_return_to_continue_>
POST-2:
POST-2: @@
POST-2: @@ SELECT CELL 3
POST-2: @@
POST-2: sel-cell 3
 CELL    3 SELECTED
POST-3:
POST-3: enter func diam3=2*poi(carb,u);
POST-3: s-d-a y diam3
POST-3: s-s-s y n 0 1.5e-6
POST-3:
POST-3: s-p-c interf carb upp
POST-3:
POST-3: set-title Figure C2.6
POST-3:
POST-3: plo
```

Figure C2.6

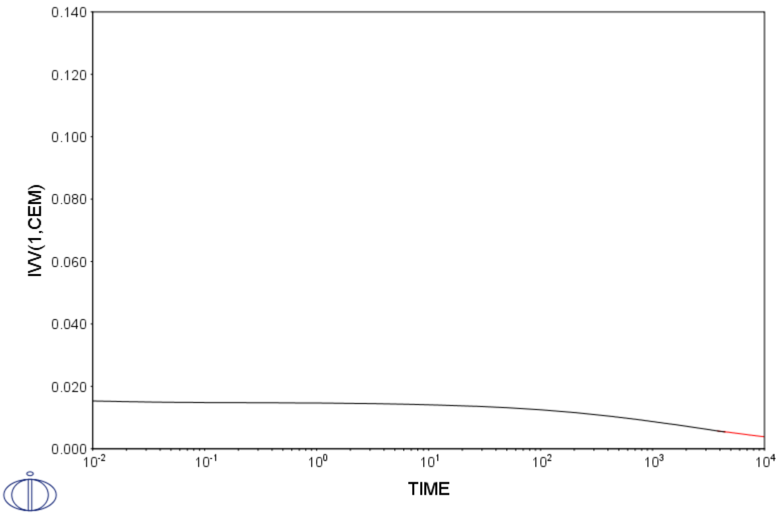
2016.05.17.15.16.20
UPPER INTERFACE OF REGION "CARB#3"
CELL #3




```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3: @@
POST-3: @@ LET US NOW PLOT THE VOLUME FRACTION OF CEMENTITE IN THE THREE
POST-3: @@ CELLS.
POST-3: @@
POST-3: s-d-a x time
  INFO: Time is set as independent variable
POST-3: s-s-s x n .01 10000
POST-3: set-axis-type x log
POST-3:
POST-3: @@
POST-3: @@ CELL 1
POST-3: @@
POST-3: s-d-a y ivv(1,cem)
POST-3: s-s-s y n 0 0.14
POST-3:
POST-3: s-p-c integral
POST-3:
POST-3: set-title Figure C2.7
POST-3: plo
```

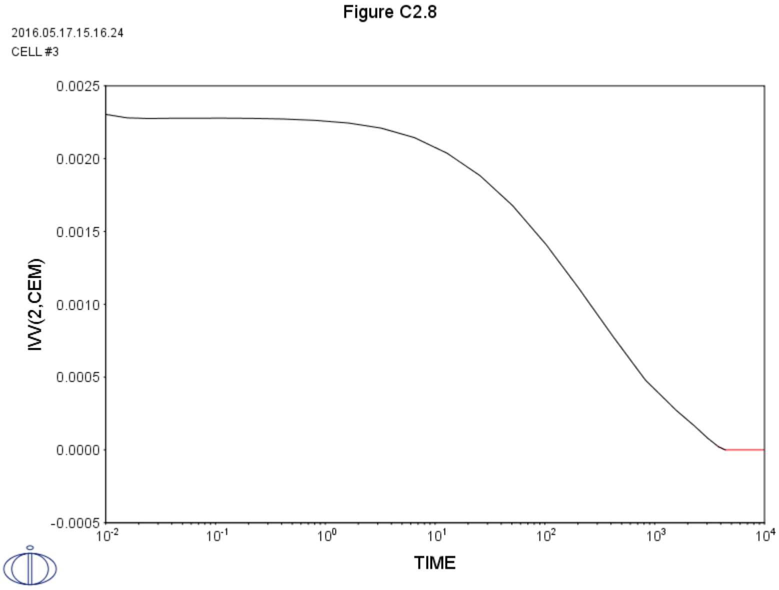
Figure C2.7

2016.05.17.15.16.22
CELL #3



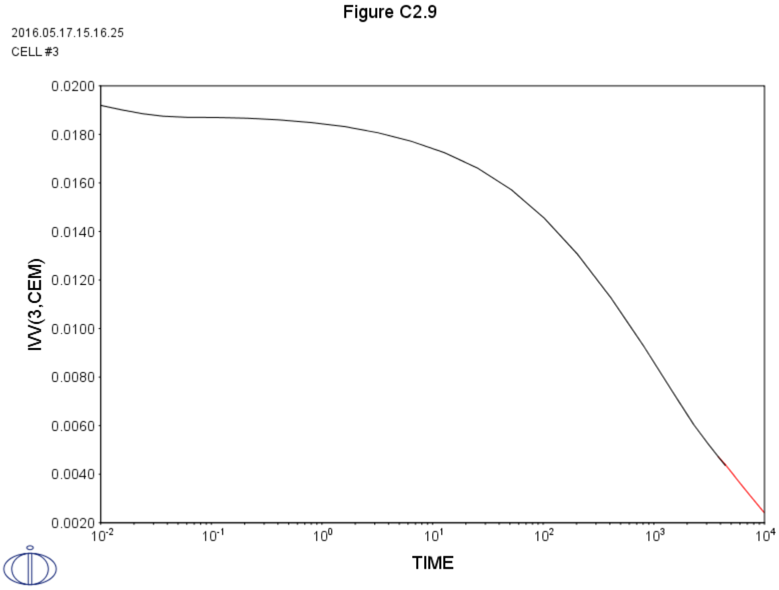

```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3: @@
POST-3: @@ CELL 2
POST-3: @@
POST-3: s-d-a y ivv(2,cem)
POST-3:
POST-3: set-title Figure C2.8
POST-3: plo
```

Figure C2.8



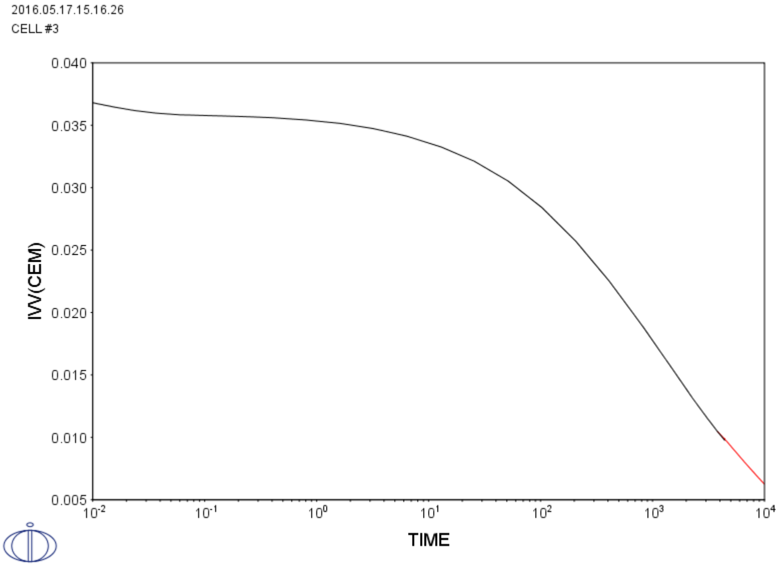

```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3: @@
POST-3: @@ CELL 3
POST-3: @@
POST-3: s-d-a y ivv(3,cem)
POST-3:
POST-3: set-title Figure C2.9
POST-3: plo
```

Figure C2.9



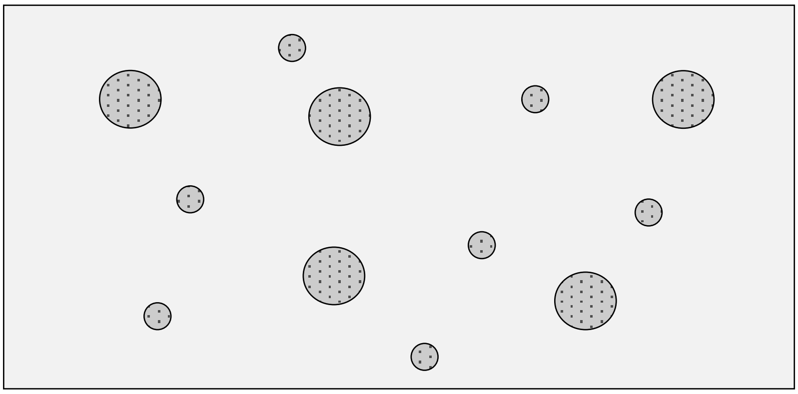

```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3: @@
POST-3: @@ FINALLY, LET US ALSO PLOT HOW THE TOTAL VOLUME FRACTION OF
POST-3: @@ CEMENTITE VARIES WITH TIME.
POST-3: @@
POST-3: s-d-a y ivv(cem)
POST-3:
POST-3: set-title Figure C2.10
POST-3: plo
```

Figure C2.10



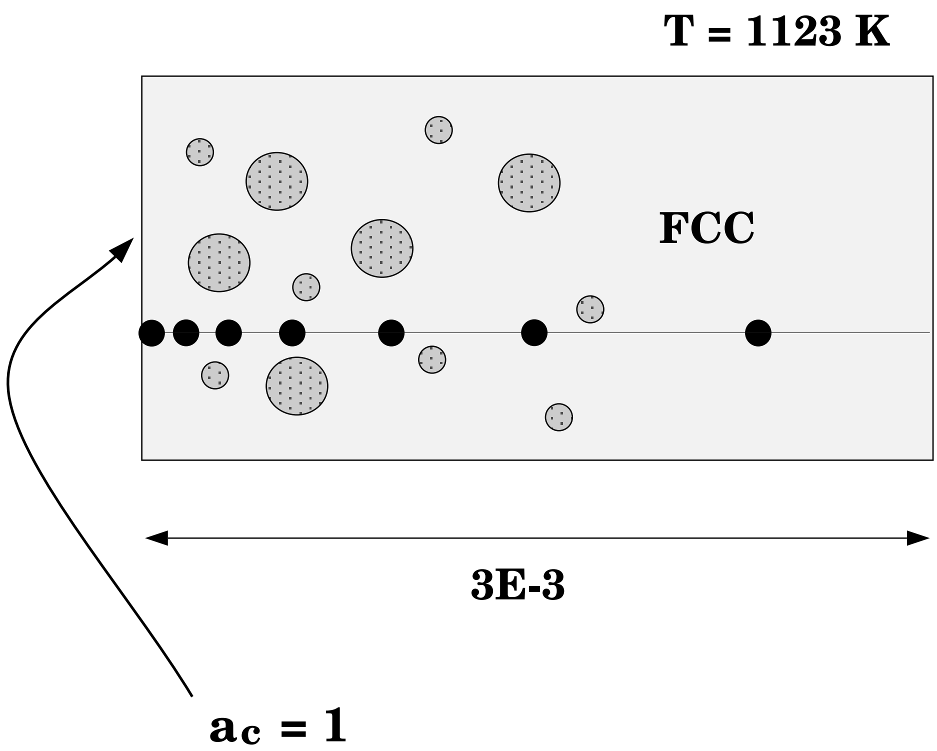

```
POST-3:
POST-3:
POST-3:
POST-3:
POST-3:@?<_hit_return_to_continue_>
POST-3:
POST-3: set-inter
 --OK---
POST-3:
```

Diffusion in dispersed systems



Example d1a

Carburization of a Ni-25% Cr alloy



exd1a-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1a\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Diffusion in dispersed systems.
SYS: @@ Setup file for carburization of a Ni-25Cr alloy.
SYS: @@ In this case the M3C2 and M7C3 carbides are entered as
SYS: @@ spheroid phases in a FCC matrix. This case is from
SYS: @@ A. Engström, L. Höglund and J. Ågren: Metall.Trans.A
SYS: @@ v. 25A (1994), pp. 1127-1134. This simulation can be run
SYS: @@ with either the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL.
SYS: @@ In this example the DISPERSED SYSTEM MODEL is used, which requires
SYS: @@ that the default HOMOGENIZATION MODEL is disabled.
SYS: @@ With the DISPERSED SYSTEM MODEL the command
SYS: @@ ENTER_LABYRINTH_FUNCTION is used to take into account the
SYS: @@ impeding effect of dispersed phases on long-range diffusion.
SYS: @@ For the HOMOGENIZATION MODEL the command
SYS: @@ ENTER_HOMOGENIZATION_FUNCTION should be used.
SYS: @@
SYS:
SYS: @@ exd1_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw ssol2
 Current database: SGTE Alloy Solutions Database v2.1

 VA  DEFINED
 B2_BCC                  L12_FCC                 AL5FE4:
   REJECTED
 GAS:G                   AQUEOUS:A               WATER:A
   REJECTED
TDB_SSOL2: def-sys ni cr c
 NI                      CR                      C
   DEFINED
TDB_SSOL2: rej ph * all
 LIQUID:L                FCC_A1                  BCC_A2
 HCP_A3                  DIAMOND_A4              CHI_A12
 CBCC_A12                CUB_A13                 SIGMA
 GRAPHITE                CEMENTITE               KSI_CARBIDE
 M23C6                   M7C3                    M3C2
 FE4N                    AL3NI2                  ALNI_B2
 CR3SI                   CRSI2  REJECTED
TDB_SSOL2: res ph fcc,m7c3,m3c2,grap
 FCC_A1                  M7C3                    M3C2
 GRAPHITE  RESTORED
TDB_SSOL2: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

   'Byeong-Joo Lee, unpublished revision (1991); C-Cr-Fe-Ni'
   'Alan Dinsdale, SGTE Data for Pure Elements, NPL Report DMA(A)195
       September 1989'
   'Alan Dinsdale, SGTE Data for Pure Elements,
       Calphad Vol 15(1991) p 317-425,
       also in NPL Report DMA(A)195 Rev. August 1990'
   'A. Gabriel, C. Chatillon, I. Ansara, to be published in
       High Temp. Sci. (parameters listed in
       Calphad Vol 11 (1987) pp 203-218); C-NI'
   'A. Dinsdale, T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
   'NPL, unpublished work (1989); C-Cr-Ni'
 -OK-
TDB_SSOL2:
TDB_SSOL2: @@
TDB_SSOL2: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_SSOL2: @@
TDB_SSOL2: app mob2
 Current database: Alloys Mobility  v2.4

 VA  DEFINED
 GAS:G  REJECTED
APP: def-sys ni c cr
 NI                      C                       CR
   DEFINED
APP: rej ph * all
 BCC_A2                  CEMENTITE               DIAMOND_A4
 FCC_A1                  FE4N                    GRAPHITE
 HCP_A3                  KSI_CARBIDE             LIQUID:L
 M23C6                   M3C2                    M7C3
 SIGMA  REJECTED
APP: res ph fcc,m7c3,m3c2,grap
 FCC_A1                  M7C3                    M3C2
 GRAPHITE  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
```

```
 List of references for assessed data

   'This parameter has not been assessed'
   'B. Jonsson: Z. Metallkunde 85(1994)502-509;
      C diffusion in fcc Cr-Fe-Ni'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Cr and Fe diffusion fcc Cr-Fe'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Cr and Ni diffusion fcc Cr-Ni'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1123; * N

DIC>
DIC> @@
DIC> @@ SET REFERENCE STATE FOR CARBON
DIC> @@
DIC> set-reference-state
Component: C
Reference state: grap
Temperature /*/: *
Pressure /100000/: 101325
DIC>
DIC> @@
DIC> @@ ENTER REGION aus
DIC> @@
DIC> enter-region aus
DIC>
DIC> @@
DIC> @@ ENTER GEOMETRICAL GRID INTO THE REGION
DIC> @@
DIC> enter-grid aus 3e-3 geo 100 1.02
DIC>
DIC> @@
DIC> @@ ENTER MATRIX PHASE IN THE REGION
DIC> @@
DIC> enter-phase act aus matrix fcc_a1#1
DIC>
DIC> @@
DIC> @@ ENTER THE START COMPOSITION FOR THE MATRIX PHASE
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: ni
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: cr
TYPE /LINEAR/: lin 25 25
PROFILE FOR /CR/: c
TYPE /LINEAR/: lin 1e-4 1e-4
DIC>
DIC> @@
DIC> @@ ENTER SPHEROIDAL PHASES IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: m7c3
DIC>
DIC> @@
DIC> @@ ENTER STOICHOMETRIC SPHEROIDAL PHASE IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: m3c2
DIC>
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR SPHEROIDAL PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: m7c3
USE EQUILIBRIUM VALUE /Y/: Y
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: m3c2
USE EQUILIBRIUM VALUE /Y/: Y
DIC>
DIC> @@
DIC> @@ SET BOUNDARY CONDITION
DIC> @@
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: boundary
BOUNDARY /LOWER/: lower
CONDITION TYPE /CLOSED_SYSTEM/: mixed
Dependent substitutional element:NI
Dependent interstitial element:VA
TYPE OF CONDITION FOR COMPONENT C /ZERO_FLUX/: activity
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 1;
HIGH TIME LIMIT /*/: *
```

```
ANY MORE RANGES /N/: N
TYPE OF CONDITION FOR COMPONENT CR /ZERO_FLUX/: zero-flux
DIC>
DIC> @@
DIC> @@ ENTER LABYRINTH FACTOR
DIC> @@
DIC> enter-lab
REGION NAME : aus
f(T,P,VOLFR,X)= volfr**2;
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME. REMEMBER WE MUST BE CAREFUL WITH THE
DIC> @@ TIMESTEP WHEN WE HAVE SPHEROIDAL PHASES PRESENT. IN THIS CASE
DIC> @@ WE DON'T ALLOW THE TIMESTEP TO BE LARGER THAN 1800s.
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 3600000
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /360000/: 1800
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ IN ORDER SO SAVE SOME SPACE ON THE DISK WE ONLY STORE THE RESULT
DIC> @@ SELECTIVELY, ELSE THE STORE-RESULT-FILE FROM THIS EXAMPLE WOULD BE
DIC> @@ VERY LARGE.
DIC> @@
DIC> set-simulation-condition
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/: 99
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC>
DIC> @@ AS OF DICTRA VERSION 27 IT IS BY DEFAULT THE "HOMOGENIZATION MODEL"
DIC> @@ THAT IS USED WHEN MULTIPLE PHASES ARE ENTERED IN A SINGLE REGION
DIC> @@ DISABLE THE HOMOGENIZATION MODEL:
DIC> ho n
 HOMOGENIZATION DISABLED
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exd1 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exd1a-run

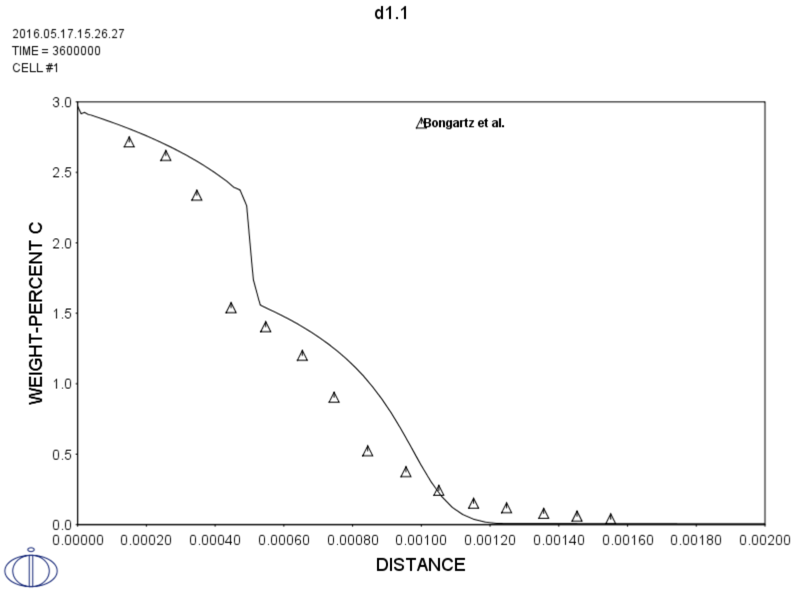
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1a\run.DCM"DIC>
DIC>
DIC> @@ exd1_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FILE AND START THE SIMULATION
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC> read exd1
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = 4.73399698964897E-06  CR = .273386070506604
                        NI = .726613929493396
 TOTAL SIZE OF SYSTEM:  .003 [m]
  WARNING:M7C3 HAS NO VOLUME FRACTION, CREATING ONE




  WARNING:M3C2 HAS NO VOLUME FRACTION, CREATING ONE




 U-FRACTION IN SYSTEM:  C = 4.73399698964897E-06  CR = .273386070506604
                        NI = .726613929493396
 TOTAL SIZE OF SYSTEM:  .003 [m]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 7.37829265031515E-05  CR = .273386070506603
                        NI = .726613929493397
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 1.07765273658405E-04  CR = .273386070506579
                        NI = .726613929493421
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 1.29844980713892E-04  CR = .273386070500949
                        NI = .726613929499051
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   61.221707     DT =   60.821607     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 1.67985911285439E-04  CR = .273386069358852
                        NI = .726613930641148
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   135.34814     DT =   74.126429     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 2.08836403949598E-04  CR = .273386068335923
                        NI = .726613931664077
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   242.67634     DT =   107.32821     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 2.57253556459831E-04  CR = .273386067317191
                        NI = .726613932682809
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   408.61790     DT =   165.94156     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 3.16829365327127E-04  CR = .273386066351576
                        NI = .726613933648424
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   677.99375     DT =   269.37585     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 3.91398501422713E-04  CR = .273386065452223
                        NI = .726613934547777
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   1140.3108     DT =   462.31701     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 4.86579533187524E-04  CR = .273386064383138
                        NI = .726613935616862
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds

 TIME =   2003.0736     DT =   862.76284     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 6.16999894543281E-04  CR = .273386063110096
                        NI = .726613936889904
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                     0  seconds
```

```
 TIME =   3701.3810     DT =   1698.3074     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 8.03308317133787E-04  CR = .273386061634911
                        NI = .726613938365089
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   5501.3810     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = 9.81626963664712E-04  CR = .27338606076244
                        NI = .72661393923756
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   7301.3810     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00114736292032063  CR = .273386060181809
                        NI = .726613939818191
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   9101.3810     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00130395620915012  CR = .273386059222276
                        NI = .726613940777724


output ignored...



... output resumed


  CPU time used in timestep                    0  seconds

 TIME =   3574559.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0315697587477024  CR = .273385942664622
                        NI = .726614057335378
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3576359.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0315776966548907  CR = .273385942634954
                        NI = .726614057365046
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    1  seconds

 TIME =   3578159.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0315856327218463  CR = .273385942605054
                        NI = .726614057394946
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3579959.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0315935669657689  CR = .273385942574922
                        NI = .726614057425078
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3581759.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316014994038047  CR = .273385942544559
                        NI = .726614057455441
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3583559.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316094300530421  CR = .273385942513963
                        NI = .726614057486037
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3585359.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316173589305067  CR = .273385942483135
                        NI = .726614057516865
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3587159.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316252860531568  CR = .273385942452074
                        NI = .726614057547926
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3588959.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316332114378795  CR = .27338594242078
                        NI = .726614057579219
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    1  seconds

 TIME =   3590759.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316411351014865  CR = .273385942389253
                        NI = .726614057610747
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3592559.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316490570607107  CR = .273385942357493
                        NI = .726614057642508
 TOTAL SIZE OF SYSTEM:  .003 [m]

  CPU time used in timestep                    0  seconds

 TIME =   3594359.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0316569773322022  CR = .273385942325498
                        NI = .726614057674502
 TOTAL SIZE OF SYSTEM:  .003 [m]
```

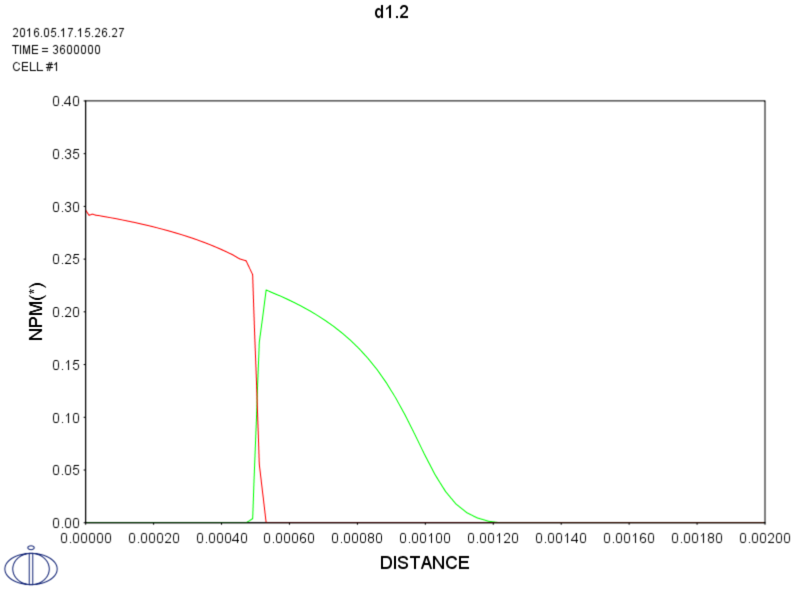
```
 CPU time used in timestep                     0  seconds

TIME =   3596159.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0316648959325255  CR = .27338594229327
                       NI = .72661405770673
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   3597959.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0316728128781563  CR = .273385942260807
                       NI = .726614057739193
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   3599759.7     DT =   1800.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .0316807281854781  CR = .273385942228109
                       NI = .726614057771891
TOTAL SIZE OF SYSTEM:  .003 [m]

 CPU time used in timestep                     0  seconds

TIME =   3600000.0     DT =   240.30748     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  C = .031681789133322  CR = .273385942223668
                       NI = .726614057776332
TOTAL SIZE OF SYSTEM:  .003 [m]

MUST SAVE WORKSPACE ON FILE
WORKSPACE SAVED ON FILE
RECLAIMING WORKSPACE
DELETING TIME-RECORD FOR TIME    0.0000000
DELETING TIME-RECORD FOR TIME   0.10000000E-06
DELETING TIME-RECORD FOR TIME   0.10010000E-03
DELETING TIME-RECORD FOR TIME    163559.69
DELETING TIME-RECORD FOR TIME    341759.69
DELETING TIME-RECORD FOR TIME    519959.69
DELETING TIME-RECORD FOR TIME    698159.69
DELETING TIME-RECORD FOR TIME    876359.69
DELETING TIME-RECORD FOR TIME    1054559.7
DELETING TIME-RECORD FOR TIME    1232759.7
DELETING TIME-RECORD FOR TIME    1410959.7
DELETING TIME-RECORD FOR TIME    1589159.7
DELETING TIME-RECORD FOR TIME    1767359.7
DELETING TIME-RECORD FOR TIME    1945559.7
DELETING TIME-RECORD FOR TIME    2123759.7
DELETING TIME-RECORD FOR TIME    2301959.7
DELETING TIME-RECORD FOR TIME    2480159.7
DELETING TIME-RECORD FOR TIME    2658359.7
DELETING TIME-RECORD FOR TIME    2836559.7
DELETING TIME-RECORD FOR TIME    3014759.7
DELETING TIME-RECORD FOR TIME    3192959.7
DELETING TIME-RECORD FOR TIME    3371159.7
DELETING TIME-RECORD FOR TIME    3549359.7
DELETING TIME-RECORD FOR TIME    3596159.7
DELETING TIME-RECORD FOR TIME    3597959.7

KEEPING TIME-RECORD FOR TIME    3599759.7
AND FOR TIME                    3600000.0
WORKSPACE RECLAIMED
DIC>
DIC> set-inter
--OK---
DIC>
```

exd1a-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1a\plot.DCM"
DIC>
DIC> @@ exd1_plot.DCM
DIC>
DIC> @@
DIC> @@ FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.60000E+06
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC> read exd1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US PLOT THE TOTAL CARBON CONCENTRATION PROFILE.
POST-1: @@
POST-1: s-d-a y w-p c
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-s-s x n 0 2e-3
POST-1: s-p-c time 3600000
POST-1:
POST-1: app y exd1.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-tit d1.1
POST-1:
POST-1: plot
```



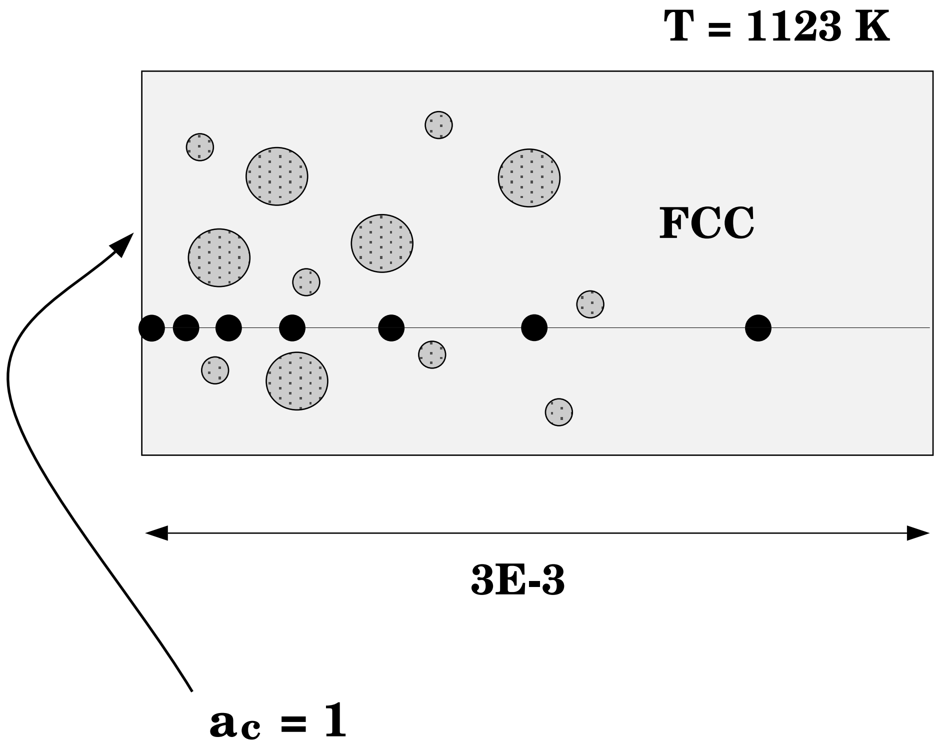

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ LET US NOW PLOT THE AMOUNT OF CARBIDES FORMED.
POST-1: @@
POST-1: s-d-a y npm(*)
POST-1: s-s-s y n 0 0.4
POST-1: app n
POST-1:
POST-1: set-tit d1.2
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example d1b

Carburization of a Ni-25% Cr alloy using homogenization model



exd1b-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1b\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Diffusion in dispersed systems.
SYS: @@ Setup file for carburization of a Ni-25Cr alloy.
SYS: @@ In this case the M3C2 and M7C3 carbides are entered as
SYS: @@ spheroid phases in a FCC matrix. This case is from
SYS: @@ A. Engström, L. Höglund and J. Ågren: Metall.Trans. A,
SYS: @@ v.25A (1994), pp. 1127-1134. This simulation can be run
SYS: @@ with either the DISPERSED SYSTEM MODEL or the
SYS: @@ HOMOGENIZATION MODEL. The default HOMOGENIZATION MODEL is used
SYS: @@ and then ENTER_HOMOGENIZATION_FUNCTION should be used instead of
SYS: @@ ENTER_LABYRINTH_FUNCTION.
SYS: @@
SYS:
SYS: @@ exd1b_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE SSOL DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw ssol2
 Current database: SGTE Alloy Solutions Database v2.1

 VA  DEFINED
 B2_BCC                   L12_FCC                  AL5FE4:
   REJECTED
 GAS:G                    AQUEOUS:A                WATER:A
   REJECTED
TDB_SSOL2: def-sys ni cr c
 NI                       CR                       C
   DEFINED
TDB_SSOL2: rej ph * all
 LIQUID:L                 FCC_A1                   BCC_A2
 HCP_A3                   DIAMOND_A4               CHI_A12
 CBCC_A12                 CUB_A13                  SIGMA
 GRAPHITE                 CEMENTITE                KSI_CARBIDE
 M23C6                    M7C3                     M3C2
 FE4N                     AL3NI2                   ALNI_B2
 CR3SI                    CRSI2  REJECTED
TDB_SSOL2: res ph fcc,m7c3,m3c2,grap
 FCC_A1                   M7C3                     M3C2
 GRAPHITE  RESTORED
TDB_SSOL2: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

   'Byeong-Joo Lee, unpublished revision (1991); C-Cr-Fe-Ni'
   'Alan Dinsdale, SGTE Data for Pure Elements, NPL Report DMA(A)195
       September 1989'
   'Alan Dinsdale, SGTE Data for Pure Elements,
       Calphad Vol 15(1991) p 317-425,
       also in NPL Report DMA(A)195 Rev. August 1990'
   'A. Gabriel, C. Chatillon, I. Ansara, to be published in
       High Temp. Sci. (parameters listed in
       Calphad Vol 11 (1987) pp 203-218); C-NI'
   'A. Dinsdale, T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
   'NPL, unpublished work (1989); C-Cr-Ni'
 -OK-
TDB_SSOL2:
TDB_SSOL2: @@
TDB_SSOL2: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_SSOL2: @@
TDB_SSOL2: app mob2
 Current database: Alloys Mobility  v2.4

 VA  DEFINED
 GAS:G  REJECTED
APP: def-sys ni c cr
 NI                       C                        CR
   DEFINED
APP: rej ph * all
 BCC_A2                   CEMENTITE                DIAMOND_A4
 FCC_A1                   FE4N                     GRAPHITE
 HCP_A3                   KSI_CARBIDE              LIQUID:L
 M23C6                    M3C2                     M7C3
 SIGMA  REJECTED
APP: res ph fcc,m7c3,m3c2,grap
 FCC_A1                   M7C3                     M3C2
 GRAPHITE  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

   'This parameter has not been assessed'
   'B. Jonsson: Z. Metallkunde 85(1994)502-509;
```

```
      C diffusion in fcc Cr-Fe-Ni'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Cr and Fe diffusion fcc Cr-Fe'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Cr and Ni diffusion fcc Cr-Ni'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1123; * N

DIC>
DIC> @@
DIC> @@ SET REFERENCE STATE FOR CARBON
DIC> @@
DIC> set-reference-state
Component: C
Reference state: grap
Temperature /*/: *
Pressure /100000/: 101325
DIC>
DIC> @@
DIC> @@ ENTER REGION aus
DIC> @@
DIC> enter-region aus
DIC>
DIC> @@
DIC> @@ ENTER GEOMETRICAL GRID INTO THE REGION
DIC> @@
DIC> enter-grid aus 3e-3 geo 100 1.02
DIC>
DIC> @@
DIC> @@ ENTER MATRIX PHASE IN THE REGION
DIC> @@
DIC> enter-phase act aus matrix fcc_a1#1
DIC>
DIC> @@
DIC> @@ ENTER THE START COMPOSITION FOR THE MATRIX PHASE
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: ni
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: cr
TYPE /LINEAR/: lin 25 25
PROFILE FOR /CR/: c
TYPE /LINEAR/: lin 1e-4 1e-4
DIC>
DIC> @@
DIC> @@ ENTER SPHEROIDAL PHASES IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: m7c3
DIC>
DIC> @@
DIC> @@ ENTER STOICHOMETRIC SPHEROIDAL PHASE IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /AUS/: aus
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: m3c2
DIC>
DIC>
DIC> @@
DIC> @@ ENTER START COMPOSITION FOR SPHEROIDAL PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: m7c3
USE EQUILIBRIUM VALUE /Y/: Y
DIC> enter-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: m3c2
USE EQUILIBRIUM VALUE /Y/: Y
DIC>
DIC> @@
DIC> @@ SET BOUNDARY CONDITION
DIC> @@
DIC> set-cond

GLOBAL OR BOUNDARY CONDITION /GLOBAL/: boundary
BOUNDARY /LOWER/: lower
CONDITION TYPE /CLOSED_SYSTEM/: mixed
Dependent substitutional element:NI
Dependent interstitial element:VA
TYPE OF CONDITION FOR COMPONENT C /ZERO_FLUX/: activity
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 1;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
TYPE OF CONDITION FOR COMPONENT CR /ZERO_FLUX/: zero-flux
DIC>
DIC> @@
DIC> @@ SELECT A HOMOGENIZATION FUNCTION
```

```
DIC> @@
DIC> enter-homo
ENTER HOMOGENIZATION FUNCTION # /5/: 8
 SELECTED FUNCTION IS LABYRINTH FACTOR f**2 WITH PRESCRIBED MATRIX PHASE
PHASE NAME:: fcc#1
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME. REMEMBER WE MUST BE CAREFUL WITH THE
DIC> @@ TIMESTEP WHEN WE HAVE SPHEROIDAL PHASES PRESENT. IN THIS CASE
DIC> @@ WE DON'T ALLOW THE TIMESTEP TO BE LARGER THAN 1800s.
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 3600000
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /360000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC>
DIC> @@
DIC> @@ IN ORDER SO SAVE SOME SPACE ON THE DISK WE ONLY STORE THE RESULT
DIC> @@ SELECTIVELY, ELSE THE STORE-RESULT-FILE FROM THIS EXAMPLE WOULD BE
DIC> @@ VERY LARGE.
DIC> @@
DIC> set-simulation-condition
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/: 99
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exd1 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exd1b-run

```
DIC>About NO SUCH COMMAND, USE HELP
DIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1b\run.DCM"DIC>
DIC>
DIC> @@ exd1_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FILE AND START THE SIMULATION
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC> read exd1
 OK
DIC> sim
 STARTING SIMULATION USING HOMOGENIZATION MODEL
 ----------------------------------------------
 INFO: PHASE WITH LIMITED SOLUBILITY OF ELEMENT(S) EXIST
 A FALLBACK PHASE ZZDICTRA_GHOST WILL BE DEFINED
 ALONG WITH THE FOLLOWING PARAMETERS:
 G(ZZDICTRA_GHOST,C;0)-H298(GRAPHITE,C;0)
 G(ZZDICTRA_GHOST,CR;0)-H298(BCC_A2,CR;0)
 G(ZZDICTRA_GHOST,NI;0)-H298(FCC_A1,NI;0)
 L(ZZDICTRA_GHOST,C,CR;0)
 L(ZZDICTRA_GHOST,C,NI;0)
 L(ZZDICTRA_GHOST,CR,NI;0)
 Saving results
 ------------------------------------------------
  WARNING:M7C3 HAS NO VOLUME FRACTION, CREATING ONE
  WARNING:M3C2 HAS NO VOLUME FRACTION, CREATING ONE
 Starting time-step t0=  0.000000000000000E+000  dt=  1.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420199165858907E-008  /   4.733974808906108E-006
 CR   8.201582115198115E-004  /   0.273384776303833
 NI   2.179841788480188E-003  /   0.726610489721358

 Time-step converged t0=  0.000000000000000E+000  dt=  1.000000000000000E-007
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=  1.000000000000000E-007  dt=  2.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420199303787341E-008  /   4.733975268663203E-006
 CR   8.201582115198115E-004  /   0.273384776303707
 NI   2.179841788480188E-003  /   0.726610489721024

 Time-step converged t0=  1.000000000000000E-007  dt=  2.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  3.000000000000000E-007  dt=  4.000000000000000E-007
 Stage 1
 Evaluating Jacobian
 Iteration #                     1  sum residual **2:   8.592925588816543E-004
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420199579644213E-008  /   4.733976188177402E-006
 CR   8.201582115198115E-004  /   0.273384776303456
 NI   2.179841788480188E-003  /   0.726610489720356

 Time-step converged t0=  3.000000000000000E-007  dt=  4.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  7.000000000000000E-007  dt=  8.000000000000000E-007
 Stage 1
 Iteration #                     1  sum residual **2:   1.435648489343165E-004
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420200131357960E-008  /   4.733978027205813E-006
 CR   8.201582115198115E-004  /   0.273384776302953
 NI   2.179841788480188E-003  /   0.726610489719020

 Time-step converged t0=  7.000000000000000E-007  dt=  8.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  1.500000000000000E-006  dt=  1.600000000000000E-006
 Stage 1
 Iteration #                     1  sum residual **2:   6.186519495508916E-005
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420201234785418E-008  /   4.733981705262516E-006
 CR   8.201582115198115E-004  /   0.273384776301948
 NI   2.179841788480188E-003  /   0.726610489716347

 Time-step converged t0=  1.500000000000000E-006  dt=  1.600000000000000E-006
 ------------------------------------------------
 Starting time-step t0=  3.100000000000000E-006  dt=  3.200000000000000E-006
 Stage 1
 Iteration #                     1  sum residual **2:   4.261106992515092E-005
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420203441639994E-008  /   4.733989061374788E-006
 CR   8.201582115198115E-004  /   0.273384776299937
 NI   2.179841788480188E-003  /   0.726610489711002

 Time-step converged t0=  3.100000000000000E-006  dt=  3.200000000000000E-006
 ------------------------------------------------
 Starting time-step t0=  6.300000000000000E-006  dt=  6.400000000000000E-006
 Stage 1
 Iteration #                     1  sum residual **2:   3.571801231626558E-005
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420207855351028E-008  /   4.734003773605606E-006
 CR   8.201582115198115E-004  /   0.273384776295914
```

```
 NI   2.179841788480188E-003  /   0.726610489700312

 Time-step converged t0=  6.300000000000000E-006  dt=  6.400000000000000E-006
 -------------------------------------------------
 Starting time-step t0=  1.270000000000000E-005  dt=  1.280000000000000E-005
 Stage 1
 Iteration #                     1  sum residual **2:   3.276725225871970E-005
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 C    1.420216682766344E-008  /   4.734033198044733E-006
 CR   8.201582115198115E-004  /   0.273384776287870
 NI   2.179841788480188E-003  /   0.726610489678932

 Time-step converged t0=  1.270000000000000E-005  dt=  1.280000000000000E-005
 -------------------------------------------------
 Starting time-step t0=  2.550000000000000E-005  dt=  2.560000000000000E-005
 Stage 1
 Iteration #                     1  sum residual **2:   3.138169207816513E-005
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
```

output ignored...

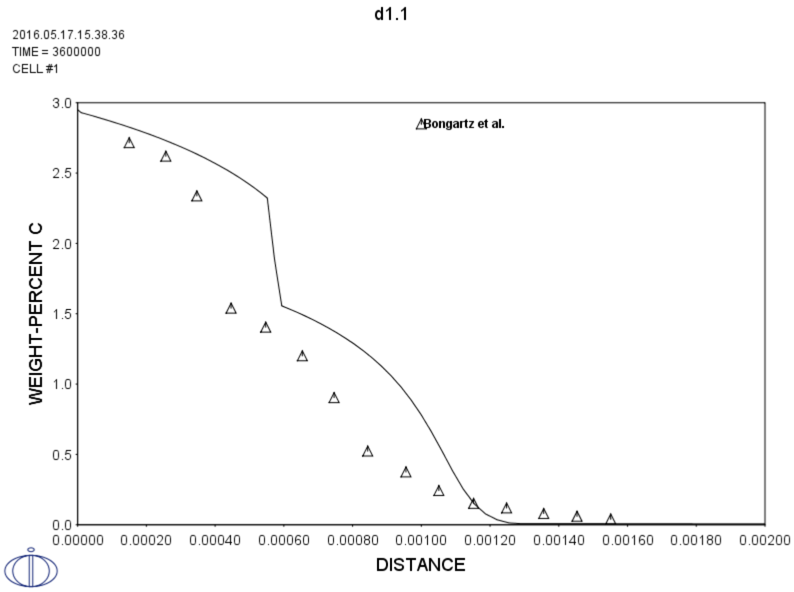
... output resumed

```
 DELETING TIME-RECORD FOR TIME    3078451.6
 DELETING TIME-RECORD FOR TIME    3082317.0
 DELETING TIME-RECORD FOR TIME    3086182.5
 DELETING TIME-RECORD FOR TIME    3090048.0
 DELETING TIME-RECORD FOR TIME    3093913.5
 DELETING TIME-RECORD FOR TIME    3097778.9
 DELETING TIME-RECORD FOR TIME    3101644.4
 DELETING TIME-RECORD FOR TIME    3105509.9
 DELETING TIME-RECORD FOR TIME    3109375.3
 DELETING TIME-RECORD FOR TIME    3113240.8
 DELETING TIME-RECORD FOR TIME    3117106.3
 DELETING TIME-RECORD FOR TIME    3120971.7
 DELETING TIME-RECORD FOR TIME    3124837.2
 DELETING TIME-RECORD FOR TIME    3128702.7
 DELETING TIME-RECORD FOR TIME    3132568.2
 DELETING TIME-RECORD FOR TIME    3136433.6
 DELETING TIME-RECORD FOR TIME    3140299.1
 DELETING TIME-RECORD FOR TIME    3144164.6
 DELETING TIME-RECORD FOR TIME    3148030.0
 DELETING TIME-RECORD FOR TIME    3151895.5
 DELETING TIME-RECORD FOR TIME    3155761.0
 DELETING TIME-RECORD FOR TIME    3159626.5
 DELETING TIME-RECORD FOR TIME    3163491.9
 DELETING TIME-RECORD FOR TIME    3167357.4
 DELETING TIME-RECORD FOR TIME    3171222.9
 DELETING TIME-RECORD FOR TIME    3175088.3
 DELETING TIME-RECORD FOR TIME    3178953.8
 DELETING TIME-RECORD FOR TIME    3182819.3
 DELETING TIME-RECORD FOR TIME    3186684.7
 DELETING TIME-RECORD FOR TIME    3190550.2
 DELETING TIME-RECORD FOR TIME    3194415.7
 DELETING TIME-RECORD FOR TIME    3198281.2
 DELETING TIME-RECORD FOR TIME    3202146.6
 DELETING TIME-RECORD FOR TIME    3206012.1
 DELETING TIME-RECORD FOR TIME    3209877.6
 DELETING TIME-RECORD FOR TIME    3213743.0
 DELETING TIME-RECORD FOR TIME    3217608.5
 DELETING TIME-RECORD FOR TIME    3221474.0
 DELETING TIME-RECORD FOR TIME    3225339.5
 DELETING TIME-RECORD FOR TIME    3229204.9
 DELETING TIME-RECORD FOR TIME    3233070.4
 DELETING TIME-RECORD FOR TIME    3236935.9
 DELETING TIME-RECORD FOR TIME    3240801.3
 DELETING TIME-RECORD FOR TIME    3244666.8
 DELETING TIME-RECORD FOR TIME    3248532.3
 DELETING TIME-RECORD FOR TIME    3252397.7
 DELETING TIME-RECORD FOR TIME    3256263.2
 DELETING TIME-RECORD FOR TIME    3260128.7
 DELETING TIME-RECORD FOR TIME    3263994.2
 DELETING TIME-RECORD FOR TIME    3267859.6
 DELETING TIME-RECORD FOR TIME    3271725.1
 DELETING TIME-RECORD FOR TIME    3275590.6
 DELETING TIME-RECORD FOR TIME    3279456.0
 DELETING TIME-RECORD FOR TIME    3283321.5
 DELETING TIME-RECORD FOR TIME    3287187.0
 DELETING TIME-RECORD FOR TIME    3291052.5
 DELETING TIME-RECORD FOR TIME    3294917.9
 DELETING TIME-RECORD FOR TIME    3298783.4
 DELETING TIME-RECORD FOR TIME    3302648.9
 DELETING TIME-RECORD FOR TIME    3306514.3
 DELETING TIME-RECORD FOR TIME    3310379.8
 DELETING TIME-RECORD FOR TIME    3314245.3
 DELETING TIME-RECORD FOR TIME    3318110.7
 DELETING TIME-RECORD FOR TIME    3321976.2
 DELETING TIME-RECORD FOR TIME    3325841.7
 DELETING TIME-RECORD FOR TIME    3329707.2
 DELETING TIME-RECORD FOR TIME    3333572.6
 DELETING TIME-RECORD FOR TIME    3337438.1
 DELETING TIME-RECORD FOR TIME    3341303.6
 DELETING TIME-RECORD FOR TIME    3345169.0
 DELETING TIME-RECORD FOR TIME    3349034.5
 DELETING TIME-RECORD FOR TIME    3352900.0
 DELETING TIME-RECORD FOR TIME    3356765.5
 DELETING TIME-RECORD FOR TIME    3360630.9
 DELETING TIME-RECORD FOR TIME    3364496.4
 DELETING TIME-RECORD FOR TIME    3368361.9
 DELETING TIME-RECORD FOR TIME    3372227.3
 DELETING TIME-RECORD FOR TIME    3376092.8
 DELETING TIME-RECORD FOR TIME    3379958.3
 DELETING TIME-RECORD FOR TIME    3383823.7
 DELETING TIME-RECORD FOR TIME    3387689.2
 DELETING TIME-RECORD FOR TIME    3391554.7
 DELETING TIME-RECORD FOR TIME    3395420.2
 DELETING TIME-RECORD FOR TIME    3399285.6
 DELETING TIME-RECORD FOR TIME    3403151.1
 DELETING TIME-RECORD FOR TIME    3407016.6
```

```
 DELETING TIME-RECORD FOR TIME    3410882.0
 DELETING TIME-RECORD FOR TIME    3414747.5
 DELETING TIME-RECORD FOR TIME    3418613.0
 DELETING TIME-RECORD FOR TIME    3422478.5
 DELETING TIME-RECORD FOR TIME    3426343.9
 DELETING TIME-RECORD FOR TIME    3430209.4
 DELETING TIME-RECORD FOR TIME    3434074.9
 DELETING TIME-RECORD FOR TIME    3437940.3
 DELETING TIME-RECORD FOR TIME    3441805.8
 DELETING TIME-RECORD FOR TIME    3445671.3
 DELETING TIME-RECORD FOR TIME    3449536.7
 DELETING TIME-RECORD FOR TIME    3453402.2
 DELETING TIME-RECORD FOR TIME    3457267.7
 DELETING TIME-RECORD FOR TIME    3461133.2
 DELETING TIME-RECORD FOR TIME    3464998.6
 DELETING TIME-RECORD FOR TIME    3468864.1
 DELETING TIME-RECORD FOR TIME    3472729.6
 DELETING TIME-RECORD FOR TIME    3476595.0
 DELETING TIME-RECORD FOR TIME    3480460.5
 DELETING TIME-RECORD FOR TIME    3484326.0
 DELETING TIME-RECORD FOR TIME    3488191.5
 DELETING TIME-RECORD FOR TIME    3492056.9
 DELETING TIME-RECORD FOR TIME    3495922.4
 DELETING TIME-RECORD FOR TIME    3499787.9
 DELETING TIME-RECORD FOR TIME    3503653.3
 DELETING TIME-RECORD FOR TIME    3507518.8
 DELETING TIME-RECORD FOR TIME    3511384.3
 DELETING TIME-RECORD FOR TIME    3515249.7
 DELETING TIME-RECORD FOR TIME    3519115.2
 DELETING TIME-RECORD FOR TIME    3522980.7
 DELETING TIME-RECORD FOR TIME    3526846.2
 DELETING TIME-RECORD FOR TIME    3530711.6
 DELETING TIME-RECORD FOR TIME    3534577.1
 DELETING TIME-RECORD FOR TIME    3538442.6
 DELETING TIME-RECORD FOR TIME    3542308.0
 DELETING TIME-RECORD FOR TIME    3546173.5
 DELETING TIME-RECORD FOR TIME    3550039.0
 DELETING TIME-RECORD FOR TIME    3553904.5
 DELETING TIME-RECORD FOR TIME    3557769.9
 DELETING TIME-RECORD FOR TIME    3561635.4
 DELETING TIME-RECORD FOR TIME    3565500.9
 DELETING TIME-RECORD FOR TIME    3569366.3
 DELETING TIME-RECORD FOR TIME    3573231.8
 DELETING TIME-RECORD FOR TIME    3577097.3
 DELETING TIME-RECORD FOR TIME    3580962.7
 DELETING TIME-RECORD FOR TIME    3584828.2
 DELETING TIME-RECORD FOR TIME    3588693.7
 DELETING TIME-RECORD FOR TIME    3592559.2

 KEEPING TIME-RECORD FOR TIME    3596424.6
 AND FOR TIME                    3600000.0
 WORKSPACE RECLAIMED
 ------------------------------------------------
 INTERPOLATION SCHEME USED THIS FRACTION OF
 THE ALLOCATED MEMORY:   0.154181976102994
 EFFICIENCY FACTOR:    92.8610080825695
 ------------------------------------------------
 DEALLOCATING
 ------------------------------------------------
DIC>
DIC> set-inter
 --OK---
DIC>
```

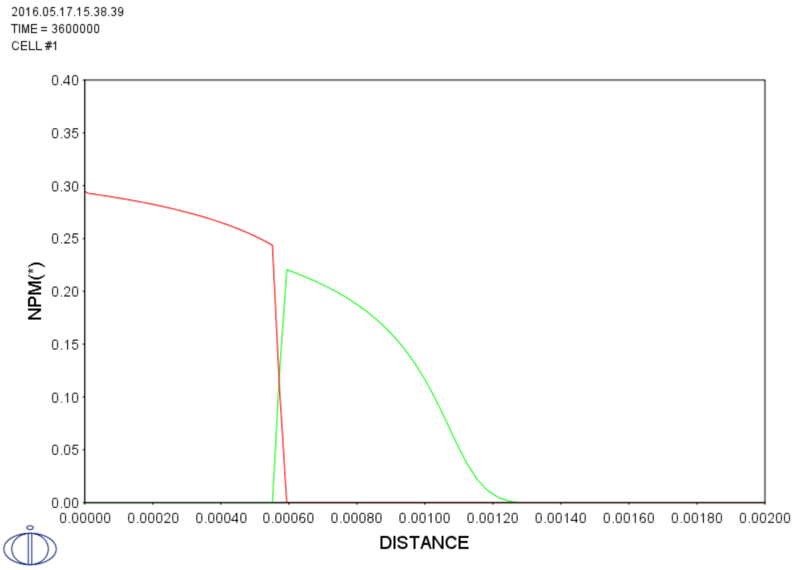
exd1b-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd1b\plot.DCM"
DIC>
DIC> @@ exd1_plot.DCM
DIC>
DIC> @@
DIC> @@ FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.60000E+06
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
 *** ENTERING M3C2 AS A DIFFUSION NONE PHASE
 *** ENTERING M7C3 AS A DIFFUSION NONE PHASE
DIC> read exd1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET US PLOT THE TOTAL CARBON CONCENTRATION PROFILE.
POST-1: @@
POST-1: s-d-a y w-p c
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-s-s x n 0 2e-3
POST-1: s-p-c time 3600000
POST-1:
POST-1: app y exd1.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-tit d1.1
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: @@
POST-1: @@ LET US NOW PLOT THE AMOUNT OF CARBIDES FORMED.
POST-1: @@
POST-1: s-d-a y npm(*)
POST-1: s-s-s y n 0 0.4
POST-1: app n
POST-1:
POST-1: set-tit d1.2
POST-1: plot
```

d1.2

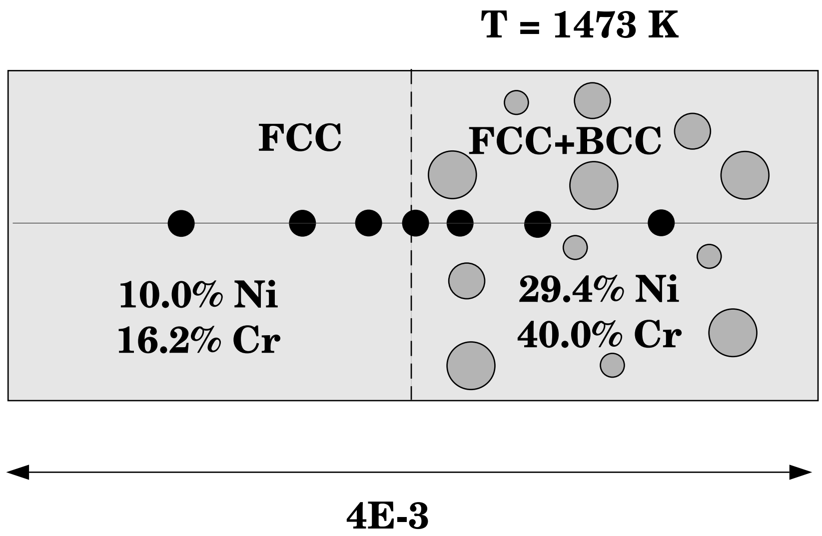



```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example d2a

diffusion couple of Fe-Ni-Cr alloys

(Initially we have a step-profile)



exd2a-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2a\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Diffusion in dispersed systems.
SYS: @@ Setup file for calculating the interdiffusion in a diffusion
SYS: @@ couple between a two-phase (FCC+BCC) and a single-phase (FCC)
SYS: @@ Fe-Ni-Cr alloy.@@ This case is from A. Engström: Scand. J. Met., v. 24,
SYS: @@ 1995, pp.12-20. This simulation can be run with either the DISPERSED
SYS: @@ SYSTEM MODEL or the HOMOGENIZATION MODEL.
SYS: @@ In this example the DISPERSED SYSTEM MODEL is used, which requires
SYS: @@ that the default HOMOGENIZATION MODEL is disabled.
SYS: @@ With the DISPERSED SYSTEM MODEL the command
SYS: @@ ENTER_LABYRINTH_FUNCTION is used to take into account the
SYS: @@ impeding effect of dispersed phases on long-range diffusion.
SYS: @@ For the HOMOGENIZATION MODEL the command
SYS: @@ ENTER_HOMOGENIZATION_FUNCTION should be used.
SYS: @@
SYS:
SYS: @@ exd2_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-sys fe ni cr
 FE                      NI                      CR
   DEFINED
TDB_FEDEMO: rej ph * all
 LIQUID:L                BCC_A2                  CHI_A12
 FCC_A1                  HCP_A3                  LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc,bcc
 FCC_A1                  BCC_A2  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe ni cr
 FE                      NI                      CR
   DEFINED
APP: rej ph * all
 BCC_A2                  FCC_A1  REJECTED
APP: res ph fcc,bcc
 FCC_A1                  BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
      Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
```

```
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1473; * N

DIC>
DIC> @@
DIC> @@ ENTER REGION fer
DIC> @@
DIC> enter-region fer
DIC>
DIC>
DIC> @@
DIC> @@ ENTER DOUBLE-GEOMETRICAL GRID INTO THE REGION
DIC> @@ THIS WILL GIVE A SHORT DISTANCE BETWEEN THE GRIDPOINTS
DIC> @@ IN THE MIDDLE OF THE REGION WHERE THE INITIAL INTERFACE IS
DIC> @@
DIC> enter-grid fer
WIDTH OF REGION /1/: 4e-3
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 200
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.97
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.03093
DIC>
DIC> @@
DIC> @@ ENTER MATRIX PHASE IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act fer matrix fcc
DIC>
DIC> @@
DIC> @@ ENTER THE START COMPOSITION FOR THE MATRIX PHASE FROM FILES
DIC> @@
DIC> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /FCC_A1/: fcc
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr
TYPE /LINEAR/: read d2cr.dat
PROFILE FOR /NI/: ni
TYPE /LINEAR/: read d2ni.dat
DIC>
DIC> @@
DIC> @@ ENTER FERRITE AS SPHEROIDAL PHASE IN THE REGION
DIC> @@ SINCE THE FRACTION OF FERRITE IS SMALL AND THEY APPEAR
DIC> @@ AS ISOLATED PARTICLES WE ENTER FERRITE AS A SPHEROIDAL PHASE
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /FER/: fer
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: bcc
DIC>
DIC> @@
DIC> @@ ENTER  COMPOSITION FOR THE SPHEROIDAL PHASE
DIC> @@ WE USE THE EQUILIBRIUM VALUE
DIC> @@
DIC> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /FCC_A1/: bcc
USE EQUILIBRIUM VALUE /Y/: y
DIC>
DIC>
DIC> @@
DIC> @@ ENTER LABYRINTH FACTOR
DIC> @@ IN THIS CASE THE LOW DIFFUSIVITY PHASE IS THE MATRIX AND THE
DIC> @@ "EFFECTIVE" DIFFUSIVITY IN THE AUSTENITE+FERRITE TWO-PHASE
DIC> @@ REGION IS EXPECTED TO BE HIGHER THAN THE DIFFUSIVITY IN THE
DIC> @@ AUSTENITE.
DIC> enter-lab
REGION NAME : fer
f(T,P,VOLFR,X)= 1+3*(1-volfr)/volfr;
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 720000
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /72000/: 5000
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> set-simulation-condition
NS01A PRINT CONTROL : /0/:
FLUX CORRECTION FACTOR : /1/:
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/:
CHECK INTERFACE POSITION /NO/:
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/:
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/:
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/: 99
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/:
MAX TIMESTEP CHANGE PER TIMESTEP : /2/:
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/:
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/:
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC> @@ AS OF DICTRA VERSION 27 IT IS BY DEFAULT THE "HOMOGENIZATION MODEL"
DIC> @@ THAT IS USED WHEN MULTIPLE PHASES ARE ENTERED IN A SINGLE REGION
DIC> @@ DISABLE THE HOMOGENIZATION MODEL:
DIC> ho n
 HOMOGENIZATION DISABLED
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC>
DIC> save exd2 y
DIC>
```

```
DIC> set-inter
 --OK---
DIC>
```

exd2a-run

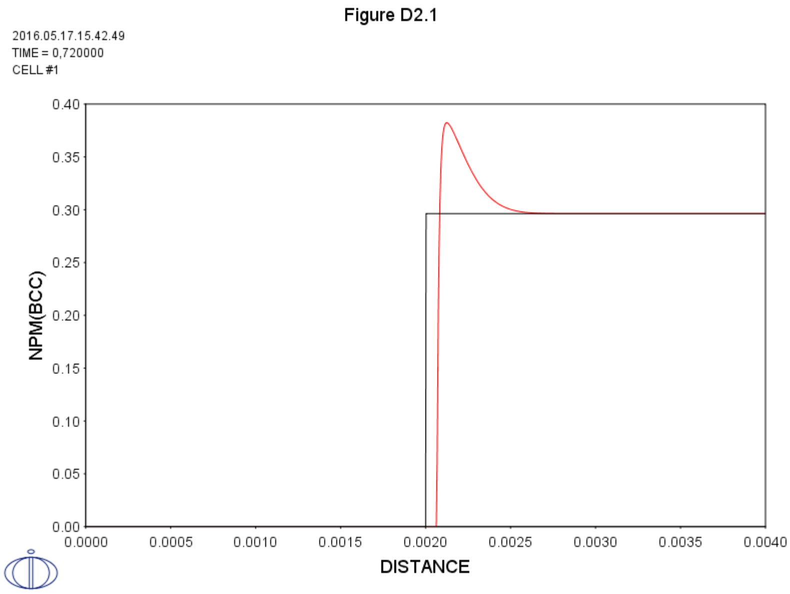
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2a\run.DCM"
DIC>
DIC> @@ exd2_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING THE SIMULATION OF EXAMPLE D2
DIC> @@
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exd2
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  CR = .297842647391914  FE = .517227320517284
                        NI = .184930032090802
 TOTAL SIZE OF SYSTEM:  .004 [m]
  WARNING:BCC_A2 HAS NO VOLUME FRACTION, CREATING ONE




 U-FRACTION IN SYSTEM:  CR = .297842647391914  FE = .517227320517284
                        NI = .184930032090802
 TOTAL SIZE OF SYSTEM:  .004 [m]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297842647356821  FE = .517227320600573
                        NI = .184930032042605
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297842648005432  FE = .517227319061158
                        NI = .18493003293341
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297842676377596  FE = .517227251840505
                        NI = .184930071781899
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   281.72200     DT =   281.32190     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297847760270076  FE = .517216317308818
                        NI = .184935922421106
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   844.36579     DT =   562.64380     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297849615184744  FE = .517212946633451
                        NI = .184937438181805
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   1969.6534     DT =   1125.2876     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297850550731974  FE = .517211426818697
                        NI = .184938022449329
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   4220.2286     DT =   2250.5752     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297851119000641  FE = .51721056932381
                        NI = .184938311675549
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     1  seconds

 TIME =   8721.3789     DT =   4501.1504     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297851494954335  FE = .517210020163941
                        NI = .184938484881724
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   13721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297851732113689  FE = .517209740277051
                        NI = .18493852760926
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   18721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297851883453515  FE = .517209584045522
                        NI = .184938532500963
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   23721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297851977657559  FE = .517209483265964
                        NI = .184938539076477
 TOTAL SIZE OF SYSTEM:  .004 [m]
```

```
 CPU time used in timestep                     0  seconds

TIME =   28721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852029985831  FE = .517209412426181
                       NI = .184938557587988
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   33721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852049766016  FE = .517209361691066
                       NI = .184938588542917
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   38721.379     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852045015483  FE = .517209326359891
                       NI = .184938628624626
TOTAL SIZE OF SYSTEM:  .004 [m]

output ignored...

... output resumed

 CPU time used in timestep                     0  seconds

TIME =   638721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852071392509  FE = .517209093083962
                       NI = .184938835523529
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   643721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852073681655  FE = .517209091807065
                       NI = .18493883451128
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   648721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852074518742  FE = .517209091190478
                       NI = .18493883429078
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   653721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .29785207388002  FE = .517209091158205
                       NI = .184938834961775
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   658721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852071737255  FE = .517209091673262
                       NI = .184938836589483
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     1  seconds

TIME =   663721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852068061676  FE = .517209092719083
                       NI = .184938839219241
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   668721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852062825016  FE = .517209094290058
                       NI = .184938842884927
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   673721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852056000114  FE = .517209096387343
                       NI = .184938847612542
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   678721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852047560686  FE = .517209099016406
                       NI = .184938853422908
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   683721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852037481124  FE = .517209102185676
                       NI = .1849388603332
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   688721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852031720885  FE = .517209104528027
                       NI = .184938863751089
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     1  seconds

TIME =   693721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852035106996  FE = .517209102778487
                       NI = .184938862114517
TOTAL SIZE OF SYSTEM:  .004 [m]

 CPU time used in timestep                     0  seconds

TIME =   698721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
U-FRACTION IN SYSTEM:  CR = .297852041265647  FE = .517209100862462
```

```
                       NI = .184938857871891
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   703721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297852048238976  FE = .517209099115032
                        NI = .184938852645992
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   708721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297852055131669  FE = .517209097666853
                        NI = .184938847201478
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   713721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297852061470695  FE = .517209096580511
                        NI = .184938841948794
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   718721.38     DT =   5000.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297852066979909  FE = .51720909589143
                        NI = .184938837128662
 TOTAL SIZE OF SYSTEM:  .004 [m]

  CPU time used in timestep                     0  seconds

 TIME =   720000.00     DT =   1278.6211     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  CR = .297852071767272  FE = .51720909542939
                        NI = .184938832803338
 TOTAL SIZE OF SYSTEM:  .004 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME    468721.38
 DELETING TIME-RECORD FOR TIME    708721.38
 DELETING TIME-RECORD FOR TIME    713721.38

 KEEPING TIME-RECORD FOR TIME    718721.38
 AND FOR TIME                    720000.00
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

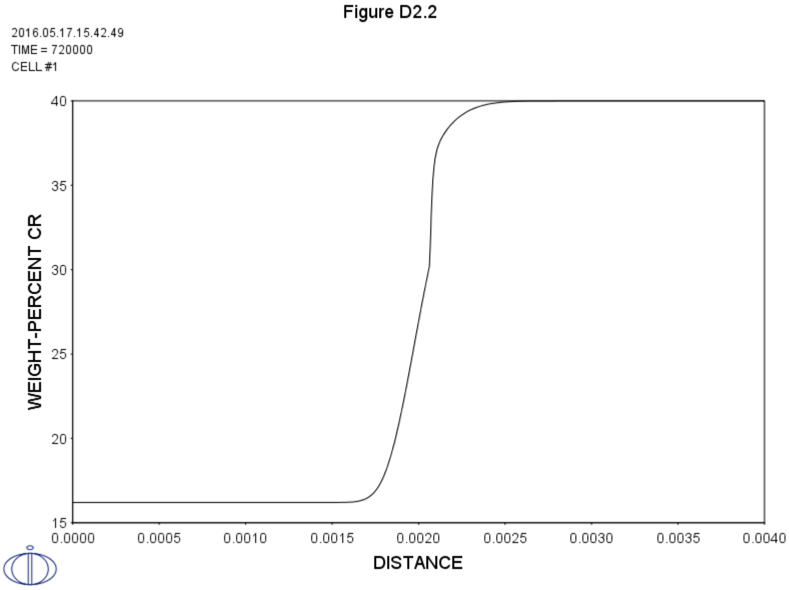
exd2a-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2a\plot.DCM"
DIC>
DIC> @@ exd2_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE D2
DIC> @@
DIC> @@ ENTER THE DICTRA MODULE AND SPECIFY THE STORE-RESULT FILE
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  7.20000E+05
DIC> read  exd2
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ LET US FIRST SEE HOW THE FRACTION OF FERRITE  HAS CHANGED
POST-1: @@ AS A RESULT OF THE DIFFUSION
POST-1: @@
POST-1: s-d-a y npm(bcc)
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-p-c time 0 720000
POST-1: set-tit Figure D2.1
POST-1: pl
```



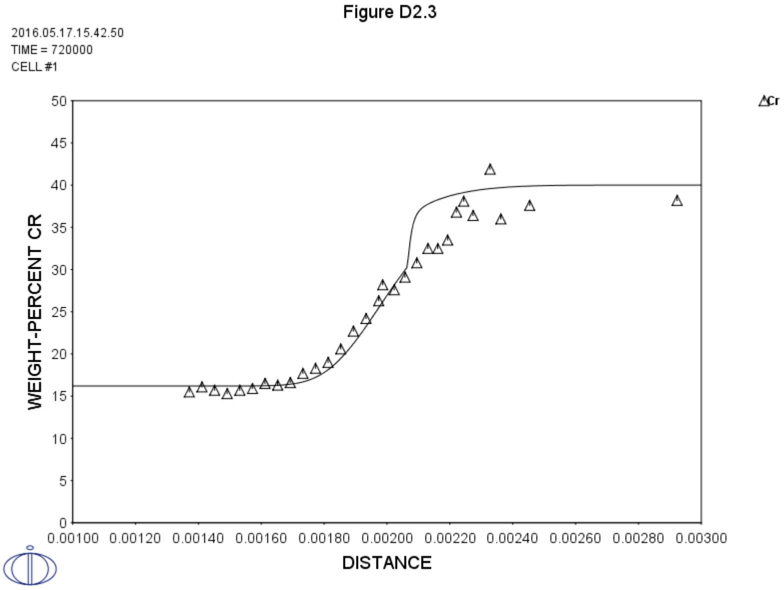

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ LET US NOW PLOT HOW THE AVERAGE CR-CONCENTRATION VARIES WITH DISTANCE
POST-1: @@
POST-1: s-d-a y w-p cr
POST-1: s-p-c time last
POST-1: set-tit Figure D2.2
POST-1: pl
```

Figure D2.2



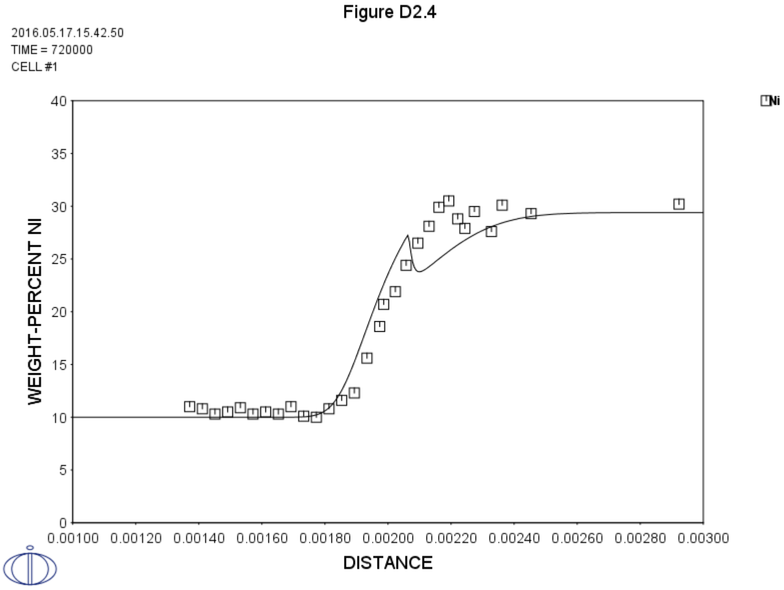

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ WE SELECT A BETTER SCALING AND APPEND EXPERIMENTAL DATA
POST-1: @@
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 4
POST-1:
POST-1: s-s-s y n 0 50
POST-1: s-s-s x n 10e-4 30e-4
POST-1:
POST-1: set-tit Figure D2.3
POST-1: pl
```

Figure D2.3



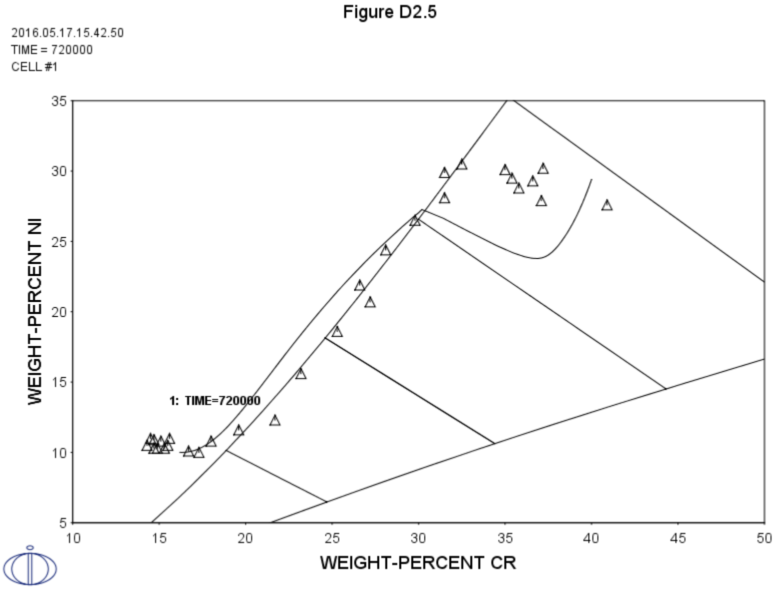

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ NOW WE DO THE SAME FOR NICKEL
POST-1: @@
POST-1:
POST-1: s-d-a y w-p ni
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /1/: 1
DATASET NUMBER(s): /-1/: 5
POST-1:
POST-1: s-s-s x n 10e-4 30e-4
POST-1: s-s-s y n 0 40
POST-1:
POST-1: set-tit Figure D2.4
POST-1: pl
```

Figure D2.4




```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ LET US PLOT THE DIFFUSION PATH FOR THE COUPLE
POST-1: @@ WE APPEND THE TERNARY PHASE-DIAGRAM CALCULATED IN THERMO-CALC
POST-1: @@ AND THE EXPERIMENTAL DATA
POST-1: @@
POST-1: s-d-a x w-p cr
POST-1: s-d-a y w-p ni
POST-1: s-i-v dis gl
POST-1: s-p-c time last
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /1/: 1
DATASET NUMBER(s): /-1/: 6 7 8
POST-1:
POST-1: s-s-s x n 10 50
POST-1: s-s-s y n 5 35
POST-1:
POST-1: set-tit Figure D2.5
POST-1: pl
```

Figure D2.5



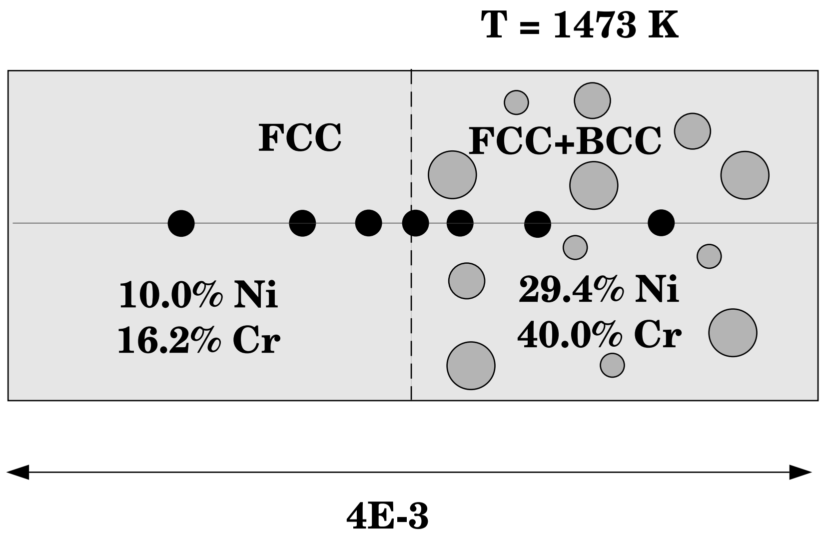

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1: set-interactive
 --OK---
POST-1:
```

Example d2b

diffusion couple of Fe-Ni-Cr alloys

using homogenization model

(Initially we have a step-profile)



exd2b-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2b\setup.DCM"SYS: ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Diffusion in dispersed systems.
SYS: @@ Setup file for calculating the interdiffusion in a diffusion
SYS: @@ couple between a two-phase (FCC+BCC) and a single-phase (FCC)
SYS: @@ Fe-Ni-Cr alloy. This case is from A. Engström: Scand. J. Met.,
SYS: @@ v. 24, 1995, pp.12-20. This simulation can be run with either
SYS: @@ the DISPERSED SYSTEM MODEL or the HOMOGENIZATION MODEL.
SYS: @@ Here the default HOMOGENIZATION MODEL is used and then
SYS: @@ ENTER_HOMOGENIZATION_FUNCTION should be used instead of
SYS: @@ ENTER_LABYRINTH_FUNCTION.
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                     /-  DEFINED
TDB_FEDEMO: def-sys fe ni cr
 FE                     NI                     CR
   DEFINED
TDB_FEDEMO: rej ph * all
 LIQUID:L               BCC_A2                 CHI_A12
 FCC_A1                 HCP_A3                 LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: res ph fcc,bcc
 FCC_A1                 BCC_A2  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
     Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
     (1986); CR-FE'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe ni cr
 FE                     NI                     CR
   DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph fcc,bcc
 FCC_A1                 BCC_A2  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
     Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
     -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
     diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
     in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 1473; * N
```

```
DIC>
DIC> @@
DIC> @@ ENTER REGION fer
DIC> @@
DIC> enter-region fer
DIC>
DIC>
DIC> @@
DIC> @@ ENTER DOUBLE-GEOMETRICAL GRID INTO THE REGION
DIC> @@ THIS WILL GIVE A SHORT DISTANCE BETWEEN THE GRIDPOINTS
DIC> @@ IN THE MIDDLE OF THE REGION WHERE THE INITIAL INTERFACE IS
DIC> @@
DIC> enter-grid fer
WIDTH OF REGION /1/: 4e-3
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 200
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.97
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.03093
DIC>
DIC> @@
DIC> @@ ENTER MATRIX PHASE IN THE REGION
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act fer matrix fcc
DIC>
DIC> @@
DIC> @@ ENTER THE START COMPOSITION FOR THE MATRIX PHASE FROM FILES
DIC> @@
DIC> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /FCC_A1/: fcc
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /CR/: cr
TYPE /LINEAR/: read d2cr.dat
PROFILE FOR /NI/: ni
TYPE /LINEAR/: read d2ni.dat
DIC>
DIC> @@
DIC> @@ ENTER FERRITE AS SPHEROIDAL PHASE IN THE REGION
DIC> @@ SINCE THE FRACTION OF FERRITE IS SMALL AND THEY APPEAR
DIC> @@ AS ISOLATED PARTICLES WE ENTER FERRITE AS A SPHEROIDAL PHASE
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /FER/: fer
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: bcc
DIC>
DIC> @@
DIC> @@ ENTER  COMPOSITION FOR THE SPHEROIDAL PHASE
DIC> @@ WE USE THE EQUILIBRIUM VALUE
DIC> @@
DIC> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /FCC_A1/: bcc
USE EQUILIBRIUM VALUE /Y/: y
DIC>
DIC> @@ SELECT A HOMOGENIZATION FUNCTION
DIC> @@ IN THIS CASE THE LOWER HASHIN-SHTRIKMAN BOUND
DIC> en-ho 1
 SELECTED FUNCTION IS HASHIN-SHTRIKMAN BOUND: GENERAL LOWER
DIC>
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 720000
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /72000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> set-simulation-condition
NS01A PRINT CONTROL : /0/: 0
FLUX CORRECTION FACTOR : /1/: 1
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/: 2
CHECK INTERFACE POSITION /NO/: n
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/: act
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/: y
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/: 99
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: 1
MAX TIMESTEP CHANGE PER TIMESTEP : /2/: 2
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/: n
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/: y
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/: n
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC>
DIC> save exd2 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exd2b-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2b\run.DCM"
DIC>
DIC> @@ exd2_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING THE SIMULATION OF EXAMPLE D2
DIC> @@
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exd2
 OK
DIC> sim
 STARTING SIMULATION USING HOMOGENIZATION MODEL
 ----------------------------------------------
 Saving results
 ------------------------------------------------
  WARNING:BCC_A2 HAS NO VOLUME FRACTION, CREATING ONE
 Starting time-step t0=  0.000000000000000E+000  dt=  1.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  0.000000000000000E+000  dt=  1.000000000000000E-007
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=  1.000000000000000E-007  dt=  2.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632081E-004  /   0.184930032090802

 Time-step converged t0=  1.000000000000000E-007  dt=  2.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  3.000000000000000E-007  dt=  4.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632081E-004  /   0.184930032090802

 Time-step converged t0=  3.000000000000000E-007  dt=  4.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  7.000000000000000E-007  dt=  8.000000000000000E-007
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069136E-003  /   0.517227320517284
 NI   7.397201283632081E-004  /   0.184930032090802

 Time-step converged t0=  7.000000000000000E-007  dt=  8.000000000000000E-007
 ------------------------------------------------
 Starting time-step t0=  1.500000000000000E-006  dt=  1.600000000000000E-006
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069136E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  1.500000000000000E-006  dt=  1.600000000000000E-006
 ------------------------------------------------
 Starting time-step t0=  3.100000000000000E-006  dt=  3.200000000000000E-006
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069136E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  3.100000000000000E-006  dt=  3.200000000000000E-006
 ------------------------------------------------
 Starting time-step t0=  6.300000000000000E-006  dt=  6.400000000000000E-006
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  6.300000000000000E-006  dt=  6.400000000000000E-006
 ------------------------------------------------
 Starting time-step t0=  1.270000000000000E-005  dt=  1.280000000000000E-005
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  1.270000000000000E-005  dt=  1.280000000000000E-005
 ------------------------------------------------
```

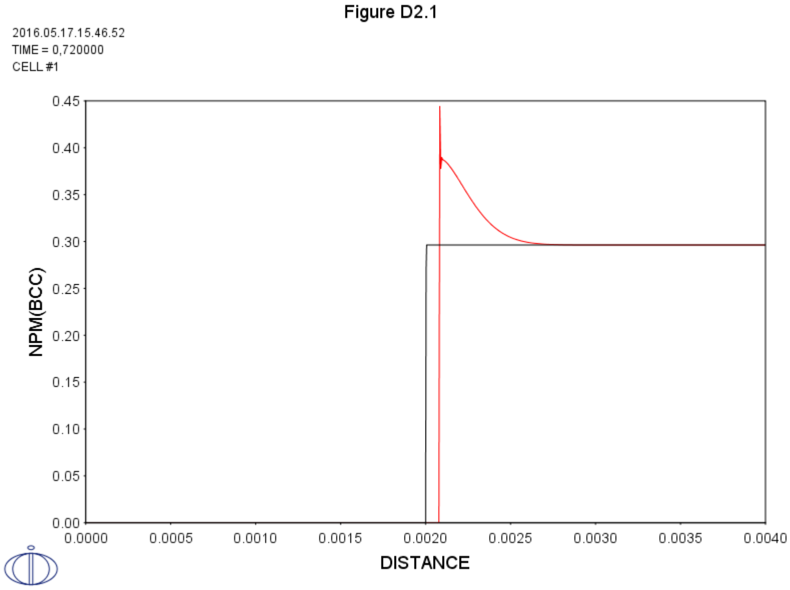
```
 Starting time-step t0=  2.550000000000000E-005  dt=  2.560000000000000E-005
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=  2.550000000000000E-005  dt=  2.560000000000000E-005
 ------------------------------------------------
 Starting time-step t0=  5.110000000000000E-005  dt=  5.120000000000000E-005
 Stage 1
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632081E-004  /   0.184930032090802


output ignored...


... output resumed

 Iteration #                      9  sum residual **2:    10.5949654297738
 Iteration #                     10  sum residual **2:    1.83745452863023
 Iteration #                     11  sum residual **2:   7.956889090290342E-002
 Stage 2
 Iteration #                      1  sum residual **2:    9.34475760894998
 Iteration #                      2  sum residual **2:   3.550793563965303E-002
 Stage 3
 Iteration #                      1  sum residual **2:    17.4169533233740
 Iteration #                      2  sum residual **2:    1.86804711772457
 Moles of elements / mole fractions in system:
 CR   1.191370589567654E-003  /   0.297842647391914
 FE   2.068909282069136E-003  /   0.517227320517284
 NI   7.397201283632080E-004  /   0.184930032090802

 Time-step converged t0=   655895.193190300       dt=   27487.7906944000
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=   683382.983884700       dt=   36617.0161153004
 Stage 1
 Iteration #                      1  sum residual **2:    1304107.30616369
 Iteration #                      2  sum residual **2:    1236219.56886241
 Iteration #                      3  sum residual **2:    1214126.05249209
 Iteration #                      4  sum residual **2:    1078718.61217706
 Iteration #                      5  sum residual **2:    7187.45775886248
 Iteration #                      6  sum residual **2:    3164.09378781401
 Iteration #                      7  sum residual **2:    882.321596874999
 Iteration #                      8  sum residual **2:    219.235531277492
 Iteration #                      9  sum residual **2:    89.0905057793363
 Iteration #                     10  sum residual **2:    22.2501116985529
 Iteration #                     11  sum residual **2:    9.34910757739606
 Iteration #                     12  sum residual **2:   9.953930560902867E-002
 Stage 2
 Iteration #                      1  sum residual **2:    92.6158974742011
 Iteration #                      2  sum residual **2:    48.2305436039771
 Iteration #                      3  sum residual **2:    26.3122645988294
 Iteration #                      4  sum residual **2:   1.423814082125506E-002
 Stage 3
 Iteration #                      1  sum residual **2:    286.706836989980
 Iteration #                      2  sum residual **2:    85.8053529583884
 Iteration #                      3  sum residual **2:   0.992772364621211
 Iteration #                      4  sum residual **2:   7.387609533560401E-002
 Moles of elements / mole fractions in system:
 CR   1.191370589567655E-003  /   0.297842647391914
 FE   2.068909282069137E-003  /   0.517227320517284
 NI   7.397201283632082E-004  /   0.184930032090802

 Time-step converged t0=   683382.983884700       dt=   36617.0161153004
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME    805.30637
 DELETING TIME-RECORD FOR TIME    1556.9256
 DELETING TIME-RECORD FOR TIME    2415.9191
 DELETING TIME-RECORD FOR TIME    3489.6609
 DELETING TIME-RECORD FOR TIME    4348.6544
 DELETING TIME-RECORD FOR TIME    5207.6478
 DELETING TIME-RECORD FOR TIME    6496.1380
 DELETING TIME-RECORD FOR TIME    7355.1315
 DELETING TIME-RECORD FOR TIME    8214.1250
 DELETING TIME-RECORD FOR TIME    9073.1184
 DELETING TIME-RECORD FOR TIME    9932.1119
 DELETING TIME-RECORD FOR TIME    10791.105
 DELETING TIME-RECORD FOR TIME    11650.099
 DELETING TIME-RECORD FOR TIME    13368.086
 DELETING TIME-RECORD FOR TIME    15086.073
 DELETING TIME-RECORD FOR TIME    16804.060
 DELETING TIME-RECORD FOR TIME    18522.046
 DELETING TIME-RECORD FOR TIME    20240.033
 DELETING TIME-RECORD FOR TIME    21958.020
 DELETING TIME-RECORD FOR TIME    25393.994
 DELETING TIME-RECORD FOR TIME    28829.968
 DELETING TIME-RECORD FOR TIME    30547.955
 DELETING TIME-RECORD FOR TIME    33983.929
 DELETING TIME-RECORD FOR TIME    37419.903
 DELETING TIME-RECORD FOR TIME    40855.876
 DELETING TIME-RECORD FOR TIME    44291.850
 DELETING TIME-RECORD FOR TIME    51163.798
 DELETING TIME-RECORD FOR TIME    58035.746
 DELETING TIME-RECORD FOR TIME    64907.693
 DELETING TIME-RECORD FOR TIME    71779.641
 DELETING TIME-RECORD FOR TIME    78651.589
 DELETING TIME-RECORD FOR TIME    85523.536
```

```
 DELETING TIME-RECORD FOR TIME    88959.510
 DELETING TIME-RECORD FOR TIME    95831.458
 DELETING TIME-RECORD FOR TIME    109575.35
 DELETING TIME-RECORD FOR TIME    116447.30
 DELETING TIME-RECORD FOR TIME    130191.20
 DELETING TIME-RECORD FOR TIME    143935.09
 DELETING TIME-RECORD FOR TIME    147371.07
 DELETING TIME-RECORD FOR TIME    150807.04
 DELETING TIME-RECORD FOR TIME    154243.01
 DELETING TIME-RECORD FOR TIME    161114.96
 DELETING TIME-RECORD FOR TIME    174858.86
 DELETING TIME-RECORD FOR TIME    181730.80
 DELETING TIME-RECORD FOR TIME    188602.75
 DELETING TIME-RECORD FOR TIME    195474.70
 DELETING TIME-RECORD FOR TIME    209218.59
 DELETING TIME-RECORD FOR TIME    222962.49
 DELETING TIME-RECORD FOR TIME    229834.44
 DELETING TIME-RECORD FOR TIME    243578.33
 DELETING TIME-RECORD FOR TIME    271066.12
 DELETING TIME-RECORD FOR TIME    277938.07
 DELETING TIME-RECORD FOR TIME    291681.97
 DELETING TIME-RECORD FOR TIME    319169.76
 DELETING TIME-RECORD FOR TIME    332913.65
 DELETING TIME-RECORD FOR TIME    360401.44
 DELETING TIME-RECORD FOR TIME    387889.23
 DELETING TIME-RECORD FOR TIME    394761.18
 DELETING TIME-RECORD FOR TIME    408505.08
 DELETING TIME-RECORD FOR TIME    422248.97
 DELETING TIME-RECORD FOR TIME    449736.76
 DELETING TIME-RECORD FOR TIME    463480.66
 DELETING TIME-RECORD FOR TIME    470352.61
 DELETING TIME-RECORD FOR TIME    477224.55
 DELETING TIME-RECORD FOR TIME    490968.45
 DELETING TIME-RECORD FOR TIME    518456.24
 DELETING TIME-RECORD FOR TIME    532200.14
 DELETING TIME-RECORD FOR TIME    545944.03
 DELETING TIME-RECORD FOR TIME    573431.82
 DELETING TIME-RECORD FOR TIME    628407.40
 DELETING TIME-RECORD FOR TIME    642151.30
 DELETING TIME-RECORD FOR TIME    655895.19

 KEEPING TIME-RECORD FOR TIME    683382.98
 AND FOR TIME                    720000.00
 WORKSPACE RECLAIMED
 ------------------------------------------------
 INTERPOLATION SCHEME USED THIS FRACTION OF
 THE ALLOCATED MEMORY:   7.642376596533051E-002
 EFFICIENCY FACTOR:    36.9740639056020
 ------------------------------------------------
 DEALLOCATING
 ------------------------------------------------
DIC>
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exd2b-plot

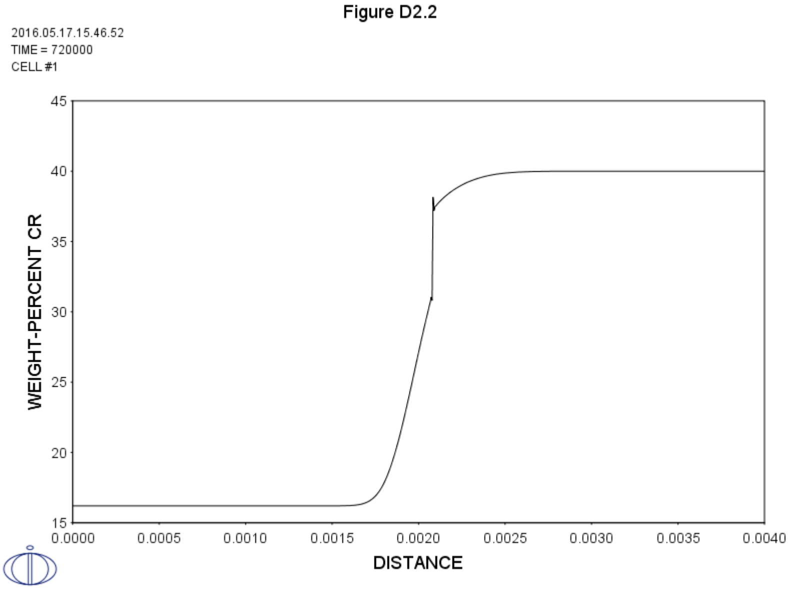
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd2b\plot.DCM"DIC>
DIC>
DIC> @@ exd2_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE D2
DIC> @@
DIC> @@ ENTER THE DICTRA MODULE AND SPECIFY THE STORE-RESULT FILE
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  7.20000E+05
DIC> read  exd2
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ LET US FIRST SEE HOW THE FRACTION OF FERRITE  HAS CHANGED
POST-1: @@ AS A RESULT OF THE DIFFUSION
POST-1: @@
POST-1: s-d-a y npm(bcc)
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-p-c time 0 720000
POST-1: set-tit Figure D2.1
POST-1: pl
```

Figure D2.1



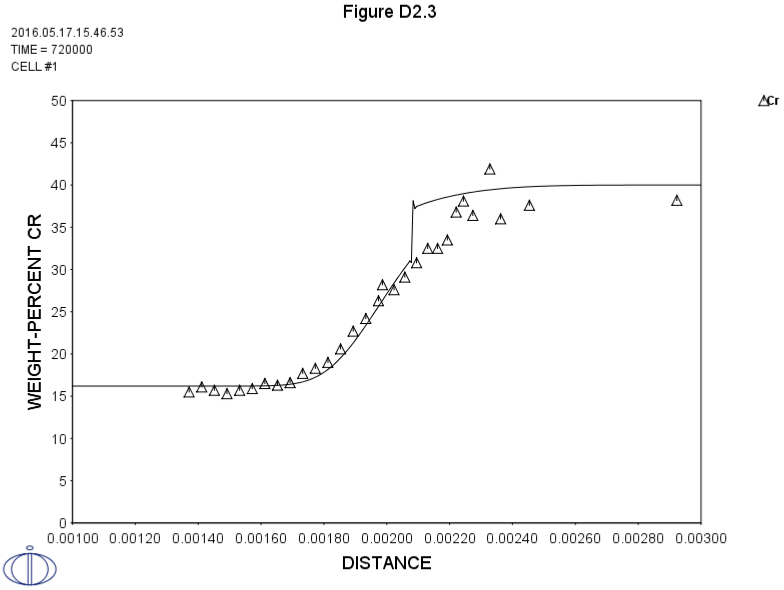

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ LET US NOW PLOT HOW THE AVERAGE CR-CONCENTRATION VARIES WITH DISTANCE
POST-1: @@
POST-1: s-d-a y w-p cr
POST-1: s-p-c time last
POST-1: set-tit Figure D2.2
POST-1: pl
```

Figure D2.2



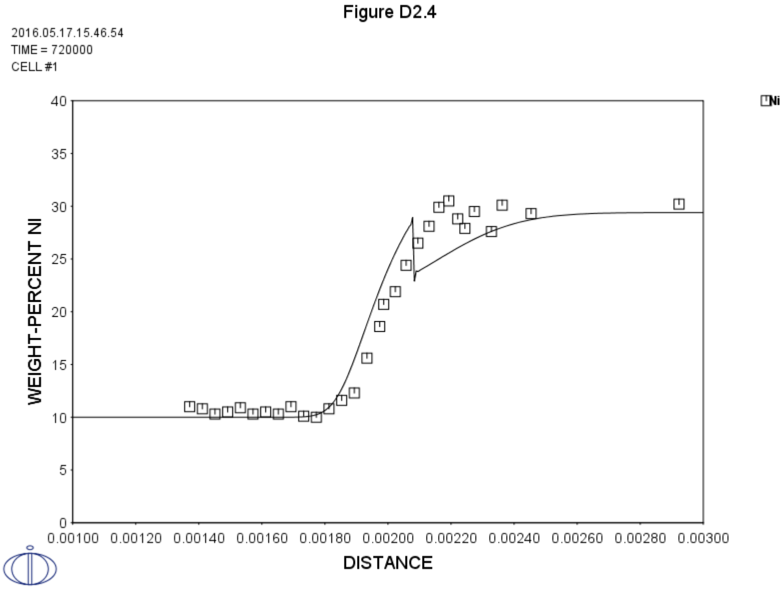

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ WE SELECT A BETTER SCALING AND APPEND EXPERIMENTAL DATA
POST-1: @@
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 4
POST-1:
POST-1: s-s-s y n 0 50
POST-1: s-s-s x n 10e-4 30e-4
POST-1:
POST-1: set-tit Figure D2.3
POST-1: pl
```

Figure D2.3



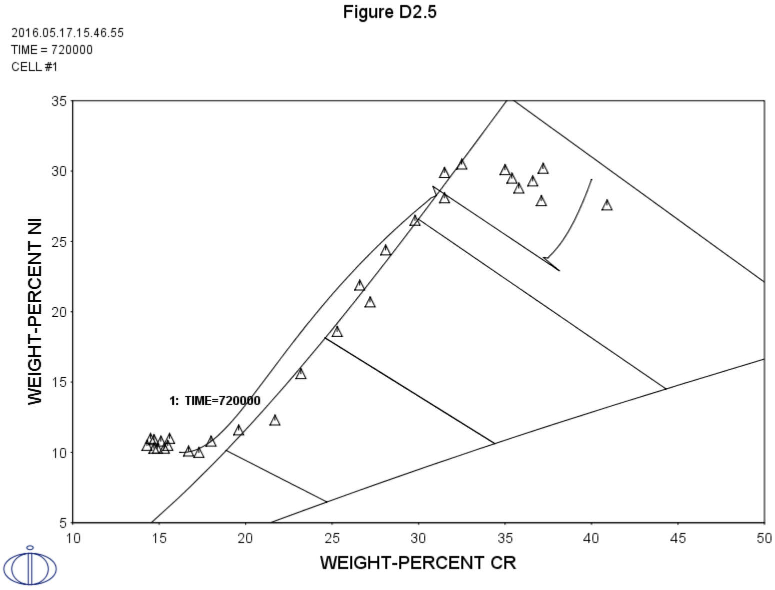

```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ NOW WE DO THE SAME FOR NICKEL
POST-1: @@
POST-1:
POST-1: s-d-a y w-p ni
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /1/: 1
DATASET NUMBER(s): /-1/: 5
POST-1:
POST-1: s-s-s x n 10e-4 30e-4
POST-1: s-s-s y n 0 40
POST-1:
POST-1: set-tit Figure D2.4
POST-1: pl
```

Figure D2.4




```
POST-1:
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ LET US PLOT THE DIFFUSION PATH FOR THE COUPLE
POST-1: @@ WE APPEND THE TERNARY PHASE-DIAGRAM CALCULATED IN THERMO-CALC
POST-1: @@ AND THE EXPERIMENTAL DATA
POST-1: @@
POST-1: s-d-a x w-p cr
POST-1: s-d-a y w-p ni
POST-1: s-i-v dis gl
POST-1: s-p-c time last
POST-1:
POST-1: app y exd2.exp
PROLOGUE NUMBER: /1/: 1
DATASET NUMBER(s): /-1/: 6 7 8
POST-1:
POST-1: s-s-s x n 10 50
POST-1: s-s-s y n 5 35
POST-1:
POST-1: set-tit Figure D2.5
POST-1: pl
```

Figure D2.5

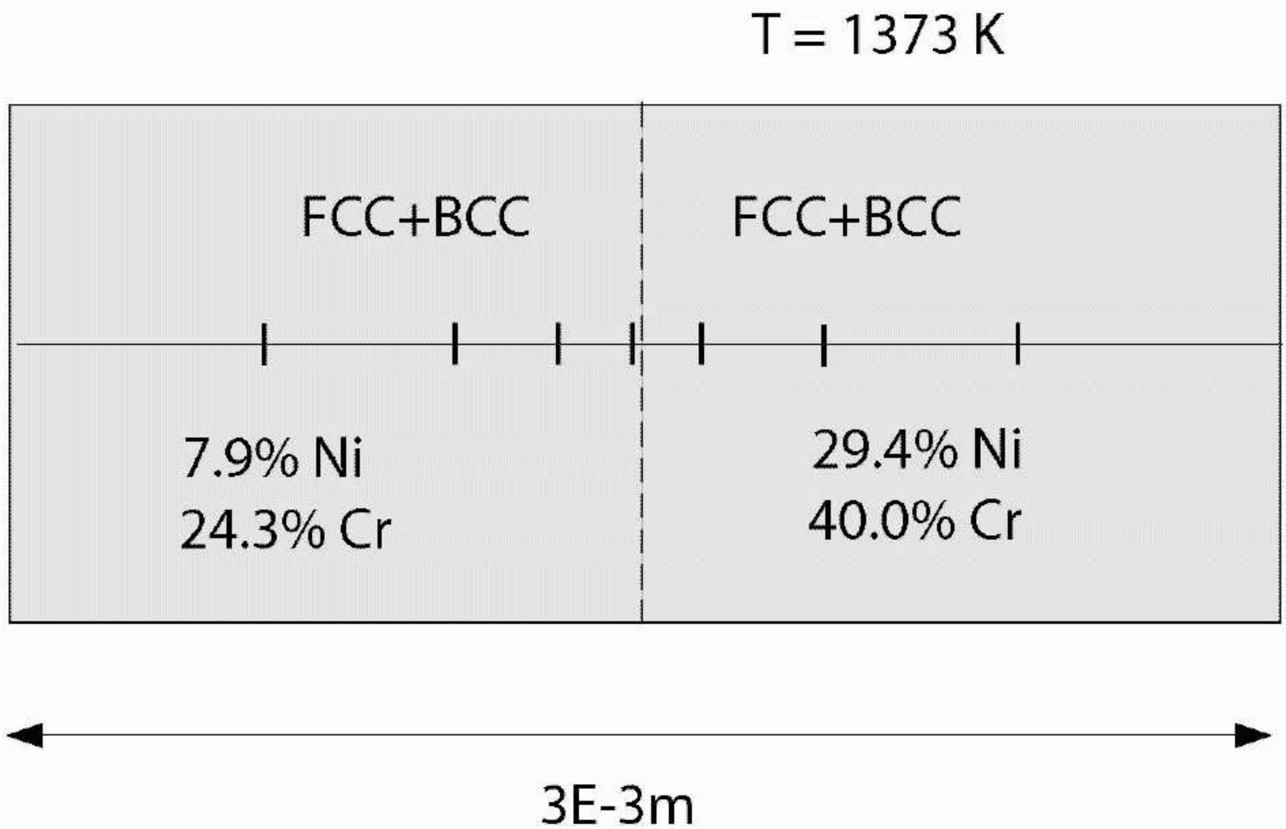



```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1: set-interactive
 --OK---
POST-1:
```

Example d3

diffusion couple of Fe-Ni-Cr alloys

Using the new model



exd3-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd3\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Diffusion in dispersed systems.
SYS: @@ This example shows the use of the homogenization model.
SYS: @@ It is taken from H. Larsson and A. Engström, Acta. Mater. v.54
SYS: @@ (2006), pp. 2431-2439. Experimental data from A. Engström,
SYS: @@ Scand J Metall, v.243 (1995), p.12.
SYS: @@ The homogenization model can be used for multiphase simulations
SYS: @@ like the dispersed system model, but unlike the dispersed system model
SYS: @@ there is no need to have a single continuous matrix phase and, furthermore,
SYS: @@ there is no need to limit the size of time-steps.
SYS: @@ The set-up is performed in the same manner as for the dispersed system
SYS: @@ model, which means that a certain phase is entered as the matrix phase
SYS: @@ and the other phases are entered as spheroidal, but the choice of matrix
SYS: @@ phase will not affect the simulation.
SYS: @@
SYS:
SYS: @@ exd3_setup.DCM
SYS:
SYS:
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                       /-  DEFINED
TDB_FEDEMO: def-sys fe cr ni
 FE                       CR                       NI
   DEFINED
TDB_FEDEMO: rej-ph *
 LIQUID:L                 BCC_A2                   CHI_A12
 FCC_A1                   HCP_A3                   LAVES_PHASE_C14
 SIGMA  REJECTED
TDB_FEDEMO: rest-ph bcc,fcc
 BCC_A2                   FCC_A1  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'J-O. Andersson and B. Sundman, Calphad, 11 (1987), 83-92; TRITA 0270
      (1986); CR-FE'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Dinsdale and T. Chart, MTDS NPL, Unpublished work (1986); CR-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe cr ni
 FE                       CR                       NI
   DEFINED
APP: rej-ph *
 BCC_A2                   FCC_A1  REJECTED
APP: rest-ph bcc,fcc
 BCC_A2                   FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Fe diffusion fcc Cr-Fe'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Cr and Ni diffusion fcc Cr-Ni'
  'B. Jonsson: Z. Metallkunde 86(1995)686-692; Cr, Fe and Ni diffusion fcc
      Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: go -m
 NO TIME STEP DEFINED
DIC>
DIC> set-cond glob T 0 1373.15; * N

DIC>
DIC> ent-geo 0
DIC>
DIC> ent-reg
REGION NAME : fecrni
```

```
DIC>
DIC> ent-grid
REGION NAME : /FECRNI/: fecrni
WIDTH OF REGION /1/: 3e-3
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 60
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.85
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.15
DIC>
DIC> ent-ph
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /FECRNI/: fecrni
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC> ent-ph
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /FECRNI/: fecrni
PHASE TYPE /MATRIX/: sph
PHASE NAME: /NONE/: bcc
DIC>
DIC> ent-comp
REGION NAME : /FECRNI/: fecrni
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: m-f
PROFILE FOR /CR/: cr
TYPE /LINEAR/: read cr.dat
PROFILE FOR /NI/: ni
TYPE /LINEAR/: read ni.dat
DIC>
DIC> ent-comp
REGION NAME : /FECRNI/: fecrni
PHASE NAME: /FCC_A1/: bcc
USE EQUILIBRIUM VALUE /Y/: y
DIC>
DIC> se-si-ti
END TIME FOR INTEGRATION /.1/: 3.6e5
AUTOMATIC TIMESTEP CONTROL /YES/: yes
MAX TIMESTEP DURING INTEGRATION /36000/: 3.6e4
INITIAL TIMESTEP : /1E-07/: 1
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/: 1e-7
DIC>
DIC> @@ Simulations will run faster if results are not saved every
DIC> @@ time-step
DIC> s-s-c
NS01A PRINT CONTROL : /0/: 0
FLUX CORRECTION FACTOR : /1/: 1
NUMBER OF DELTA TIMESTEPS IN CALLING MULDIF: /2/: 2
CHECK INTERFACE POSITION /NO/: n
VARY POTENTIALS OR ACTIVITIES : /ACTIVITIES/: act
ALLOW AUTOMATIC SWITCHING OF VARYING ELEMENT : /YES/: y
SAVE WORKSPACE ON FILE (YES,NO,0-999) /YES/: 99
DEGREE OF IMPLICITY WHEN INTEGRATING PDEs (0 -> 0.5 -> 1): /.5/: .5
MAX TIMESTEP CHANGE PER TIMESTEP : /2/: 2
USE FORCED STARTING VALUES IN EQUILIBRIUM CALCULATION /NO/: n
ALWAYS CALCULATE STIFFNES MATRIX IN MULDIF /YES/: y
CALCULATE RESIDUAL FOR DEPENDENT COMPONENT /NO/:
DIC>
DIC> @@ There are a certain number of options available for the
DIC> @@ homogenization model. In particular, an interpolation
DIC> @@ scheme has been implemented that may drastically reduce
DIC> @@ simulation times. Here, however, default settings are
DIC> @@ chosen and the interpolation scheme is then turned off.
DIC> homogen yes yes
 INFO: HOMOGENIZATION MODEL ENABLED
DIC>
DIC> @@ There are a large number of homogenization functions
DIC> @@ available. These determine how the average kinetics
DIC> @@ of the multiphase mixture is evaluated. For this example
DIC> @@ the General lower Hashin-Shtrikman bound is a good choice.
DIC> en-ho
ENTER HOMOGENIZATION FUNCTION # /5/: 1
 SELECTED FUNCTION IS HASHIN-SHTRIKMAN BOUND: GENERAL LOWER
DIC>
DIC>
DIC>
DIC> save exd3 Y
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exd3-run

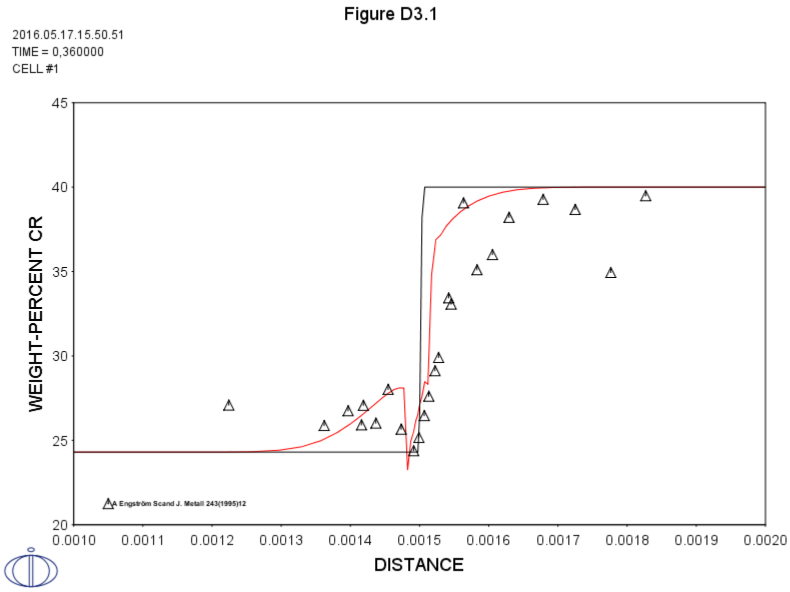
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd3\run.DCM"
DIC>
DIC> @@ exd3_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exd3
 OK
DIC>
DIC> sim
 STARTING SIMULATION USING HOMOGENIZATION MODEL
 ----------------------------------------------
 Saving results
 ------------------------------------------------
  WARNING:BCC_A2 HAS NO VOLUME FRACTION, CREATING ONE
 Starting time-step t0=  0.000000000000000E+000  dt=   1.00000000000000
 Stage 1
 Evaluating Jacobian
 Iteration #                     1  sum residual **2:   9.120180692461437E-004
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=  0.000000000000000E+000  dt=   1.00000000000000
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=   1.00000000000000       dt=   2.00000000000000
 Stage 1
 Iteration #                     1  sum residual **2:   4.906132674490165E-004
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088493E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   1.00000000000000       dt=   2.00000000000000
 ------------------------------------------------
 Starting time-step t0=   3.00000000000000       dt=   4.00000000000000
 Stage 1
 Iteration #                     1  sum residual **2:   3.518450668248791E-004
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   3.00000000000000       dt=   4.00000000000000
 ------------------------------------------------
 Starting time-step t0=   7.00000000000000       dt=   8.00000000000000
 Stage 1
 Iteration #                     1  sum residual **2:   3.227494704476395E-003
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   7.00000000000000       dt=   8.00000000000000
 ------------------------------------------------
 Starting time-step t0=   15.0000000000000       dt=   16.0000000000000
 Stage 1
 Iteration #                     1  sum residual **2:   0.824146500821795
 Stage 2
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   15.0000000000000       dt=   16.0000000000000
 ------------------------------------------------
 Starting time-step t0=   31.0000000000000       dt=   32.0000000000000
 Stage 1
 Iteration #                     1  sum residual **2:    22.4929239945914
 Iteration #                     2  sum residual **2:   4.124859032316958E-003
 Stage 2
 Iteration #                     1  sum residual **2:   2.468606119379819E-003
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088493E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   31.0000000000000       dt=   32.0000000000000
 ------------------------------------------------
 Starting time-step t0=   63.0000000000000       dt=   64.0000000000000
 Stage 1
 Iteration #                     1  sum residual **2:    318.007588765841
 Iteration #                     2  sum residual **2:   1.143675460340569E-002
 Stage 2
 Iteration #                     1  sum residual **2:   1.198981658779452E-003
 Stage 3
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088493E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   63.0000000000000       dt=   64.0000000000000
 ------------------------------------------------
 Starting time-step t0=   127.000000000000       dt=   128.000000000000
 Stage 1
```

```
 Iteration #                        1  sum residual **2:    3329.18606777667
 Iteration #                        2  sum residual **2:    7.78905658243195
 Iteration #                        3  sum residual **2:   3.277409708192316E-003
 Stage 2
 Iteration #                        1  sum residual **2:   0.131821299564790
 Stage 3
 Iteration #                        1  sum residual **2:   2.021182073329573E-003
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   127.000000000000       dt=   128.000000000000
 ------------------------------------------------
 Starting time-step t0=   255.000000000000       dt=   256.000000000000
 Stage 1
 Iteration #                        1  sum residual **2:    24934.0181206745
 Iteration #                        2  sum residual **2:    200.021723882755
 Iteration #                        3  sum residual **2:    2.78303386129339
 Iteration #                        4  sum residual **2:   1.436363601918922E-003
 Stage 2
 Iteration #                        1  sum residual **2:    6.61692545357813
 Iteration #                        2  sum residual **2:   1.134594313711652E-003


output ignored...



... output resumed

 Starting time-step t0=   311295.000000000       dt=   16384.0000000000
 Stage 1
 Evaluating Jacobian
 Iteration #                        1  sum residual **2:    5.04579235754691
 Iteration #                        2  sum residual **2:   7.144338116212557E-003
 Stage 2
 Iteration #                        1  sum residual **2:    6340.13239260283
 Iteration #                        2  sum residual **2:    1343.81814122525
 Iteration #                        3  sum residual **2:    282.612831519302
 Iteration #                        4  sum residual **2:    57.9695537308309
 Iteration #                        5  sum residual **2:    11.0412213585912
 Iteration #                        6  sum residual **2:    1.55155584115148
 Stage 3
 Iteration #                        1  sum residual **2:    19400.5990898574
 Iteration #                        2  sum residual **2:    4113.03056823437
 Iteration #                        3  sum residual **2:    869.379099128735
 Iteration #                        4  sum residual **2:    181.960139720020
 Iteration #                        5  sum residual **2:    36.7927001776137
 Iteration #                        6  sum residual **2:    6.72835802136405
 Iteration #                        7  sum residual **2:   0.716574642299992
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   311295.000000000       dt=   16384.0000000000
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=   327679.000000000       dt=   16384.0000000000
 Stage 1
 Iteration #                        1  sum residual **2:    176030.691861265
 Iteration #                        2  sum residual **2:    24035.0649731424
 Iteration #                        3  sum residual **2:    7124.61675258476
 Iteration #                        4  sum residual **2:    564.586194990616
 Iteration #                        5  sum residual **2:    105.115302482051
 Iteration #                        6  sum residual **2:   4.866832143001540E-002
 Stage 2
 Iteration #                        1  sum residual **2:    11001.3819366730
 Iteration #                        2  sum residual **2:    3019.32450671631
 Iteration #                        3  sum residual **2:    859.084477980905
 Iteration #                        4  sum residual **2:    234.781844941611
 Iteration #                        5  sum residual **2:    64.5946850350146
 Iteration #                        6  sum residual **2:    16.1027456011829
 Iteration #                        7  sum residual **2:    3.41754650996110
 Iteration #                        8  sum residual **2:   0.302291652033423
 Stage 3
 Iteration #                        1  sum residual **2:    22504.2645057328
 Iteration #                        2  sum residual **2:    4840.61817319829
 Iteration #                        3  sum residual **2:    1078.72054371367
 Iteration #                        4  sum residual **2:    261.696656725679
 Iteration #                        5  sum residual **2:    73.6212132204934
 Iteration #                        6  sum residual **2:    19.3120823146564
 Iteration #                        7  sum residual **2:    4.40540030018689
 Iteration #                        8  sum residual **2:   0.453553215480303
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088491E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   327679.000000000       dt=   16384.0000000000
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 Starting time-step t0=   344063.000000000       dt=   15937.0000000000
 Stage 1
 Iteration #                        1  sum residual **2:    10940.8312332506
 Iteration #                        2  sum residual **2:    153.143340937849
 Iteration #                        3  sum residual **2:    1.50494994979681
 Iteration #                        4  sum residual **2:   9.470986838788640E-003
 Stage 2
 Iteration #                        1  sum residual **2:    7736.77956338268
 Iteration #                        2  sum residual **2:    1725.26553495295
 Iteration #                        3  sum residual **2:    387.821364315184
 Iteration #                        4  sum residual **2:    86.8309516842719
 Iteration #                        5  sum residual **2:    19.5473752227189
 Iteration #                        6  sum residual **2:    4.18104706213747
 Iteration #                        7  sum residual **2:   0.407886772520074
 Stage 3
 Iteration #                        1  sum residual **2:    23690.1243298681
 Iteration #                        2  sum residual **2:    5268.12545317751
 Iteration #                        3  sum residual **2:    1169.35333919327
 Iteration #                        4  sum residual **2:    257.568314827242
 Iteration #                        5  sum residual **2:    55.4483217806892
 Iteration #                        6  sum residual **2:    11.0880710980353
```

```
 Iteration #                      7  sum residual **2:    1.68207940209971
 Moles of elements / mole fractions in system:
 CR   1.020229808644575E-003  /   0.340076602881525
 FE   1.469790221088492E-003  /   0.489930073696164
 NI   5.099799702669337E-004  /   0.169993323422311

 Time-step converged t0=   344063.000000000       dt=   15937.0000000000
 ------------------------------------------------
 Saving results
 ------------------------------------------------
 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   1.00000000
 DELETING TIME-RECORD FOR TIME    511.00000
 DELETING TIME-RECORD FOR TIME    1023.0000
 DELETING TIME-RECORD FOR TIME    1535.0000
 DELETING TIME-RECORD FOR TIME    2047.0000
 DELETING TIME-RECORD FOR TIME    3071.0000
 DELETING TIME-RECORD FOR TIME    4095.0000
 DELETING TIME-RECORD FOR TIME    6143.0000
 DELETING TIME-RECORD FOR TIME    8191.0000
 DELETING TIME-RECORD FOR TIME    10239.000
 DELETING TIME-RECORD FOR TIME    12287.000
 DELETING TIME-RECORD FOR TIME    16383.000
 DELETING TIME-RECORD FOR TIME    24575.000
 DELETING TIME-RECORD FOR TIME    32767.000
 DELETING TIME-RECORD FOR TIME    40959.000
 DELETING TIME-RECORD FOR TIME    49151.000
 DELETING TIME-RECORD FOR TIME    65535.000
 DELETING TIME-RECORD FOR TIME    81919.000
 DELETING TIME-RECORD FOR TIME    98303.000
 DELETING TIME-RECORD FOR TIME    114687.00
 DELETING TIME-RECORD FOR TIME    131071.00
 DELETING TIME-RECORD FOR TIME    147455.00
 DELETING TIME-RECORD FOR TIME    163839.00
 DELETING TIME-RECORD FOR TIME    180223.00
 DELETING TIME-RECORD FOR TIME    196607.00
 DELETING TIME-RECORD FOR TIME    212991.00
 DELETING TIME-RECORD FOR TIME    229375.00
 DELETING TIME-RECORD FOR TIME    245759.00
 DELETING TIME-RECORD FOR TIME    262143.00
 DELETING TIME-RECORD FOR TIME    278527.00
 DELETING TIME-RECORD FOR TIME    294911.00
 DELETING TIME-RECORD FOR TIME    311295.00
 DELETING TIME-RECORD FOR TIME    327679.00

 KEEPING TIME-RECORD FOR TIME    344063.00
 AND FOR TIME                    360000.00
 WORKSPACE RECLAIMED
 ------------------------------------------------
 INTERPOLATION SCHEME USED THIS FRACTION OF
 THE ALLOCATED MEMORY:   5.081755192199146E-002
 EFFICIENCY FACTOR:    7.15882455833479
 ------------------------------------------------
 DEALLOCATING
 ------------------------------------------------
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

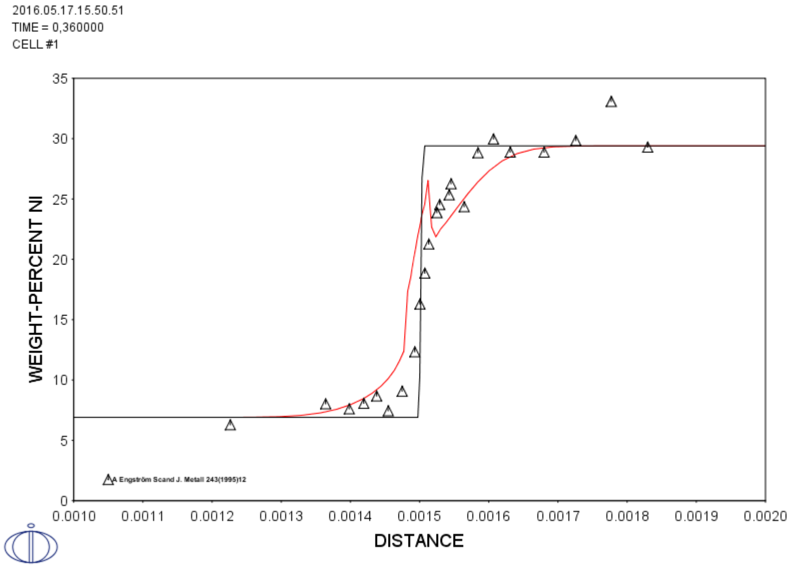
exd3-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exd3\plot.DCM"DIC>
DIC>
DIC> @@ exd3_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE D3
DIC> @@
DIC> @@ ENTER THE DICTRA MODULE AND SPECIFY THE STORE-RESULT FILE
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  3.60000E+05
DIC> read  exd3
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ First study the composition profiles of Cr and Ni
POST-1: @@
POST-1: s-d-a x distance global
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p Cr
POST-1: s-p-c time 0 360000
POST-1: set-tit Figure D3.1
POST-1:
POST-1: app yes k5k7cr.exp 0; 1
POST-1:
POST-1: s-s-s x n 1e-3 2e-3
POST-1:
POST-1: s-s-s y n 20 45
POST-1:
POST-1: plot
```



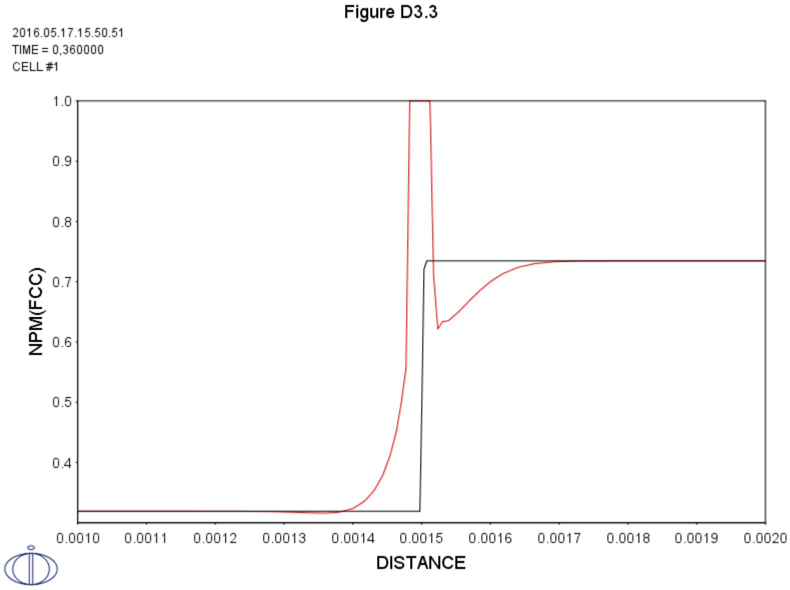

```
POST-1:
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: s-d-a y w-p Ni
POST-1: set-tit Figure D3.2
POST-1:
POST-1: app yes k5k7ni.exp 0; 1
POST-1:
POST-1: s-s-s y n 0 35
POST-1:
POST-1: plot
```

Figure D3.2



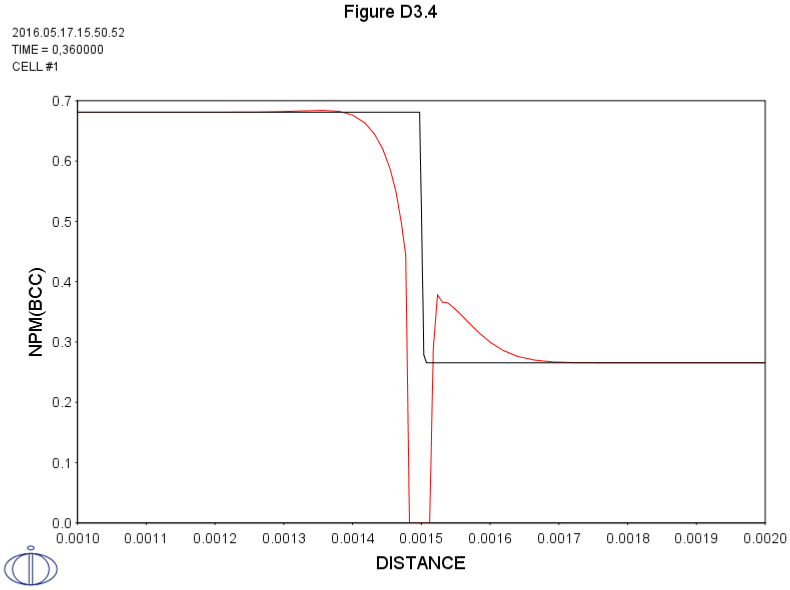

```
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ then study the amount of FCC and BCC
POST-1: @@
POST-1: app no
POST-1: s-d-a y npm(fcc)
POST-1: set-tit Figure D3.3
POST-1: plot
```

Figure D3.3




```
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: s-d-a y npm(bcc)
POST-1: set-tit Figure D3.4
POST-1: plot
```

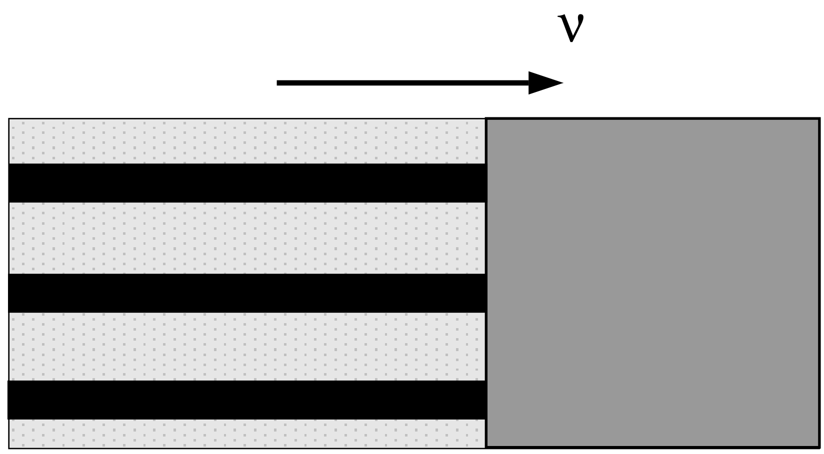
Figure D3.4




```
POST-1:
```

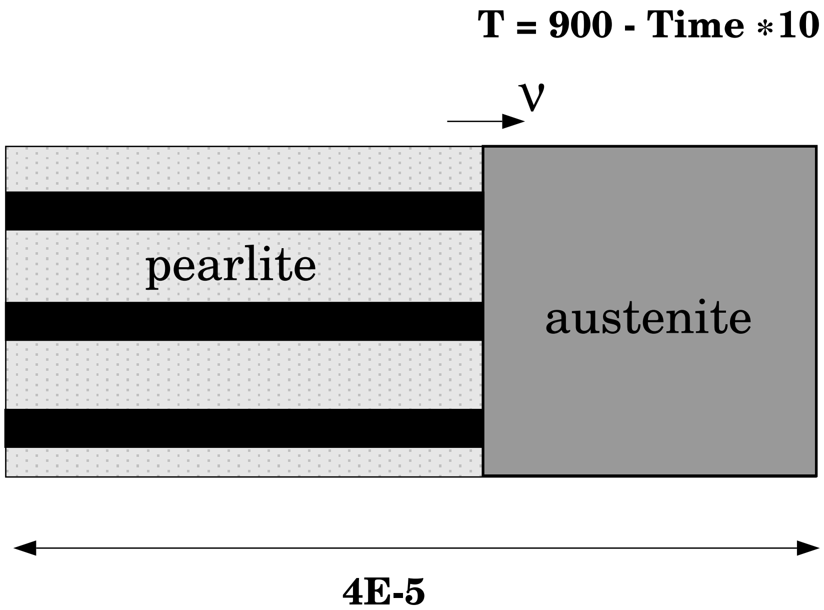
```
POST-1:
POST-1: set-interactive
 --OK---
POST-1:
```

Cooperative growth



Example e1

Growth of pearlite in an Fe-Mn-C alloy



exe1-setup

```
About Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exe1\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Cooperative growth.
SYS: @@ This is an example file for a setup and a calculation of PEARLITE
SYS: @@ growth in an Fe-0.50wt%C - 0.91wt%Mn steel.
SYS: @@
SYS:
SYS: @@ exe1_setup.DCM
SYS:
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1   REJECTED
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1   REJECTED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ DEFINE THE SYSTEM
TDB_TCFE7: @@
TDB_TCFE7: def-sys fe c mn
 FE                       C                        MN
   DEFINED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ KEEP ONLY THE AUSTENITE, FERRITE AND CEMENTITE PHASES
TDB_TCFE7: @@
TDB_TCFE7: rej-ph /all
 GAS:G                    LIQUID:L                 BCC_A2
 FCC_A1                   HCP_A3                   DIAMOND_FCC_A4
 GRAPHITE                 CEMENTITE                M23C6
 M7C3                     M5C2                     KSI_CARBIDE
 A1_KAPPA                 KAPPA                    FE4N_LP1
 FECN_CHI                 LAVES_PHASE_C14          G_PHASE
   REJECTED
TDB_TCFE7: rest-ph fcc,bcc,cem
 FCC_A1                   BCC_A2                   CEMENTITE
   RESTORED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ GET THE THERMODYNAMIC DATA
TDB_TCFE7: @@
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo-C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'W. Huang, Metall. Trans. A, 21A (1990), 2115-2123; TRITA-MAC 411 (Rev
      1989); C-FE-MN'
  'W. Huang, Calphad, 13 (1989), 243-252; TRITA-MAC 388 (rev 1989); FE-MN'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'P. Villars and L.D. Calvert (1985). Pearsons handbook of crystallographic
      data for intermetallic phases. Metals park, Ohio. American Society for
      Metals; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ APPEND THE KINETIC DATA FROM THE MOBILITY DATABASE
TDB_TCFE7: @@
TDB_TCFE7: append mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe c mn
 FE                       C                        MN
   DEFINED
APP: rej-ph /all
 BCC_A2                   CEMENTITE                FCC_A1
 FE4N_LP1                 HCP_A3                   LIQUID:L
   REJECTED
APP: rest-ph bcc,fcc,cem
 BCC_A2                   FCC_A1                   CEMENTITE
   RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
```

```
 List of references for assessed data

   'This parameter has not been assessed'
   'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
   'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
       -Ni'
   'Bae et al.: Z. Metallkunde 91(2000)672-674; fcc Fe-Mn
        Mn-Ni'
   'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
       -Fe-Ni'
   'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
       in bcc Fe'
   'Assessed from data presented in Landholt-Bornstein, Vol. 26, ed. H.
       Mehrer, springer (1990); Impurity diff of Mn in bcc Fe.'
 -OK-
APP:
APP: @@
APP: @@ OK, ALL THERMODYNAMIC AND KINETIC DATA HAVE BEEN FETCHED
APP: @@ GO TO THE DICTRA MONITOR TO SETUP YOUR PROBLEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ SET CONDITION ON TEMPERATURE
DIC> @@
DIC> set-cond glob t 0 900-time*10; * n

DIC>
DIC> @@
DIC> @@ ENTER REGIONS
DIC> @@
DIC> enter-reg pearlite
DIC>
DIC> @@
DIC> @@ ENTER A SMALL INITIAL SIZE OF THE GRID IN THE 'PEARLITE' REGION
DIC> @@
DIC> enter-grid pearlite 5e-10 lin 5
DIC>
DIC> @@
DIC> @@ ENTER INTO THE 'PEARLITE' REGION THE PHASES 'BCC' AND 'CEM' AND SPECIFY
DIC> @@ THAT THEY WILL BE PRESENT IN THE FORM OF A 'LAMELLAR AGGREGATE. SET THEIR
DIC> @@ STATUS TO 'ACTIVE'. A LOT OF QUESTIONS FOLLOWS CONCERNING THE VALUES OF
DIC> @@ THE PARAMETERS PRESENT IN THE PEARLITE GROWTH MODEL SUCH AS SURFACE
DIC> @@ TENSION, OPTIMUM GROWTH RATE FACTOR, BOUNDARY DIFFUSION COEFFICIENTS.
DIC> @@ CARBON(C) IS TREATED IN A SPECIAL WAY INSIDE THE PROGRAM, IF AUTOMATIC
DIC> @@ IS ENTERED THE DIFFUSION OF C IS CALCULATED ACCORDING TO AN EQUATION
DIC> @@ FOR MIXED BOUNDARY AND VOLUME DIFFUSION. YOU MAY CHOOSE BETWEEN
DIC> @@ MANUAL OR AUTOMATIC START VALUES FOR ALL VARIABLES EXCEPT THE GROWTH
DIC> @@ RATE, LET US TRY 1E-6
DIC> @@
DIC> @@ FOR MORE INFORMATION ABOUT THE PEARLITE GROWTH MODEL CONSULT
DIC> @@ B. JÖNSSON: INTERNAL REPORT, TRITA-MAC-0478, DIVISION OF
DIC> @@ PHYSICAL METALLURGY, ROYAL INSTITUTE OF TECHNOLOGY, S-10044
DIC> @@ STOCKHOLM, SWEDEN, 1992.
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /PEARLITE/: pearlite
PHASE TYPE /MATRIX/: lam

      Eutectiod reaction is "GAMMA" ==> "ALPHA" + "BETA"

 Enter name of "ALPHA" phase /BCC_A2/: bcc_a2
 Enter name of "BETA" phase /CEMENTITE/: cementite
 Enter name of "GAMMA" phase /FCC_A1/: fcc_a1
 Enter "ALPHA"/"BETA" surface tension
LOW TIME LIMIT /0/: 0
 Surface tension(T,P,TIME)= 1;
HIGH TIME LIMIT /*/: 1000
ANY MORE RANGES /N/: N
 Enter "ALPHA"/"GAMMA" surface tension
LOW TIME LIMIT /0/: 0
 Surface tension(T,P,TIME)= 1;
HIGH TIME LIMIT /*/: 1000
ANY MORE RANGES /N/: N
 Enter "BETA"/"GAMMA" surface tension
LOW TIME LIMIT /0/: 0
 Surface tension(T,P,TIME)= 1;
HIGH TIME LIMIT /*/: 1000
ANY MORE RANGES /N/: N
 Optimum growth condition factor /2/: 2
 Name of dependent element /FE/: fe
 INPUT OF DIFFUSION DATA
   Growth model (VOLUME/BOUNDARY/KIRKALDY) for element C /BOUNDARY/: boundary
     DF(C) = /value/AUTOMATIC/MIXED/: auto
   Growth model (VOLUME/BOUNDARY/KIRKALDY) for element MN /BOUNDARY/: boundary
     DF(MN) = /value/MIXED/: 5.4e-14
     DQ(MN): 155000
 Automatic start values for the S0 determination /Y/: Y
 Growth rate V: 1E-6
 Automatic start values on other variables /Y/: Y
DIC>
DIC> @@
DIC> @@ INITIATE THE COMPOSITION RECORDS FOR THE 'PEARLITE'
DIC> @@
DIC> enter-composition
REGION NAME : /PEARLITE/: pearlite
DIC>
DIC> @@
DIC> @@ WE WILL NOW CONTINUE BY DEFINING A MATRIX PHASE INTO WHICH THE PEARLITE
DIC> @@ WILL GROW. START BY ENTERING A REGION NAME, LET US CALL IT 'AUSTENITE'
DIC> @@
DIC> enter-region austenite
ATTACH TO REGION NAMED /PEARLITE/:
ATTACHED TO THE RIGHT OF PEARLITE /YES/:
DIC> @@
DIC> @@ SPECIFY WHAT PHASE 'FCC' WILL BE PRESENT IN THE 'AUSTENITE' REGION
DIC> @@ AND WHAT TYPE OF PHASE 'MATRIX' IT IS AND ITS INITIAL STATE 'ACTIVE'
DIC> @@
DIC> enter-phase act austenite matrix fcc
```

```
DIC>
DIC> @@
DIC> @@ WE ALSO NEED TO HAVE A SPATIAL GRID IN THE 'AUSTENITE' REGION.
DIC> @@ CHOSE SIZE '4E-5' GRIDTYPE 'GEOMETRICAL', '30' GRIDPOINTS AND '1.5'
DIC> @@ AS VALUE FOR THE GEOMETRICAL FACTOR OF THE GRID.
DIC> @@
DIC> enter-grid austenite 4e-5 geo 30 1.5
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL CONCENTRATION PROFILES IN THE 'FCC' PHASE OF THE
DIC> @@ 'AUSTENITE' REGION. CONCENTRATIONS MUST BE GIVEN IN Y-FRACTIONS.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc
DEPENDENT COMPONENT ? /MN/: fe
COMPOSITION TYPE /MOLE_FRACTION/: site-fraction
 PROFILE FOR MN
TYPE /LINEAR/: lin 9.29232973E-3 9.29232973E-3
 PROFILE FOR C
TYPE /LINEAR/: lin 2.3384332E-2 2.3384332E-2
DIC>
DIC> @@
DIC> @@ WE ARE NOW FINNISHED WHITH THE MATRIX PHASE
DIC> @@
DIC>
DIC> @@
DIC> @@ SPECIFY A SPHERICAL '2' GEOMETRY
DIC> @@
DIC> enter-geo 2
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 5
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /.5/: 0.1
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exe1 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exe1-run

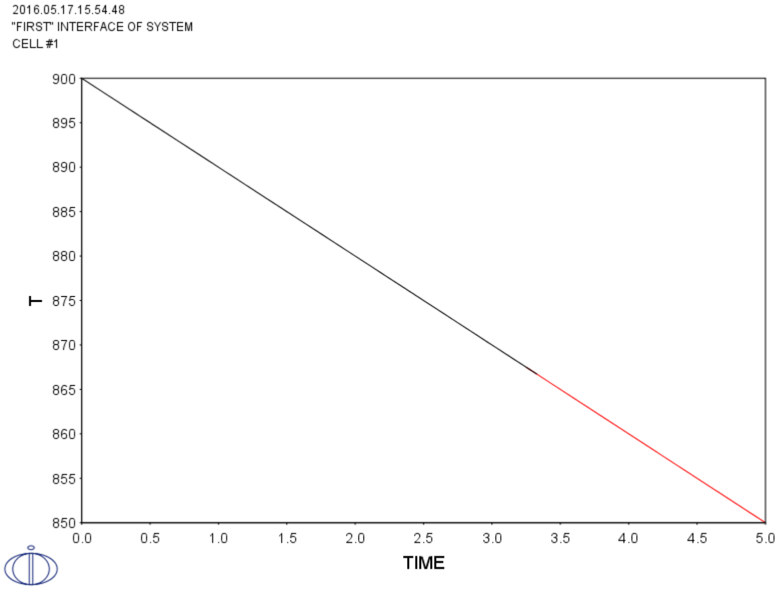
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exe1\run.DCM"
DIC>
DIC> @@ exe1_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE e1
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> @@
DIC> @@ READ SETUP FROM FILE
DIC> @@
DIC> read exe1
 OK
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> simulate
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                      4
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399293
                        MN = .0092923297312127
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399293
                        MN = .0092923297312127
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]
  17  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76465982E-05 AND   0.76465982E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.50076466E-09
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121271
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  10  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                     0  seconds
  4  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76468335E-05 AND   0.76468335E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.12654480E-08
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399295
                        MN = .00929232973121271
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
  1 GRIDPOINT(S) REMOVED FROM CELL#  1 REGION #  1

 TIME =  0.13026424E-02 DT =  0.12025424E-02 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76496630E-05 AND   0.76496630E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.10464492E-07
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
  2 GRIDPOINT(S) REMOVED FROM CELL#  1 REGION #  1

 TIME =  0.37077272E-02 DT =  0.24050848E-02 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76553232E-05 AND   0.76553232E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.28876193E-07
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
  1 GRIDPOINT(S) REMOVED FROM CELL#  1 REGION #  1

 TIME =  0.85178967E-02 DT =  0.48101695E-02 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76666480E-05 AND   0.76666480E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.65754070E-07
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399295
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
  1 GRIDPOINT(S) REMOVED FROM CELL#  1 REGION #  1

 TIME =  0.18138236E-01 DT =  0.96203391E-02 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.76893156E-05 AND   0.76893156E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.13972789E-06
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =  0.37378914E-01 DT =  0.19240678E-01 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.77347217E-05 AND   0.77347217E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.28854918E-06
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
```

```
 TIME =  0.75860270E-01 DT =  0.38481356E-01 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.78258159E-05 AND   0.78258159E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.58969719E-06
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399294
                        MN = .00929232973121272
 TOTAL SIZE OF SYSTEM:  2.68092626329E-13 [m^3]

  CPU time used in timestep                     0  seconds
  1 GRIDPOINT(S) REMOVED FROM CELL#  1 REGION #  1

 TIME =  0.15282298     DT =  0.76962713E-01 SUM OF SQUARES =   0.0000000
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.80091146E-05 AND   0.80091146E-05
 POSITION OF INTERFACE PEARLITE / AUSTENITE IS   0.12061004E-05
 U-FRACTION IN SYSTEM:  C = .0233843320030519  FE = .990707670399294


output ignored...



... output resumed


 TIME =   3.5362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   3.6362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   3.7362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   3.8362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   3.9362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.0362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.1362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.2362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.3362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.4362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.5362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.6362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.7362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.8362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
```

```
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   4.9362705     DT =  0.10000000E+00 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

  CPU time used in timestep                     0  seconds

 TIME =   5.0000000     DT =  0.63729517E-01 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .0233843320030518  FE = .990707670399277
                        MN = .00929232973122968
 TOTAL SIZE OF SYSTEM:  2.67725268543E-13 [m^3]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    3.2528230
 DELETING TIME-RECORD FOR TIME    3.3262605
 DELETING TIME-RECORD FOR TIME    3.3262705
 DELETING TIME-RECORD FOR TIME    3.3362705
 DELETING TIME-RECORD FOR TIME    3.4362705
 DELETING TIME-RECORD FOR TIME    3.5362705
 DELETING TIME-RECORD FOR TIME    3.6362705
 DELETING TIME-RECORD FOR TIME    3.7362705
 DELETING TIME-RECORD FOR TIME    3.8362705
 DELETING TIME-RECORD FOR TIME    3.9362705
 DELETING TIME-RECORD FOR TIME    4.0362705
 DELETING TIME-RECORD FOR TIME    4.1362705
 DELETING TIME-RECORD FOR TIME    4.2362705
 DELETING TIME-RECORD FOR TIME    4.3362705
 DELETING TIME-RECORD FOR TIME    4.4362705
 DELETING TIME-RECORD FOR TIME    4.5362705
 DELETING TIME-RECORD FOR TIME    4.6362705
 DELETING TIME-RECORD FOR TIME    4.7362705
 DELETING TIME-RECORD FOR TIME    4.8362705

 KEEPING TIME-RECORD FOR TIME    4.9362705
 AND FOR TIME                    5.0000000
 WORKSPACE RECLAIMED
DIC>
DIC> @@
DIC> @@ THE SIMULATION IS FINISHED
DIC> @@
DIC>
DIC> set-inter
 --OK---
DIC>
```

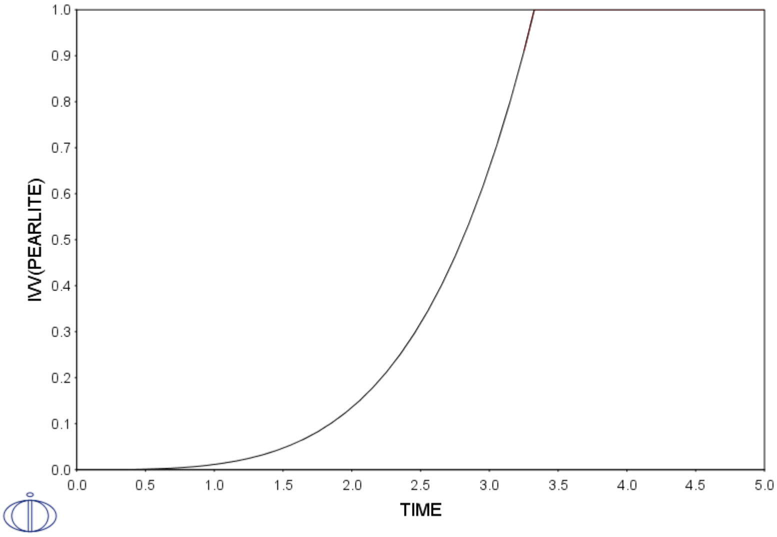
exe1-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exe1\plot.DCM"DIC>
DIC>
DIC> @@ exe1_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  5.00000E+00
DIC> read exe1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ PLOT THE TEMPERATURE AS FUNCTION OF TIME.
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y t
POST-1: s-p-c interface first
POST-1:
POST-1: plot
```



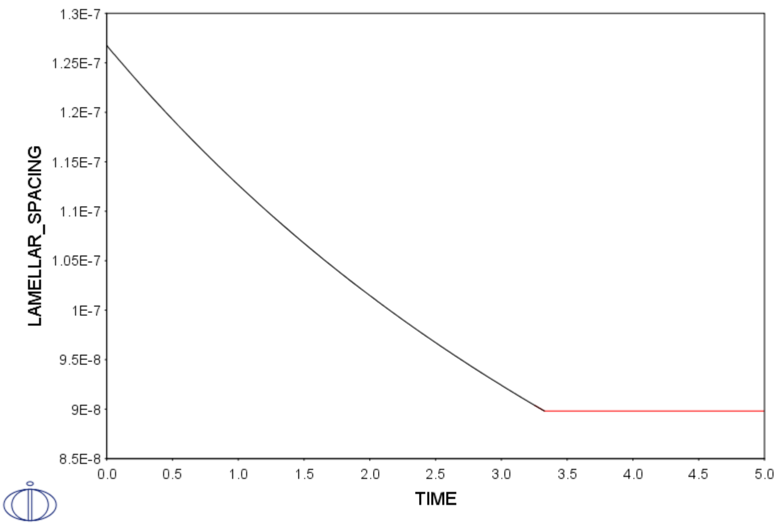

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ NOW LETS PLOT THE FRACTION OF PEARLITE VS. TIME.
POST-1: @@
POST-1: s-d-a y ivv(pearlite)
POST-1:
POST-1: plot
```

2016.05.17.15.54.49
CELL #1



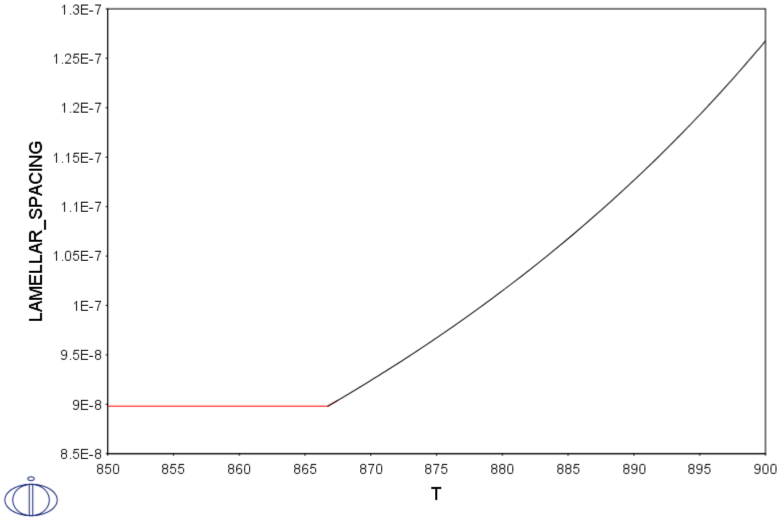

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ THE LAMELLAR SPACING AS FUNCTION OF TIME.
POST-1: @@
POST-1: s-d-a
AXIS (X, Y OR Z) : y
VARIABLE : lamellar-sp
IN REGION: /*/: pearlite
POST-1:
POST-1: s-p-c
CONDITION /INTEGRAL/: interface
INTERFACE : pearlite
UPPER OR LOWER INTERFACE OF REGION PEARLITE#1 /LOWER/: upper
POST-1:
POST-1: plot
```

2016.05.17.15.54.49
UPPER INTERFACE OF REGION "PEARLITE#1"
CELL #1



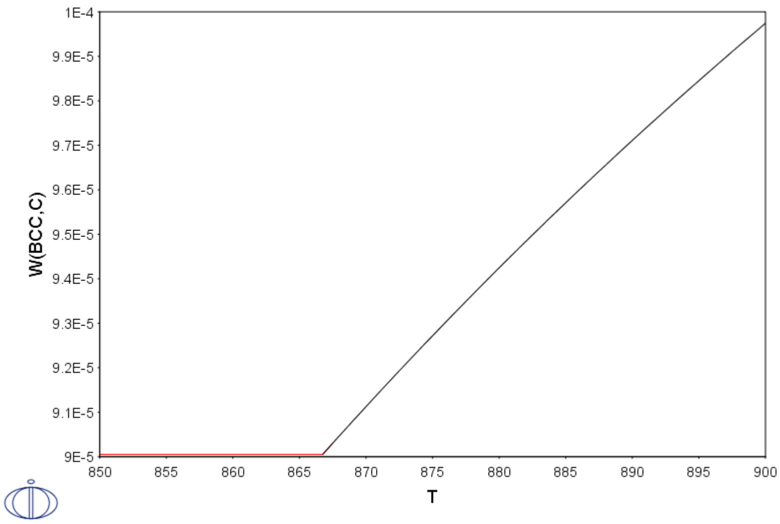

```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ LET'S LOOK AT THE LAMELLAR SPACING VS. TEMPERATURE INSTEAD.
POST-1: @@
POST-1: s-d-a x t
POST-1:
POST-1: s-p-c interface pearlite upper
POST-1:
POST-1: plot
```

2016.05.17.15.54.49
UPPER INTERFACE OF REGION "PEARLITE#1"
CELL #1




```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ AND THE C COMPOSITION IN THE FERRITE VS. TEMP.
POST-1: @@
POST-1: s-d-a y w(bcc,c)
POST-1:
POST-1:
POST-1: plot
```

2016.05.17.15.54.49
UPPER INTERFACE OF REGION "PEARLITE#1"
CELL #1

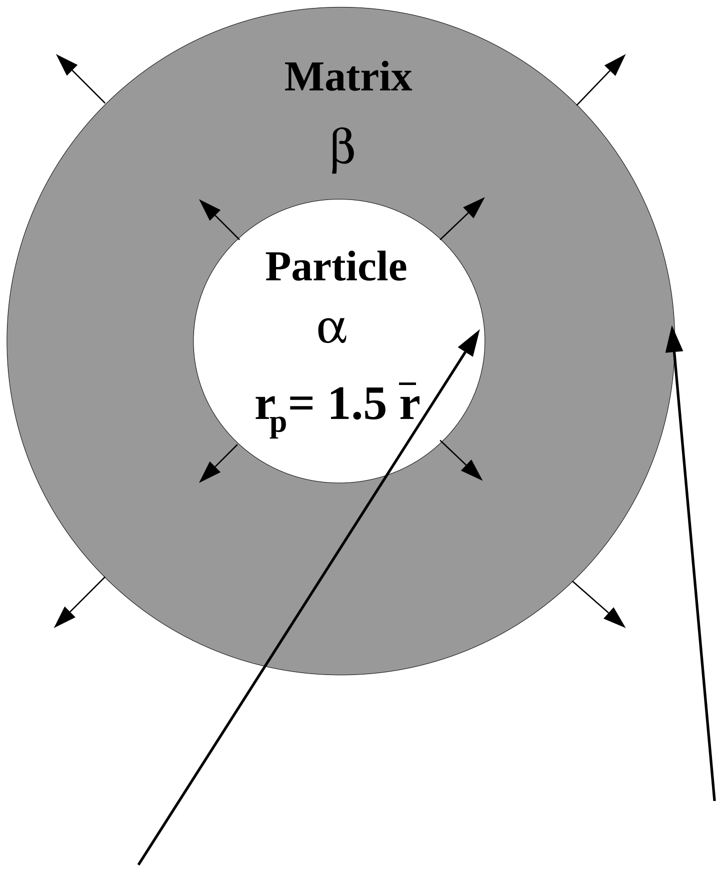



```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1: @@
POST-1: @@ FINALLY, LETS LOOK AT THE VELOCITY OF THE INTERFACE VS. TEMP.
POST-1: @@
POST-1: s-d-a y t
POST-1: s-d-a x velocity
INTERFACE : pearlite
UPPER OR LOWER INTERFACE OF REGION PEARLITE#1 /LOWER/: upper
POST-1: set-ax-ty x log
POST-1: s-s-s x n 1e-6 1e-4
POST-1:
POST-1: plot
```

2016.05.17.15.54.50
UPPER INTERFACE OF REGION "PEARLITE#1"
CELL #1


```
POST-1:
POST-1:
POST-1:
POST-1:Hit RETURN to continue
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Coarsening



Moving phase interface with α and β in local equilibrium.

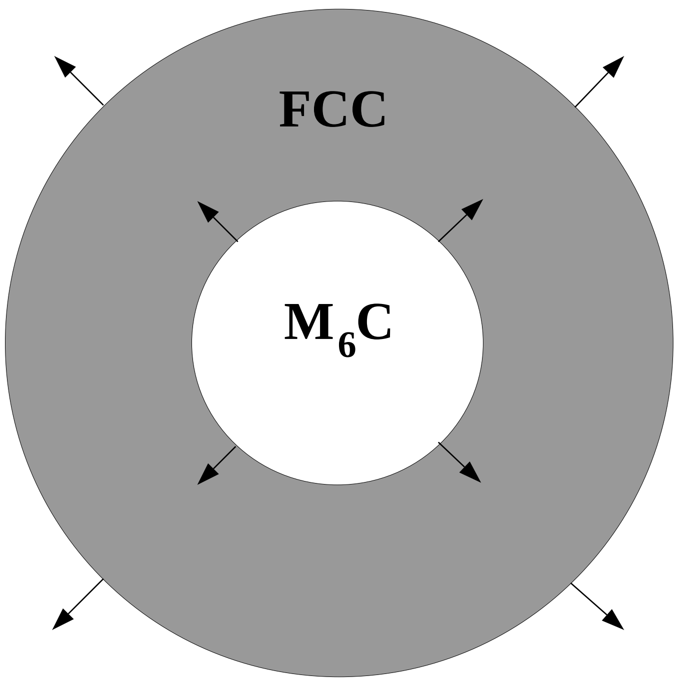
Equilibrium as defined by the average composition in the system.

$\frac{2\sigma V_m}{r_p}$ **interfacial energy contribution for α-phase**

$\frac{2\sigma V_m}{\bar{r}}$ **interfacial energy contribution for α-phase**

Example f1

Coarsening of a M_6C precipitate in an Fe-Mo-C alloy



T = 1173K **r_p = 0.228 μm**

exf1-setup

```
About Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exf1\setup.DCM" @@
SYS: @@ Coarsening.
SYS: @@ Setup file for calculating the Ostwald-ripening
SYS: @@ of a spherical M6C carbide in an AUSTENITE matrix.
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1   REJECTED
TDB_TCFE8: switch tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1   REJECTED
TDB_TCFE7: def-sys fe mo c
 FE                      MO                      C
   DEFINED
TDB_TCFE7: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 FCC_A1                  HCP_A3                  DIAMOND_FCC_A4
 GRAPHITE                CEMENTITE               M23C6
 M7C3                    M6C                     M5C2
 M3C2                    MC_ETA                  MC_SHP
 KSI_CARBIDE             A1_KAPPA                KAPPA
 Z_PHASE                 FE4N_LP1                FECN_CHI
 SIGMA                   MU_PHASE                P_PHASE
 R_PHASE                 CHI_A12                 LAVES_PHASE_C14
   REJECTED
TDB_TCFE7: res ph fcc m6c
 FCC_A1                  M6C  RESTORED
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'J-O. Andersson, Calphad, 12 (1988), 1-8; TRITA 0317 (1986); C-MO'
  'A. Fernandez Guillermet, Calphad, 6 (1982), 127-140; (sigma phase revised
      1986); TRITA-MAC 200 (1982); FE-MO'
  'J-O. Andersson, Calphad, 12 (1988), 9-23; TRITA 0321 (1986); C-FE-MO'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'S. Nagakura (1968), Transactions of the Iron and Steel Institute of Japan,
       Vol.8, pp. 265-294; Molar volumes'
  'Kupalova, IK and Pavlova, VI (988); High speed steels: Physical
      Properties, Prop. Data Update, 2, 67-78; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_TCFE7: @@
TDB_TCFE7: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
```

```
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /TCFE7/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe mo c
 FE                        MO                        C
   DEFINED
APP: rej ph * all
 BCC_A2                    CEMENTITE                 FCC_A1
 FE4N_LP1                  HCP_A3                    LIQUID:L
   REJECTED
APP: res ph fcc m6c
 *** M6C INPUT IGNORED
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'H. Oikawa, Tech. Rept. Tohoku Univ., 48(1983)7.; Impurity diffusion of Mo
      in fcc Fe.'
 -OK-
APP:
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING M6C AS A DIFFUSION NONE PHASE
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> s-cond glob t 0 1173; * N
```

```
DIC>
DIC> @@
DIC> @@ ENTER REGIONS part AND aus
DIC> @@
DIC> enter-region
REGION NAME : part
DIC> enter-region aus
ATTACH TO REGION NAMED /PART/:
ATTACHED TO THE RIGHT OF PART /YES/:
DIC> @@
DIC> @@ ENTER GEOMTRICAL GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ THE INITIAL SIZE OF THE CARBIDE PARTICLE IS ASSUMED TO BE KNOWN
DIC> @@ (IN THIS CASE WE TAKE OUR VALUE FROM NISHIZAWA ET. AL.)  THE
DIC> @@ AVERAGE PARTICLE SIZE IS ASSUMED TO BE 0.152E-6 METERS, WE WILL HOWEVER
DIC> @@ PERFORM THE CALCULATIONS ON A MAXIMUM SIZE PARTICLE WHICH IS ASSUMED TO
DIC> @@ 1.5 TIMES THE AVERAGE SIZE. THE SURROUNDING AUSTENITIC MATRIX SIZE IS
DIC> @@ CHOOSEN TO MAINTAIN THE AVERAGE COMPOSITION.
DIC> @@
DIC> enter-grid
REGION NAME : /PART/: part
WIDTH OF REGION /1/: 0.228E-6
TYPE /LINEAR/: lin 50
DIC>
DIC> enter-grid
REGION NAME : /AUS/: aus
WIDTH OF REGION /1/: 4.53147041E-7
TYPE /LINEAR/: lin 50
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> enter-phase active part matrix m6c
DIC>
DIC> enter-phase active aus matrix fcc#1
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /PART/: part
PHASE NAME: /M6C/: m6c
DEPENDENT COMPONENT ? /MO/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-f
PROFILE FOR /MO/: mo lin 6.20117E-01 6.20117E-01
DIC>
DIC> ent-composition
REGION NAME : /AUS/: aus
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /MO/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-f
PROFILE FOR /C/: mo lin 1.82099E-02 1.82099E-02
PROFILE FOR /MO/: c lin 2.83351E-03 2.83351E-03
DIC>
DIC>
DIC> @@
DIC> @@ SET SPHERICAL GEOMETRY
DIC> @@
DIC> ent-geo 2
DIC>
DIC> @@
DIC> @@ ENTER THE SURFACE TENSION ENERGY CONTRIBUTION AS A FUNCTION OF
DIC> @@ THE INTERFACE POSITION (THE RADIUS OF THE PARTICLE)
DIC> @@ WE WILL ALSO ENTER THE MOLAR VOLUME OF THE PHASE CORRECTED
DIC> @@ TO BE THE MOLAR VOLUME PER SUBSTITIONAL ATOM
DIC> @@ THE SURFACE TENSION IS 0.7, THE MOLAR VOLUME IS 0.71 AND THE
DIC> @@ TRANSFORMATION TO MOLAR VOLUME PER SUBSTITIONAL ATOM IS 7/6
DIC> @@
DIC> set-surf 2*0.7*0.71*(7/6)/X;
 ENTERED FUNCTION :2*.7*.71*7/6/X FOR CELL #1
DIC>
DIC>
DIC> @@
DIC> @@ ENABLE THE SIMPLIFIED MODEL THE COARSENING (OSTWALD-RIPENING)
DIC> @@
DIC> coarse  YES
DIC>
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME AND VARIOUS SIMULATION PARAMETERS
DIC> @@
DIC> set-simulation-time 1E6
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /100000/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exf1 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exf1-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exf1\run.DCM"DIC> @@ exf1_run.DCM
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING M6C AS A DIFFUSION NONE PHASE
DIC> read exf1
 OK
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                      4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .0190652843033664  FE = .970761291162784
                        MO = .0292387089677228
 TOTAL SIZE OF SYSTEM:  1.32376603026E-18 [m^3]
 U-FRACTION IN SYSTEM:  C = .0190652843033664  FE = .970761291162784
                        MO = .0292387089677228
 TOTAL SIZE OF SYSTEM:  1.32376603026E-18 [m^3]
 0.610218686341790        0.610386763302538        0.610212754562795        2.577439251116885E-002        2.204026974008048E-
003        2.511101571596205E-005        6.202108280413697E-008        2.619773115661363E-009        1.181243458996165E-
009        8.640042993716350E-010        3.488136626588954E-009        4.880010803609078E-010        7.631332703845558E-
011        1.073882854914204E-014        9.616361541258593E-021
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.96163615E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.72886389E-05 AND  -0.72886389E-05
 POSITION OF INTERFACE PART / AUS IS   0.22799927E-06
 U-FRACTION IN SYSTEM:  C = .019066077989355  FE = .970758567486736
                        MO = .0292414326437702
 TOTAL SIZE OF SYSTEM:  1.32375333496E-18 [m^3]

  CPU time used in timestep                    0  seconds
 1.247393297473699E-005        1.247642752846073E-005        1.247555555148743E-005        1.194633165580483E-006        5.572570235589888E-
007        9.459235504737183E-009        1.194773067602253E-012        1.492591937203126E-014        9.800831139960738E-
017       SWITCHING ACTIVITIES FOR INTERFACE #2, CELL #1
 FROM: C  TO: MO


 TIME =  0.51722608E-05 DT =  0.50722608E-05 SUM OF SQUARES =  0.98008311E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.28448353E-08 AND  -0.28448353E-08
 POSITION OF INTERFACE PART / AUS IS   0.22799926E-06
 U-FRACTION IN SYSTEM:  C = .0190669251813878  FE = .970758545807002
                        MO = .0292414543235047
 TOTAL SIZE OF SYSTEM:  1.32375308363E-18 [m^3]

  CPU time used in timestep                    0  seconds
 1.981717619774803E-012        1.982094654904473E-012        1.092857918490165E-012        1.057142240413414E-012        1.056834511634861E-
012        1.056219703484371E-012        1.054990109534616E-012        1.651101170376211E-012        1.052533513310341E-
012        1.047628606890552E-012        1.037853456594443E-012        1.634600230190247E-012        1.018440283633567E-
012        9.801639891935364E-013        9.058100739753741E-013        1.502036397971311E-012        7.658987103353221E-
013        5.212620244296235E-013        1.727476209539347E-013        1.456226216916591E-021
 TIME =  0.15316783E-04 DT =  0.10144522E-04 SUM OF SQUARES =  0.14562262E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.40380765E-09 AND  -0.40380765E-09
 POSITION OF INTERFACE PART / AUS IS   0.22799925E-06
 U-FRACTION IN SYSTEM:  C = .0190678327596393  FE = .970758541244064
                        MO = .0292414588864431
 TOTAL SIZE OF SYSTEM:  1.32375301228E-18 [m^3]

  CPU time used in timestep                    1  seconds
 4.559555744735687E-014        4.560305684874453E-014        1.465166416270655E-013        2.090526655550763E-014        2.091054414556866E-
014        2.089998772213471E-014        2.824776966133181E-013        2.089471204029009E-014        2.088416016117839E-
014        2.086306688932830E-014        2.824631546310546E-013        2.082090980032321E-014        2.073672606061375E-
014        2.056886765334059E-014        2.821478507474395E-013        2.023519972281791E-014        1.957604701181261E-
014        1.829048636917020E-014

output ignored...



... output resumed

 1.142512563289157E-014        1.118081803200462E-014        1.070020436054221E-014        9.770600056737860E-015        5.709893862993964E-
013        8.038248608528668E-015        5.080423857245265E-015        1.191975415609428E-015        1.846806816132095E-022
 TIME =   774281.56      DT =  100000.000      SUM OF SQUARES =  0.18468068E-21
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.42232322E-13 AND   0.42232322E-13
 POSITION OF INTERFACE PART / AUS IS   0.26511941E-06
 U-FRACTION IN SYSTEM:  C = .0195361337250528  FE = .97070071544692
                        MO = .0292992846835869
 TOTAL SIZE OF SYSTEM:  2.08128369835E-18 [m^3]

  23  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: PART
  23  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUS

  CPU time used in timestep                    0  seconds
 8.903353114672830E-013        8.942337036868494E-013        2.818002097329602E-012        8.245062139580557E-015        8.059980202748808E-
015        7.696203276141083E-015        6.993787855810988E-015        5.504219384476050E-013        5.689918979043227E-
015        3.485378034875575E-015        6.892662857594977E-016        6.203794033035725E-023
 TIME =   874281.56      DT =  100000.000      SUM OF SQUARES =  0.62037940E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.40920227E-13 AND   0.40920227E-13
 POSITION OF INTERFACE PART / AUS IS   0.26921144E-06
 U-FRACTION IN SYSTEM:  C = .0195379504997532  FE = .970703186929402
                        MO = .0292968132011065
 TOTAL SIZE OF SYSTEM:  2.17915040492E-18 [m^3]

  21  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: PART
  21  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUS

  CPU time used in timestep                    0  seconds
 7.387199788394374E-013        7.421456602256449E-013        2.508134483889649E-012        5.911859525059236E-015        5.771538489455730E-
```

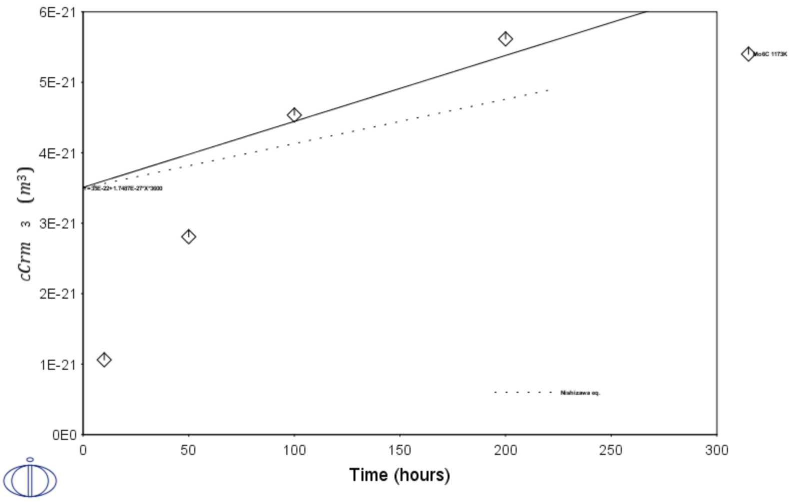
```
015        5.495901343264641E-015        4.965005360152692E-015        5.319198893336870E-013        3.984162251480447E-
015        2.346016940606032E-015        3.636837068060717E-016        2.735263117511465E-023
 TIME =   974281.56     DT =   100000.000     SUM OF SQUARES =  0.27352631E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.39705524E-13 AND   0.39705524E-13
 POSITION OF INTERFACE PART / AUS IS   0.27318199E-06
 U-FRACTION IN SYSTEM:  C = .0195396954479369  FE = .97070553898551
                        MO = .0292944611449967
 TOTAL SIZE OF SYSTEM:  2.27699919988E-18 [m^3]

  20  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: PART
  20  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUS

  CPU time used in timestep                    1  seconds
 7.837996905437424E-014        7.936304332204428E-014        1.001954033732713E-012        3.250609830189901E-016        2.992028130133000E-
016        2.506969585202179E-016        5.208317797195455E-013        1.733615835597377E-016
 TIME =  1000000.00     DT =   25718.445     SUM OF SQUARES =  0.17336158E-15
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.38940847E-13 AND   0.38940847E-13
 POSITION OF INTERFACE PART / AUS IS   0.27418349E-06
 U-FRACTION IN SYSTEM:  C = .0195417239487887  FE = .970706774312103
                        MO = .0292932258184042
 TOTAL SIZE OF SYSTEM:  2.3021338784E-18 [m^3]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.51722608E-05
 DELETING TIME-RECORD FOR TIME   0.15316783E-04
 DELETING TIME-RECORD FOR TIME   0.35605826E-04
 DELETING TIME-RECORD FOR TIME   0.76183913E-04
 DELETING TIME-RECORD FOR TIME   0.15734009E-03
 DELETING TIME-RECORD FOR TIME   0.31965243E-03
 DELETING TIME-RECORD FOR TIME   0.64427713E-03
 DELETING TIME-RECORD FOR TIME   0.12935265E-02
 DELETING TIME-RECORD FOR TIME   0.25920253E-02
 DELETING TIME-RECORD FOR TIME   0.51890228E-02
 DELETING TIME-RECORD FOR TIME   0.10383018E-01
 DELETING TIME-RECORD FOR TIME   0.20771008E-01
 DELETING TIME-RECORD FOR TIME   0.41546989E-01
 DELETING TIME-RECORD FOR TIME   0.83098949E-01
 DELETING TIME-RECORD FOR TIME   0.16620287
 DELETING TIME-RECORD FOR TIME   0.33241071
 DELETING TIME-RECORD FOR TIME   0.66482640
 DELETING TIME-RECORD FOR TIME    1.3296578
 DELETING TIME-RECORD FOR TIME    2.6593205
 DELETING TIME-RECORD FOR TIME    5.3186460
 DELETING TIME-RECORD FOR TIME    10.637297
 DELETING TIME-RECORD FOR TIME    21.274599
 DELETING TIME-RECORD FOR TIME    42.549203
 DELETING TIME-RECORD FOR TIME    85.098411
 DELETING TIME-RECORD FOR TIME    170.19683
 DELETING TIME-RECORD FOR TIME    340.39366
 DELETING TIME-RECORD FOR TIME    680.78732
 DELETING TIME-RECORD FOR TIME    1361.5746
 DELETING TIME-RECORD FOR TIME    2723.1493
 DELETING TIME-RECORD FOR TIME    5446.2986
 DELETING TIME-RECORD FOR TIME    10892.597
 DELETING TIME-RECORD FOR TIME    21785.194
 DELETING TIME-RECORD FOR TIME    43570.389
 DELETING TIME-RECORD FOR TIME    87140.778
 DELETING TIME-RECORD FOR TIME    174281.56
 DELETING TIME-RECORD FOR TIME    274281.56
 DELETING TIME-RECORD FOR TIME    374281.56
 DELETING TIME-RECORD FOR TIME    474281.56
 DELETING TIME-RECORD FOR TIME    574281.56
 DELETING TIME-RECORD FOR TIME    674281.56
 DELETING TIME-RECORD FOR TIME    774281.56
 DELETING TIME-RECORD FOR TIME    874281.56

 KEEPING TIME-RECORD FOR TIME    974281.56
 AND FOR TIME                   1000000.00
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exf1-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exf1\plot.DCM" @@ exf1_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE f1
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  1.00000E+06
 *** ENTERING M6C AS A DIFFUSION NONE PHASE
DIC> read exf1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ WE WILL NOW PLOT THE AVERAGE PARTICLE SIZE CUBED AS THIS ASSUMED
POST-1: @@ TO SCALE LINEARLY WITH TIME, WE MUST THEN ENTER A FUNCTION TO BE
POST-1: @@ ABLE TO ACCESS THIS QUANTITY
POST-1: @@ WE ALSO WANT TO PLOT THIS QUANTITY VERSUS TIME IN HOURS SO WILL
POST-1: @@ ENTER A FUNCTION FOR THIS
POST-1: @@
POST-1: enter-symbol func rr3=(poi(part,u)/1.5)**3;
POST-1: enter-symbol func hours=time/3600;
POST-1: s-d-a x hours
POST-1: s-d-a y rr3
POST-1:
POST-1: @@
POST-1: @@ AS WE ARE PLOTTING FUNCTIONS ON BOTH AXIS WE MUST EXPLICITLY
POST-1: @@ DEFINE THE INDEPENDENT VARIABLE AND THE PLOT CONDITION
POST-1: @@
POST-1: s-ind time
POST-1: s-p-c inter
INTERFACE : part upper
POST-1:
POST-1:
POST-1: set-axis-text-status x n
AXIS TEXT : Time (hours)
POST-1:
POST-1: @@
POST-1: @@ THIS FAIRLY CRYPTICAL AXIS TEXT WILL GIVE A NICE NOTATION FOR
POST-1: @@ THE AVERAGE RADIUS CUBED ONCE THE PLOT IS PRINTED. FOR MORE
POST-1: @@ INFORMATION ON TEXT MANIPULATION IN THE POST PROCESSOR THE READER
POST-1: @@ IS REFERRED TO THE SECTION ON DATAPLOT GRAPHICAL LANGUAGE IN THE
POST-1: @@ THERMO-CALC USERS GUIDE.
POST-1: @@
POST-1:
POST-1:
POST-1: set-axis-text-status y n
AXIS TEXT : ^CCrm ^UP3$ (m^UP3$)
POST-1:
POST-1:
POST-1: @@
POST-1: @@ WE ALSO WANT TO COMPARE WITH EXPERIMENTAL DATA FROM NISHIZAWA ET AL.
POST-1: @@ TRANS. JPN. INST. MET. VOL. 22 1981 PP. 733-742.
POST-1: @@
POST-1:
POST-1: app y exf1
PROLOGUE NUMBER: /0/: 0
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: s-s-s y n 0 6e-21
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-title Figure f1.1
POST-1:
POST-1: plot
```

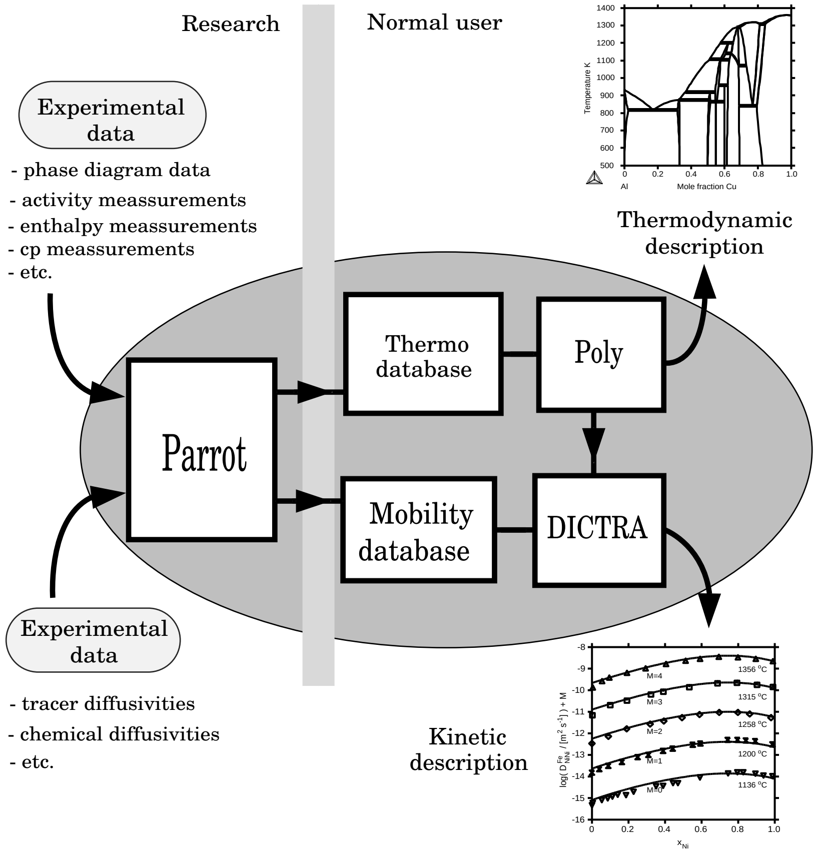
Figure f1.1

2016.05.17.15.59.01
UPPER INTERFACE OF REGION "PART#1"
CELL #1



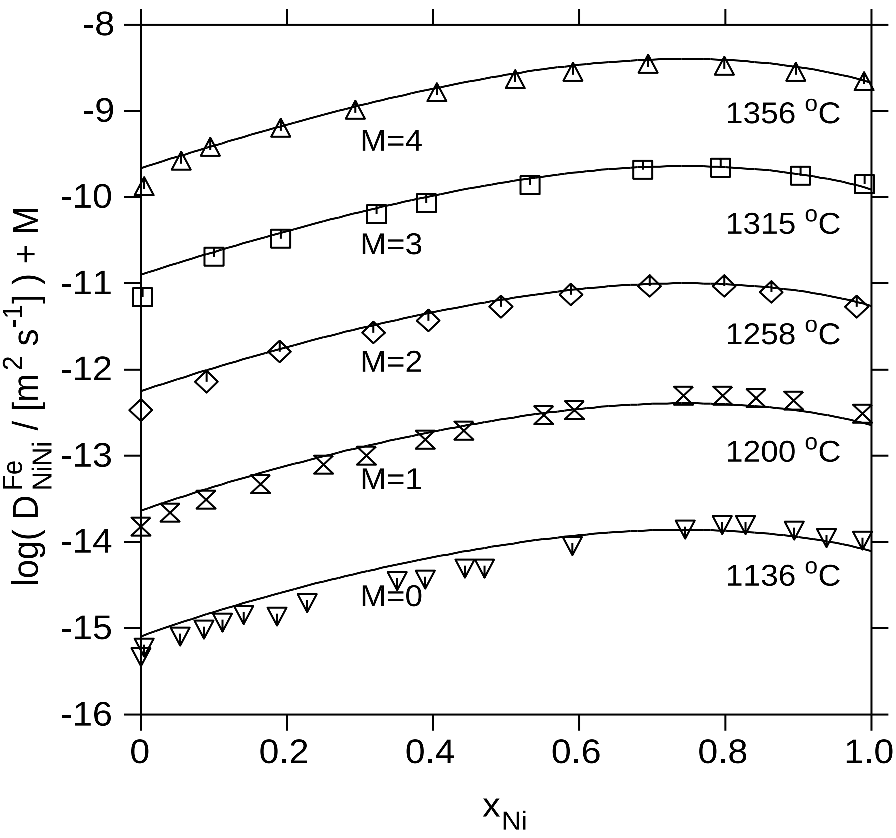

```
POST-1:
POST-1:@?<_hit_return_to_continue_>
POST-1: @@
POST-1: @@ THE DIFFENCE BETWEEN THE CALCULATION AND THE EQUATION USED BY
POST-1: @@ NISHIZAWA ET AL IS MAINLY DUE TO DIFFERENT THERMODYNAMIC DESCRIPTIONS
POST-1: @@ AND DIFFERENT DIFFUSIVITIES.
POST-1: @@
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Kinetic data



Example g1

Checking diffusivities in an Fe-Ni alloy



exg1-setup

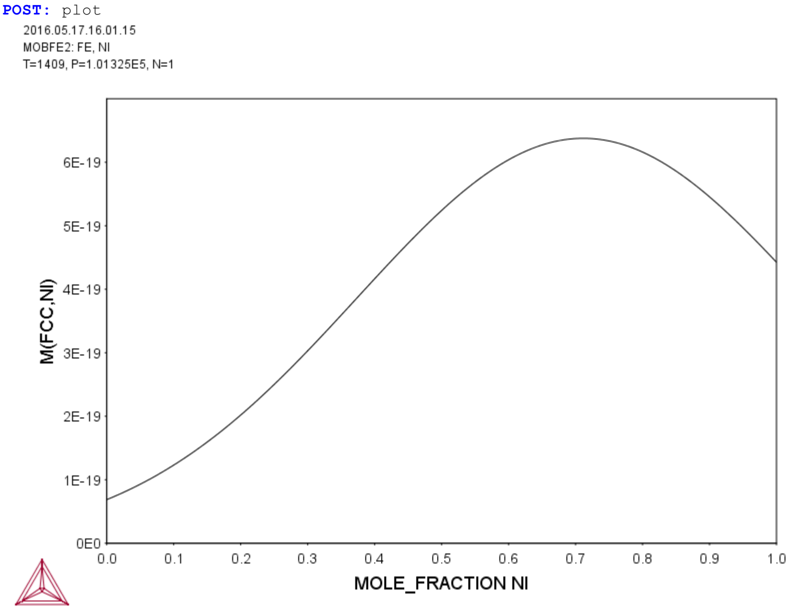
About

```
 Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exg1\setup.DCM" @@
SYS: @@ Kinetic data.
SYS: @@ A file for checking mobilities and diffusivities in an Fe-Ni alloy.
SYS: @@
SYS:
SYS: @@ exg1_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE.
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ SELECT DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw tcfe7
 Current database: Steels/Fe-Alloys  v7.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ DEFINE WHAT SYSTEM WE WANT TO WORK WITH
TDB_TCFE7: @@
TDB_TCFE7: def-sys fe ni
 FE                     NI  DEFINED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ EXCLUDE THE THERMODYNAMIC DATA FOR THE PHASES THAT IS NOT NEEDED
TDB_TCFE7: @@
TDB_TCFE7: rej ph * all
 LIQUID:L               BCC_A2                 FCC_A1
 HCP_A3                 A1_KAPPA               KAPPA
 LAVES_PHASE_C14        NBNI3                  NI3TI
   REJECTED
TDB_TCFE7: res ph fcc
 FCC_A1  RESTORED
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ RETRIEVE DATA FROM DATABASE FILE
TDB_TCFE7: @@
TDB_TCFE7: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317-425'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
     Molar volumes'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
 -OK-
TDB_TCFE7:
TDB_TCFE7: @@
TDB_TCFE7: @@ MOBILITY/DIFFUSIVITY DATA ARE STORED ON A SEPARATE DATABASE FILE.
TDB_TCFE7: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE DATA
TDB_TCFE7: @@
TDB_TCFE7: app
 Use one of these databases

 TCFE8   =  Steels/Fe-Alloys  v8.0
 TCFE9   =  Steels/Fe-Alloys  v9.0 SNAPSHOT
 FROST1  =  FROST database v1.0
 TCFE7   =  Steels/Fe-Alloys  v7.0
 TCFE6   =  Steels/Fe-Alloys  v6.2
 TCFE5   =  Steels/Fe-Alloys  v5.0
 TCFE4   =  Steels/Fe-Alloys  v4.1
 TCFE3   =  Steels/Fe-Alloys  v3.1
 TCFE2   =  Steels/Fe-Alloys  v2.1
 TCFE1   =  Steels/Fe-Alloys  v1.0
 FEDAT   =  TCS/TT Steels Database v1.0
 TCNI9   =  Ni-Alloys  v9.0 SNAPSHOT
 TCNI8   =  Ni-Alloys  v8.0
 TCNI7   =  Ni-Alloys  v7.1
 TCNI6   =  Ni-Alloys  v6.0
 TCNI5   =  Ni-Alloys  v5.1
 TCNI4   =  Ni-Alloys  v4.0
 TCNI1   =  Ni-Alloys  v1.3
 TCAL4   =  Al-Alloys  v4.0
 TCAL3   =  Al-Alloys  v3.0
 TCAL2   =  Al-Alloys  v2.0
 TCAL1   =  Al-Alloys  v1.2
 TCMG5   =  Mg-Alloys  v5.0 SNAPSHOT
 TCMG4   =  Mg-Alloys  v4.0
 TCMG3   =  Mg-Alloys  v3.0
 TCMG2   =  Mg-Alloys  v2.0
 TCMG1   =  Mg-Alloys  v1.1
 TCTI1   =  Ti-Alloys  v1.0 SNAPSHOT
 TCCU1   =  Copper  v1.0 SNAPSHOT
 TCCC1   =  Cemented carbide  v1.0
 TCHEA1  =  High Entropy Alloy  v1.0
```

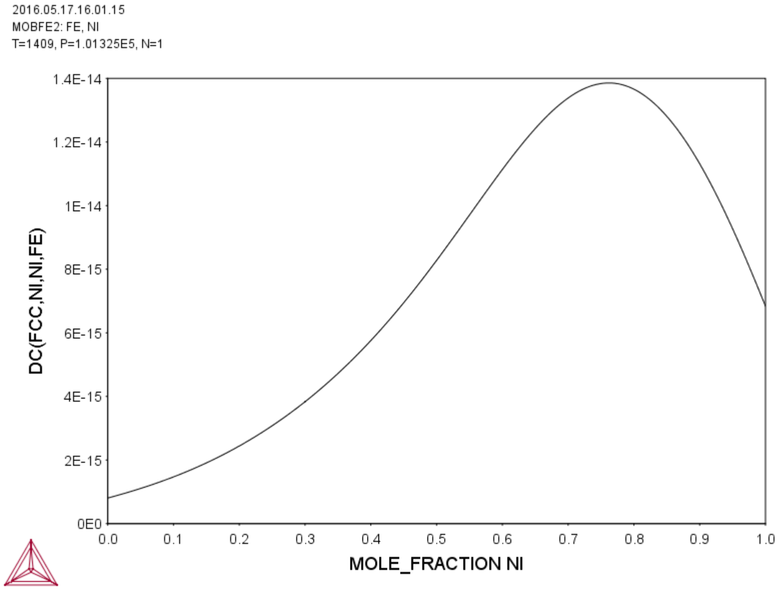
```
 SSOL5   =  SGTE Alloy Solutions Database v5.0
 SSOL4   =  SGTE Alloy Solutions Database v4.9f
 SSOL2   =  SGTE Alloy Solutions Database v2.1
 SSUB5   =  SGTE Substances Database v5.1
 SSUB4   =  SGTE Substances Database v4.1
 SSUB3   =  SGTE Substances Database v3.3
 SSUB2   =  SGTE Substances Database v2.2
 SNOB3   =  SGTE Nobel Metal Alloys Database v3.1
 SNOB2   =  SGTE Nobel Metal Alloys Database v2.1
 SNOB1   =  SGTE Nobel Metal Alloys Database v1.2
 STBC2   =  SGTE Thermal Barrier Coating TDB v2.2
 STBC1   =  SGTE Thermal Barrier Coating TDB v1.1
 SALT1   =  SGTE Molten Salts Database v1.2
 SNUX6   =  SGTE In-Vessel Nuclear Oxide TDB v6.2
 SEMC2   =  TC Semi-Conductors  v2.1
 SLAG4   =  Fe-containing Slag  v4.0 snapshot
 SLAG3   =  Fe-containing Slag  v3.2
 SLAG2   =  Fe-containing Slag  v2.2
 SLAG1   =  Fe-containing Slag  v1.2
 TCOX7   =  Metal Oxide Solutions  v7.0 SNAPSHOT
 TCOX6   =  Metal Oxide Solutions  v6.1
 TCOX5   =  Metal Oxide Solutions  v5.1
 TCOX4   =  Metal Oxide Solutions  v4.1
 ION3    =  Ionic Solutions  v3.0
 ION2    =  Ionic Solutions  v2.6
 ION1    =  Ionic Solutions  v1.5
 NOX2    =  NPL Oxide Solutions Database v2.1
 TCSLD3  =  Solder Alloys  v3.1
 TCSLD2  =  Solder Alloys  v2.0
 TCSLD1  =  Solder Alloys  v1.0
 TCSI1   =  Ultrapure Silicon  v1.1
 TCMP2   =  Materials Processing  v2.5
 TCES1   =  Combustion/Sintering  v1.1
 TCSC1   =  Super Conductor  v1.0
 TCFC1   =  SOFC Database v1.0
 TCNF2   =  Nuclear Fuels  v2.1b
 NUMT2   =  Nuclear Materials  v2.1
 NUOX4   =  Nuclear Oxides  v4.2
 NUTO1   =  U-Zr-Si Ternary Oxides TDB v1.1
 NUTA1   =  Ag-Cd-In Ternary Alloys TDB v1.1
 NUCL10  =  ThermoData NUCLEA Alloys-oxides TDB v10.2
 MEPH11  =  ThermoData MEPHISTA Nuclear Fuels TDB v11
 TCAQ2   =  Aqueous Solution  v2.5
 AQS2    =  TGG Aqueous Solution Database v2.5
 GCE2    =  TGG Geochemical/Environmental TDB v2.3
 PURE5   =  SGTE Unary (Pure Elements) TDB v5.1
 ALDEMO  =  Aluminum Demo Database
 FEDEMO  =  Iron Demo Database
 NIDEMO  =  Nickel Demo Database
 SLDEMO  =  Solder Demo Database
 OXDEMO  =  Oxide Demo Database
 SUBDEMO =  Substance Demo Database
 PTERN   =  Public Ternary Alloys TDB v1.3
 PAQ2    =  Public Aqueous Soln (SIT) TDB v2.4
 PG35    =  G35 Binary Semi-Conductors TDB v1.2
 MOB2    =  Alloys Mobility  v2.4
 MOB1    =  Alloys Mobility  v1.3
 MOBFE1  =  Steels/Fe-Alloys Mobility  v1.0
 MOBFE2  =  Steels/Fe-Alloys Mobility  v2.0
 MOBFE3  =  Steels/Fe-Alloys Mobility  v3.0
 MOBNI4  =  Ni-Alloys Mobility  v4.0
 MOBNI3  =  Ni-Alloys Mobility  v3.1
 MOBNI2  =  Ni-Alloys Mobility  v2.4
 MOBNI1  =  Ni-Alloys Mobility  v1.0
 MOBAL3  =  Al-Alloys Mobility  v3.0
 MOBAL2  =  Al-Alloys Mobility  v2.0
 MOBAL1  =  Al-Alloys Mobility  v1.0
 MOBCU1  =  Cu-Alloys Mobility  v1.0 SNAPSHOT
 MOBMG1  =  Mg-Alloys Mobility  v1.0
 MOBSI1  =  Si-Alloys Mobility  v1.0
 MOBTI1  =  Ti-Alloys Mobility  v1.0
 MALDEMO =  Al-Alloys Mobility demo database
 MFEDEMO =  Fe-Alloys Mobility demo database
 MNIDEMO =  Ni-Alloys Mobility demo database
 USER    =  User defined Database

DATABASE NAME /TCFE7/: mobfe2
 Current database: Steels/Fe-Alloys Mobility  v2.0

   TCS Steel Mobility Database Version 2.0 from 2011-12-09.

 VA  DEFINED
APP: def-sys fe ni
 FE                       NI  DEFINED
APP: rej ph * all
 BCC_A2                   FCC_A1                   HCP_A3
 LIQUID:L  REJECTED
APP: res ph fcc
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR WHERE WE WILL SETUP OUR SYSTEM
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ LIST MOBILITY DATA
DIC> @@
DIC> list-mob-data
 Sorry, LIST-DATA disabled for this database
```

```
DIC>
DIC>
DIC>
DIC>@?<Hit_return_to_continue>
DIC>
DIC> @@
DIC> @@ CHECK DIFFUSIVITES
DIC> @@
DIC> check
OUTPUT TO SCREEN OR FILE /SCREEN/:
PHASE NAME : fcc
DEPENDENT COMPONENT ? /NI/: fe
CONCENTRATION OF NI IN U-FRACTION /1/: 0.3
Pressure /100000/: 101325
Temperature /298.15/: 1409
OPTION ( dlpbmx0ez or * ) /D/: dl

  Dkj (reduced n=FE)
  k / j NI
  NI   +3.83335E-15
   L0kj = Uk*Mvak IF (kES) ELSE Uk*Yva*Mvak
  k / j   FE             NI
  FE    +9.0901E-20
  NI                   +9.0901E-20


  Volume =   1.000000000000000E-005

DIC>
DIC>
DIC>@?<Hit_return_to_continue>
DIC>
DIC> @@
DIC> @@ CALCULATE DIFFUSIVITIES VS. COMPOSITION BY STEPPING IN POLY-3
DIC> @@
DIC> go p-3
POLY_3: s-c t=1409,p=101325,n=1,x(ni)=0.3
POLY_3: c-e
 Using global minimization procedure
Calculated           137 grid points in             0 s
Found the set of lowest grid points in              0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3:
POLY_3: s-a-v
Axis number: /1/: 1
Condition /NONE/: x(ni)
Min value /0/: 0
Max value /1/: 1
Increment /.025/: 1e-3
POLY_3:
POLY_3: step
Option? /NORMAL/: normal
 No initial equilibrium, using default
 Step will start from axis value   0.300000
...OK

 Phase Region from   0.300000     for:
     FCC_A1
 Global test at  3.08000E-01 .... OK
 Global test at  3.18000E-01 .... OK
 Global test at  3.28000E-01 .... OK
 Global test at  3.38000E-01 .... OK
 Global test at  3.48000E-01 .... OK
 Global test at  3.58000E-01 .... OK
 Global test at  3.68000E-01 .... OK
 Global test at  3.78000E-01 .... OK
 Global test at  3.88000E-01 .... OK
 Global test at  3.98000E-01 .... OK
 Global test at  4.08000E-01 .... OK
 Global test at  4.18000E-01 .... OK
 Global test at  4.28000E-01 .... OK
 Global test at  4.38000E-01 .... OK
 Global test at  4.48000E-01 .... OK
 Global test at  4.58000E-01 .... OK
 Global test at  4.68000E-01 .... OK
 Global test at  4.78000E-01 .... OK
 Global test at  4.88000E-01 .... OK
 Global test at  4.98000E-01 .... OK
 Global test at  5.08000E-01 .... OK
 Global test at  5.18000E-01 .... OK
 Global test at  5.28000E-01 .... OK
 Global test at  5.38000E-01 .... OK
 Global test at  5.48000E-01 .... OK
 Global test at  5.58000E-01 .... OK
 Global test at  5.68000E-01 .... OK
 Global test at  5.78000E-01 .... OK
 Global test at  5.88000E-01 .... OK
 Global test at  5.98000E-01 .... OK
 Global test at  6.08000E-01 .... OK
 Global test at  6.18000E-01 .... OK
 Global test at  6.28000E-01 .... OK
 Global test at  6.38000E-01 .... OK
 Global test at  6.48000E-01 .... OK
 Global test at  6.58000E-01 .... OK
 Global test at  6.68000E-01 .... OK
 Global test at  6.78000E-01 .... OK
 Global test at  6.88000E-01 .... OK
 Global test at  6.98000E-01 .... OK
 Global test at  7.08000E-01 .... OK
 Global test at  7.18000E-01 .... OK
 Global test at  7.28000E-01 .... OK
 Global test at  7.38000E-01 .... OK
 Global test at  7.48000E-01 .... OK
 Global test at  7.58000E-01 .... OK
 Global test at  7.68000E-01 .... OK
 Global test at  7.78000E-01 .... OK
 Global test at  7.88000E-01 .... OK
 Global test at  7.98000E-01 .... OK
 Global test at  8.08000E-01 .... OK
 Global test at  8.18000E-01 .... OK
 Global test at  8.28000E-01 .... OK
 Global test at  8.38000E-01 .... OK
 Global test at  8.48000E-01 .... OK
 Global test at  8.58000E-01 .... OK
```

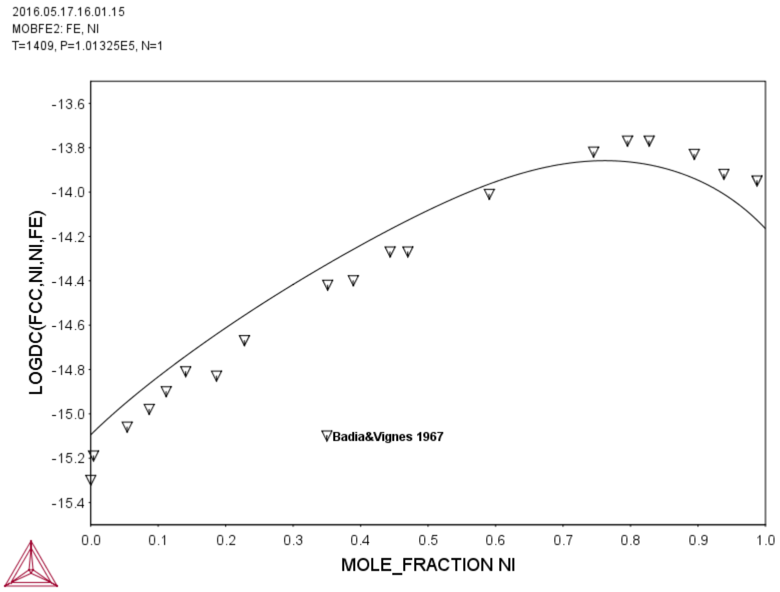
```
 Global test at  8.68000E-01 .... OK
 Global test at  8.78000E-01 .... OK
 Global test at  8.88000E-01 .... OK
 Global test at  8.98000E-01 .... OK
 Global test at  9.08000E-01 .... OK
 Global test at  9.18000E-01 .... OK
 Global test at  9.28000E-01 .... OK
 Global test at  9.38000E-01 .... OK
 Global test at  9.48000E-01 .... OK
 Global test at  9.58000E-01 .... OK
 Global test at  9.68000E-01 .... OK
 Global test at  9.78000E-01 .... OK
 Global test at  9.88000E-01 .... OK
 Global test at  9.98000E-01 .... OK
 Terminating at    1.00000
 Calculated   703 equilibria

 Phase Region from   0.300000     for:
     FCC_A1
 Global test at  2.92000E-01 .... OK
 Global test at  2.82000E-01 .... OK
 Global test at  2.72000E-01 .... OK
 Global test at  2.62000E-01 .... OK
 Global test at  2.52000E-01 .... OK
 Global test at  2.42000E-01 .... OK
 Global test at  2.32000E-01 .... OK
 Global test at  2.22000E-01 .... OK
 Global test at  2.12000E-01 .... OK
 Global test at  2.02000E-01 .... OK
 Global test at  1.92000E-01 .... OK
 Global test at  1.82000E-01 .... OK
 Global test at  1.72000E-01 .... OK
 Global test at  1.62000E-01 .... OK
 Global test at  1.52000E-01 .... OK
 Global test at  1.42000E-01 .... OK
 Global test at  1.32000E-01 .... OK
 Global test at  1.22000E-01 .... OK
 Global test at  1.12000E-01 .... OK
 Global test at  1.02000E-01 .... OK
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-11
 Calculated   303 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3:
POLY_3: @@
POLY_3: @@ ENTER POST-PROCESSOR AND PLOT THE RESULT
POLY_3: @@
POLY_3: post

  POLY-3 POSTPROCESSOR VERSION 3.2

POST:
POST:
POST:
POST: @@
POST: @@ PLOT MOBILITY OF NI VS. X(NI)
POST: @@
POST: s-d-a y m(fcc,ni)
POST: s-d-a x m-f ni
POST:
POST: plot
```




```
POST:
POST:
POST:
POST:@?<Hit_return_to_continue>
POST:
POST: @@
POST: @@ THEN PLOT DIFFUSIVITY OF NI VS. X(NI)
POST: @@
POST: s-d-a y dc(fcc,ni,ni,fe)
POST:
POST: plot
```



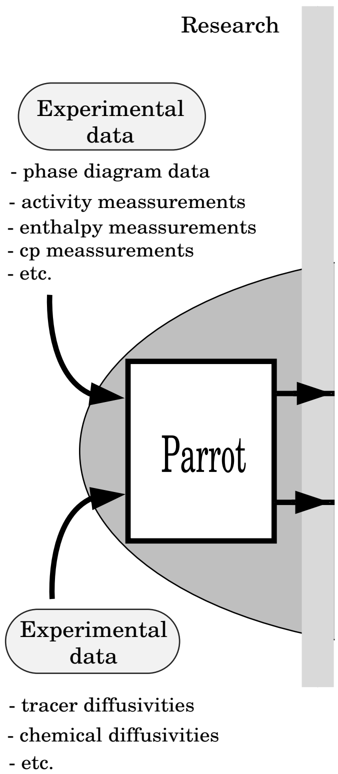

```
POST:
POST:
POST:@?<Hit_return_to_continue>
POST:
POST: @@
POST: @@ PLOT THE LOGARITHM OF DC AND APPEND EXPERIMENTAL DATA
POST: @@
POST: s-d-a y logdc(fcc,ni,ni,fe)
POST:
POST: app y feni.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 1
POST:
POST: s-s-s y n -15.5 -13.5
POST:
POST: plot
```




```
POST:
POST:
POST:@?<Hit_return_to_continue>
POST:
POST: set-inter
POST:
```

Example g2

Optimization of mobilities in Ni-Al fcc alloys



exg2-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exg2\setup.DCM" ?@@
 NO SUCH COMMAND, USE HELP
SYS: @@ Kinetic data.
SYS: @@ A file for reading thermodynamic data and setting up the kinetic
SYS: @@ parameters which are needed for an optimization of the FCC phase
SYS: @@ in the binary Ni-Al system.
SYS: @@ See also A. Engström and J. Ågren: ("Assessment of Diffusional
SYS: @@ Mobilities in Face-Centered Cubic Ni-Cr-Al Alloys" in
SYS: @@ Z. METALLKUNDE, Feb. 1996).
SYS: @@
SYS:
SYS: @@ exg2_setup.DCM
SYS:
SYS: @@
SYS: @@ RETRIEVE THERMODYNAMIC DATA FROM A USER DEFINED DATABASE
SYS: @@
SYS: go data
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw us tdata
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA  DEFINED
TDB_USER: def-sys el
ELEMENTS: al ni
 AL                       NI  DEFINED
TDB_USER: rej ph *
 LIQUID                   B2_BCC                   BCC_A2
 FCC_A1                   GAMMA_PRIME  REJECTED
TDB_USER: rest ph fcc_a1
 FCC_A1  RESTORED
TDB_USER: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
 -OK-
TDB_USER:
TDB_USER: @@
TDB_USER: @@ APPEND THE KINETIC DATA FROM THE MOBILITY DATABASE IN ORDER TO
TDB_USER: @@ HAVE SOME DUMMY PARAMETERS.
TDB_USER: @@
TDB_USER: app mob2
 Current database: Alloys Mobility  v2.4

 VA  DEFINED
 GAS:G  REJECTED
APP: def-sys
ELEMENTS: al ni
 AL                       NI  DEFINED
APP: rej ph *
 BCC_A2                   FCC_A1                   FE4N
 HCP_A3                   LIQUID:L  REJECTED
APP: res ph fcc_a1
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

   'This parameter has not been assessed'
   'A. Engstrom and J. Agren: Z. Metallkunde 87(1996)92-97;
      Al, Cr and Ni diffusion in fcc Al-Cr-Ni'
   'B. Jonsson: Scand. J. Metall. 24(1995)21-27;
      Ni self-diffusion'
 -OK-
APP:
APP: @@
APP: @@ GO TO THE DICTRA MODULE AND DEFINE THE KINETIC PARAMETERS. THE
APP: @@ VARIABLE V1,V2,V3 AND V4 ARE TO BE OPTIMIZED. NOTE THAT IF
APP: @@ YOU ARE OPTIMIZING PARAMETERS FOR A PHASE WITH MAGNETIC
APP: @@ CONTRIBUTION. I.E. USING BOTH MF- AND MQ-PARAMETERS YOU
APP: @@ MIGHT HAVE TO ENTER THE PARROT MODULE AND GO BACK BEFORE
APP: @@ ENTERING PARAMETERS CONTAINING VARIABLES.
APP: @@
APP: go dic_par

 PARROT  VERSION 5.3d RUNNING ON PC/WINDOWS NT

 Developed at the Division of Physical Metallurgy
 Royal Institute of Technology Stockholm, Sweden

PARROT:
PARROT:
PARROT: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@ MOBILITY OF AL IN AL
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&AL,AL:VA) 298.15 -142000+R*T*LN(1.71E-4); 6000 N
 MQ(FCC_A1&AL#1,AL:VA;0)
DIC>
DIC> @@ MOBILITY OF AL IN NI
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&AL,NI:VA) 298.15 -284000+R*T*LN(7.5E-4); 6000 N
 MQ(FCC_A1&AL#1,NI:VA;0)
```

```
DIC>
DIC> @@ MOBILITY OF AL INTERACTION BETWEEN AL AND NI
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&AL,AL,NI:VA;0) 298.15 V1+V2*T; 6000 N
 MQ(FCC_A1&AL#1,AL,NI:VA;0)
DIC>
DIC> @@ MOBILITY OF NI IN AL
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&NI,AL:VA) 298.15 -145900+R*T*LN(4.4E-4); 6000 N
 MQ(FCC_A1&NI#1,AL:VA;0)
DIC>
DIC> @@ MOBILITY OF NI IN NI
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&NI,NI:VA) 298.15 -287000-69.8*T; 6000 N
 MQ(FCC_A1&NI#1,NI:VA;0)
DIC>
DIC> @@ MOBILITY OF NI INTERACTION BETWEEN NI AND AL
DIC> ENTER-MOB-DATA
PARAMETER: MQ(FCC_A1&NI,NI,AL:VA;0) 298.15 V3+V4*T; 6000 N
 MQ(FCC_A1&NI#1,AL,NI:VA;0)
DIC>
DIC> @@
DIC> @@ GO TO PARROT AND SAVE SETUP ON FILE
DIC> @@
DIC> go dic_parrot

 PARROT  VERSION 5.3d RUNNING ON PC/WINDOWS NT

PARROT: create-new-store-file opt
PARROT:
PARROT: set-inter
 --OK---
PARROT:
```

exg2-run

```
PARROT:AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exg2\run.DCM" @@ exg2_run.DCM
PARROT:
PARROT: @@-----------------------------------------------------------------
PARROT: @@ FILE FOR DOING THE OPTIMIZATION IN PARROT
PARROT: @@-----------------------------------------------------------------
PARROT:
PARROT: @@
PARROT: @@ GO TO PARROT AND READ THE SETUP
PARROT: @@
PARROT: go dic_parrot

 PARROT  VERSION 5.3d RUNNING ON PC/WINDOWS NT

PARROT: set-store-file opt
PARROT:
PARROT:
PARROT: @@
PARROT: @@ COMPILE THE EXPERIMENTAL DATA IN exp.DOP INTO STRUCTURED BINARY DATA.
PARROT: @@
PARROT: compile-experiments exp
OUTPUT TO SCREEN OR FILE /SCREEN/:
INITIATE STORE FILE: /Y/:

 $---------------------------------------------------------------------
 $ DOP-FILE CONTAINING EXPERIMENTAL INFORMATION USED DURING THE
 $ OTIMIZATION IN PARROT (COMPARE WITH POP-FILE USED WHEN EVALUATING
 $ THERMODYNAMIC DATA). THE EXPERIMENTAL DATA HERE STEAM FROM A STUDY BY
 $ YAMAMOTO ET AL. TRANS. JPN. INST. MET. VOL. 21,NO. 9 (1980), P. 601.
 $
 $ CONSULT THE THERMO-CALC USER'S GUIDE TO LEARN MORE ABOUT SYNTAXES
 $ FOR OPTIMIZATION OF THERMODYNAMIC DATA.
 $---------------------------------------------------------------------

 ENTER CONST P0=101325

 TABLE_HEAD 10
 CREATE_NEW 0010,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.01055
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.6:.1
 CREATE_NEW 0011,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.02032
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.56:.1
 CREATE_NEW 0012,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.02957
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.65:.1
 CREATE_NEW 0013,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.03884
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.52:.1
 CREATE_NEW 0014,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.03884
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.52:.1
 CREATE_NEW 0015,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.04927
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.48:.1
 CREATE_NEW 0016,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.06062
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.43:.1
 CREATE_NEW 0017,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.07029
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.41:.1
 CREATE_NEW 0018,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.08113
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.37:.1
 CREATE_NEW 0019,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.09166
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.32:.1
 CREATE_NEW 0020,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.09945
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.28:.1
 CREATE_NEW 0021,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1099
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.24:.1
 CREATE_NEW 0022,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1207
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.2:.1
 CREATE_NEW 0023,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.129
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.18:.1
 CREATE_NEW 0024,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1392
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.16:.1
 CREATE_NEW 0025,1
```

```
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1503
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.1:.1
 CREATE_NEW 0026,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1589
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.08:.1
 CREATE_NEW 0027,1
 C-S PH FCC=ENT 1
 S-C T=1573,N=1,P=P0
 S-C X(AL)=.1716
 EXPERIMENT LOGDC(FCC_A1,AL,AL,NI)=-12.02:.1
 CREATE_NEW 0028,1
 C-S PH FCC=ENT 1

output ignored...



... output resumed

 V3     -1.32972465E+05   -1.32959170E+05   -1.32959170E+05    3.97449913E+00    3.97449913E+00
    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00
    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00
    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00    3.97449913E+00
    3.97449913E+00    3.97449913E+00    3.97449913E+00
 V4      7.81857821E+01    7.81857821E+01    7.81857821E+01    4.46532133E+00    4.46532133E+00
    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00
    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00
    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00    4.46532133E+00
    4.46532133E+00    4.46532133E+00    4.46532133E+00

 NUMBER OF OPTIMIZING VARIABLES :   4
 ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO
 THE SUM OF SQUARES HAS CHANGED FROM   2.02977003E+01 TO   2.02976923E+01
 DEGREES OF FREEDOM  113.  REDUCED SUM OF SQUARES   1.79625595E-01


 Sorry, LIST-DATA disabled for this database

 ======  BLOCK NUMBER  1

 DEFINED CONSTANTS
    P0=101325
10 LOGDC(F...,AL,NI)=-12.6       -12.56      1.00E-01  4.3827E-02  0.4383
11 LOGDC(F...,AL,NI)=-12.56      -12.55      1.00E-01  1.2341E-02  0.1234
12 LOGDC(F...,AL,NI)=-12.65      -12.53      1.00E-01  0.1167       1.167
13 LOGDC(F...,AL,NI)=-12.52      -12.51      1.00E-01  5.8843E-03  5.8843E-02
14 LOGDC(F...,AL,NI)=-12.52      -12.51      1.00E-01  5.8843E-03  5.8843E-02
15 LOGDC(F...,AL,NI)=-12.48      -12.49      1.00E-01 -8.4123E-03 -8.4123E-02
16 LOGDC(F...,AL,NI)=-12.43      -12.46      1.00E-01 -2.6851E-02 -0.2685
17 LOGDC(F...,AL,NI)=-12.41      -12.43      1.00E-01 -1.7821E-02 -0.1782
18 LOGDC(F...,AL,NI)=-12.37      -12.39      1.00E-01 -2.3493E-02 -0.2349
19 LOGDC(F...,AL,NI)=-12.32      -12.36      1.00E-01 -3.8626E-02 -0.3863
20 LOGDC(F...,AL,NI)=-12.28      -12.33      1.00E-01 -5.1933E-02 -0.5193
21 LOGDC(F...,AL,NI)=-12.24      -12.29      1.00E-01 -5.4919E-02 -0.5492
22 LOGDC(F...,AL,NI)=-12.2       -12.26      1.00E-01 -5.5128E-02 -0.5513
23 LOGDC(F...,AL,NI)=-12.18      -12.22      1.00E-01 -4.3400E-02 -0.4340
24 LOGDC(F...,AL,NI)=-12.16      -12.18      1.00E-01 -2.2927E-02 -0.2293
25 LOGDC(F...,AL,NI)=-12.1       -12.14      1.00E-01 -3.6888E-02 -0.3689
26 LOGDC(F...,AL,NI)=-12.08      -12.10      1.00E-01 -1.9703E-02 -0.1970
27 LOGDC(F...,AL,NI)=-12.02      -12.04      1.00E-01 -2.2272E-02 -0.2227
28 LOGDC(F...,AL,NI)=-11.98      -11.99      1.00E-01 -9.0917E-03 -9.0917E-02
29 LOGDC(F...,AL,NI)=-11.94      -11.95      1.00E-01 -9.1618E-03 -9.1618E-02
30 LOGDC(F...,AL,NI)=-13         -12.86      1.00E-01  0.1360       1.360
31 LOGDC(F...,AL,NI)=-12.96      -12.85      1.00E-01  0.1059       1.059
32 LOGDC(F...,AL,NI)=-12.92      -12.84      1.00E-01  8.4799E-02  0.8480
33 LOGDC(F...,AL,NI)=-12.9       -12.82      1.00E-01  8.2549E-02  0.8255
34 LOGDC(F...,AL,NI)=-12.77      -12.79      1.00E-01 -1.5173E-02 -0.1517
35 LOGDC(F...,AL,NI)=-12.74      -12.75      1.00E-01 -1.1192E-02 -0.1119
36 LOGDC(F...,AL,NI)=-12.82      -12.72      1.00E-01  9.5223E-02  0.9522
37 LOGDC(F...,AL,NI)=-12.82      -12.72      1.00E-01  9.5223E-02  0.9522
38 LOGDC(F...,AL,NI)=-12.69      -12.68      1.00E-01  5.7856E-03  5.7856E-02
39 LOGDC(F...,AL,NI)=-12.65      -12.65      1.00E-01  2.6229E-03  2.6229E-02
40 LOGDC(F...,AL,NI)=-12.64      -12.62      1.00E-01  1.9222E-02  0.1922
41 LOGDC(F...,AL,NI)=-12.61      -12.58      1.00E-01  2.8492E-02  0.2849
42 LOGDC(F...,AL,NI)=-12.55      -12.54      1.00E-01  9.2091E-03  9.2091E-02
43 LOGDC(F...,AL,NI)=-12.53      -12.51      1.00E-01  2.1034E-02  0.2103
44 LOGDC(F...,AL,NI)=-12.47      -12.47      1.00E-01  2.0402E-03  2.0402E-02
45 LOGDC(F...,AL,NI)=-12.41      -12.42      1.00E-01 -1.0621E-02 -0.1062
46 LOGDC(F...,AL,NI)=-12.38      -12.38      1.00E-01 -1.6323E-04 -1.6323E-03
47 LOGDC(F...,AL,NI)=-12.36      -12.32      1.00E-01  3.6255E-02  0.3626
48 LOGDC(F...,AL,NI)=-12.36      -12.32      1.00E-01  3.6255E-02  0.3626
49 LOGDC(F...,AL,NI)=-12.3       -12.27      1.00E-01  3.0494E-02  0.3049
50 LOGDC(F...,AL,NI)=-13.23      -13.19      1.00E-01  3.8416E-02  0.3842
51 LOGDC(F...,AL,NI)=-13.23      -13.19      1.00E-01  3.8416E-02  0.3842
52 LOGDC(F...,AL,NI)=-13.19      -13.18      1.00E-01  1.3209E-02  0.1321
53 LOGDC(F...,AL,NI)=-13.15      -13.16      1.00E-01 -5.5530E-03 -5.5530E-02
54 LOGDC(F...,AL,NI)=-13.12      -13.13      1.00E-01 -1.4736E-02 -0.1474
55 LOGDC(F...,AL,NI)=-13.09      -13.10      1.00E-01 -1.2913E-02 -0.1291
56 LOGDC(F...,AL,NI)=-13.06      -13.07      1.00E-01 -6.9397E-03 -6.9397E-02
57 LOGDC(F...,AL,NI)=-13.02      -13.04      1.00E-01 -1.5971E-02 -0.1597
58 LOGDC(F...,AL,NI)=-12.99      -13.00      1.00E-01 -1.0672E-02 -0.1067
59 LOGDC(F...,AL,NI)=-12.96      -12.96      1.00E-01  8.0069E-04  8.0069E-03
60 LOGDC(F...,AL,NI)=-12.91      -12.93      1.00E-01 -2.0470E-02 -0.2047
61 LOGDC(F...,AL,NI)=-12.88      -12.89      1.00E-01 -7.3947E-03 -7.3947E-02
62 LOGDC(F...,AL,NI)=-12.86      -12.85      1.00E-01  1.4312E-02  0.1431
63 LOGDC(F...,AL,NI)=-12.86      -12.85      1.00E-01  1.4312E-02  0.1431
64 LOGDC(F...,AL,NI)=-12.83      -12.81      1.00E-01  1.7076E-02  0.1708
65 LOGDC(F...,AL,NI)=-12.8       -12.77      1.00E-01  2.9558E-02  0.2956
66 LOGDC(F...,AL,NI)=-12.75      -12.72      1.00E-01  2.5293E-02  0.2529
67 LOGDC(F...,AL,NI)=-12.71      -12.68      1.00E-01  2.6861E-02  0.2686
68 LOGDC(F...,AL,NI)=-12.67      -12.63      1.00E-01  4.4033E-02  0.4403
70 LOGDC(F...,AL,NI)=-13.5       -13.54      1.00E-01 -4.2456E-02 -0.4246
71 LOGDC(F...,AL,NI)=-13.47      -13.53      1.00E-01 -5.5160E-02 -0.5516
72 LOGDC(F...,AL,NI)=-13.45      -13.50      1.00E-01 -5.0384E-02 -0.5038
73 LOGDC(F...,AL,NI)=-13.42      -13.48      1.00E-01 -5.6139E-02 -0.5614
74 LOGDC(F...,AL,NI)=-13.39      -13.44      1.00E-01 -5.3034E-02 -0.5303
75 LOGDC(F...,AL,NI)=-13.36      -13.40      1.00E-01 -4.3865E-02 -0.4386
76 LOGDC(F...,AL,NI)=-13.34      -13.37      1.00E-01 -3.0943E-02 -0.3094
77 LOGDC(F...,AL,NI)=-13.31      -13.33      1.00E-01 -2.0539E-02 -0.2054
78 LOGDC(F...,AL,NI)=-13.24      -13.29      1.00E-01 -5.0311E-02 -0.5031
79 LOGDC(F...,AL,NI)=-13.22      -13.26      1.00E-01 -4.1318E-02 -0.4132
```

```
80 LOGDC(F...,AL,NI)=-13.19       -13.22      1.00E-01 -2.6230E-02 -0.2623
81 LOGDC(F...,AL,NI)=-13.13       -13.17      1.00E-01 -4.3098E-02 -0.4310
82 LOGDC(F...,AL,NI)=-13.12       -13.14      1.00E-01 -1.9773E-02 -0.1977
83 LOGDC(F...,AL,NI)=-13.08       -13.09      1.00E-01 -1.4457E-02 -0.1446
84 LOGDC(F...,AL,NI)=-13.04       -13.05      1.00E-01 -9.4618E-03 -9.4618E-02
85 LOGDC(F...,AL,NI)=-13.03       -13.01      1.00E-01  2.2404E-02  0.2240
90 LOGDC(F...,AL,NI)=-13.97       -13.92      1.00E-01  5.1241E-02  0.5124
91 LOGDC(F...,AL,NI)=-13.92       -13.90      1.00E-01  2.0497E-02  0.2050
92 LOGDC(F...,AL,NI)=-13.88       -13.87      1.00E-01  9.3921E-03  9.3921E-02
93 LOGDC(F...,AL,NI)=-13.85       -13.84      1.00E-01  5.2264E-03  5.2264E-02
94 LOGDC(F...,AL,NI)=-13.82       -13.81      1.00E-01  1.1044E-02  0.1104
95 LOGDC(F...,AL,NI)=-13.78       -13.77      1.00E-01  1.3352E-02  0.1335
96 LOGDC(F...,AL,NI)=-13.9        -13.74      1.00E-01  0.1640       1.640
97 LOGDC(F...,AL,NI)=-13.85       -13.69      1.00E-01  0.1576       1.576
98 LOGDC(F...,AL,NI)=-13.65       -13.65      1.00E-01  1.5604E-03  1.5604E-02
99 LOGDC(F...,AL,NI)=-13.62       -13.62      1.00E-01  4.9052E-03  4.9052E-02
100 LOGDC(F...,AL,NI)=-13.57      -13.57      1.00E-01 -9.5589E-04 -9.5589E-03
101 LOGDC(F...,AL,NI)=-13.52      -13.53      1.00E-01 -5.4819E-03 -5.4819E-02
102 LOGDC(F...,AL,NI)=-13.47      -13.49      1.00E-01 -2.0262E-02 -0.2026
103 LOGDC(F...,AL,NI)=-13.45      -13.45      1.00E-01  4.9628E-03  4.9628E-02
104 LOGDC(F...,AL,NI)=-13.4       -13.40      1.00E-01  2.8073E-03  2.8073E-02
110 LOGDC(F...,AL,NI)=-14.32      -14.32      1.00E-01 -3.9760E-03 -3.9760E-02
111 LOGDC(F...,AL,NI)=-14.32      -14.30      1.00E-01  1.9307E-02  0.1931
112 LOGDC(F...,AL,NI)=-14.28      -14.27      1.00E-01  1.0664E-02  0.1066
113 LOGDC(F...,AL,NI)=-14.25      -14.24      1.00E-01  8.9412E-03  8.9412E-02
114 LOGDC(F...,AL,NI)=-14.22      -14.20      1.00E-01  1.8329E-02  0.1833
115 LOGDC(F...,AL,NI)=-14.17      -14.16      1.00E-01  1.2994E-02  0.1299
116 LOGDC(F...,AL,NI)=-14.15      -14.12      1.00E-01  3.0489E-02  0.3049
117 LOGDC(F...,AL,NI)=-14.1       -14.07      1.00E-01  2.5593E-02  0.2559
118 LOGDC(F...,AL,NI)=-14.03      -14.03      1.00E-01 -1.4483E-03 -1.4483E-02
119 LOGDC(F...,AL,NI)=-14         -14.00      1.00E-01  2.7074E-03  2.7074E-02
120 LOGDC(F...,AL,NI)=-13.95      -13.95      1.00E-01 -1.9282E-03 -1.9282E-02
121 LOGDC(F...,AL,NI)=-13.9       -13.91      1.00E-01 -5.2349E-03 -5.2349E-02
122 LOGDC(F...,AL,NI)=-13.85      -13.87      1.00E-01 -1.6645E-02 -0.1665
130 LOGDC(F...,AL,NI)=-14.73      -14.76      1.00E-01 -3.0493E-02 -0.3049
131 LOGDC(F...,AL,NI)=-14.71      -14.73      1.00E-01 -2.3757E-02 -0.2376
132 LOGDC(F...,AL,NI)=-14.68      -14.70      1.00E-01 -1.9446E-02 -0.1945
133 LOGDC(F...,AL,NI)=-14.66      -14.67      1.00E-01 -8.5997E-03 -8.5997E-02
134 LOGDC(F...,AL,NI)=-14.61      -14.63      1.00E-01 -1.5314E-02 -0.1531
135 LOGDC(F...,AL,NI)=-14.61      -14.63      1.00E-01 -1.5314E-02 -0.1531
136 LOGDC(F...,AL,NI)=-14.58      -14.58      1.00E-01  3.4408E-03  3.4408E-02
137 LOGDC(F...,AL,NI)=-14.54      -14.54      1.00E-01  3.3897E-03  3.3897E-02
138 LOGDC(F...,AL,NI)=-14.5       -14.49      1.00E-01  9.2113E-03  9.2113E-02
139 LOGDC(F...,AL,NI)=-14.46      -14.44      1.00E-01  1.5390E-02  0.1539
140 LOGDC(F...,AL,NI)=-14.41      -14.41      1.00E-01  1.6005E-03  1.6005E-02
141 LOGDC(F...,AL,NI)=-14.35      -14.36      1.00E-01 -1.1351E-02 -0.1135
142 LOGDC(F...,AL,NI)=-14.27      -14.31      1.00E-01 -4.1487E-02 -0.4149
143 LOGDC(F...,AL,NI)=-14.2       -14.27      1.00E-01 -7.4774E-02 -0.7477


PARROT:
PARROT:
PARROT: set-inter
 --OK---
PARROT:
```

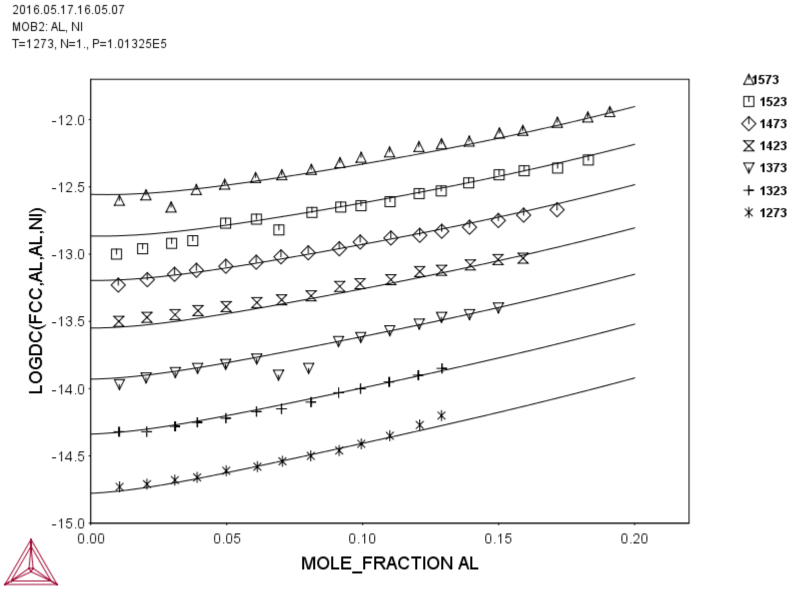
exg2-plot

```
PARROT:AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exg2\plot.DCM" @@ exg2_plot.DCM
PARROT:
PARROT: @@----------------------------------------------------------------
PARROT: @@ FILE FOR PLOTTING THE RESULT AFTER THE OPTIMIZATION. HERE
PARROT: @@ DIFFUSIVITIES CALCULATED FROM THE OPTIMIZED VARIABLES ARE
PARROT: @@ COMPARED WITH EXPERIMENTALLY MEASURED ONES.
PARROT: @@----------------------------------------------------------------
PARROT:
PARROT: @@
PARROT: @@ GO TO PARROT AND READ THE FILE CONTAINING THE RESULT FROM
PARROT: @@ THE OPTIMIZATION.
PARROT: @@
PARROT: go dic_parrot

 PARROT  VERSION 5.3d RUNNING ON PC/WINDOWS NT

PARROT: set-store-file opt
PARROT:
PARROT: @@
PARROT: @@ GO TO POLY3 AND STEP IN X(AL)
PARROT: @@
PARROT: go p-3
POLY_3: s-c n=1,p=101325,t=1573
POLY_3: s-c x(al)=.1
POLY_3: c-e,,,,
 Using global minimization procedure
Calculated           137 grid points in              0 s
Found the set of lowest grid points in               0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3: add,,
POLY_3:
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: @@
POLY_3: @@ REPEATE THE PROCEDURE FOR SOME OTHER TEMPERATURES.
POLY_3: @@
POLY_3: s-c t=1523,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in              0 s
Found the set of lowest grid points in               0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
```

```
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: s-c t=1473,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in              0 s
Found the set of lowest grid points in               0 s
 Calculated POLY solution       0 s, total time      0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: s-c t=1423,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in              0 s
Found the set of lowest grid points in               0 s
 Calculated POLY solution       0 s, total time      0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: s-c t=1373,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in              0 s
Found the set of lowest grid points in               0 s
 Calculated POLY solution       0 s, total time      0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
```

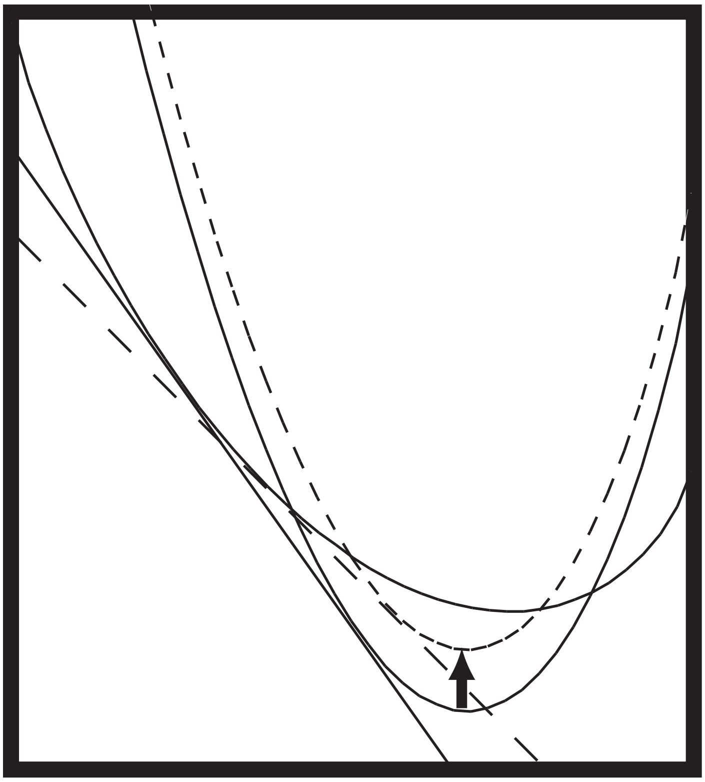
```
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: s-c t=1323,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in             0 s
Found the set of lowest grid points in              0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: s-c t=1273,x(al)=.1
POLY_3: c-e,,,,,,,
 Using global minimization procedure
Calculated           137 grid points in             0 s
Found the set of lowest grid points in              0 s
 Calculated POLY solution       0 s, total time     0  s
POLY_3: s-a-v 1 x(al) 0.0001 .20 0.001,,,,
POLY_3: step
Option? /NORMAL/:
 Step will start from axis value   0.100000E+00
...OK

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  1.08000E-01 .... OK
 Global test at  1.18000E-01 .... OK
 Global test at  1.28000E-01 .... OK
 Global test at  1.38000E-01 .... OK
 Global test at  1.48000E-01 .... OK
 Global test at  1.58000E-01 .... OK
 Global test at  1.68000E-01 .... OK
 Global test at  1.78000E-01 .... OK
 Global test at  1.88000E-01 .... OK
 Global test at  1.98000E-01 .... OK
 Terminating at   0.200000
 Calculated   103 equilibria

 Phase Region from   0.100000E+00 for:
     FCC_A1
 Global test at  9.20000E-02 .... OK
 Global test at  8.20000E-02 .... OK
 Global test at  7.20000E-02 .... OK
 Global test at  6.20000E-02 .... OK
 Global test at  5.20000E-02 .... OK
 Global test at  4.20000E-02 .... OK
 Global test at  3.20000E-02 .... OK
 Global test at  2.20000E-02 .... OK
 Global test at  1.20000E-02 .... OK
 Global test at  2.00000E-03 .... OK
 Terminating at   0.100000E-03
 Calculated   103 equilibria
 *** Buffer saved on file: C:\Users\test\Documents\RESULT_002.POLY3
POLY_3: @@
POLY_3: @@ ENTER THE POST MODULE, PLOT THE DIFFUSIVITY ON THE Y-AXIS
POLY_3: @@ AND MOLE-FRACTION AL ON THE X-AXIS.
POLY_3: @@
POLY_3: post

  POLY-3 POSTPROCESSOR VERSION 3.2

POST:
POST: s-d-a x m-f al
POST: s-d-a y logdc(fcc,al,al,ni)
POST:
POST: app y yama.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 1 2 3 4 5 6 7
POST:
POST:
POST: s-t-m-s y
```

```
 COMMAND NOT SUPPORTED IN THIS PLOT DRIVER
POST: s-s-s y n -15 -11.7
POST: s-t-m-s y
 COMMAND NOT SUPPORTED IN THIS PLOT DRIVER
POST: plo
```



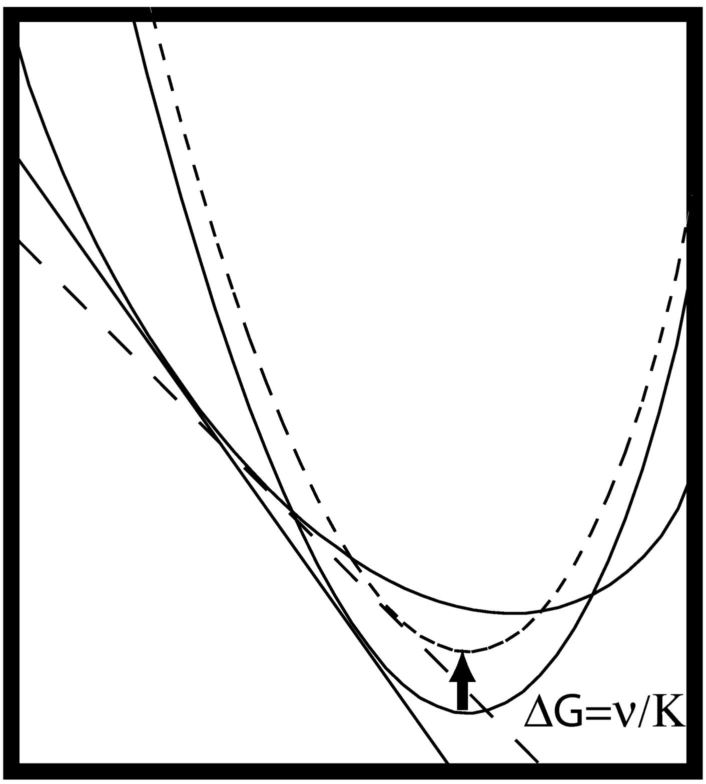

```
POST:
POST:@?<Hit_return_to_continue>
POST:
POST: set-inter
POST:
```

Deviation from local equilibrium



Example h1

α/γ diffusion couple with limited interface mobilitty



exh1-setup

```
About Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh1\setup.DCM" @@
SYS: @@ Deviation from local equilibrium.
SYS: @@ Setup file for calculating the growth of FERRITE into AUSTENITE
SYS: @@ with a limited interface mobility.
SYS: @@ This is achieved by adding a Gibbs-energy contribution to the FERRITE
SYS: @@ using the SET-SURFACE-ENERGY command.
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE A PUBLIC DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw FEDEMO
 Current database: Iron Demo Database

 VA                      /-  DEFINED
TDB_FEDEMO: def-sys fe c
 FE                      C  DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                   LIQUID:L                BCC_A2
 CEMENTITE               DIAMOND_FCC_A4          FCC_A1
 GRAPHITE                HCP_A3                  KSI_CARBIDE
 LAVES_PHASE_C14         M23C6                   M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph bcc fcc
 BCC_A2                  FCC_A1  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe c
 FE                      C  DEFINED
APP: rej ph * all
 BCC_A2                  FCC_A1  REJECTED
APP: res ph bcc fcc
 BCC_A2                  FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> set-cond glob T 0 1000; * N

DIC>
DIC> @@
DIC> @@ WE START BY ENTERING REGION ferrite AND austenite WHEREIN WE
DIC> @@ PUT THE BCC AND FCC PHASE RESPECTIVELY. THE ferrite REGION IS
DIC> @@ ASSUMED INITIALLY TO BE VERY THIN, 1E-9 METERS.
DIC> @@
```

```
DIC> enter-region
REGION NAME : ferrite
DIC>
DIC> enter-region
REGION NAME : austenite
ATTACH TO REGION NAMED /FERRITE/:
ATTACHED TO THE RIGHT OF FERRITE /YES/:
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO THE REGIONS.
DIC> @@
DIC> enter-grid
REGION NAME : /FERRITE/: ferrite
WIDTH OF REGION /1/: 1e-9
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC> enter-grid austenite
WIDTH OF REGION /1/: 0.999e-6
TYPE /LINEAR/: geo
NUMBER OF POINTS /50/: 100
VALUE OF R IN THE GEOMETRICAL SERIE : 1.07
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /FERRITE/: ferrite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: bcc
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO BCC.
DIC> @@
DIC> enter-composition
REGION NAME : /FERRITE/: ferrite
PHASE NAME: /BCC_A2/: bcc
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.019091893
VALUE OF LAST POINT : /1.9091893E-2/: 0.019091893
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO FCC.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: C
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.0191
VALUE OF LAST POINT : /1.91E-2/: 0.0191
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exh1 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

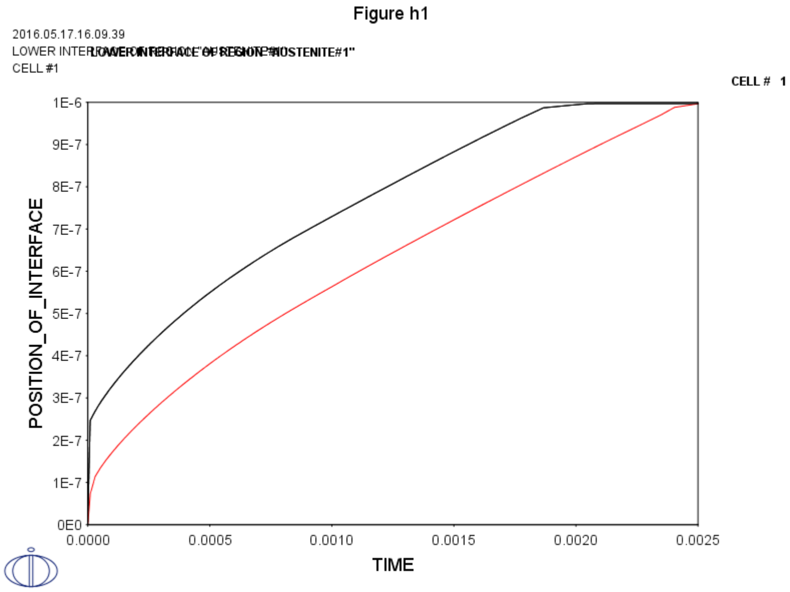
exh1-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh1\run.DCM"
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE AND START THE SIMULATION
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> read exh1
 OK
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-sim-time
END TIME FOR INTEGRATION /.1/: 2.5E-3
AUTOMATIC TIMESTEP CONTROL /YES/: YES
MAX TIMESTEP DURING INTEGRATION /2.5E-04/:
INITIAL TIMESTEP : /1E-07/: 1E-5
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/: 1E-5
DIC>
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                    3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = 8.8825328568563E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]
 U-FRACTION IN SYSTEM:  C = 8.8825328568563E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]
  5  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE
 3.006330312986683E-004        3.006931962453742E-004        1.304180087454165E-013        1.025447727641494E-013        3.501040022804090E-
013        3.797883399461895E-014        1.922744244304792E-014        1.204461245269159E-014        9.962062312514671E-
015        9.187320485702992E-015        8.901938292313761E-015        8.720949022385942E-015        8.572889944889849E-
015        8.381986793162124E-015        8.092563277096857E-015        7.529295395707530E-015        6.463714405341433E-
015        4.576341385080086E-015        1.776751180038911E-015        2.938735877055719E-039
 TIME =  0.10000000E-04 DT =  0.10000000E-04 SUM OF SQUARES =  0.29387359E-38
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.24565862E-01 AND   0.24565862E-01
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.24665862E-06
 U-FRACTION IN SYSTEM:  C = 8.88752500162862E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  42  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE
  18  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: AUSTENITE

  CPU time used in timestep                    0  seconds
 9.532451674653315E-005        9.534358260332851E-005        1.08068355139141        9.355887468730204E-005        9.268224311091822E-
005        9.224547468668426E-005        9.180973783778337E-005        9.135136304817618E-005        9.043805482947239E-
005        8.862520383435556E-005        8.505456360627941E-005        7.813353020599762E-005        6.517245162593472E-
005        4.277424734667033E-005        1.207365031208958E-005         1.08068361880292        1.199334109354295E-
005        1.195328697855534E-005        1.193328504467595E-005        1.192329035865516E-005        1.191329985982370E-
005        1.190201907839078E-005        1.187947354573692E-005        1.183444660191465E-005        1.174464920011318E-
005        1.156608033936990E-005        1.121304638995359E-005        1.052339357861011E-005        9.209748306814319E-
006        6.845099166613397E-006        3.166366499126212E-006        0.201290660352526        3.142878360386035E-
006        3.131167081907244E-006        3.125319640347416E-006        3.121868524172648E-006        3.118419314444282E-
006        3.114459257272476E-006        3.106546690998612E-006        3.090751750654812E-006        3.059282638977348E-
006        2.996827491681859E-006        2.873849501365664E-006        2.635622737834605E-006        2.190086078898572E-006

output ignored...


... output resumed

 7.432103844403325E-023
 TIME =  0.21222907E-02 DT =  0.73893022E-04 SUM OF SQUARES =  0.74321038E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.28748209E-03 AND   0.28748209E-03
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.90647166E-06
 U-FRACTION IN SYSTEM:  C = 8.88677648686597E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  CPU time used in timestep                    1  seconds
 3.476561673007225E-010        3.369925775545053E-010        9.855477591072671E-014        3.133800791222436E-017
 TIME =  0.21979081E-02 DT =  0.75617426E-04 SUM OF SQUARES =  0.31338008E-16
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.28358648E-03 AND   0.28358648E-03
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.92791574E-06
 U-FRACTION IN SYSTEM:  C = 8.88677711773427E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  CPU time used in timestep                    1  seconds
 1.676034518872772E-011        1.455040594044796E-011        2.494553079822881E-016        4.357153774937472E-021
 TIME =  0.22762186E-02 DT =  0.78310462E-04 SUM OF SQUARES =  0.43571538E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.27832349E-03 AND   0.27832349E-03
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.94971138E-06
 U-FRACTION IN SYSTEM:  C = 8.88677740521273E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  CPU time used in timestep                    0  seconds
 1.827911728200690E-008        1.818060630557994E-008        2.026154225165427E-010        2.925430507729969E-012        4.389232822615135E-
016        1.011131006932736E-021
 TIME =  0.23501472E-02 DT =  0.73928623E-04 SUM OF SQUARES =  0.10111310E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.28421956E-03 AND   0.28421956E-03
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.97072335E-06
 U-FRACTION IN SYSTEM:  C = 8.8867770869897E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    2  seconds
```

```
 7.644226645544558E-007        7.630848369033542E-007        3.904226658072104E-008        3.839482368746872E-009        4.710604457588960E-
011        6.898694713180442E-014        1.463131832719687E-018
 TIME =  0.24038784E-02 DT =  0.53731175E-04 SUM OF SQUARES =  0.14631318E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.32443557E-03 AND   0.32443557E-03
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.98815565E-06
 U-FRACTION IN SYSTEM:  C = 9.84480196224329E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                     1  seconds
 7.143507555829430E-007        7.145371615626139E-007        1.504569084566396E-009        5.632607488467333E-011        1.078019278208866E-
016        1.762934283370765E-023
 TIME =  0.25000000E-02 DT =  0.96121601E-04 SUM OF SQUARES =  0.17629343E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.87504498E-04 AND   0.87504498E-04
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.99656672E-06
 U-FRACTION IN SYSTEM:  C = 9.8570447651847E-04  FE = 1
 TOTAL SIZE OF SYSTEM:  1E-06 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-04
 DELETING TIME-RECORD FOR TIME   0.30000000E-04
 DELETING TIME-RECORD FOR TIME   0.51388245E-04
 DELETING TIME-RECORD FOR TIME   0.74132378E-04
 DELETING TIME-RECORD FOR TIME   0.98215846E-04
 DELETING TIME-RECORD FOR TIME   0.12363037E-03
 DELETING TIME-RECORD FOR TIME   0.15037637E-03
 DELETING TIME-RECORD FOR TIME   0.17846378E-03
 DELETING TIME-RECORD FOR TIME   0.20791109E-03
 DELETING TIME-RECORD FOR TIME   0.23874321E-03
 DELETING TIME-RECORD FOR TIME   0.27099622E-03
 DELETING TIME-RECORD FOR TIME   0.30472410E-03
 DELETING TIME-RECORD FOR TIME   0.33998236E-03
 DELETING TIME-RECORD FOR TIME   0.37686107E-03
 DELETING TIME-RECORD FOR TIME   0.41545357E-03
 DELETING TIME-RECORD FOR TIME   0.45589548E-03
 DELETING TIME-RECORD FOR TIME   0.49832718E-03
 DELETING TIME-RECORD FOR TIME   0.54296201E-03
 DELETING TIME-RECORD FOR TIME   0.59004053E-03
 DELETING TIME-RECORD FOR TIME   0.63985836E-03
 DELETING TIME-RECORD FOR TIME   0.69282670E-03
 DELETING TIME-RECORD FOR TIME   0.74940641E-03
 DELETING TIME-RECORD FOR TIME   0.80982854E-03
 DELETING TIME-RECORD FOR TIME   0.87440408E-03
 DELETING TIME-RECORD FOR TIME   0.94129663E-03
 DELETING TIME-RECORD FOR TIME   0.10083126E-02
 DELETING TIME-RECORD FOR TIME   0.10754807E-02
 DELETING TIME-RECORD FOR TIME   0.11430974E-02
 DELETING TIME-RECORD FOR TIME   0.12110373E-02
 DELETING TIME-RECORD FOR TIME   0.12791546E-02
 DELETING TIME-RECORD FOR TIME   0.13473364E-02
 DELETING TIME-RECORD FOR TIME   0.14157626E-02
 DELETING TIME-RECORD FOR TIME   0.14844509E-02
 DELETING TIME-RECORD FOR TIME   0.15533751E-02
 DELETING TIME-RECORD FOR TIME   0.16226888E-02
 DELETING TIME-RECORD FOR TIME   0.16923387E-02
 DELETING TIME-RECORD FOR TIME   0.17624052E-02
 DELETING TIME-RECORD FOR TIME   0.18328775E-02
 DELETING TIME-RECORD FOR TIME   0.19039611E-02
 DELETING TIME-RECORD FOR TIME   0.19757368E-02
 DELETING TIME-RECORD FOR TIME   0.20483977E-02
 DELETING TIME-RECORD FOR TIME   0.21222907E-02
 DELETING TIME-RECORD FOR TIME   0.21979081E-02
 DELETING TIME-RECORD FOR TIME   0.22762186E-02
 DELETING TIME-RECORD FOR TIME   0.23501472E-02

 KEEPING TIME-RECORD FOR TIME   0.24038784E-02
 AND FOR TIME                   0.25000000E-02
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exh1-plot

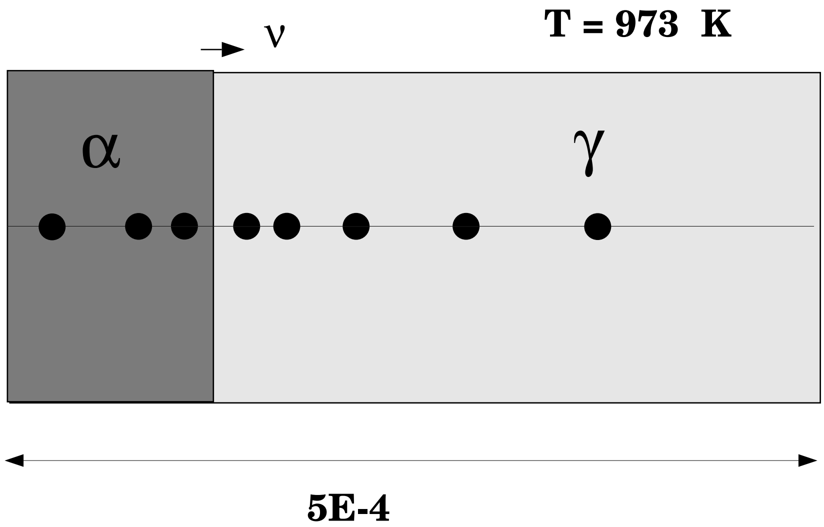
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh1\plot.DCM"
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  2.50000E-03
DIC>
DIC> read exh1
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ SET THE DATA APPENDED FROM THE "EXP"-FILE TO BE RED
POST-1: @@
POST-1: set-col for for red
 COMMAND NOT SUPPORTED IN THIS PLOT DRIVER
POST-1:
POST-1: @@
POST-1: @@ WE WILL COMPARE THE POSITION OF THE INTERFACE AS A FUNCTION OF TIME
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y posi aus low
POST-1:
POST-1: @@
POST-1: @@ APPEND THE SIMULATION WITHOUT THE ENERGY CONTRIBUTION FROM FILE
POST-1: @@
POST-1: app y noadd.exp 1; 1
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-title Figure h1
POST-1:
POST-1: @@
POST-1: @@ PLOT THE RESULTS
POST-1: @@
POST-1: plot
```




```
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example h2

α/γ para-equilibrium in a Fe-Ni-C alloy



exh2-setup

About Linked: Tue May 17 11:39:18 2016

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh2\setup.DCM" @@
SYS: @@ Deviation from local equilibrium
SYS: @@ Setup file for calculation of the growth of FERRITE into AUSTENITE
SYS: @@ in an Fe-2.02%Ni-0.0885%C alloy using the para-equilibrium model.
SYS: @@ The results are compared with experimental information from
SYS: @@ Hutchinson, C. R., A. Fuchsmann, and Yves Brechet. "The diffusional
SYS: @@ formation of ferrite from austenite in Fe-C-Ni alloys." Metall.
SYS: @@  Mat. Trans. A 35.4 (2004): 1211-1221.
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ SELECT DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                     /-  DEFINED
TDB_FEDEMO: def-sys fe ni c
 FE                     NI                     C
   DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                  LIQUID:L               BCC_A2
 CEMENTITE              DIAMOND_FCC_A4         FCC_A1
 GRAPHITE               HCP_A3                 KSI_CARBIDE
 LAVES_PHASE_C14        M23C6                  M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph bcc fcc
 BCC_A2                 FCC_A1  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Gabriel, C. Chatillon, and I. Ansara, published in High Temp. Sci.
      (parameters listed in Calphad, 11 (1987), 203-218); C-NI'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'A. Gabriel, P. Gustafson, and I. Ansara, Calphad, 11 (1987), 203 -218;
      TRITA-MAC 285 (1986); C-FE-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe ni c
 FE                     NI                     C
   DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph bcc fcc
 BCC_A2                 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Z. Metallkunde 85(1994)502-509; C diffusion in fcc Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
  'B. Jonsson: Z. Metallkunde 85(1994)498-501; C and N diffusion in bcc Cr
      -Fe-Ni'
  'B. Jonsson: Z. Metallkunde 83(1992)349-355; Cr, Co, Fe and Ni diffusion
      in bcc Fe'
  'B. Jonsson: ISIJ International, 35(1995)1415-1421; Cr, Fe and Ni
      diffusion bcc Cr-Fe-Ni'
 -OK-
APP:
APP: @@
APP: @@ ENTER THE DICTRA MONITOR
APP: @@
APP: go d-m
 NO TIME STEP DEFINED
```

```
DIC>
DIC> @@
DIC> @@ ENTER GLOBAL CONDITION T.
DIC> @@
DIC> set-cond glob T 0 973; * N

DIC>
DIC> @@
DIC> @@ WE START BY ENTERING REGION ferrite AND austenite WHEREIN WE
DIC> @@ PUT THE BCC AND FCC PHASE RESPECTIVELY. THE ferrite REGION IS
DIC> @@ ASSUMED INITIALLY TO BE VERY THIN, 1E-9 METERS.
DIC> @@
DIC> enter-region
REGION NAME : ferrite
DIC>
DIC> enter-region
REGION NAME : austenite
ATTACH TO REGION NAMED /FERRITE/:
ATTACHED TO THE RIGHT OF FERRITE /YES/:
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO THE REGIONS.
DIC> @@
DIC> enter-grid
REGION NAME : /FERRITE/: ferrite
WIDTH OF REGION /1/: 1e-9
TYPE /LINEAR/: linear
NUMBER OF POINTS /50/: 50
DIC>
DIC> enter-grid austenite
WIDTH OF REGION /1/: 50e-6
TYPE /LINEAR/: geo 50 1.05
DIC> @@geo
DIC> @@64
DIC> @@1.05
DIC>
DIC> @@
DIC> @@ ENTER ACTIVE PHASES INTO REGIONS
DIC> @@
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /FERRITE/: ferrite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: bcc
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: active
REGION NAME : /AUSTENITE/: austenite
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: fcc#1
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO BCC.
DIC> @@
DIC> enter-composition
REGION NAME : /FERRITE/: ferrite
PHASE NAME: /BCC_A2/: bcc
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.0885
VALUE OF LAST POINT : /8.85E-2/: 0.0885
PROFILE FOR /NI/: ni
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 2.02
VALUE OF LAST POINT : /2.02/: 2.02
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITION INTO FCC.
DIC> @@
DIC> enter-composition
REGION NAME : /AUSTENITE/: austenite
PHASE NAME: /FCC_A1/: fcc#1
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: w-p
PROFILE FOR /C/: c
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 0.0885
VALUE OF LAST POINT : /8.85E-2/: 0.0885
PROFILE FOR /NI/: ni
TYPE /LINEAR/: linear
VALUE OF FIRST POINT : 2.02
VALUE OF LAST POINT : /2.02/: 2.02
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exh2 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

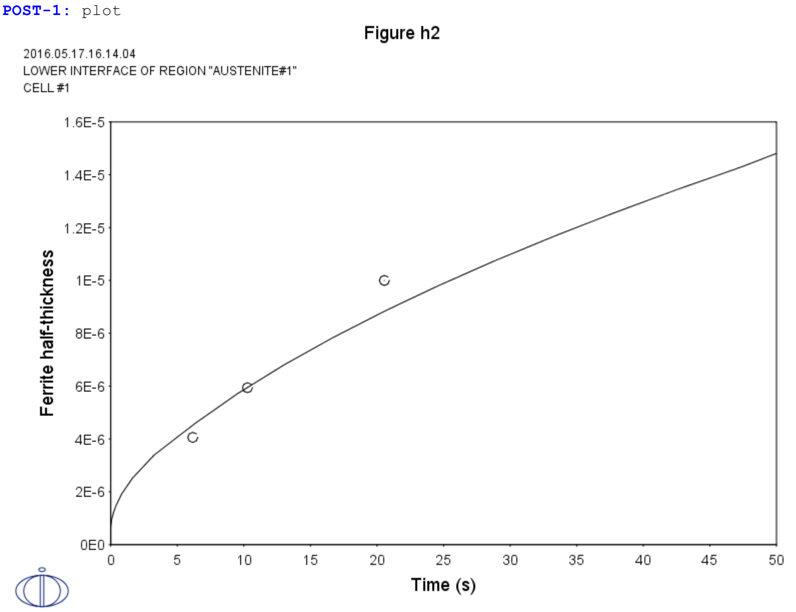
exh2-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh2\run.DCM"
DIC>
DIC>
DIC> @@
DIC> @@ FILE TO RUN EXH2
DIC> @@
DIC>
DIC> @@
DIC> @@ READ THE SETUP FROM FILE
DIC> @@
DIC>
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC>
DIC> read exh2
 OK
DIC>
DIC> @@
DIC> @@ SET THE SIMULATION TIME
DIC> @@
DIC> set-sim-time 50,,,,,,,,,,
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ ENABLE PARA-EQUILIBRIUM MODEL
DIC> @@
DIC> para
ENABLE PARAEQ : /NO/: YES
U-FRACTION OF COMPONENT FE /AUTO/: AUTO
U-FRACTION OF COMPONENT NI /AUTO/: AUTO
DIC>
DIC> @@
DIC> @@ START THE SIMULATION
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                  4
 GENERATING STARTING VALUES FOR CELL #  1 INTERFACE #  2
 DETERMINING INITIAL EQUILIBRIUM VALUES
CALCULATING STARTING VALUES:       9   EQUILIBRIUM CALCULATIONS               DONE     6 OUT OF     9

 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00412262676333  FE = .980742621143594
                        NI = .0192573788564064
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]
 U-FRACTION IN SYSTEM:  C = .00412262676333  FE = .980742621143594
                        NI = .0192573788564064
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]
  2  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: AUSTENITE
  13403.6959346809         13403.6959346759         13396.6825259383         181.653141195922         1.24995838856407         6.13070458006353
003        5.231515575903294E-004        2.136843547467712E-004        1.399791418590323E-004        3.952351410500741E-
005        2.316371120323491E-016
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.23163711E-15
 CELL # 1 VELOCITY AT INTERFACE # 2 IS    6.1709093     AND    6.1709093
 POSITION OF INTERFACE FERRITE / AUSTENITE IS    0.61809093E-06
 U-FRACTION IN SYSTEM:  C = .00414504645885367  FE = .980742621143596
                        NI = .0192573788564041
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]

  46  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    1  seconds
 4.810840690039439E-004        4.811802904599878E-004        2.876981682733878E-003        4.809878574305101E-004        4.808916549463522E-
004        4.806992796234162E-004        2.876843604636588E-003        4.803146435752126E-004        4.795458337960006E-
004        4.780100595429073E-004        2.873908372540977E-003        4.749458987620731E-004        4.688471331468752E-
004        4.567678328651244E-004        2.852911887798853E-003        4.330821563891156E-004        3.876025044437493E-
004        3.042100894217160E-004        2.700155452866472E-003        1.676926889900403E-004        1.572595579477210E-
005        3.647089510512923E-015        2.118476582424958E-022

output ignored...


... output resumed


 TIME =   37.827961     DT =   4.5652762     SUM OF SQUARES =  0.74442899E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.19672607E-06 AND   0.19672607E-06
 POSITION OF INTERFACE FERRITE / AUSTENITE IS    0.12563479E-04
 U-FRACTION IN SYSTEM:  C = .00412586743729131  FE = .980742621140488
                        NI = .0192573788595122
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]

  27  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    1  seconds
 2.612513340904137E-008        2.628064121682700E-008        2.396123823798855E-003        2.597008506306419E-008        2.581549624936868E-
008        2.550769714420020E-008        2.396123708024823E-003        2.489761372999800E-008        2.369950979250122E-
008        2.139158531748651E-008        2.396119204398370E-003        1.713674970718683E-008        1.001971031861954E-
008        1.473435403192718E-009        2.562969300352045E-015        6.491536283274548E-022
 TIME =   42.542090     DT =   4.7141286     SUM OF SQUARES =  0.64915363E-21
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.18632318E-06 AND   0.18632318E-06
 POSITION OF INTERFACE FERRITE / AUSTENITE IS    0.13441830E-04
 U-FRACTION IN SYSTEM:  C = .00412525911016849  FE = .980742621140273
                        NI = .0192573788597273
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]

  22  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                    1  seconds
 1.898852878642338E-008        1.911554905461583E-008        2.396116742195260E-003        1.886193039324907E-008        1.873575394848762E-
008        1.848466679699807E-008        2.396116666473206E-003        1.798755617066886E-008        1.701359276056368E-
```

```
008        1.514817374059558E-008        2.396112931395243E-003        1.174419062819423E-008        6.217509594556301E-
009        3.899102519973399E-010        4.130828265546513E-016        1.423889621169843E-018
 TIME =   47.398794     DT =    4.8567038     SUM OF SQUARES =  0.14238896E-17
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.17714221E-06 AND   0.17714221E-06
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.14302157E-04
 U-FRACTION IN SYSTEM:  C = .00412468672479594  FE = .980742621140063
                        NI = .0192573788599376
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]

  11  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: FERRITE

  CPU time used in timestep                     2  seconds
 3.578517559368621E-009        3.572951945222275E-009        2.396101741933899E-003        3.572951945000027E-009        3.567390692847176E-
009        3.556281273629098E-009        2.396101705365143E-003        3.534114775811928E-009        3.489991125819875E-
009        3.402581080891163E-009        2.396101570294999E-003        3.231108987171687E-009        2.901548593943389E-
009        2.295548745164632E-009        2.396100439961225E-003        1.297157421585491E-009        1.519494545009704E-
010        1.553389099115502E-015        5.313232428327676E-021
 TIME =   50.000000     DT =    2.6012064     SUM OF SQUARES =  0.53132324E-20
 CELL # 1 VELOCITY AT INTERFACE # 2 IS   0.19601581E-06 AND   0.19601581E-06
 POSITION OF INTERFACE FERRITE / AUSTENITE IS   0.14812035E-04
 U-FRACTION IN SYSTEM:  C = .00412421757309798  FE = .980742621139952
                        NI = .0192573788600479
 TOTAL SIZE OF SYSTEM:  5.0001E-05 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.20000000E-06
 DELETING TIME-RECORD FOR TIME   0.39398177E-06
 DELETING TIME-RECORD FOR TIME   0.78194531E-06
 DELETING TIME-RECORD FOR TIME   0.15578724E-05
 DELETING TIME-RECORD FOR TIME   0.31097266E-05
 DELETING TIME-RECORD FOR TIME   0.62134349E-05
 DELETING TIME-RECORD FOR TIME   0.12420852E-04
 DELETING TIME-RECORD FOR TIME   0.24835685E-04
 DELETING TIME-RECORD FOR TIME   0.49665351E-04
 DELETING TIME-RECORD FOR TIME   0.99324685E-04
 DELETING TIME-RECORD FOR TIME   0.19864335E-03
 DELETING TIME-RECORD FOR TIME   0.39728068E-03
 DELETING TIME-RECORD FOR TIME   0.79455535E-03
 DELETING TIME-RECORD FOR TIME   0.15891047E-02
 DELETING TIME-RECORD FOR TIME   0.31782033E-02
 DELETING TIME-RECORD FOR TIME   0.63564007E-02
 DELETING TIME-RECORD FOR TIME   0.12712795E-01
 DELETING TIME-RECORD FOR TIME   0.25425585E-01
 DELETING TIME-RECORD FOR TIME   0.50851163E-01
 DELETING TIME-RECORD FOR TIME   0.10170232
 DELETING TIME-RECORD FOR TIME   0.20340463
 DELETING TIME-RECORD FOR TIME   0.40680926
 DELETING TIME-RECORD FOR TIME   0.81361852
 DELETING TIME-RECORD FOR TIME    1.6272370
 DELETING TIME-RECORD FOR TIME    3.2544741
 DELETING TIME-RECORD FOR TIME    6.4062551
 DELETING TIME-RECORD FOR TIME    9.5427438
 DELETING TIME-RECORD FOR TIME    12.998617
 DELETING TIME-RECORD FOR TIME    16.666036
 DELETING TIME-RECORD FOR TIME    20.544636
 DELETING TIME-RECORD FOR TIME    24.610646
 DELETING TIME-RECORD FOR TIME    28.854023
 DELETING TIME-RECORD FOR TIME    33.262685
 DELETING TIME-RECORD FOR TIME    37.827961
 DELETING TIME-RECORD FOR TIME    42.542090

 KEEPING TIME-RECORD FOR TIME    47.398794
 AND FOR TIME                    50.000000
 WORKSPACE RECLAIMED
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

exh2-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh2\plot.DCM"DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  5.00000E+01
DIC>
DIC> read exh2
 OK
DIC>
DIC> @@
DIC> @@ GO TO THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1:
POST-1: @@
POST-1: @@ WE WANT TO PLOT THE POSITION OF THE INTERFACE AS A FUNCTION OF TIME
POST-1: @@ I.E. THE FERRITE HALF-THICKNESS
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1: s-d-a y posi aus low
POST-1:
POST-1: @@
POST-1: @@ APPEND THE EXPERIMENTAL INFORMATION
POST-1: @@
POST-1: app y exh2.exp 1; 1
POST-1:
POST-1: @@
POST-1: @@ SET TITLE ON DIAGRAM
POST-1: @@
POST-1: set-title Figure h2
POST-1:
POST-1: @@
POST-1: @@ RENAME AXIS LABELS
POST-1: @@
POST-1: set-axis-text-status
AXIS (X, Y OR Z) : x
AUTOMATIC AXIS TEXT  (Y OR N) /N/: NO
AXIS TEXT : Time (s)
POST-1:
POST-1: set-axis-text-status
AXIS (X, Y OR Z) : y
AUTOMATIC AXIS TEXT  (Y OR N) /N/: NO
AXIS TEXT : Ferrite half-thickness
POST-1:
POST-1: @@
POST-1: @@ PLOT THE RESULTS
POST-1: @@
POST-1: plot
```




```
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

exh3-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh3\setup.DCM" @@
SYS: @@ Deviation from local equilibrium.
SYS: @@ This calculation shows how a temperature gradient induces
SYS: @@ diffusion.
SYS: @@
SYS:
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw fedemo
 Current database: Iron Demo Database

 VA                     /-  DEFINED
TDB_FEDEMO: def-sys fe ni c
 FE                     NI                     C
   DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                  LIQUID:L               BCC_A2
 CEMENTITE              DIAMOND_FCC_A4         FCC_A1
 GRAPHITE               HCP_A3                 KSI_CARBIDE
 LAVES_PHASE_C14        M23C6                  M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph fcc graph
 FCC_A1                 GRAPHITE  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'B.-J. Lee, unpublished revision (1991); C-Cr-Fe-Ni'
  'A. Gabriel, C. Chatillon, and I. Ansara, published in High Temp. Sci.
      (parameters listed in Calphad, 11 (1987), 203-218); C-NI'
  'A. Markstrom, Swerea KIMAB, Sweden; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'A. Gabriel, P. Gustafson, and I. Ansara, Calphad, 11 (1987), 203 -218;
      TRITA-MAC 285 (1986); C-FE-NI'
  'A. Dinsdale, T. Chart, MTDS NPL, unpublished work (1986); FE-NI'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
      hard mater, Vol. 12, pp. 121-127; Molar volumes'
 -OK-
TDB_FEDEMO: app mfedemo
 Current database: Fe-Alloys Mobility demo database

 VA  DEFINED
APP: def-sys fe ni c
 FE                     NI                     C
   DEFINED
APP: rej ph * all
 BCC_A2                 FCC_A1  REJECTED
APP: res ph fcc
 FCC_A1  RESTORED
APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'This parameter has not been assessed'
  'J. Agren: Scripta Met. 20(1986)1507-1510; C diff in fcc C-Fe'
  'B. Jonsson: Z. Metallkunde 85(1994)502-509; C diffusion in fcc Cr-Fe-Ni'
  'B. Jonsson: Scand. J. Metall. 23(1994)201-208; Fe and Ni diffusion fcc Fe
      -Ni'
  'B. Jonsson: Scand. J. Metall. 24(1995)21-27; Ni self-diffusion'
 -OK-
APP:
APP: go d-m
 NO TIME STEP DEFINED
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC> @@ Enter a gaussian shaped temperature gradient
DIC> set-cond glob T 0 1000+400*exp(-3.35074E4*(x-11e-3)**2); * N

DIC>
DIC> set-ref C grap,,,,,,,,,,
DIC>
DIC> ent-reg aus,,,,,,
DIC>
DIC> ent-grid aus 25e-3 lin 50
DIC>
DIC> ent-pha act aus matrix fcc#1
DIC>
DIC> ent-comp aus fcc#1 fe w-p
PROFILE FOR /C/: c lin 0.14 0.14
PROFILE FOR /NI/: ni lin 32.5 32.5
DIC>
DIC> s-s-time 5E7,,,,,,,,,,,
DIC>
DIC> @@ Enter the heat of transfer parameter for Carbon
```

```
DIC> ent-heat-tra-p
HEAT TRANSFER PARAMETER FOR PHASE: fcc
ELEMENT: C
PARAMETER /0/: -42000
DIC>
DIC>
DIC> save exh3 y
DIC>
DIC> set-inter
 --OK---
DIC>
```

exh3-run

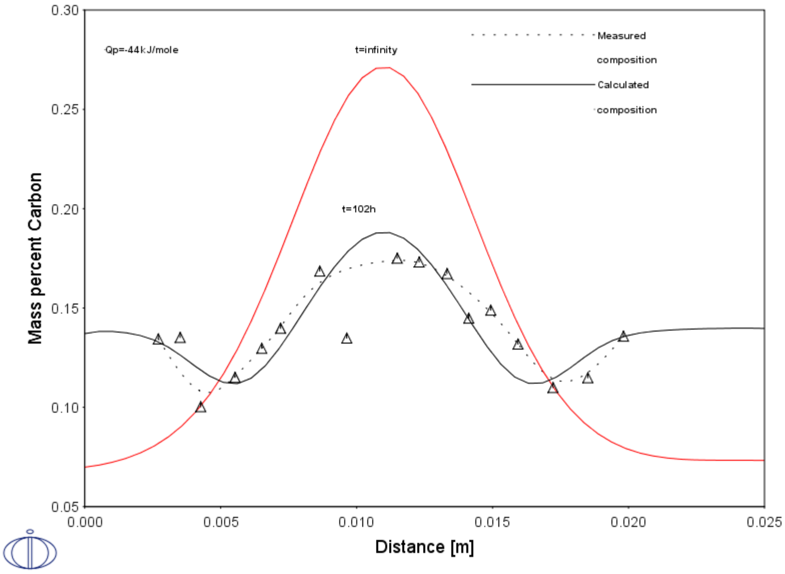
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh3\run.DCM"
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC> read exh3
 OK
DIC>
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .685349154604931
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .685349154604931
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   1600.4001     DT =   1600.0000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741857946  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   169987.44     DT =   168387.04     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741841045  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   506761.51     DT =   336774.07     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741841416  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     1   seconds

 TIME =   1180309.6     DT =   673548.14     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741841567  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   2527405.9     DT =   1347096.3     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741842162  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   5221598.5     DT =   2694192.6     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741843884  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   8073739.7     DT =   2852141.2     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741845336  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   10498680.     DT =   2424940.6     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741846301  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds

 TIME =   12740296.     DT =   2241615.3     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741846919  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     1   seconds

 TIME =   14883972.     DT =   2143676.4     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741847356  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0   seconds
```

```
 TIME =   16967480.     DT =   2083508.3     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741810029  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   19011098.     DT =   2043617.7     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741742159  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   21026872.     DT =   2015774.1     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741685344  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   23022415.     DT =   1995543.3     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741637009  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds


output ignored...


... output resumed


  CPU time used in timestep                     1  seconds

 TIME =   26971452.     DT =   1968669.1     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741559536  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   28930874.     DT =   1959422.1     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741528248  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   30882826.     DT =   1951951.9     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741500935  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   32828629.     DT =   1945803.0     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741476973  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   34769285.     DT =   1940656.0     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741455873  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   36705574.     DT =   1936289.5     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741437374  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     1  seconds

 TIME =   38638119.     DT =   1932545.2     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741420948  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   40567426.     DT =   1929307.0     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741406392  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   42493915.     DT =   1926488.6     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741393458  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   44417939.     DT =   1924024.0     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741381902  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds

 TIME =   46339800.     DT =   1921860.9     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741371548  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                     0  seconds
```

```
 TIME =   47886492.     DT =   1546692.3     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741364789  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                    0  seconds

 TIME =   49160264.     DT =   1273771.6     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741360206  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                    1  seconds

 TIME =   49798301.     DT =   638037.24     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741359151  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

  CPU time used in timestep                    0  seconds

 TIME =   50000000.     DT =   201699.09     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  C = .00662305741359122  FE = .68534915460493
                        NI = .31465084539507
 TOTAL SIZE OF SYSTEM:  .025 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.40010010
 DELETING TIME-RECORD FOR TIME    1600.4001
 DELETING TIME-RECORD FOR TIME    169987.44
 DELETING TIME-RECORD FOR TIME    506761.51
 DELETING TIME-RECORD FOR TIME    1180309.6
 DELETING TIME-RECORD FOR TIME    2527405.9
 DELETING TIME-RECORD FOR TIME    5221598.5
 DELETING TIME-RECORD FOR TIME    8073739.7
 DELETING TIME-RECORD FOR TIME    10498680.
 DELETING TIME-RECORD FOR TIME    12740296.
 DELETING TIME-RECORD FOR TIME    14883972.
 DELETING TIME-RECORD FOR TIME    16967480.
 DELETING TIME-RECORD FOR TIME    19011098.
 DELETING TIME-RECORD FOR TIME    21026872.
 DELETING TIME-RECORD FOR TIME    23022415.
 DELETING TIME-RECORD FOR TIME    25002783.
 DELETING TIME-RECORD FOR TIME    26971452.
 DELETING TIME-RECORD FOR TIME    28930874.
 DELETING TIME-RECORD FOR TIME    30882826.
 DELETING TIME-RECORD FOR TIME    32828629.
 DELETING TIME-RECORD FOR TIME    34769285.
 DELETING TIME-RECORD FOR TIME    36705574.
 DELETING TIME-RECORD FOR TIME    38638119.
 DELETING TIME-RECORD FOR TIME    40567426.
 DELETING TIME-RECORD FOR TIME    42493915.
 DELETING TIME-RECORD FOR TIME    44417939.
 DELETING TIME-RECORD FOR TIME    46339800.
 DELETING TIME-RECORD FOR TIME    47886492.
 DELETING TIME-RECORD FOR TIME    49160264.

 KEEPING TIME-RECORD FOR TIME    49798301.
 AND FOR TIME                    50000000.
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

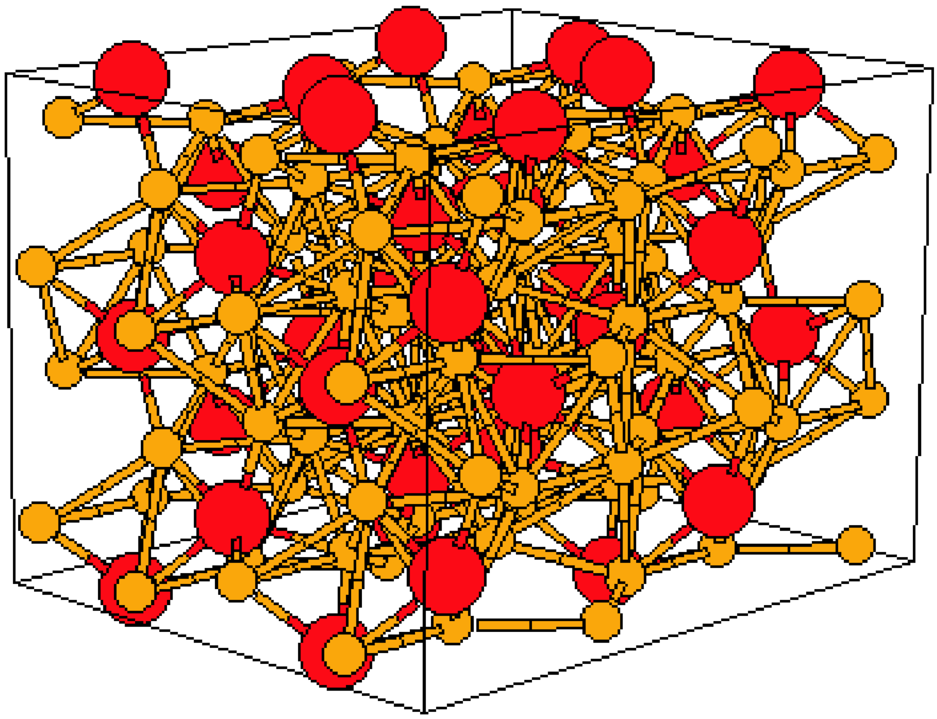
exh3-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exh3\plot.DCM"DIC> go d-m
 TIME STEP AT TIME  5.00000E+07
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC>
DIC> read exh3
 OK
DIC>
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1: s-d-a x dist glob
  INFO: Distance is set as independent variable
POST-1: s-d-a y w-p c
POST-1: s-p-c time 367200 5E7
POST-1: s-s-s y n 0.1 0.18
POST-1: s-s-s y y
POST-1: app y exh3.exp 0; 1 3;
POST-1: s-p-o n y y n y n n,,,,,,,,
POST-1:
POST-1: set-ax-text-st x
AUTOMATIC AXIS TEXT  (Y OR N) /N/: n
AXIS TEXT : Distance [m]
POST-1:
POST-1: set-ax-text-st y
AUTOMATIC AXIS TEXT  (Y OR N) /N/: n
AXIS TEXT : Mass percent Carbon
POST-1:
POST-1: plot
```



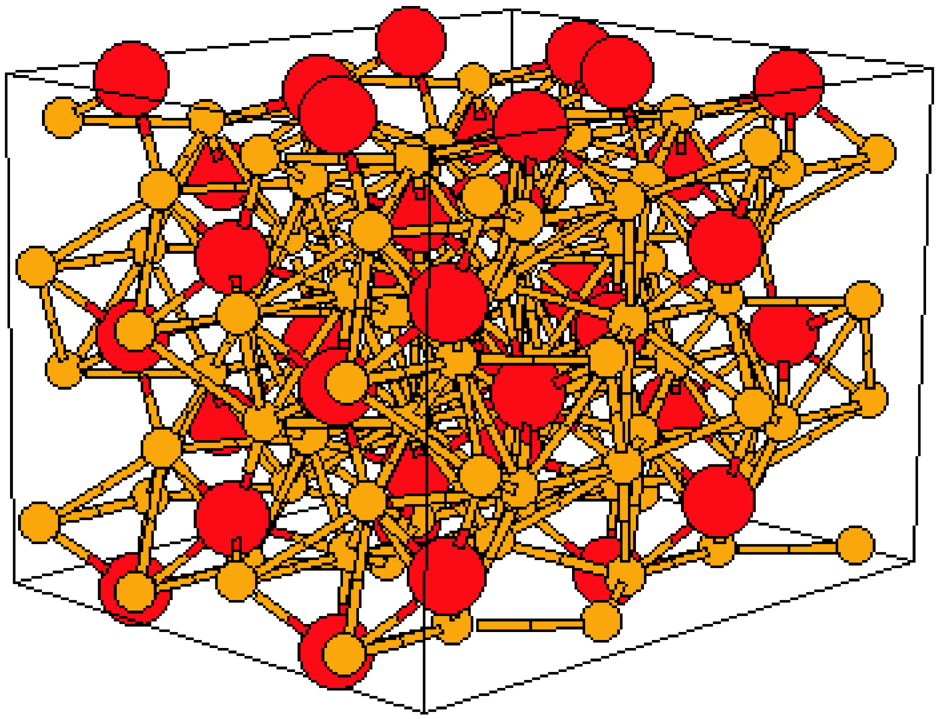

```
POST-1:
POST-1:
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Diffusion in complex phases



Example i1

Diffusion in system with B2 ordering



exi1-setup

```
About License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016


SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi1\setup.DCM" @@
SYS: @@ Diffusion in complex phases.
SYS: @@ Diffusion in including effects from chemical ordering.
SYS: @@ The datafile AlFeNi-data.TDB contains both a thermodynamic
SYS: @@ and kinetic description for the ordered and disordered BCC.
SYS: @@
SYS:
SYS: @@ exi1_setup.DCM
SYS:
SYS:
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8: sw user AlFeNi-data.TDB
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA  DEFINED
TDB_USER: def-sys fe al ni
 FE                      AL                      NI
   DEFINED
TDB_USER: rej ph *
 B2_BCC                  BCC_DIS                 B2_ORD
   REJECTED
TDB_USER: res ph bcc_dis b2_ord
 BCC_DIS                 B2_ORD  RESTORED
TDB_USER: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....
 -OK-
TDB_USER: go -m
 NO TIME STEP DEFINED
DIC> set-cond glob T 0 1277; * N

DIC>
DIC> enter-region beta
DIC>
DIC> enter-grid beta
WIDTH OF REGION /1/: 2e-3
TYPE /LINEAR/: double
NUMBER OF POINTS /50/: 50
VALUE OF R IN THE GEOMETRICAL SERIE FOR LOWER PART OF REGION: 0.85
VALUE OF R IN THE GEOMETRICAL SERIE FOR UPPER PART OF REGION: 1.1765
DIC>
DIC> enter-phase
ACTIVE OR INACTIVE PHASE /ACTIVE/: act
REGION NAME : /BETA/: beta
PHASE TYPE /MATRIX/: matrix
PHASE NAME: /NONE/: b2_ord
DIC>
DIC> enter-composition
REGION NAME : /BETA/: beta
PHASE NAME: /B2_ORD/: b2_ord
DEPENDENT COMPONENT ? /NI/: fe
COMPOSITION TYPE /MOLE_FRACTION/: mole-fraction
PROFILE FOR /AL/: ni
TYPE /LINEAR/: function
Function F(X)= 0.28-0.277*erf((x-1e-3)/3e-6);
PROFILE FOR /NI/: al
TYPE /LINEAR/: function
Function F(X)= 0.4295-0.0105*erf((x-1e-3)/3e-6);
DIC>
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 345600
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /34560/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC> s-a-s-v
AUTOMATIC STARTING VALUES FOR PHASE COMPOSITIONS /YES/: no
START VALUES FOR PHASES IN REGION BETA
PHASE: B2_ORD
MAJOR CONSTITUENTS IN PHASE B2_ORD: NI;AL

DIC>
DIC>
DIC>
DIC>
DIC>
DIC> save exi1 yes
DIC>
DIC> set-inter
 --OK---
DIC>
```

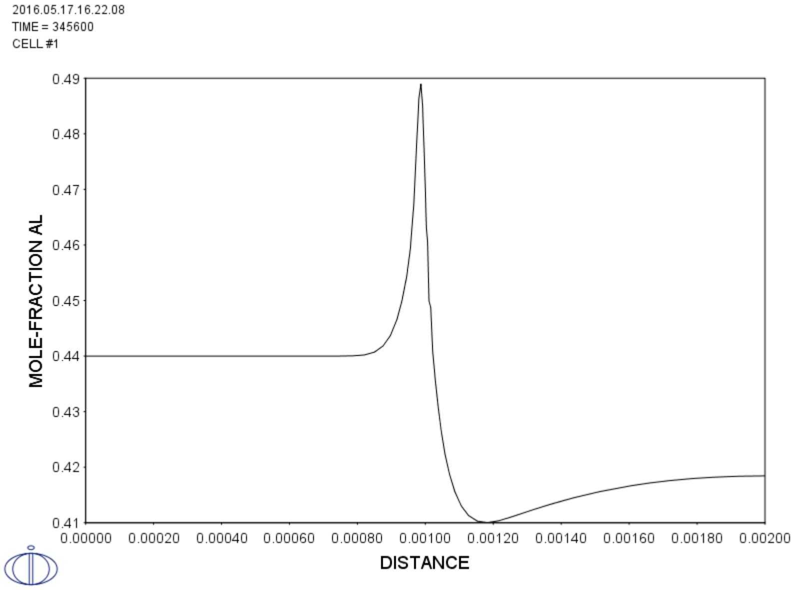
exi1-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi1\run.DCM"DIC>
DIC>
DIC> @@ exi1_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE i1
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exi1
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  AL = .429499345639258  FE = .290517917020317
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]
 U-FRACTION IN SYSTEM:  AL = .429499345639258  FE = .290517917020317
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]
  6  GRIDPOINT(S) ADDED   TO   CELL #1   REGION: BETA

 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639258  FE = .290517917020317
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639258  FE = .290517917020317
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     0  seconds

 TIME =  0.40010010     DT =  0.40000000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639258  FE = .290517917020317
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     0  seconds

 TIME =   513.08868     DT =   512.68858     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639255  FE = .29051791702032
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     0  seconds

 TIME =   1538.4658     DT =   1025.3772     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639253  FE = .290517917020322
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     1  seconds

 TIME =   3589.2202     DT =   2050.7543     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .42949934563925  FE = .290517917020325
                        NI = .279982737340425
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     1  seconds

 TIME =   7690.7288     DT =   4101.5087     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639246  FE = .290517917020326
                        NI = .279982737340428
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     1  seconds

 TIME =   15893.746     DT =   8203.0173     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639252  FE = .290517917020322
                        NI = .279982737340426
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     1  seconds

 TIME =   32299.781     DT =   16406.035     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639229  FE = .290517917020343
                        NI = .279982737340427
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     2  seconds

 TIME =   65111.850     DT =   32812.069     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639193  FE = .290517917020363
                        NI = .279982737340444
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     1  seconds

 TIME =   99671.850     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639108  FE = .290517917020477
                        NI = .279982737340416
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                     2  seconds

 TIME =   134231.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
```

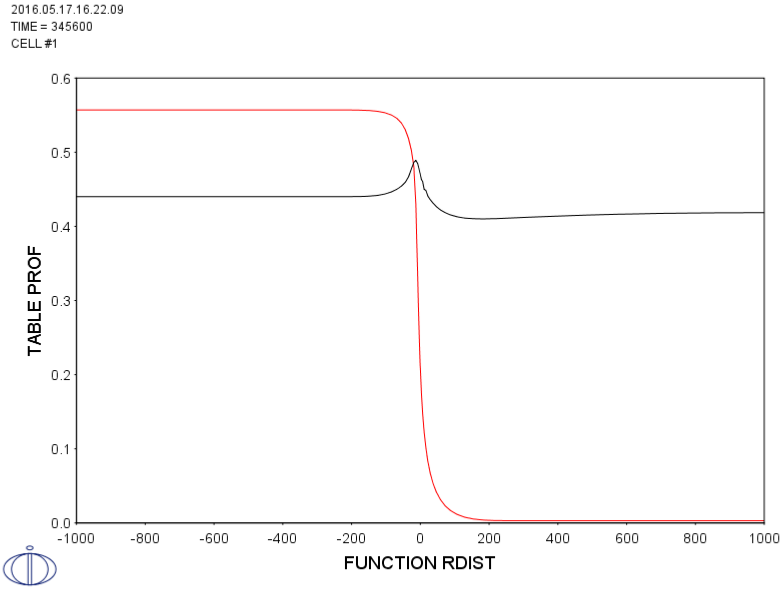
```
 U-FRACTION IN SYSTEM:  AL = .42949934563915  FE = .290517917020464
                        NI = .279982737340386
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      1  seconds

 TIME =   168791.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639186  FE = .290517917020427
                        NI = .279982737340387
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      2  seconds

 TIME =   203351.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639176  FE = .290517917020437
                        NI = .279982737340388
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      1  seconds

 TIME =   237911.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .42949934563917  FE = .290517917020422
                        NI = .279982737340408
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      1  seconds

 TIME =   272471.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .42949934563917  FE = .290517917020406
                        NI = .279982737340424
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      2  seconds

 TIME =   307031.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639179  FE = .290517917020393
                        NI = .279982737340428
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      1  seconds

 TIME =   341591.85     DT =   34560.000     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639191  FE = .290517917020383
                        NI = .279982737340426
 TOTAL SIZE OF SYSTEM:  .002 [m]

  CPU time used in timestep                      2  seconds

 TIME =   345600.00     DT =   4008.1500     SUM OF SQUARES =   0.0000000
 U-FRACTION IN SYSTEM:  AL = .429499345639192  FE = .290517917020382
                        NI = .279982737340426
 TOTAL SIZE OF SYSTEM:  .002 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.40010010
 DELETING TIME-RECORD FOR TIME    513.08868
 DELETING TIME-RECORD FOR TIME    1538.4658
 DELETING TIME-RECORD FOR TIME    3589.2202
 DELETING TIME-RECORD FOR TIME    7690.7288
 DELETING TIME-RECORD FOR TIME    15893.746
 DELETING TIME-RECORD FOR TIME    32299.781
 DELETING TIME-RECORD FOR TIME    65111.850
 DELETING TIME-RECORD FOR TIME    99671.850
 DELETING TIME-RECORD FOR TIME    134231.85
 DELETING TIME-RECORD FOR TIME    168791.85
 DELETING TIME-RECORD FOR TIME    203351.85
 DELETING TIME-RECORD FOR TIME    237911.85
 DELETING TIME-RECORD FOR TIME    272471.85
 DELETING TIME-RECORD FOR TIME    307031.85

 KEEPING TIME-RECORD FOR TIME    341591.85
 AND FOR TIME                   345600.00
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exi1-plot

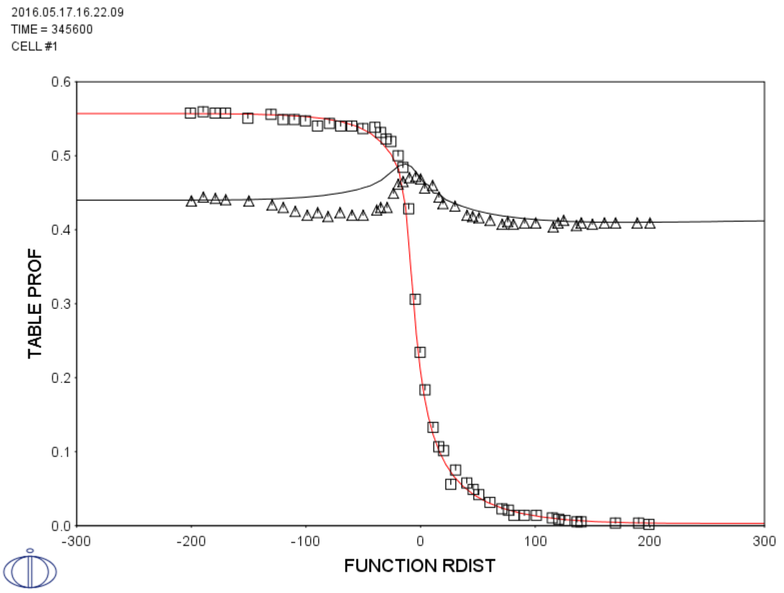
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi1\plot.DCM"DIC>
DIC> @@ exi1_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE i1
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  3.45600E+05
DIC> read exi1
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: s-d-a x dist glob
  INFO: Distance is set as independent variable
POST-1:
POST-1: s-d-a y m-f al
POST-1:
POST-1: s-p-c time last
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: ent tab prof
Variable(s) x(al) x(ni)
POST-1:
POST-1: ent fun rdist
FUNCTION: 1e6*(gd-10e-4)
&
POST-1: s-d-a y prof
COLUMN NUMBER /*/: 1 2
POST-1:
POST-1: s-d-a x rdist
POST-1:
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: app y exi1.exp
PROLOGUE NUMBER: /0/: 1
DATASET NUMBER(s): /-1/: 1
POST-1:
POST-1: s-s-s x n -300 300
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example i2

Diffusion of carbon in cementite

$$J \sim \Delta\mu$$

exi2-setup

```
About Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi2\setup.DCM" @@
SYS: @@ Diffusion in complex phases.
SYS: @@ This example demonstrates the use of the model for calculation of
SYS: @@ diffusion through a stoichiometric phase. The flux of a component in
SYS: @@ the stoichiometric phase is assumed to be proportional to the
SYS: @@ difference in chemical potential at each side of the stoichiometric
SYS: @@ phase multiplied with the mobility for the component in the phase. The
SYS: @@ mobility is assessed from experimental information and is basically
SYS: @@ the tracer diffusivity for the component.
SYS: @@ This calculation is compared with experimental data where a sample of
SYS: @@ pure iron has been exposed to a gas atmosphere with a certain carbon
SYS: @@ activity. The weight gain is then measured as a function of time.
SYS: @@ The experimental data is obtained from Ozturk B., Fearing V. L.,
SYS: @@ Ruth A. Jr. and Simkovich G., Met. Trans A, vol 13A (1982), pp. 1871-1873.
SYS: @@
SYS:
SYS: @@
SYS: @@ RETRIEVE DATA FROM DATABASE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                B2_BCC                 B2_VACANCY
 HIGH_SIGMA             DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ USE A PUBLIC DATABASE FOR THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: switch FEDEMO
 Current database: Iron Demo Database

 VA                     /-  DEFINED
TDB_FEDEMO: def-sys fe c
 FE                     C  DEFINED
TDB_FEDEMO: rej ph * all
 GAS:G                  LIQUID:L               BCC_A2
 CEMENTITE              DIAMOND_FCC_A4         FCC_A1
 GRAPHITE               HCP_A3                 KSI_CARBIDE
 LAVES_PHASE_C14        M23C6                  M5C2
 M7C3  REJECTED
TDB_FEDEMO: res ph bcc fcc cementite grap
 BCC_A2                 FCC_A1                 CEMENTITE
 GRAPHITE  RESTORED
TDB_FEDEMO: get
 REINITIATING GES5 .....
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data

  'P. Franke, estimated parameter within SGTE, 2007; Fe-C, Ni-C, Mo -C, C-Mn'
  'P. Gustafson, Scan. J. Metall., 14 (1985), 259-267; TRITA 0237 (1984); C
      -FE'
  'X.-G. Lu, Thermo-Calc Software AB, Sweden,2006; Molar volumes'
  'A. Dinsdale, SGTE Data for Pure Elements, Calphad, 15 (1991), 317 -425'
  'X.-G. Lu, M. Selleby and B. Sundman, CALPHAD, Vol. 29, 2005, pp. 68-89;
      Molar volumes'
  'X.-G. Lu et al. Calphad 29 (2005) 49-55, Fe P-T diagram'
  'B. Uhrenius (1993-1994), International journal of refractory metals and
      hard mater, Vol. 12, pp. 121-127; Molar volumes'
 -OK-
TDB_FEDEMO:
TDB_FEDEMO: @@
TDB_FEDEMO: @@ SWITCH TO MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_FEDEMO: @@
TDB_FEDEMO: app user cembccstoik.TDB
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA  DEFINED
TDB_APP: def-sys fe c
 FE                     C  DEFINED
TDB_APP: rej ph * all
 BCC_A2                 CEMENTITE  REJECTED
TDB_APP: res ph fcc bcc cementite
 *** FCC INPUT IGNORED
 BCC_A2                 CEMENTITE  RESTORED
TDB_APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data


 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE

 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE
 -OK-
TDB_APP:
TDB_APP: @@
TDB_APP: @@ ENTER THE DICTRA MONITOR
TDB_APP: @@
TDB_APP: go d-m
 NO TIME STEP DEFINED
```

```
 *** ENTERING FCC_A1 AS A DIFFUSION NONE PHASE
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC> set-ref c grap,,,,,,,,,,
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob t 0 723; * n

DIC>
DIC> @@
DIC> @@ ENTER REGIONS carb AND fer
DIC> @@
DIC> enter-region
REGION NAME : fer
DIC>
DIC> enter-region
REGION NAME : carb
ATTACH TO REGION NAMED /FER/:
ATTACHED TO THE RIGHT OF FER /YES/:
DIC> @@
DIC> @@ ENTER LINEAR GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ Enter a size for the ferrite
DIC> @@
DIC> @@
DIC> @@
DIC> enter-grid
REGION NAME : /FER/: fer
WIDTH OF REGION /1/: 3.3E-6
TYPE /LINEAR/: lin 50
DIC>
DIC> @@
DIC> @@ Enter a size (very small) for the cementite layer
DIC> @@
DIC> @@
DIC> enter-grid
REGION NAME : /CARB/: carb
WIDTH OF REGION /1/: 1E-12
TYPE /LINEAR/: lin
NUMBER OF POINTS /50/: 50
DIC>
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> enter-phase act carb  matrix   cementite
  COMPOSITION RECORD FOR STOICHIOMETRIC PHASE CEMENTITE IN REGION CARB CREATED
DIC> enter-phase act fer   matrix   bcc#1
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE PHASES
DIC> @@
DIC> enter-composition
REGION NAME : /FER/: carb
PHASE NAME: /CEMENTITE/: cementite
DIC>
DIC> enter-composition
REGION NAME : /FER/: fer
PHASE NAME: /BCC_A2/: bcc#1
COMPOSITION TYPE /MOLE_FRACTION/: weig-fraction
PROFILE FOR /C/: C  lin 1E-5 1E-5
DIC>
DIC> set-cond bound upp

CONDITION TYPE /CLOSED_SYSTEM/: mix
Dependent substitutional element:FE
Dependent interstitial element:VA
LOW TIME LIMIT /0/: 0
ACR(C)(TIME)= 9;
HIGH TIME LIMIT /*/: *
ANY MORE RANGES /N/: N
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ simulate for 150 minutes
DIC> @@
DIC> set-simulation-time
END TIME FOR INTEGRATION /.1/: 9000
AUTOMATIC TIMESTEP CONTROL /YES/:
MAX TIMESTEP DURING INTEGRATION /900/:
INITIAL TIMESTEP : /1E-07/:
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exi2 Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

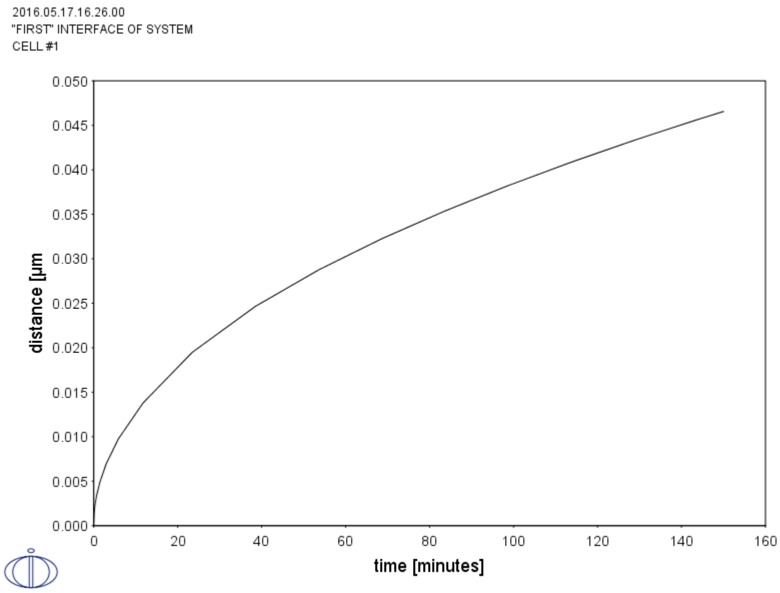
exi2-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi2\run.DCM"
DIC>
DIC> @@ exi2_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE i2
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
 *** ENTERING FCC_A1 AS A DIFFUSION NONE PHASE
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC> read exi2
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> sim
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                      3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  C = 4.6598005784384E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]
 U-FRACTION IN SYSTEM:  C = 4.6598005784384E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]
 0.111097719336122        0.111119939666141        9.691611001729309E-020
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.96916110E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14617667E-04 AND  -0.14617667E-04
 POSITION OF INTERFACE FER / CARB IS   0.32999985E-05
 U-FRACTION IN SYSTEM:  C = 4.65258726084234E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  CPU time used in timestep                     0  seconds
 2.264815947299283E-005        2.265270404058167E-005        5.027029744041694E-020
 TIME =  0.10010000E-03 DT =  0.10000000E-03 SUM OF SQUARES =  0.50270297E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.46363829E-07 AND  -0.46363829E-07
 POSITION OF INTERFACE FER / CARB IS   0.32999939E-05
 U-FRACTION IN SYSTEM:  C = 4.6983979832688E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  40  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 9.341884324918429E-007        9.344812013987095E-007        5.105702713691625E-022
 TIME =  0.27909042E-02 DT =  0.26908042E-02 SUM OF SQUARES =  0.51057027E-21
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.16437826E-07 AND  -0.16437826E-07
 POSITION OF INTERFACE FER / CARB IS   0.32999497E-05
 U-FRACTION IN SYSTEM:  C = 5.1259189348325E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  24  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 2.027667719630935E-005        2.028151298271774E-005        2.494890787506013E-020
 TIME =  0.81725125E-02 DT =  0.53816084E-02 SUM OF SQUARES =  0.24948908E-19
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.25990487E-08 AND  -0.25990487E-08
 POSITION OF INTERFACE FER / CARB IS   0.32999357E-05
 U-FRACTION IN SYSTEM:  C = 5.24416018574873E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 3.430896436428050E-006        3.434042115366487E-006        1.199997662236344E-024
 TIME =  0.18935729E-01 DT =  0.10763217E-01 SUM OF SQUARES =  0.11999977E-23
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.19913385E-08 AND  -0.19913385E-08
 POSITION OF INTERFACE FER / CARB IS   0.32999143E-05
 U-FRACTION IN SYSTEM:  C = 5.43000196720892E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  CPU time used in timestep                     0  seconds
 2.458075289389597E-006        2.460115293765368E-006        7.938345378483446E-025
 TIME =  0.40462163E-01 DT =  0.21526433E-01 SUM OF SQUARES =  0.79383454E-24
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14811247E-08 AND  -0.14811247E-08
 POSITION OF INTERFACE FER / CARB IS   0.32998824E-05
 U-FRACTION IN SYSTEM:  C = 5.70985325141529E-05  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  1  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 1.561011360975503E-006        1.562220505705435E-006        7.902659183851734E-025


output ignored...



... output resumed

 POSITION OF INTERFACE FER / CARB IS   0.32712688E-05
 U-FRACTION IN SYSTEM:  C = .00292698161361244  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  44  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 3.382688098895244E-008        3.388131432326961E-008        6.785189226500377E-027
 TIME =   4110.7537     DT =   900.00000    SUM OF SQUARES =  0.67851892E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.38861536E-11 AND  -0.38861536E-11
```

```
 POSITION OF INTERFACE FER / CARB IS   0.32677713E-05
 U-FRACTION IN SYSTEM:  C = .00328024173504251  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  15  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 1.314638216622111E-008        1.317550715160612E-008        2.302815081917142E-027
 TIME =   5010.7537     DT =   900.00000     SUM OF SQUARES =  0.23028151E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.34644318E-11 AND  -0.34644318E-11
 POSITION OF INTERFACE FER / CARB IS   0.32646533E-05
 U-FRACTION IN SYSTEM:  C = .00359516638951726  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  11  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 6.203563492420051E-009        6.221402339803630E-009        5.499631429432279E-028
 TIME =   5910.7537     DT =   900.00000     SUM OF SQUARES =  0.54996314E-27
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.31588373E-11 AND  -0.31588373E-11
 POSITION OF INTERFACE FER / CARB IS   0.32618104E-05
 U-FRACTION IN SYSTEM:  C = .00388231182053254  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  19  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 3.285493905069299E-009        3.297333056967420E-009        1.314960496745230E-028
 TIME =   6810.7537     DT =   900.00000     SUM OF SQUARES =  0.13149605E-27
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.29236900E-11 AND  -0.29236900E-11
 POSITION OF INTERFACE FER / CARB IS   0.32591790E-05
 U-FRACTION IN SYSTEM:  C = .00414808183129118  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  6  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 1.876998084839297E-007        1.885282157885578E-007        3.560216365444161E-027
 TIME =   7710.7537     DT =   900.00000     SUM OF SQUARES =  0.35602164E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.27352335E-11 AND  -0.27352335E-11
 POSITION OF INTERFACE FER / CARB IS   0.32567173E-05
 U-FRACTION IN SYSTEM:  C = .00439672071771767  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  5  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 1.130104476038546E-007        1.136119435657704E-007        1.101146356111536E-027
 TIME =   8610.7537     DT =   900.00000     SUM OF SQUARES =  0.11011464E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.25796702E-11 AND  -0.25796702E-11
 POSITION OF INTERFACE FER / CARB IS   0.32543956E-05
 U-FRACTION IN SYSTEM:  C = .0046312185494458  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

  4  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: CARB

  CPU time used in timestep                     0  seconds
 7.062813212487258E-008        7.107671560567722E-008        1.741355107882138E-028
 TIME =   9000.0000     DT =   389.24625     SUM OF SQUARES =  0.17413551E-27
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.24483429E-11 AND  -0.24483429E-11
 POSITION OF INTERFACE FER / CARB IS   0.32534426E-05
 U-FRACTION IN SYSTEM:  C = .00472747477504805  FE = 1
 TOTAL SIZE OF SYSTEM:  3.300001E-06 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.10010000E-03
 DELETING TIME-RECORD FOR TIME   0.27909042E-02
 DELETING TIME-RECORD FOR TIME   0.81725125E-02
 DELETING TIME-RECORD FOR TIME   0.18935729E-01
 DELETING TIME-RECORD FOR TIME   0.40462163E-01
 DELETING TIME-RECORD FOR TIME   0.83515029E-01
 DELETING TIME-RECORD FOR TIME   0.16962076
 DELETING TIME-RECORD FOR TIME   0.34183223
 DELETING TIME-RECORD FOR TIME   0.68625517
 DELETING TIME-RECORD FOR TIME    1.3751010
 DELETING TIME-RECORD FOR TIME    2.7527928
 DELETING TIME-RECORD FOR TIME    5.5081762
 DELETING TIME-RECORD FOR TIME    11.018943
 DELETING TIME-RECORD FOR TIME    22.040477
 DELETING TIME-RECORD FOR TIME    44.083545
 DELETING TIME-RECORD FOR TIME    88.169681
 DELETING TIME-RECORD FOR TIME    176.34195
 DELETING TIME-RECORD FOR TIME    352.68649
 DELETING TIME-RECORD FOR TIME    705.37558
 DELETING TIME-RECORD FOR TIME    1410.7537
 DELETING TIME-RECORD FOR TIME    2310.7537
 DELETING TIME-RECORD FOR TIME    3210.7537
 DELETING TIME-RECORD FOR TIME    4110.7537
 DELETING TIME-RECORD FOR TIME    5010.7537
 DELETING TIME-RECORD FOR TIME    5910.7537
 DELETING TIME-RECORD FOR TIME    6810.7537
 DELETING TIME-RECORD FOR TIME    7710.7537

 KEEPING TIME-RECORD FOR TIME    8610.7537
 AND FOR TIME                    9000.0000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

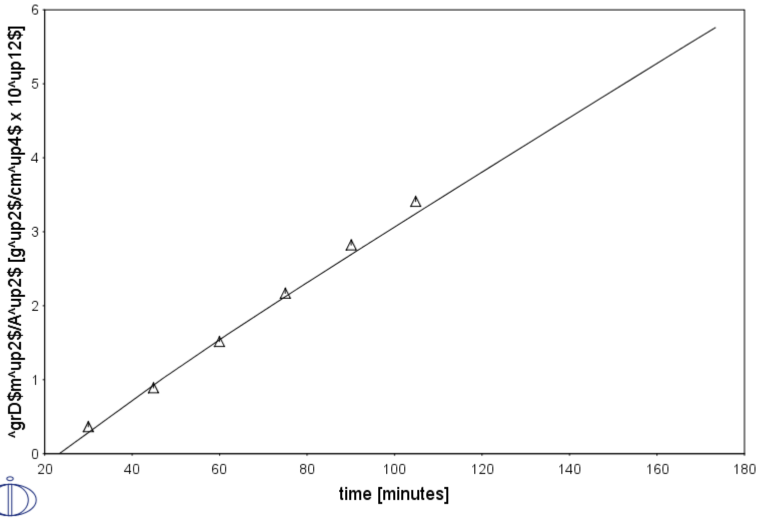
exi2-plot

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi2\plot.DCM"DIC>
DIC> @@ exi2_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE i2
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  9.00000E+03
 *** ENTERING FCC_A1 AS A DIFFUSION NONE PHASE
 *** ENTERING GRAPHITE AS A DIFFUSION NONE PHASE
DIC>
DIC>
DIC> read exi2 Y
 OK
DIC>
DIC>
DIC> @@
DIC> @@ Plot the size of cementite layer as a function of time
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: ent-symb func csize
FUNCTION: 1e6*(poi(car,u)-poi(car,l));
POST-1:
POST-1: ent-symb func minutes
FUNCTION: time/60;
POST-1:
POST-1: s-d-a x minutes
POST-1: s-d-a y csize
POST-1:
POST-1: s-p-c inter first
POST-1:
POST-1: s-a-t-s x n time [minutes]
POST-1: s-a-t-s y n distance [?m]
POST-1:
POST-1: plot
```



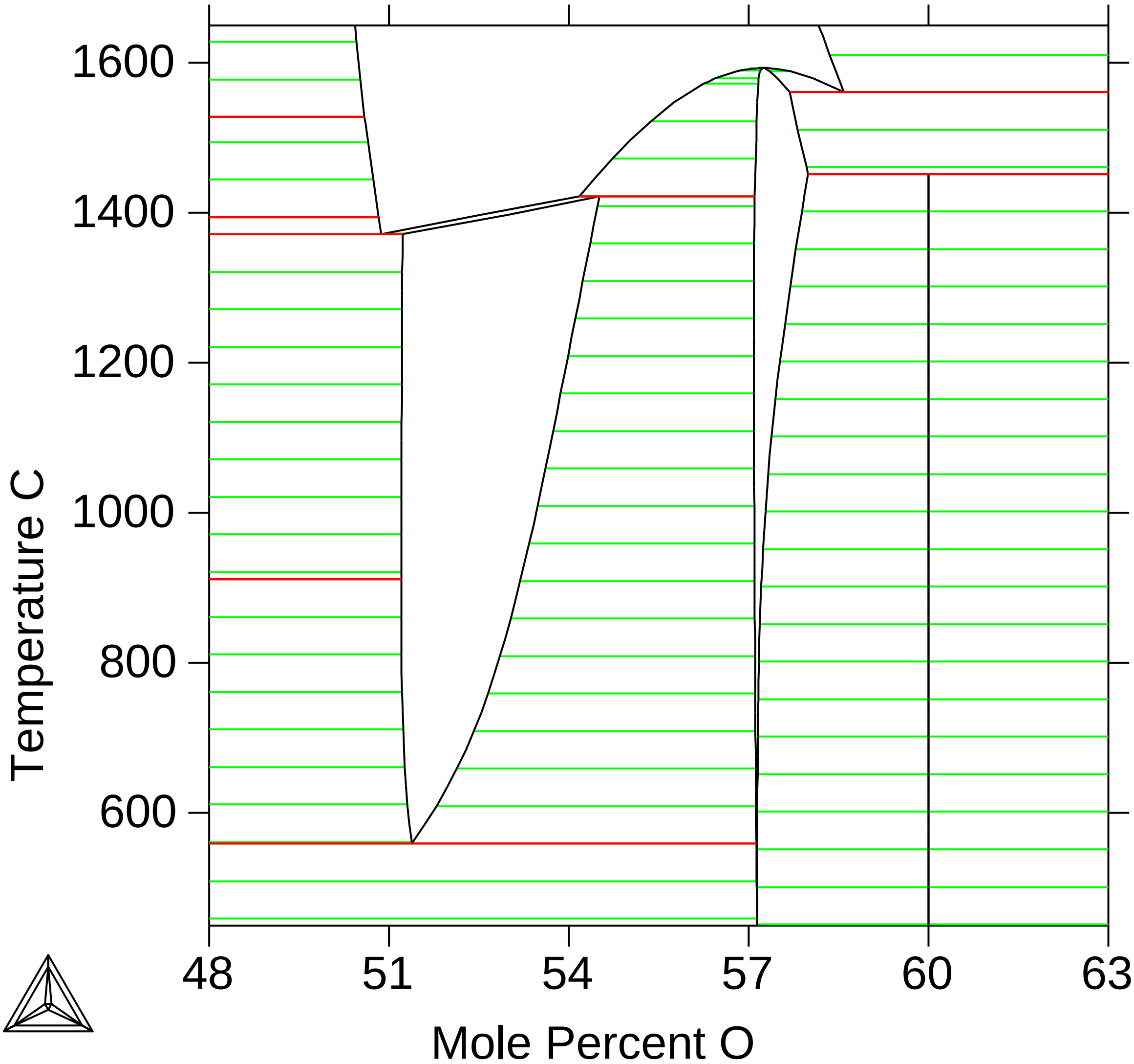

```
POST-1:
POST-1: @<? HIT_RETURN_TO_CONTINUE>
Felaktig kommandosyntax.
POST-1:
POST-1: @@
POST-1: @@ Assume a certain time for nucleation of the cementite layer
POST-1: @@
POST-1: ent-symb func cortim
FUNCTION: (time+1400)/60;
POST-1:
POST-1: @@
POST-1: @@ plot the weight gain as a function of time
POST-1: @@
POST-1: ent-symb func cwei
FUNCTION: 1e12*((poi(car,u)-poi(car,l)-1E-12)*12.01/2.33E-5*1e-4)**2;
POST-1:
POST-1:
POST-1: s-d-a x cortim
POST-1: s-d-a y cwei
POST-1:
POST-1: @@
POST-1: @@ Compare with experimental data
POST-1: @@
POST-1: app y exi2.exp 0; 1
POST-1:
POST-1:
POST-1: s-a-t-s x n time [minutes]
POST-1: s-a-t-s y n ^grD$m^up2$/A^up2$ [g^up2$/cm^up4$ x 10^up12$]
POST-1:
POST-1: plot
```

2016.05.17.16.26.02
"FIRST" INTERFACE OF SYSTEM
CELL #1



POST-1:
POST-1:
POST-1: set-inter
--OK---
POST-1:

Example i3 Diffusion in FeO



exi3a-setup

About Linked: Tue May 17 11:39:18 2016

```
SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3a\setup.DCM" @@
SYS: @@ Diffusion in complex phases.
SYS: @@ Oxidation of iron sample and consequent growth of an oxide layer.
SYS: @@
SYS:
SYS: @@ exi3_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                 B2_BCC                  B2_VACANCY
 HIGH_SIGMA              DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ WE SELECT A USER DATABASE FOR READING THE THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw user FeO.TDB
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA                      /-  DEFINED
TDB_USER: def-sys fe o
 FE                      O  DEFINED
TDB_USER: rej sp *
 /-                      VA                      FE
 O                       FE+2                    FE+3
 FE+4                    FE2O3                   FEO
 FEO3/2                  O-2                     O2
   REJECTED
TDB_USER: res sp fe fe+2 fe+3 o o2 o-2 va
 FE                      FE+2                    FE+3
 O                       O2                      O-2
 VA  RESTORED
TDB_USER: rej ph * all
 GAS:G                   BCC_A2                  SPINEL:I
   REJECTED
TDB_USER: res ph bcc spinel gas
 BCC_A2                  SPINEL:I                GAS:G
   RESTORED
TDB_USER:
TDB_USER: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 Reference REF368     missing close to line                      55
 Reference REF304     missing close to line                      63
 Reference REF420     missing close to line                      65
 Reference REF420     missing close to line                      67
 Reference REF368     missing close to line                      69
 Reference REF368     missing close to line                      70
 Reference REF368     missing close to line                      72
 Reference REF418     missing close to line                      74
 Reference REF418     missing close to line                      76
 Reference REF304     missing close to line                      78
 Reference REF30      missing close to line                      86
 Reference REF30      missing close to line                      88
 Reference REF30      missing close to line                      90
 Reference REF420     missing close to line                      92
 Reference REF420     missing close to line                      94
 Reference REF30      missing close to line                      96
 Reference REF30      missing close to line                      98
 Reference REF30      missing close to line                     100
 Reference REF420     missing close to line                     102
 Reference REF420     missing close to line                     104
 Reference REF30      missing close to line                     106
 Reference REF30      missing close to line                     108
 Reference REF30      missing close to line                     110
 Reference REF420     missing close to line                     112
 Reference REF420     missing close to line                     114
 Reference REF30      missing close to line                     116
 Reference REF30      missing close to line                     118
 Reference REF30      missing close to line                     120
 Reference REF420     missing close to line                     122
 Reference REF420     missing close to line                     124
 Reference REF30      missing close to line                     126
 Reference REF30      missing close to line                     128
 Reference REF30      missing close to line                     130
 Reference REF420     missing close to line                     132
 Reference REF420     missing close to line                     134
 Reference REF30      missing close to line                     136
 Reference REF30      missing close to line                     138
 Reference REF30      missing close to line                     140
 Reference REF420     missing close to line                     142
 Reference REF420     missing close to line                     144
 Reference REF30      missing close to line                     146
 Reference REF30      missing close to line                     148
 Reference REF30      missing close to line                     150
 Reference REF420     missing close to line                     152
 Reference REF420     missing close to line                     154
 Reference REF30      missing close to line                     156
 Reference REF30      missing close to line                     158
 Reference REF30      missing close to line                     160
 Reference REF420     missing close to line                     162
 Reference REF420     missing close to line                     164
 Reference REF30      missing close to line                     166
 Reference REF30      missing close to line                     168
 Reference REF30      missing close to line                     170
 Reference REF420     missing close to line                     172
 Reference REF420     missing close to line                     174
 Reference REF30      missing close to line                     176
 Reference REF30      missing close to line                     178
 Reference REF30      missing close to line                     180
```

```
 Reference REF420     missing close to line                    182
 Reference REF420     missing close to line                    184
 Reference REF30      missing close to line                    186
 Reference REF30      missing close to line                    188
 Reference REF30      missing close to line                    190
 Reference REF420     missing close to line                    192
 Reference REF420     missing close to line                    194
 Reference REF30      missing close to line                    196
 Reference REF30      missing close to line                    198
 Reference REF30      missing close to line                    200
 Reference REF420     missing close to line                    202
 Reference REF420     missing close to line                    204
 FUNCTIONS ....
 -OK-
TDB_USER:
TDB_USER: @@
TDB_USER: @@ SWITCH TO A USER DEFINED MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_USER: @@
TDB_USER: app user FeOmob.TDB
 Current database: User defined Database

         test database

 VA                       /-                       O
   DEFINED
TDB_APP: def-sys fe o
 FE  DEFINED
TDB_APP: rej sp *
 /-                       VA                       FE
 O                        FE+2                     FE+3
 FE2O3                    FEO                      FEO3/2
 O-2                      O2  REJECTED
TDB_APP: res sp fe fe+2 fe+3 o o2 o-2 va
 FE                       FE+2                     FE+3
 O                        O2                       O-2
 VA  RESTORED
TDB_APP: rej ph * all
 SPINEL:I                 GAS:G                    BCC_A2
   REJECTED
TDB_APP: res ph bcc spinel gas
 BCC_A2                   SPINEL:I                 GAS:G
   RESTORED
TDB_APP:
TDB_APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data


 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE

 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE
 -OK-
TDB_APP:
TDB_APP:
TDB_APP: @@
TDB_APP: @@ ENTER THE DICTRA MONITOR
TDB_APP: @@
TDB_APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 823; * N

DIC>
DIC>
DIC> @@
DIC> @@  SET REFERENCE STATE FOR O TO O2 (GAS)
DIC> @@
DIC> set-ref o gas,,,,,,,
DIC>
DIC>
DIC> @@
DIC> @@ ENTER REGIONS fer AND sp
DIC> @@
DIC> ent-reg fer
DIC> ent-reg sp,,,,,,,,
DIC>
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> ent-phase act fer matrix bcc#1
DIC> ent-phase act sp matrix spinel
DIC>
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ Enter a size for the ferrite
DIC> @@
DIC> @@
DIC> ent-grid fer 4.99999e-3 lin 50
DIC>
DIC> @@
DIC> @@ Enter a thin initial size for the oxide
DIC> @@
DIC> @@
DIC> ent-grid sp  1.00e-10 lin 50
DIC>
DIC>
```

```
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN BCC
DIC> @@
DIC> ent-comp fer bcc#1 m-f
PROFILE FOR /O/: o lin 1e-9 1e-9
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE OXIDE
DIC> @@
DIC> ent-comp sp spinel m-f
  this is a phase with charged species
  with more than 2 sublattices
PROFILE FOR /FE/: FE lin 4.28771E-01 4.28549E-01
DIC>
DIC>
DIC> @@
DIC> @@ ENTER A BOUNDARY CONDITION "GAS" ON THE UPPER (RIGHTMOST) INTERFACE
DIC> @@ OF THE OXIDE. THIS WILL ALLOW THE SYSTEM TO EXPAND AND THE OXIDE LAYER
DIC> @@ TO GROW EXTERNALY. FOR THIS EXAMPLE WE HAVE SPECIFIED AN OXYGEN ACTIVITY
DIC> @@ THAT IS LOW ENOUGH IN ORDER NOT TO FORM CORUNDUM (FE2O3). WE ALSO
DIC> @@ SPECIFY THAT THERE IS NO FLUX OF FE ACROSS THIS INTERFACE, I.E. NO FE
DIC> @@ IS ALLOWED TO ENTER OR LEAVE THE SYSTEM.
DIC> @@
DIC> set-cond boundary upper gas

TYPE OF CONDITION FOR COMPONENT FE /ZERO_FLUX/: zero-flux
TYPE OF CONDITION FOR COMPONENT O /ZERO_FLUX/: act
LOW TIME LIMIT /0/: 0 4.5e-4; * N
DIC>
DIC>
DIC> @@
DIC> @@ ENTER START VALUES FOR INITIAL INTERFACE VELOCITIES
DIC> @@
DIC> s-a-s-v -1e-5 1e-5 yes
 STARTING VALUES WILL BE TAKEN FROM PROFILES
DIC>
DIC>
DIC> @@
DIC> @@ SIMULATE FOR 24 HOURS
DIC> @@
DIC> s-s-time 86400,,,,
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> @@
DIC> @@ SPECIFY THAT POTENTIALS AND NOT ACTIVITIES ARE VARIED AT THE PHASE
DIC> @@ INTERFACE, AND ALSO USE A FULLY IMPLICIT SCHEME FOR TIME INTEGRATION.
DIC> @@
DIC> s-s-c 0 1 1 YES POT YES YES 1 2,,,,,,,,,,
 RELEASING OLD STARTING VALUES
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exi3.DIC Y
DIC>
DIC> set-inter
 --OK---
DIC>
```

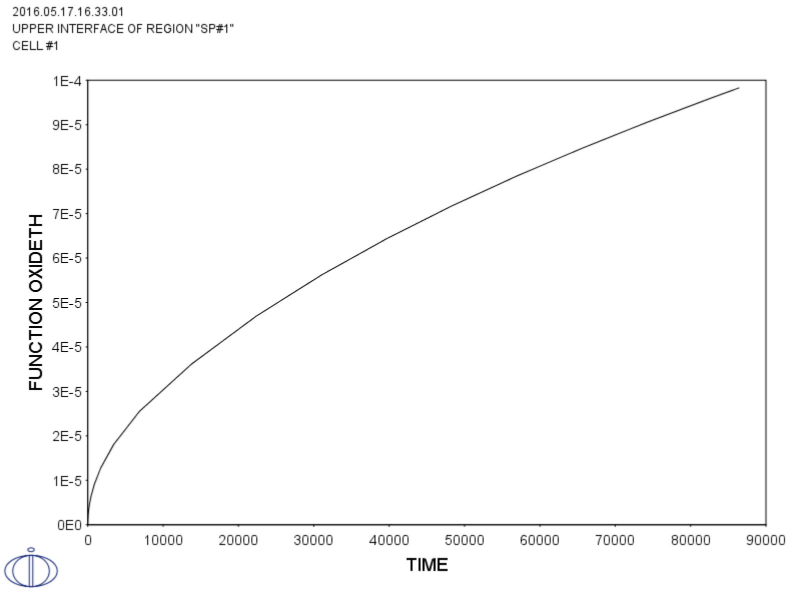
exi3a-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3a\run.DCM"DIC>
DIC>
DIC> @@ exi3_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE i3
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exi3
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> sim y
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                        3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  FE = 0.999999999500542  O = 2.10000395011622E-08
 TOTAL SIZE OF SYSTEM:  .0049999901 [m]
 U-FRACTION IN SYSTEM:  FE = 0.999999999500542  O = 2.10000395011622E-08
 TOTAL SIZE OF SYSTEM:  .0049999901 [m]
 0.348221385917084        0.348220611977384        0.348220784986088        4.628762193141661E-002       3.066957688797020E-023
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.30669577E-22
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.39697035E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999900E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.67055898E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999901E-02
 U-FRACTION IN SYSTEM:  FE = 0.999999998954458  O = 3.64455426696996E-08
 TOTAL SIZE OF SYSTEM:  .00499999012736 [m]

  CPU time used in timestep                      1  seconds
 1.517929307240013E-005        1.518116782832156E-005        1.518200725383254E-005        2.236110879216593E-007        2.496140624875072E-
030
 TIME =  0.30000000E-06 DT =  0.20000000E-06 SUM OF SQUARES =  0.24961406E-29
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14809836E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999899E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.20317373E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999901E-02
 U-FRACTION IN SYSTEM:  FE = 0.999999998714658  O = 4.45725081862438E-08
 TOTAL SIZE OF SYSTEM:  .00499999013837 [m]

  CPU time used in timestep                      1  seconds
 1.697188451126333E-008        1.696117664156532E-008        1.693835140251377E-008        3.448739422006715E-011        1.222514124855179E-
036
 TIME =  0.70000000E-06 DT =  0.40000000E-06 SUM OF SQUARES =  0.12225141E-35
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14883255E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999899E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.21763279E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.999999998165685  O = 6.1983166037256E-08
 TOTAL SIZE OF SYSTEM:  .00499999016589 [m]

  CPU time used in timestep                      3  seconds
 5.775245065901706E-007        5.776794622582910E-007        5.776808970079075E-007        1.113839840259837E-010        1.572717401852155E-
033
 TIME =  0.15000000E-05 DT =  0.80000000E-06 SUM OF SQUARES =  0.15727174E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.92743756E-04 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999898E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.13268868E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.999999997509654  O = 8.32133984608031E-08
 TOTAL SIZE OF SYSTEM:  .00499999019785 [m]

  41  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                      3  seconds
 7.810089590422916E-006        7.813782184860066E-006        7.813457615429235E-006        9.867932325942762E-009        1.905617402005027E-
032
 TIME =  0.31000000E-05 DT =  0.16000000E-05 SUM OF SQUARES =  0.19056174E-31
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.70579640E-04 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999897E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.10157046E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.9999999649525  O = 1.1571601122368E-07
 TOTAL SIZE OF SYSTEM:  .00499999024744 [m]

  31  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                      2  seconds
 8.025849191851695E-006        8.028649710112110E-006        8.028510994403175E-006

output ignored...



... output resumed

  CPU time used in timestep                      2  seconds
 2.795494825223063E-007        2.799988598826527E-007        2.799589408028554E-007        3.840510181353415E-010        6.536877065148708E-
033
 TIME =   56943.895     DT =   8640.0000     SUM OF SQUARES =  0.65368771E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.54604964E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49451130E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.78104770E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50235788E-02
 U-FRACTION IN SYSTEM:  FE = .995206298130482  O = .0156196803359364
 TOTAL SIZE OF SYSTEM:  .00502357883043 [m]
```

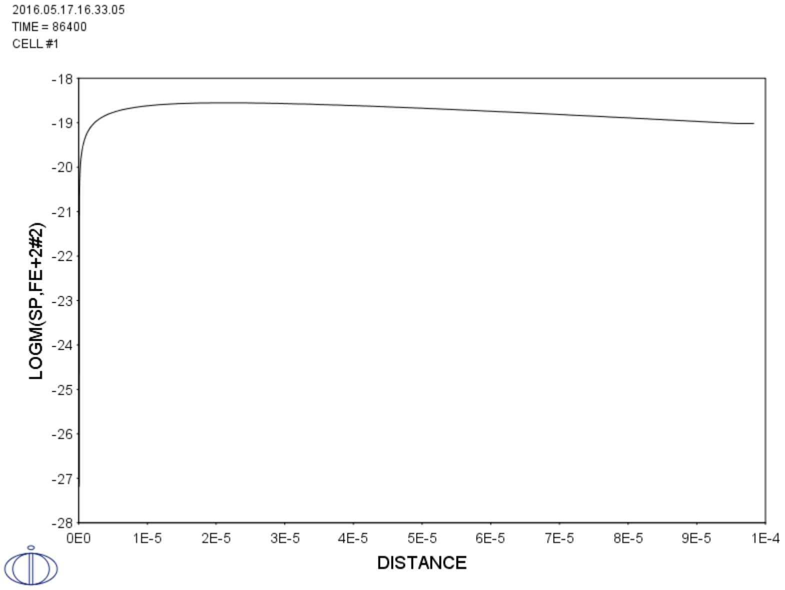
```
  9  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                    2  seconds
 1.758352078734065E-007       1.761476448628660E-007       1.761465359979151E-007       2.270518618218001E-010       4.156548024507609E-
033
 TIME =   65583.895     DT =   8640.0000     SUM OF SQUARES =  0.41565480E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.50596271E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49407415E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.72075536E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50254346E-02
 U-FRACTION IN SYSTEM:  FE = .994832079329009  O = .0168530738288614
 TOTAL SIZE OF SYSTEM:  .00502543463891 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                    3  seconds
 8.203405562105583E-008       8.225224927149764E-008       8.220927275354629E-008       1.252500846596262E-010       1.276680904541070E-
033
 TIME =   74223.895     DT =   8640.0000     SUM OF SQUARES =  0.12766809E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.47386598E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49366473E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.67910782E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50272079E-02
 U-FRACTION IN SYSTEM:  FE = .994479333673158  O = .0180142761374323
 TOTAL SIZE OF SYSTEM:  .00502720792844 [m]

  10  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                    1  seconds
 6.077304940780635E-008       6.093856315328823E-008       6.092638454212013E-008       6.987755380893316E-011       1.736346215486617E-
033
 TIME =   82863.895     DT =   8640.0000     SUM OF SQUARES =  0.17363462E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.44659127E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49327887E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.64023312E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50288810E-02
 U-FRACTION IN SYSTEM:  FE = .994146961515746  O = .0191082519928364
 TOTAL SIZE OF SYSTEM:  .00502888099401 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                    3  seconds
 9.196818475465287E-009       9.142970979109809E-009       9.259105356680440E-009       1.023010000813023E-011       6.770847460736376E-
035
 TIME =   86400.000     DT =   3536.1047     SUM OF SQUARES =  0.67708475E-34
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.44383308E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49312193E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.61654746E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50294917E-02
 U-FRACTION IN SYSTEM:  FE = .994025798981642  O = .0195394100646522
 TOTAL SIZE OF SYSTEM:  .0050294917301 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.30000000E-06
 DELETING TIME-RECORD FOR TIME   0.70000000E-06
 DELETING TIME-RECORD FOR TIME   0.15000000E-05
 DELETING TIME-RECORD FOR TIME   0.31000000E-05
 DELETING TIME-RECORD FOR TIME   0.63000000E-05
 DELETING TIME-RECORD FOR TIME   0.12700000E-04
 DELETING TIME-RECORD FOR TIME   0.25500000E-04
 DELETING TIME-RECORD FOR TIME   0.51100000E-04
 DELETING TIME-RECORD FOR TIME   0.10230000E-03
 DELETING TIME-RECORD FOR TIME   0.20470000E-03
 DELETING TIME-RECORD FOR TIME   0.40950000E-03
 DELETING TIME-RECORD FOR TIME   0.81910000E-03
 DELETING TIME-RECORD FOR TIME   0.16383000E-02
 DELETING TIME-RECORD FOR TIME   0.32767000E-02
 DELETING TIME-RECORD FOR TIME   0.65535000E-02
 DELETING TIME-RECORD FOR TIME   0.13107100E-01
 DELETING TIME-RECORD FOR TIME   0.26214300E-01
 DELETING TIME-RECORD FOR TIME   0.52428700E-01
 DELETING TIME-RECORD FOR TIME   0.10485750
 DELETING TIME-RECORD FOR TIME   0.20971510
 DELETING TIME-RECORD FOR TIME   0.41943030
 DELETING TIME-RECORD FOR TIME   0.83886070
 DELETING TIME-RECORD FOR TIME    1.6777215
 DELETING TIME-RECORD FOR TIME    3.3554431
 DELETING TIME-RECORD FOR TIME    6.7108863
 DELETING TIME-RECORD FOR TIME    13.421773
 DELETING TIME-RECORD FOR TIME    26.843545
 DELETING TIME-RECORD FOR TIME    53.687091
 DELETING TIME-RECORD FOR TIME    107.37418
 DELETING TIME-RECORD FOR TIME    214.74836
 DELETING TIME-RECORD FOR TIME    429.49673
 DELETING TIME-RECORD FOR TIME    858.99346
 DELETING TIME-RECORD FOR TIME    1717.9869
 DELETING TIME-RECORD FOR TIME    3435.9738
 DELETING TIME-RECORD FOR TIME    6871.9477
 DELETING TIME-RECORD FOR TIME    13743.895
 DELETING TIME-RECORD FOR TIME    22383.895
 DELETING TIME-RECORD FOR TIME    31023.895
 DELETING TIME-RECORD FOR TIME    39663.895
 DELETING TIME-RECORD FOR TIME    48303.895
 DELETING TIME-RECORD FOR TIME    56943.895
 DELETING TIME-RECORD FOR TIME    65583.895
 DELETING TIME-RECORD FOR TIME    74223.895

 KEEPING TIME-RECORD FOR TIME    82863.895
 AND FOR TIME                    86400.000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exi3a-plot

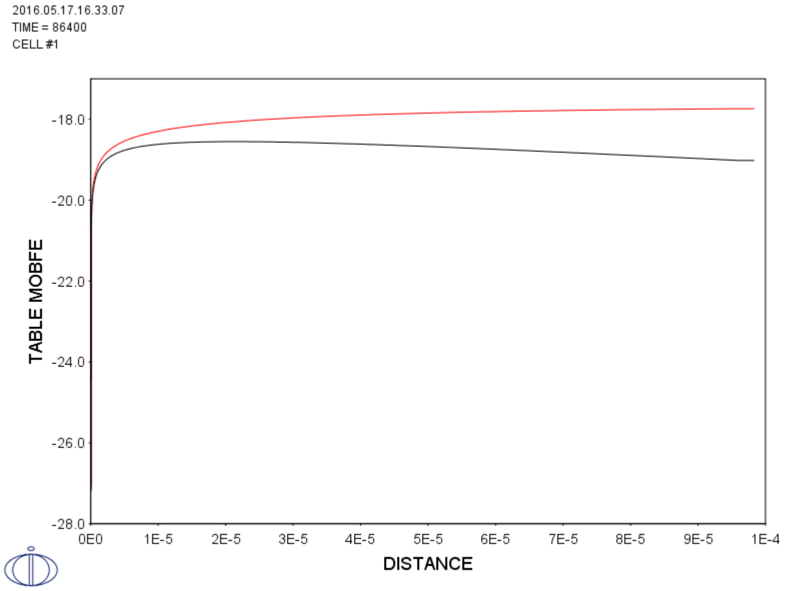
```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3a\plot.DCM"
DIC> @@ exi3_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE i3
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  8.64000E+04
DIC> read exi3
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET'S PLOT THICKNESS OF THE OXIDE LAYER GROWING AT THE SURFACE.
POST-1: @@ FOR THIS PURPOSE WE NEED TO ENTER A FUNCTION ACCORDING TO BELOW.
POST-1: ent func oxideth
FUNCTION: poi(sp,upper)-poi(sp,lower)
&
POST-1: @@
POST-1: @@ WE PUT THIS FUNCTION ON Y-AXIS
POST-1: @@
POST-1: s-d-a y oxideth
POST-1:
POST-1: @@
POST-1: @@ AND PLOT OXIDE THICKNESS VERSUS TIME
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1:
POST-1: @@
POST-1: @@ SINCE WE ARE PLOTTING A FUNCTION WE NEED TO SPECIFY A PLOT CONDITION
POST-1: @@
POST-1: s-p-c interface sp upper
POST-1:
POST-1: plot
```



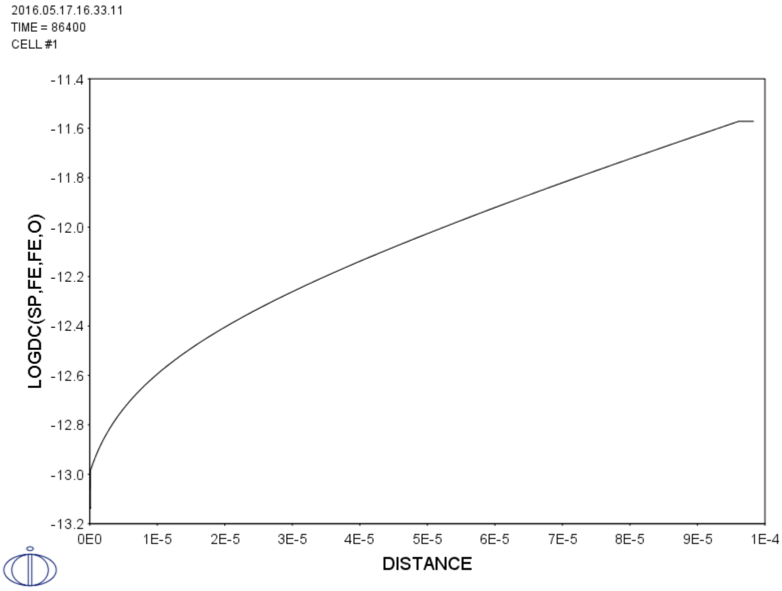

```
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ NOW LET'S PLOT THE MOBILITY IN SPINEL FOR FE+2 ON THE SECOND SUB-LATTICE
POST-1: @@
POST-1: s-d-a y logm(sp,fe+2#2)
POST-1:
POST-1: @@
POST-1: @@ LIMIT THE PLOT TO THE SPINEL PHASE
POST-1: @@
POST-1: s-d-a x dis local sp
  INFO: Distance is set as independent variable
POST-1:
POST-1: s-p-c time 86400
POST-1:
POST-1: plo
```




```
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ LET'S COMPARE MOBILITIES IN SPINEL FOR FE+2 AND FE+3 SPECIES PRESENT ON THE
POST-1: @@ SECOND SUB-LATTICE. FOR THIS COMPARISON WE NEED TO ENTER A TABLE.
POST-1: @@
POST-1: ent table mobfe
Variable(s) logm(sp,fe+2#2) logm(sp,fe+3#2)
POST-1:
POST-1: s-d-a y mobfe
COLUMN NUMBER /*/:
POST-1:
POST-1: plot
```

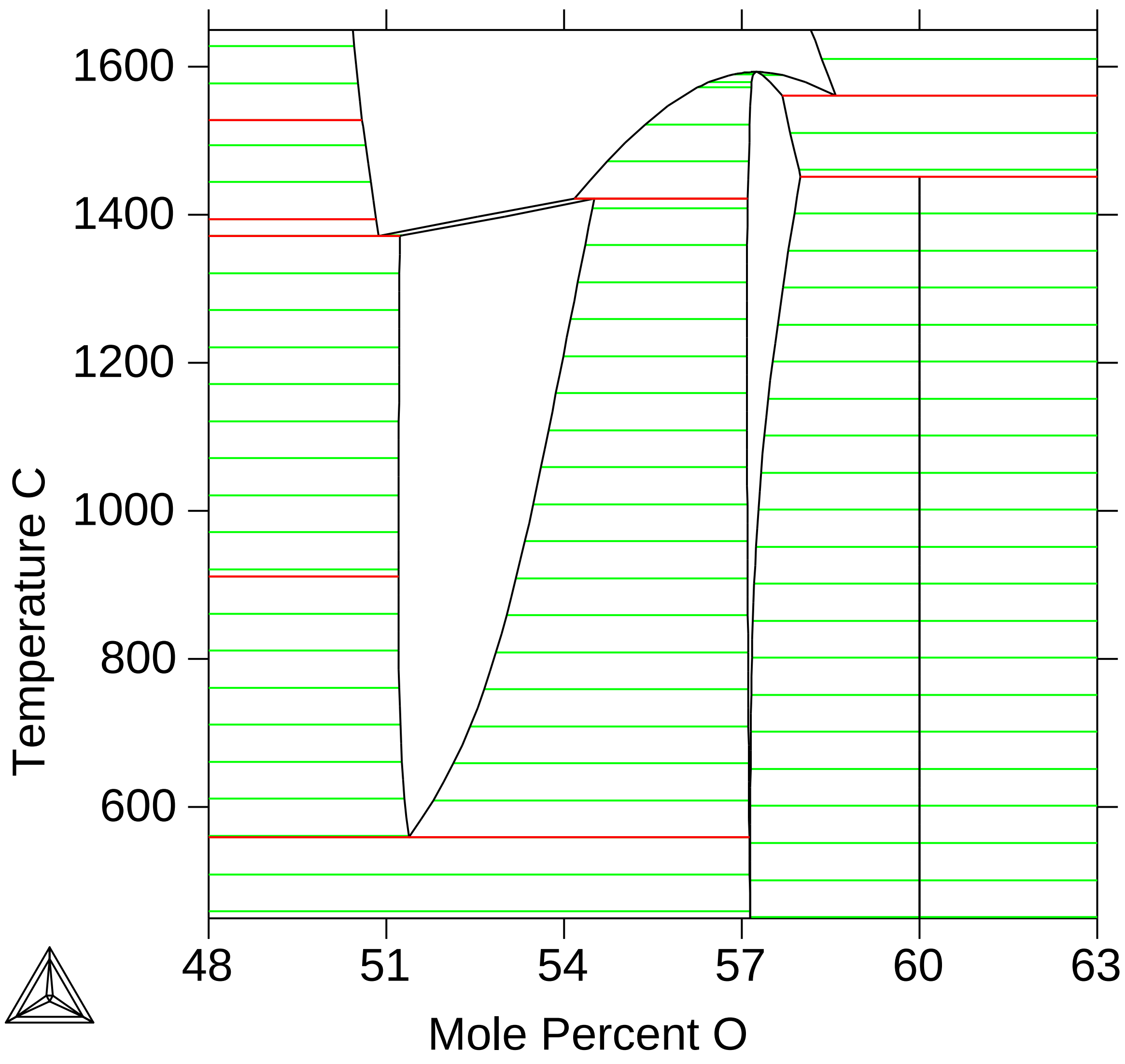



```
POST-1:
POST-1:
POST-1:@?<Hit_return_to_continue>
POST-1:
POST-1: @@
POST-1: @@ NOW LET'S PLOT THE INTERDIFFUSION COEFFICIENT OF FE IN SPINEL
POST-1: @@
POST-1: s-d-a y logdc(sp,fe,fe,o)
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Example i3 Diffusion in FeO with GB contribution



exi3b-setup

```
About Software  (build 9595) running on WinNT 64-bit wordlength
 Compiler: Intel(R) Visual Fortran Composer Version 13.1.0.149 Build 20130118
 License library version: 8.5.1.0017
 Linked: Tue May 17 11:39:18 2016

SYS:SYS:MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3b\setup.DCM" @@
SYS: @@ Diffusion in complex phases.
SYS: @@ Oxidation of iron sample and consequent growth of an oxide layer
SYS: @@ using the grain boundary diffusion contribution model.
SYS: @@
SYS:
SYS: @@ exi3_setup.DCM
SYS:
SYS: @@
SYS: @@ WE START BY GOING TO THE DATABASE MODULE
SYS: @@
SYS: go da
 THERMODYNAMIC DATABASE module
 Current database: Steels/Fe-Alloys  v8.0

 VA  DEFINED
 L12_FCC                  B2_BCC                   B2_VACANCY
 HIGH_SIGMA               DICTRA_FCC_A1  REJECTED
TDB_TCFE8:
TDB_TCFE8: @@
TDB_TCFE8: @@ WE SELECT A USER DATABASE FOR READING THE THERMODYNAMIC DATA
TDB_TCFE8: @@
TDB_TCFE8: sw user FeO.TDB
 Current database: User defined Database
This database does not support the DATABASE_INFORMATION command

 VA                       /-  DEFINED
TDB_USER: def-sys fe o
 FE                       O  DEFINED
TDB_USER: rej sp *
 /-                       VA                       FE
 O                        FE+2                     FE+3
 FE+4                     FE2O3                    FEO
 FEO3/2                   O-2                      O2
   REJECTED
TDB_USER: res sp fe fe+2 fe+3 o o2 o-2 va
 FE                       FE+2                     FE+3
 O                        O2                       O-2
 VA  RESTORED
TDB_USER: rej ph * all
 GAS:G                    BCC_A2                   SPINEL:I
   REJECTED
TDB_USER: res ph bcc spinel gas
 BCC_A2                   SPINEL:I                 GAS:G
   RESTORED
TDB_USER:
TDB_USER: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 Reference REF368     missing close to line                      55
 Reference REF304     missing close to line                      63
 Reference REF420     missing close to line                      65
 Reference REF420     missing close to line                      67
 Reference REF368     missing close to line                      69
 Reference REF368     missing close to line                      70
 Reference REF368     missing close to line                      72
 Reference REF418     missing close to line                      74
 Reference REF418     missing close to line                      76
 Reference REF304     missing close to line                      78
 Reference REF30      missing close to line                      86
 Reference REF30      missing close to line                      88
 Reference REF30      missing close to line                      90
 Reference REF420     missing close to line                      92
 Reference REF420     missing close to line                      94
 Reference REF30      missing close to line                      96
 Reference REF30      missing close to line                      98
 Reference REF30      missing close to line                     100
 Reference REF420     missing close to line                     102
 Reference REF420     missing close to line                     104
 Reference REF30      missing close to line                     106
 Reference REF30      missing close to line                     108
 Reference REF30      missing close to line                     110
 Reference REF420     missing close to line                     112
 Reference REF420     missing close to line                     114
 Reference REF30      missing close to line                     116
 Reference REF30      missing close to line                     118
 Reference REF30      missing close to line                     120
 Reference REF420     missing close to line                     122
 Reference REF420     missing close to line                     124
 Reference REF30      missing close to line                     126
 Reference REF30      missing close to line                     128
 Reference REF30      missing close to line                     130
 Reference REF420     missing close to line                     132
 Reference REF420     missing close to line                     134
 Reference REF30      missing close to line                     136
 Reference REF30      missing close to line                     138
 Reference REF30      missing close to line                     140
 Reference REF420     missing close to line                     142
 Reference REF420     missing close to line                     144
 Reference REF30      missing close to line                     146
 Reference REF30      missing close to line                     148
 Reference REF30      missing close to line                     150
 Reference REF420     missing close to line                     152
 Reference REF420     missing close to line                     154
 Reference REF30      missing close to line                     156
 Reference REF30      missing close to line                     158
 Reference REF30      missing close to line                     160
 Reference REF420     missing close to line                     162
 Reference REF420     missing close to line                     164
 Reference REF30      missing close to line                     166
 Reference REF30      missing close to line                     168
 Reference REF30      missing close to line                     170
 Reference REF420     missing close to line                     172
```

```
 Reference REF420     missing close to line                    174
 Reference REF30      missing close to line                    176
 Reference REF30      missing close to line                    178
 Reference REF30      missing close to line                    180
 Reference REF420     missing close to line                    182
 Reference REF420     missing close to line                    184
 Reference REF30      missing close to line                    186
 Reference REF30      missing close to line                    188
 Reference REF30      missing close to line                    190
 Reference REF420     missing close to line                    192
 Reference REF420     missing close to line                    194
 Reference REF30      missing close to line                    196
 Reference REF30      missing close to line                    198
 Reference REF30      missing close to line                    200
 Reference REF420     missing close to line                    202
 Reference REF420     missing close to line                    204
 FUNCTIONS ....
 -OK-
TDB_USER:
TDB_USER: @@
TDB_USER: @@ SWITCH TO A USER DEFINED MOBILITY DATABASE TO RETRIEVE MOBILITY DATA
TDB_USER: @@
TDB_USER: app user FeOmob.TDB
 Current database: User defined Database

         test database

 VA                       /-                       O
   DEFINED
TDB_APP: def-sys fe o
 FE  DEFINED
TDB_APP: rej sp *
 /-                       VA                       FE
 O                        FE+2                     FE+3
 FE2O3                    FEO                      FEO3/2
 O-2                      O2  REJECTED
TDB_APP: res sp fe fe+2 fe+3 o o2 o-2 va
 FE                       FE+2                     FE+3
 O                        O2                       O-2
 VA  RESTORED
TDB_APP: rej ph * all
 SPINEL:I                 GAS:G                    BCC_A2
   REJECTED
TDB_APP: res ph bcc spinel gas
 BCC_A2                   SPINEL:I                 GAS:G
   RESTORED
TDB_APP:
TDB_APP: get
 ELEMENTS .....
 SPECIES ......
 PHASES .......
 PARAMETERS ...
 FUNCTIONS ....

 List of references for assessed data


 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE

 *** ERROR  1000 IN TDBGRT
 *** NO REFERENCE FILE
 -OK-
TDB_APP:
TDB_APP:
TDB_APP: @@
TDB_APP: @@ ENTER THE DICTRA MONITOR
TDB_APP: @@
TDB_APP: go d-m
 NO TIME STEP DEFINED
DIC>
DIC>
DIC> @@
DIC> @@  ENTER GLOBAL CONDITION T
DIC> @@
DIC> set-cond glob T 0 823; * N

DIC>
DIC>
DIC> @@
DIC> @@  SET REFERENCE STATE FOR O TO O2 (GAS)
DIC> @@
DIC> set-ref o gas,,,,,,,
DIC>
DIC>
DIC> @@
DIC> @@ ENTER REGIONS fer AND sp
DIC> @@
DIC> ent-reg fer
DIC> ent-reg sp,,,,,,,,
DIC>
DIC>
DIC> @@
DIC> @@ ENTER PHASES INTO REGIONS
DIC> @@
DIC> ent-phase act fer matrix bcc#1
DIC> ent-phase act sp matrix spinel
DIC>
DIC>
DIC> @@
DIC> @@ ENTER GRIDS INTO THE REGIONS
DIC> @@
DIC>
DIC> @@
DIC> @@ Enter a size for the ferrite
DIC> @@
DIC> @@
DIC> ent-grid fer 4.99999e-3 lin 50
DIC>
DIC> @@
DIC> @@ Enter a thin initial size for the oxide
DIC> @@
DIC> @@
```

```
DIC> ent-grid sp  1.00e-10 lin 50
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN BCC
DIC> @@
DIC> ent-comp fer bcc#1 m-f
PROFILE FOR /O/: o lin 1e-9 1e-9
DIC>
DIC>
DIC> @@
DIC> @@ ENTER INITIAL COMPOSITIONS IN THE OXIDE
DIC> @@
DIC> ent-comp sp spinel m-f
  this is a phase with charged species
  with more than 2 sublattices
PROFILE FOR /FE/: FE lin 4.28771E-01 4.28549E-01
DIC>
DIC>
DIC> @@
DIC> @@ ENTER A BOUNDARY CONDITION "GAS" ON THE UPPER (RIGHTMOST) INTERFACE
DIC> @@ OF THE OXIDE. THIS WILL ALLOW THE SYSTEM TO EXPAND AND THE OXIDE LAYER
DIC> @@ TO GROW EXTERNALY. FOR THIS EXAMPLE WE HAVE SPECIFIED AN OXYGEN ACTIVITY
DIC> @@ THAT IS LOW ENOUGH IN ORDER NOT TO FORM CORUNDUM (FE2O3). WE ALSO
DIC> @@ SPECIFY THAT THERE IS NO FLUX OF FE ACROSS THIS INTERFACE, I.E. NO FE
DIC> @@ IS ALLOWED TO ENTER OR LEAVE THE SYSTEM.
DIC> @@
DIC> set-cond boundary upper gas

TYPE OF CONDITION FOR COMPONENT FE /ZERO_FLUX/: zero-flux
TYPE OF CONDITION FOR COMPONENT O /ZERO_FLUX/: act
LOW TIME LIMIT /0/: 0 4.5e-4; * N
DIC>
DIC>
DIC> @@
DIC> @@ ENTER START VALUES FOR INITIAL INTERFACE VELOCITIES
DIC> @@
DIC> s-a-s-v -1e-5 1e-5 yes
 STARTING VALUES WILL BE TAKEN FROM PROFILES
DIC>
DIC>
DIC> @@
DIC> @@ SIMULATE FOR 24 HOURS
DIC> @@
DIC> s-s-time 86400,,,,
SMALLEST ACCEPTABLE TIMESTEP : /1E-07/:
DIC>
DIC> @@
DIC> @@ SPECIFY THAT POTENTIALS AND NOT ACTIVITIES ARE VARIED AT THE PHASE
DIC> @@ INTERFACE, AND ALSO USE A FULLY IMPLICIT SCHEME FOR TIME INTEGRATION.
DIC> @@
DIC> s-s-c 0 1 1 YES POT YES YES 1 2,,,,,,,,,,
 RELEASING OLD STARTING VALUES
DIC>
DIC>
DIC> @@ Enable the grainboundary diffusion contribution model
DIC> GB
REGION NAME : /SP/: SP
PHASE NAME: /SPINEL/: SPINEL
Enable model for grainboundary contribution to diffusion /YES/: YES
Grainboundary thickness /5E-10/: 5e-10
Grainsize(T,P,TIME)= 10.0e-6;
Bulkdiffusion activation energy multiplier /.5/: 0.333333
Enable model for dislocation contribution to diffusion /YES/: NO
DIC>
DIC>
DIC>
DIC> @@
DIC> @@ SAVE THE SETUP ON A NEW STORE FILE AND EXIT
DIC> @@
DIC> save exi3b.DIC Y
DIC>
DIC>
DIC>
DIC> set-inter
 --OK---
DIC>
```

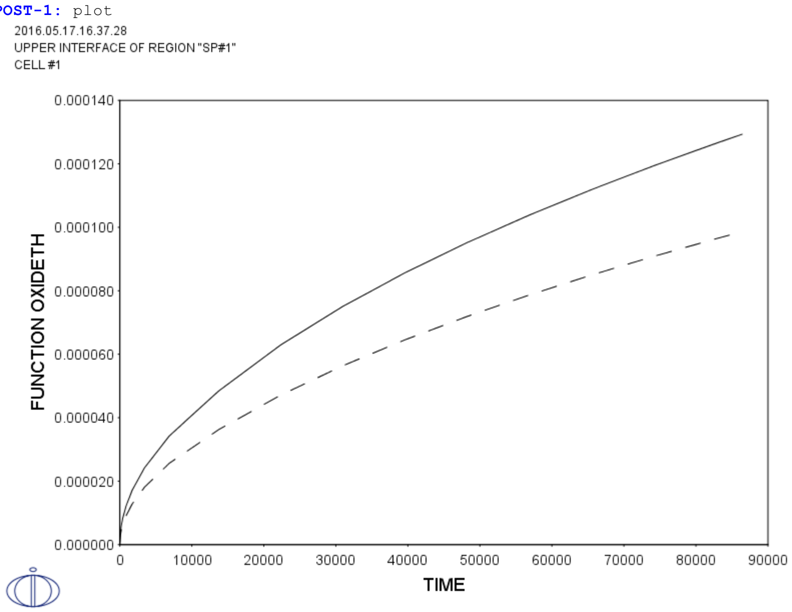
exi3b-run

```
DIC>AboutMACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3b\run.DCM"
DIC>
DIC> @@ exi3_run.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR RUNNING EXAMPLE i3
DIC> @@
DIC>
DIC> @@
DIC> @@ ENTER THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  0.00000E+00
DIC> read exi3b
 OK
DIC>
DIC> @@
DIC> @@ Start the simulation
DIC> @@
DIC> sim y
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
  Trying old scheme                     3
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 Old start values kept
 Automatic start values will be set
 U-FRACTION IN SYSTEM:  FE = 0.99999999500542  O = 2.10000395011622E-08
 TOTAL SIZE OF SYSTEM:  .0049999901 [m]
 U-FRACTION IN SYSTEM:  FE = 0.99999999500542  O = 2.10000395011622E-08
 TOTAL SIZE OF SYSTEM:  .0049999901 [m]
 3.754012022235730E-002       3.754011717918598E-002       3.753985968178590E-002       2.214551768329310E-004       8.343190148843726E-
027
 TIME =  0.10000000E-06 DT =  0.10000000E-06 SUM OF SQUARES =  0.83431901E-26
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.16215907E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999900E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.29639326E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999901E-02
 U-FRACTION IN SYSTEM:  FE = 0.9999999923282  O = 2.89622134513929E-08
 TOTAL SIZE OF SYSTEM:  .00499999011342 [m]

  CPU time used in timestep                   0  seconds
 8.130039137525315E-006       8.129356847939127E-006       8.129303947595926E-006       3.439442535642660E-007       3.607676798266860E-
032
 TIME =  0.30000000E-06 DT =  0.20000000E-06 SUM OF SQUARES =  0.36076768E-31
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.37359387E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999899E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.56870214E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.99999998464714  O = 5.17103442902628E-08
 TOTAL SIZE OF SYSTEM:  .00499999015245 [m]

  CPU time used in timestep                   0  seconds
 8.253489276921684E-006       8.254889055599710E-006       8.255101752627559E-006       5.598357468399940E-009       5.023804246657276E-
032
 TIME =  0.70000000E-06 DT =  0.40000000E-06 SUM OF SQUARES =  0.50238042E-31
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.17497742E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999898E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.23932537E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.99999997898039  O = 7.08564117688965E-08
 TOTAL SIZE OF SYSTEM:  .00499999017818 [m]

  22  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                   0  seconds
 9.729908171215534E-008       9.738993488286263E-008       9.735125014216650E-008       4.714125841858521E-010       5.045221753729258E-
035
 TIME =  0.15000000E-05 DT =  0.80000000E-06 SUM OF SQUARES =  0.50452218E-34
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.14499522E-03 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999897E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.21010054E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999902E-02
 U-FRACTION IN SYSTEM:  FE = 0.99999996837576  O = 1.04472565347718E-07
 TOTAL SIZE OF SYSTEM:  .00499999023027 [m]

  CPU time used in timestep                   1  seconds
 4.655614600663251E-007       4.656979205865371E-007       4.656961329764379E-007       2.459594013142007E-010       5.544489469354008E-
033
 TIME =  0.31000000E-05 DT =  0.16000000E-05 SUM OF SQUARES =  0.55444895E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.94085177E-04 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49999896E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.13401135E-03 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.49999903E-02
 U-FRACTION IN SYSTEM:  FE = 0.99999995513277  O = 1.47356281891432E-07
 TOTAL SIZE OF SYSTEM:  .00499999029415 [m]

  35  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                   0  seconds
 1.112705457832001E-005       1.113153854177326E-005       1.113110151447614E-005       1.773309272209452E-008

output ignored...


... output resumed

  CPU time used in timestep                   0  seconds
 5.293743519964927E-007       5.301697672488749E-007       5.301054992486230E-007       7.125704321662739E-010       4.641792092527049E-
034
 TIME =   56943.895     DT =   8640.0000     SUM OF SQUARES =  0.46417921E-33
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.70214915E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49272875E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.10042297E-08 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50312157E-02
 U-FRACTION IN SYSTEM:  FE = .993662009801502  O = .0206568677840276
 TOTAL SIZE OF SYSTEM:  .00503121573903 [m]
```

```
  10  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                     1  seconds
 2.897339947907232E-007       2.902720955064225E-007       2.902260023459213E-007       3.731355838770750E-010       2.528535368392773E-
033
 TIME =   65583.895     DT =   8640.0000     SUM OF SQUARES =  0.25285354E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.64904044E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49216797E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.92924029E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50336367E-02
 U-FRACTION IN SYSTEM:  FE = .993181211093957  O = .0222419298164428
 TOTAL SIZE OF SYSTEM:  .00503363666571 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                     0  seconds
 1.645599404527048E-007       1.649350681228161E-007       1.649029196813604E-007       1.998603507767561E-010       2.892656498503485E-
033
 TIME =   74223.895     DT =   8640.0000     SUM OF SQUARES =  0.28926565E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.60660303E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49164387E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.86930101E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50359064E-02
 U-FRACTION IN SYSTEM:  FE = .992731292721503  O = .0237233467729492
 TOTAL SIZE OF SYSTEM:  .00503590637626 [m]

  11  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                     1  seconds
 1.054646105158804E-007       1.057429582007526E-007       1.057239264696965E-007       1.216534152760058E-010       1.327086102304330E-
033
 TIME =   82863.895     DT =   8640.0000     SUM OF SQUARES =  0.13270861E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.57108932E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49115045E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.81849808E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50380440E-02
 U-FRACTION IN SYSTEM:  FE = .992308185941587  O = .0251169652507858
 TOTAL SIZE OF SYSTEM:  .0050380439879 [m]

  8  GRIDPOINT(S) REMOVED FROM CELL #1   REGION: SP

  CPU time used in timestep                     0  seconds
 1.547097681221731E-008       1.541227861501759E-008       1.559316088470541E-008       1.096898577181496E-011       1.032930395954560E-
033
 TIME =   86400.000     DT =   3536.1047     SUM OF SQUARES =  0.10329304E-32
 CELL # 1 VELOCITY AT INTERFACE # 2 IS  -0.56493954E-09 AND    0.0000000
 POSITION OF INTERFACE FER / SP IS   0.49095068E-02
 CELL # 1 VELOCITY AT INTERFACE # 3 IS   0.78542429E-09 AND    0.0000000
 POSITION OF INTERFACE SP / gas interface IS   0.50388236E-02
 U-FRACTION IN SYSTEM:  FE = .992154105704416  O = .0256642674961308
 TOTAL SIZE OF SYSTEM:  .00503882364503 [m]

 MUST SAVE WORKSPACE ON FILE
 WORKSPACE SAVED ON FILE
 RECLAIMING WORKSPACE
 DELETING TIME-RECORD FOR TIME    0.0000000
 DELETING TIME-RECORD FOR TIME   0.10000000E-06
 DELETING TIME-RECORD FOR TIME   0.30000000E-06
 DELETING TIME-RECORD FOR TIME   0.70000000E-06
 DELETING TIME-RECORD FOR TIME   0.15000000E-05
 DELETING TIME-RECORD FOR TIME   0.31000000E-05
 DELETING TIME-RECORD FOR TIME   0.63000000E-05
 DELETING TIME-RECORD FOR TIME   0.12700000E-04
 DELETING TIME-RECORD FOR TIME   0.25500000E-04
 DELETING TIME-RECORD FOR TIME   0.51100000E-04
 DELETING TIME-RECORD FOR TIME   0.10230000E-03
 DELETING TIME-RECORD FOR TIME   0.20470000E-03
 DELETING TIME-RECORD FOR TIME   0.40950000E-03
 DELETING TIME-RECORD FOR TIME   0.81910000E-03
 DELETING TIME-RECORD FOR TIME   0.16383000E-02
 DELETING TIME-RECORD FOR TIME   0.32767000E-02
 DELETING TIME-RECORD FOR TIME   0.65535000E-02
 DELETING TIME-RECORD FOR TIME   0.13107100E-01
 DELETING TIME-RECORD FOR TIME   0.26214300E-01
 DELETING TIME-RECORD FOR TIME   0.52428700E-01
 DELETING TIME-RECORD FOR TIME   0.10485750
 DELETING TIME-RECORD FOR TIME   0.20971510
 DELETING TIME-RECORD FOR TIME   0.41943030
 DELETING TIME-RECORD FOR TIME   0.83886070
 DELETING TIME-RECORD FOR TIME    1.6777215
 DELETING TIME-RECORD FOR TIME    3.3554431
 DELETING TIME-RECORD FOR TIME    6.7108863
 DELETING TIME-RECORD FOR TIME    13.421773
 DELETING TIME-RECORD FOR TIME    26.843545
 DELETING TIME-RECORD FOR TIME    53.687091
 DELETING TIME-RECORD FOR TIME    107.37418
 DELETING TIME-RECORD FOR TIME    214.74836
 DELETING TIME-RECORD FOR TIME    429.49673
 DELETING TIME-RECORD FOR TIME    858.99346
 DELETING TIME-RECORD FOR TIME    1717.9869
 DELETING TIME-RECORD FOR TIME    3435.9738
 DELETING TIME-RECORD FOR TIME    6871.9477
 DELETING TIME-RECORD FOR TIME    13743.895
 DELETING TIME-RECORD FOR TIME    22383.895
 DELETING TIME-RECORD FOR TIME    31023.895
 DELETING TIME-RECORD FOR TIME    39663.895
 DELETING TIME-RECORD FOR TIME    48303.895
 DELETING TIME-RECORD FOR TIME    56943.895
 DELETING TIME-RECORD FOR TIME    65583.895
 DELETING TIME-RECORD FOR TIME    74223.895

 KEEPING TIME-RECORD FOR TIME    82863.895
 AND FOR TIME                    86400.000
 WORKSPACE RECLAIMED
DIC>
DIC> set-inter
 --OK---
DIC>
```

exi3b-plot

```
DIC>About NO SUCH COMMAND, USE HELP
DIC>DIC>MACRO "c:\jenkins\workspace\DICTRA-Generate-Console-Examples\examples\exi3b\plot.DCM"DIC>
DIC> @@ exi3_plot.DCM
DIC>
DIC> @@
DIC> @@ FILE FOR GENERATING GRAPHICAL OUTPUT FOR EXAMPLE i3
DIC> @@
DIC>
DIC> @@
DIC> @@ GO TO THE DICTRA MONITOR AND READ THE STORE RESULT FILE
DIC> @@
DIC> go d-m
 TIME STEP AT TIME  8.64000E+04
DIC> read exi3b
 OK
DIC>
DIC> @@
DIC> @@ ENTER THE POST PROCESSOR
DIC> @@
DIC> post

  POST PROCESSOR VERSION   1.7
  Implemented by Bjorn Jonsson

POST-1:
POST-1: @@
POST-1: @@ LET'S PLOT THICKNESS OF THE OXIDE LAYER GROWING AT THE SURFACE.
POST-1: @@ FOR THIS PURPOSE WE NEED TO ENTER A FUNCTION ACCORDING TO BELOW.
POST-1: ent func oxideth
FUNCTION: poi(sp,upper)-poi(sp,lower)
&
POST-1: @@
POST-1: @@ WE PUT THIS FUNCTION ON Y-AXIS
POST-1: @@
POST-1: s-d-a y oxideth
POST-1:
POST-1: @@
POST-1: @@ AND PLOT OXIDE THICKNESS VERSUS TIME
POST-1: @@
POST-1: s-d-a x time
  INFO: Time is set as independent variable
POST-1:
POST-1: @@
POST-1: @@ SINCE WE ARE PLOTTING A FUNCTION WE NEED TO SPECIFY A PLOT CONDITION
POST-1: @@
POST-1: s-p-c interface sp upper
POST-1:
POST-1: app y exi3a.exp 0; 1;
POST-1:
POST-1: plot
```




```
POST-1:
POST-1:
POST-1: set-inter
 --OK---
POST-1:
```

Thermo-Calc User Guide

Version 2016a

Thermo-Calc Software

Introduction to Thermo-Calc

This guide is an introduction to working with both *Graphical Mode* and *Console Mode* in Thermo-Calc.

In this section:

The Thermo-Calc Software

The Thermo-Calc software is a sophisticated database and programming interface package used to perform thermodynamic calculations. It can calculate complex homogeneous and heterogeneous phase equilibria, and then plot the results as property diagrams and phase diagrams.

The Thermo-Calc software (also referred to as the *Thermo-Calc program* or *the software* to distinguish it from the company name) fully supports stoichiometric and non-ideal solution models and databases. These models and databases can be used to make calculations on a large variety of materials such as steels, alloys, slags, salts, ceramics, solders, polymers, subcritical aqueous solutions, supercritical electrolyte solutions, non-ideal gases and hydrothermal fluids or organic substances. The calculations take into account a wide range of temperature, pressure and compositions conditions.

A Software Development Kit (SDK) is available. This SDK is used to plug the calculation engine into your own application or into other third-party applications.

The Thermo-Calc program also includes modules to run simulations of:

- Diffusion controlled transformations in multicomponent systems: Diffusion Module (DICTRA)and DIC_PARROT.
- Diffusion controlled multi-particle precipitation process in multicomponent and multi-phase alloy systems: Precipitation Module (TC-PRISMA).

All Thermo-Calc users have access to the Precipitation Calculator to use it with two elements. To fully realise the power of the Precipitation Module you need an additional license key. There are also two examples available to all users. All examples use demonstration databases, which are included with every installation.

Use online help for information about how to use Diffusion Module (DICTRA) and Precipitation Module (TC-PRISMA).

Typographical Conventions

The following typographical conventions are used throughout the documentation:

Convention	Definition
Forward arrow symbol →	The forward arrow symbol → instructs you to select a series of menu items in a specific order. For example, **Tools → Options** is equivalent to: From the **Tools** menu, select **Options**.
Boldface font	A **boldface** font indicates that the given word(s) are shown that way on a toolbar button or as a menu selection. For example, if you are told to select a menu item in a particular order, such as **Tools→Options**, or to click **Save**.

Convention	Definition
Italic font	An *italic* font indicates the introduction of important terminology. Expect to find an explanation in the same paragraph or elsewhere in the guide.
COMMAND	For features in Thermo-Calc that use the command line, this font and all capital letters indicates that this is a COMMAND used in the Console Mode terminal. Examples of how you can use a command are written with code font: Use DEFINE_ELEMENTS followed by a list of the elements that you want in your system. (To list the elements that are available in your current database, use LIST_DATABASE and choose `Elements`).
HELP	Text in blue and underline and a page number is a link to another topic in the current or referenced guide. Often command names are also topics. Clicking the link takes you to more detail about a particular command or subject in the PDF.
<Enter>	Text with <angle brackets> indicates a keyboard entry. Usually to press <Enter> (or Return).
`code` and **`code bold`**	A `code` font shows a programming code or code example. The **`code bold`** font highlights the entry.
Important	Provides important information and indicates that more detail is located in the linked or named topic.
Note	The information can be of use to you. It is recommended that you read the text or follow the link to more information.
Also see	Go to more information about the topic being discussed.
Examples	Go to the example collection to learn more.

Graphical Mode vs Console Mode

There are two interfaces in Thermo-Calc: *Graphical Mode* with a graphical user interface (GUI) and *Console Mode*, which uses a command line interface.

- In Graphical Mode calculations are set up, carried out, and visualized as part of a *project*. The steps in the project are performed with *activities*. There are templates and a

Wizard available to guide you through the process of defining the project.

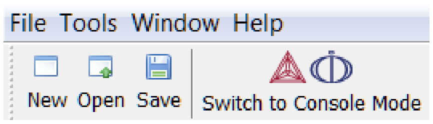
- In Console Mode you work with *modules*, which are managed using commands typed at a prompt. Some modules, called *response-driven modules*, prompt you with a series of questions that typically take you through the process of defining your system, setting calculation conditions, performing calculations and plotting the results.

Switching Between Modes

The first time you open Thermo-Calc, it defaults to Graphical Mode. For any future instance it defaults to the last mode used.

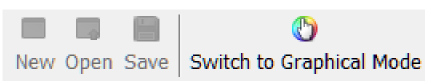
Along the top of the GUI is the toolbar where you switch between modes.

The toolbar in Graphical Mode



The toolbar in Console Mode

In Console Mode, except for the **Switch to Graphical Mode** button, the toolbar is not used.



See *Main Menu and Toolbar* on page 32 for information about the **Tools**, **Window** and **Help** menus, which are available in both modes.

The two modes can be run simultaneously, but there is no communication between them. What you do in Graphical Mode does not affect the state of the Console Mode session and vice versa. One exception is the plot settings. See *Global Settings: Graphical and Console Mode-Plotting* on page 200.

Although many calculations can be done in either mode, only data optimization and thermodynamic or kinetic assessments are available in Console Mode.

Help Resources

Online and Context Help

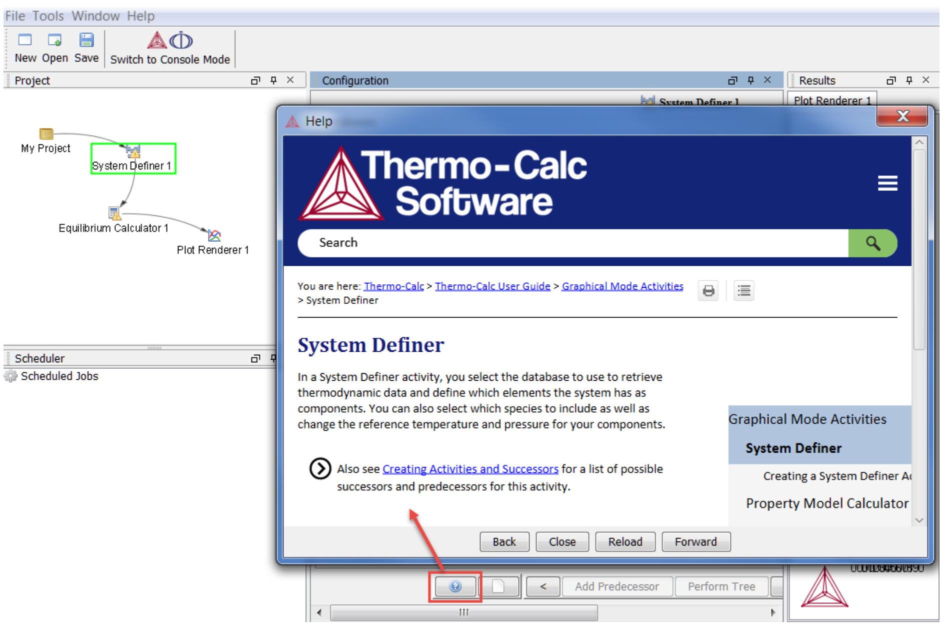
Online Help

To access online help in a browser, open Thermo-Calc and select **Help →Online Help**. The content opens in a browser but uses local content so you don't need an internet connection.

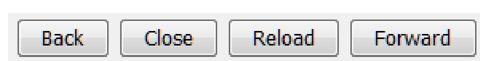
Context Help

When you are in Graphical Mode, you can access feature help (also called *topic-sensitive* or *context help*) for the activity nodes in the tree.

1. In the Project window, click a node. For example, **System Definer**.
2. In the lower left corner of the **Configuration** window, click the help button .
3. The **Help** window opens to the relevant topic.



The window that opens has the same content as the help you access in the browser. There are these extra navigation buttons in this view.

- Use the buttons on the bottom of the window, , to navigate **Back** and **Forward** (these are only active once you have started using the help to load pages and create a history), to **Close** the window, and **Reload** the original content.

Console Mode Help

Console Mode is for Thermo-Calc and the Diffusion Module (DICTRA).

In Console Mode at the command line prompt, you can access help in these ways:

- For a list of all the available commands in the current module, at the prompt type a question mark (`?`) and press <Enter>.
- For a description of a specific command, type `Help` followed by the name of the command. You can only get online help about a command related to the current module you are in.
- For general system information type `Information`. Specify the subject or type `?` and the available subjects are listed. This subject list is specific to the current module.

Examples, Documentation and Materials

To learn more about how to use the software you can open and run the example projects (Graphical Mode) and macro files (Console Mode). These are in the format of project files (*.TCU) and macro files (*.TCM), respectively. There are also PDF documentation sets and database information sheets included with your installation. These are also available on the Thermo-Calc Software website.

Installation Resources

On Windows, once Thermo-Calc is installed, you can also locate the Examples and Materials folders, plus all the Manuals using the shortcuts located in the Start menu. Go to **Start →** **All Programs >Thermo-Calc** and click **Examples**, **Manuals**, or **Materials** as required to open the applicable folder.

In the table, *<user>* stands for the username and *<version>* for the version of Thermo-Calc, for example 2016a.

OS	User type	Default directory
Windows	Normal user	Users*<user>*\Thermo-Calc*<version>* Users*<user>*\Documents\Thermo- Calc*<version>* My documents
	Administrator	Program Files\Thermo-Calc*<version>* Users\Public\Documents\Thermo- Calc*<version>* Public documents
Mac	Administrator (user name and password required)	Examples and manuals in /Users/Shared/Thermo-Calc/*<version>* To go to this folder, in Finder, from the **Go** main menu select **Go to folder**. Enter the above file path and click **Go**.
Linux	Non root user	home/*<user>*/Thermo-Calc/*<version>*
	Root user	usr/local/Thermo-Calc/*<version>*

PDF Documentation

Based on your needs, there is PDF documentation available as part of your installation. These do not include the Database Information Sheets (see below).

- **Thermo-Calc Documentation Set**: Includes all the manuals needed to work with Thermo-Calc including the installation guides. For the other add-on module documentation, see the separate PDFs.
- **Installation Guides**: Includes the quick install guides for Standalone Windows and Mac installations as well as the full installation guide for all platforms and installation types.
- **Diffusion Module (DICTRA) Documentation Set**: Includes Diffusion Module (DICTRA)-specific manuals plus the Thermo-Calc User Guide, Console Mode Command Reference, Data Optimization User Guide, Database Manager User Guide, and the DATAPLOT User

Guide.

- **Precipitation Module (TC-PRISMA) User Guide**
- **SDK (Software Development Kit) Documentation Set**: Includes TQ-Interface Programmer Guide, TC Toolbox for MATLAB® Programmer Guide, and TC-API Programmer Guide.

Database Information Sheets

The database information sheets are included in the online help system and in a **Manuals** sub-folder . You can also view these on the Thermo-Calc website.

Graphical Mode Example Collection

After you have accessed the *Examples, Documentation and Materials* on page 7, go to the Graphical Mode Example folder to open one of the project files in Thermo-Calc.

Opening a Project File

Navigate to the file location and double-click the `*.tcu` file to open it. It launches a new session of the software. Or you can select **File →Open Project** from an open session of Thermo-Calc.

Also see the *Thermo-Calc Graphical Mode Examples Guide*.

The project files have the extension `*.tcu`.

- Ex_01_Single-point_equilibrium.tcu
- Ex_02_Step_in_temperature_in_Fe-C.tcu
- Ex_03_Fe-C_phase_diagram.tcu
- Ex_04_Fe-Cr-C_ternary_phase_diagram.tcu
- Ex_05_Fe-C_stable_and_metastable_phase_diagram.tcu
- Ex_06_Serial_equilibrium_calculators.tcu
- Ex_07_User_defined_functions.tcu
- Ex_08_Scheil_and_equilibrium_solidification.tcu
- Ex_09_Heat_map_of_carbide_driving_force.tcu
- Ex_10_Phase_Transition.tcu
- Ex_11_Coarsening_and_Interfacial_energy.tcu
- Ex_12_Driving_force_and_Interfacial_energy.tcu
- Ex_13_Precipitation_Al-Sc_AL3SC.tcu
- Ex_14_Precipitation_Fe-C-Cr_Cementite-M7C3-M23C6.tcu
- Ex_15_Precipitation_Fe-C-Cr_TTT_Cementite-M7C3-M23C6.tcu
- Ex_16_Precipitation_Fe-C_Cemetite.tcu
- Ex_17_Precipitation_Ni-Al-Cr_Isothermal_Gamma-Gamma_prime.tcu
- Ex_18_Precipitation_Ni-Al-Cr_Non-isothermal_Gamma-Gamma_prime.tcu

Examples of Thermo-Calc Activities

Example 1: Calculating a Single-Point Equilibrium

This example shows the result from a single-point equilibrium calculation in the Fe-C system. The number of equilibrium conditions is C+2 where C is the number of components, meaning that four conditions are needed in this example:

- **Temperature** is 1000 K
- **Pressure** is 101325 Pa
- **System size** is 1 mole
- **Mass percent carbon** is 0.1%

The displayed results of the calculation show that the BCC_A2 (ferrite) and GRAPHITE phases are stable for this set of equilibrium conditions.

File Name	Activity Example
Ex_01_Single-point_equilibrium	Equilibrium Calculator

Example 2: Stepping in Temperature in the Fe-C System

This example shows how the fractions of stable phases vary for an Fe-0.1 mass-% C alloy when the temperature is varied between 500 and 2000 K. To allow temperature to vary, the corresponding **Axis Definition** check box is selected. In this example, results are displayed graphically using a **Plot Renderer** activity. To get text results a **Table Renderer** activity is used instead.

File Name	Activity Examples
Ex_02_Step_in_temperature_in_Fe-C	Equilibrium Calculator, Plot Renderer and a Table Renderer

Example 3: Fe-C Phase Diagrams

This example shows the stable Fe-C phase diagram (stable meaning that the graphite phase is entered in the calculation). The same diagram is calculated using a:

- **Binary Calculator** activity and a
- **System Definer** and an **Equilibrium Calculator** activity. In both cases, a **Plot Renderer** activity is used to display results. The purpose of the **Binary Calculator** is to simplify common calculations for binary systems. In the **Equilibrium Calculator** two axes are defined: *Mass percent C* and *Temperature*. An axis variable must also be an equilibrium condition.

File Name	Activity Examples
Ex_03_Fe-C_phase_diagram	Binary Calculator, System Definer and Equilibrium Calculator

Example 4: Ternary Phase Diagram in the Fe-Cr-C System at 1000 K

This example shows a ternary phase diagram in the Fe-Cr-C system at 1000 K. Similar to Example 3, the same diagram is calculated using:

- a **Ternary Calculator** activity and
- a **System Definer** and an **Equilibrium Calculator** activity. To toggle between a triangular and a rectangular diagram, press the **Show Triangular** button. To toggle the X and Y axis variables, press the **Switch Axes** button.

File Name	Activity Examples
Ex_04_Fe-Cr-C_ternary_phase_diagram	Ternary Calculator, System Definer and Equilibrium Calculator

Example 5: Stable and the Metastable Fe-C Phase Diagrams

This example shows how to overlay results from two calculations in the same plot. In *Equilibrium Calculator 1* the stable phase diagram is calculated and in *Equilibrium Calculator 2* the metastable. The Plot Renderer activity is first created as a successor to Equilibrium Calculator 1. Then right-click the Plot Renderer node and select **Add predecessor** and the results from both equilibrium calculators are in the same plot. The metastable diagram is obtained by deselecting the graphite phase in the System Definer under the **Phases and phase constitution** tab.

File Name	Activity Example
Ex_05_Fe-C_stable_and_metastable_phase_diagram	Equilibrium Calculator

Example 6: Serially Coupled Equilibrium Calculators

This example shows how to use serially coupled Equilibrium Calculators for more complex equilibrium conditions. Sometimes there are multiple possible solutions for a given set of equilibrium conditions. In other cases the equilibrium calculation does not converge. You can then aid the final calculation by, in effect, telling the software where it should start the search for the equilibrium. Here are examples of two such cases.

- In the first example, the final objective is to calculate the equilibrium at 850◦C where the mole fractions of the FCC A1 and BCC A2 phases are 0.5 each for an Fe-Cr-C steel with 10% chromium. In Equilibrium Calculator 1 a simple set of equilibrium conditions (temperature, pressure, system size and composition) are used to find a carbon content where only the FCC A1 and BCC A2 phases are stable. The results from this preliminary calculation are displayed in Table Renderer 1.

- In Equilibrium Calculator 2 the carbon content equilibrium condition is replaced by the condition that the numbers of moles of the BCC A2 phase should be 0.5. The final result is displayed in Table Renderer 2. In this case, the final objective is to calculate the solidus temperature of an Fe-Cr-C steel with 10% Cr and 0.01% C.
- In Equilibrium Calculator 3 the state at 2000 K is calculated and the result is displayed in Table Renderer 3.
- In Equilibrium Calculator 4 the temperature equilibrium condition is replaced by the condition *Fix phase / liquid / 0.0*, meaning that liquid should be stable in an amount of zero moles, i.e. the solidus temperature. The final result is displayed in Table Renderer 4.

File Name	Activity Example
Ex_06_Serial_equilibrium_calculators	Equilibrium Calculator

Example 7: User-Defined Functions

This basic example shows the use of user-defined functions. A series of equilibria for an Fe-Cr-C alloy are calculated by varying temperature between 500 and 3000 K. In the configuration window of the **Equilibrium Calculator** under the **Functions** tab, two identically meaning functions are defined, *fraction solid* and *f solid*, the values of which are plotted against temperature in the Plot Renderers. Functions can be entered in terms of *QuantitiesQ1*, *Q2*, *Q3* etc., or, for using the Thermo-Calc syntax.

File Name	Activity Example
Ex_07_User_defined_functions	Equilibrium Calculator

Example 8: Scheil and Equilibrium Solidification

This example shows a comparison for an Al-Si alloy solidified under full local equilibrium and under the Scheil assumptions, i.e. zero diffusion in the solidified material and infinitely fast diffusion in the liquid. There are three Plot Renderers created: Scheil solidification, equilibrium solidification and the third Plot Renderer includes both in the same plot.

File Name	Activity Example
Ex_08_Scheil_and_equilibrium_solidification	Scheil Calculator

Example 9: Carbide Driving Force Heat Maps

This is an example of using *property grid* calculations to plot the driving force for a carbide as a function of two composition variables. With the property grid calculation type, a 2D grid is generated from the two calculation axes. After the calculation is done, an equilibrium is calculated in each grid point. A Plot

Renderer connected to a property grid calculator plots the *z*-axis property for each equilibrium as a function of the two calculation axes. The final plot can be either a heat map or a contour plot.

File Name	Activity Example
Ex_09_Heat_map_of_carbide_driving_force	Plot Renderer

Example 10: Phase Transition

The example uses the Property Model Calculator to predict the transition temperature to the unwanted brittle Sigma phase. The example shows how the temperature is influenced by changes to a steel alloy's composition using the uncertainty calculation type and how to create either a histogram or probability plot.

File Name	Activity Example
Ex_10_Phase_Transition	Property Model Calculator

Example 11: Coarsening and Interfacial Energy

The example uses the Property Model Calculator and both thermodynamic and kinetic demonstration steel databases. Using a grid calculation type it produces three plots: a heat map, a contour plot and a cross plot with contour (where both the interfacial energy and coarsening rate is shown).

File Name	Activity Example
Ex_11_Coarsening_and_Interfacial_energy	Property Model Calculator

Example 12: Driving Force and Interfacial Energy

The example uses the Property Model Calculator and a thermodynamic demonstration steel database. Using a grid calculation type it produces a contour plot comparing the driving force and interfacial energy.

File Name	Activity Example
Ex_12_Driving_force_and_Interfacial_energy	Property Model Calculator

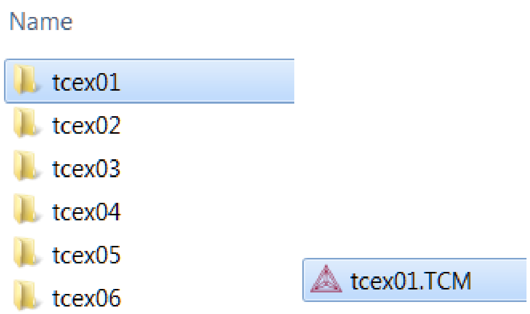
Console Mode Example Collection

After you have accessed the *Examples, Documentation and Materials* on page 7, go to the Console Mode Example folder to open one of the macro files in Thermo-Calc.

Opening a Macro File

These macro files (with the extension `*.TCM`) include comments, which you can either run in Thermo-Calc or open and read in a text editor. If you read the macro file in a text editor, you do not see the output that Thermo-Calc gives in response to the commands stored in the macro file.

Navigate to the file location, where each example is in its own folder.



In the folder, double-click the *.`TCM` file to open it. It launches a new session of Thermo-Calc and opens in Console Mode. If you have an instance of Thermo-Calc open in Console Mode, you can also drag and drop the file to run the macro. You can also open any text file editor and drag the file into it to see its contents.

Console Mode Example Descriptions

#	Example Description
1	Calculation of the binary Fe-C phase diagram.
2	Plotting of thermodynamic functions in unary, binary and ternary systems and working with partial derivatives and partial quantities
3	Calculation of an isothermal section using the TERNARY module.
4	Calculation of the miscibility gap in Fe-Cr
5	Calculation of a vertical section from Al to 30% Mg2Si
6	Calculation of an isopleth in low alloyed Fe-Mn-Si-Cr-Ni-C steel. This example calculates a multicomponent phase diagram using the define-material command in POLY and the TCFE steel database. The material contains 1.5 %Cr + 0.4 %Mn + 3.5 %Ni + 0.3 %Si and 1 %C (by weight). These conditions and the temperature are set by the command. Hidden commands set the pressure to 1 bar and so that iron is "the rest". After calculating the first equilibrium a phase diagram is calculated with one axis variable as temperature and the other as the carbon content.
7	Calculation of single equilibria in low alloyed Fe-Mn-Si-Cr-Ni-C steel. There are two general ways to perform a single equilibrium calculation: • Get data from a database, then in POLY use SET_CONDITION and COMPUTE_EQUILIBRIUM. • Go directly to POLY and use DEFINE_MATERIAL. The COMPUTE_TRANSITION command is also used to determine the temperature or

#	Example Description
	composition where one phase forms or disappears.
8	Calculation of property diagrams for a high speed steel i.e. phase fraction plots, activity vs temperature, etc.
9	Calculation of dew point.
10	The example calculates an equilibrium with suspended or dormant phases and shows how to avoid clogging of Cr2O3 in a continuous casting process. The origin of this example is that a manufacturer wanted to increase the Cr content of a material from 18 to 25 weight percent but the continuous casting of this material was problematic because solid Cr2O3 was formed. Using Thermo-Calc to calculate the equilibria in the steel/slag system, a simple correction was found: modify the Mn or Si content to decrease the oxygen potential. In Thermo-Calc, you can FIX a phase with zero amount to simulate how to avoid forming this phase. You then release one of the conditions, usually one of the compositions, and this composition is determined by the equilibrium calculation.
11	This example demonstrates the oxidation of Cu2S with H2O/O2 gas mixture. Thermo-Calc is used to find the optimum O/H ratio (i.e. oxygen potential) as certain oxygen potential values can desulphurize Cu2S without forming copper oxides. In Thermo-Calc, the problem reduces to perform equilibria calculations in a Cu-S-H-O system. The amounts of the components should be kept to correct ratio corresponding to Cu2S and H2O using a command SET_INPUT_AMOUNTS in POLY_3. Initially, O/H = 0.5 is given. Optimum O/H ratio is calculated with the desired calculation conditions. For example, to simulate one phase disappearing, you can FIX the phase with zero amount.
12	This example shows a number of independent cases using the TAB- module to tabulate thermodynamic data for reactions.
13	Binary calculation of Al-Ti phase diagram and G curve.
14	Calculation of heat and heat capacity variations during solidification of an Al-Mg-Si alloy.
15	Solidification simulation of a Cr-Ni alloy using the SCHEIL module. There is no back diffusion in the solid.
16	Calculation of the second order transition line in the Bcc field of the Al-Fe system.
17	Calculation of pseudo- or quasi-binary phase diagram in the CaO-SiO2 system. This database can be used both for pseudobinary systems like the one in this case, CaO-SiO2, or for full ternary systems like Ca-Fe-O.
18	Calculation of the A3 temperature of a steel and the influence of each alloying element on this temperature.

#	Example Description
	A3 temperature is the temperature where ferrite starts to form from austenite. You can easily read A3 from an Fe-C phase diagram. But for complex multicomponent steels, no simple diagram can be used. This example shows how to calculate the A3 temperature of a steel. Using POLY, it is easy to find out the influence of each alloying element on A3 temperature. This information is useful if you want to modify the compositions of a steel but keep A3 unchanged.
19A	Mapping of univariant equilibria with the liquid in Al-Cu-Si. Part A. Step-by-step calculation using POLY.
19B	Mapping of univariant equilibria with the liquid in Al-Cu-Si. Part B. Using the TERNARY module, you can get the information on invariant reactions, such as temperature and compositions.
20	Calculation of adiabatic decompression in a geological system using the geochemical database.
21	Calculate a ternary isotherm in Fe-Cr-Ni with a user-defined database.
22	Calculation of a heat balance. In this case C3H8 is burned in oxygen and the adiabatic flame temperature is calculated.
23	Calculation of a para-equilibrium and the T0 temperature in a low alloyed steel.
24	Simulation of the silicon arc furnace using the REACTOR module. This is a simple reactor model with output of gases at the top and output of condensed phases at the bottom. The gas phase from one segment flows to higher segments, 80 % reacts in the first above, 15% in the second above and 5 % in the third above. The condensed phases flow downwards and all of it goes to the next lowest segment. Heat can be added at any module. The only way to specify the initial state of the reactants added to the reactor is to specify their heat content.
25	Simulation of steel refining. Raw iron used to produce steel usually has high carbon and silicon content, which is why oxygen is blown into the furnace to burn off carbon. Lime (CaO) is added to form a slag with silica, and the slag can be removed. Alloying elements, such as Mn, Ni, Cr and V are added to produce the desired steel. Since the reaction between O and C increases the temperature, scrap iron is added in order to keep the temperature constant (it is assumed the furnace is isolated and no heat is lost to the environment). This is a typical steel refining process. This example simulates blowing oxygen into a liquid steel of one metric ton (1e6 grams) with 4 w/o C, 2 w/o Si and 1 w/o Mn. 100 moles of CaO (equivalent to 5.6 kg) is added. Keeping the enthalpy constant is the way to simulate the isolation of the furnace. The oxygen reacts with carbon and increases the temperature. After blowing a certain amount of oxygen, scrap iron is added to keep the temperature constant.

#	Example Description
26	Example of plotting the partial pressures of a gas species along the solubility lines in the As-Ga phase diagram. The calculation makes it possible to monitor the input gases to a process of depositing solid AsGa.
27	CVD calculations
28	Example showing calculation of PRE (Pitting Resistance Equivalence) for a duplex stainless steel.
29	Calculation of speciation of a gas
30A	Scheil solidification simulation for Al-4Mg-2Si-2Cu alloy. Part A. Step-by-step calculation using POLY and the STEP_WITH_OPTION command, EVALUATE option.
30B	Scheil solidification simulation for Al-4Mg-2Si-2Cu alloy. Part B. Simulation performed using SCHEIL module.
31	Calculations of CVM and comparisons with sublattices of a fictitious A B system. Also shows how to overlay diagrams from two calculations.
32	Calculation of oxide layers on steel. Also shows how to append databases.
33	Benchmark calculation - An isopleth in the Fe-Cr-C system.
34	Calculation of the phase diagram and G curves in the Al-Zn system. An example of using the BINARY module.
35	Calculation of potential diagram
36	Assessment - The use of the PARROT module The example is divided into four parts corresponding to the files in the folder. The problem is presented with "selected" experimental data (see the *tcex.36.readme* file). The example uses a fictitious binary A-B system where element A is BCC until it melts. Element B is BCC up to 1100 K and FCC above. There is a compound A2B stable in a limited temperature range. • The creation of the "setup" file is described (*tcex36a.TCM*) • The creation of the experimental data file is described (*tcex 36b.TCM*) • The actual run in the PARROT module in order to obtain the result (*tcex 36cp-d.TCM*)
37	Calculation of an isothermal section using command lines.
38	Calculation of the Morral “rose” (miscibility gaps).

#	Example Description
39	This example shows how Thermo-Calc calculates the reversible Carnot cycle of a heat engine using one mole of an ideal gas with two fictitious species A and A2. The Reversible Carnot cycles are usually drawn schematically. Using Thermo-Calc and realistic data it is possible to calculate a reversible Carnot cycle of a heat engine and to relate it to different thermodynamic quantities. One application of the Second Law is to the efficiencies of heat engines, pumps and refrigerators. Whenever there is a difference of temperature, work can be produced - the principle of heat engines. The Gibbs energy also enables the prediction of the maximum work that a process may achieve.
40	A series of examples designed to demonstrate the advanced POURBAIX module calculations and graphical processing. Also see the text file the Extended folder (*TCEX40-README.txt*). The first example automatically calculates and plots a Pourbaix diagram for 0.001 m Fe in a 0.1 m NaCl aqueous solution at 25C and 1 bar. A so-called Pourbaix diagram is actually a phase diagram with independently-varied acidity (pH) and electropotential (Eh), for a heterogeneous interaction system at a certain bulk composition (that is by default always set as 1 kg of water dissolving a specified amount of metals and other solutes), under a defined temperature and pressure condition. For details about the other examples, see *Pourbaix Examples TCEX40, 40A to 40E, and TCEX53* on page 168.
41	Calculation of a solubility product. Shows the STEP_WITH_OPTION command, T-ZERO option.
42	Formation of para-pearlite (isopleth calculation) Fe-Mn-C system at 2.5%Mn Mass u-fraction (an example of a paraequilibrium calculation).
43	Formation of para-pearlite (calculation of isothermal section) Fe-Mn-C system at 700 C (an example of a paraequilibrium calculation).
44	This example uses variables and functions to predict properties e.g. proof strength for an austenitic stainless steel at elevated temperatures (20-550C).
45	3D-diagram that calculates the gamma volume in the Fe-Cr-C system. To view the generated file, *tcex45.wrl*, install a WRML (Wirtual Reality Modelling Language) viewer to the web browser. WRML viewers can be downloaded from various sites, e.g. www.parallelgraphics.com or www.sim.no.
46	3D-diagram that calculates the liquidus surface of the Fe-Cr-C system. To view the generated file, *tcex46_tri.wrl* and *tcex46_sqrt*, install a WRML (Wirtual Reality Modelling Language) viewer to the web browser. WRML viewers can be downloaded from various sites, e.g. www.parallelgraphics.com or www.sim.no..

#	Example Description
47	3D-quaternary diagram that calculates the gamma volume in the Fe-Cr-V-C system at 1373K. To view the generated file, *tcex47.wrl*, install a WRML (Wirtual Reality Modelling Language) viewer to the web browser. WRML viewers can be downloaded from various sites, e.g. www.parallelgraphics.com or www.sim.no..
48	This is an example of Scheil solidification with C back diffusion in solid phases and compares it with simple Scheil and equilibrium calculations.
49	This example shows how to enter parameters for a FACT quasichemical liquid model and how to calculate the sulfur activity using the GIBBS_ENERGY_SYSTEM (GES) module commands.
50	This example shows the quasichemical model using TDB to replicate figures 3 and 4 from Kongoli, F., Pelton, A. D. & Dessureault, Y. Thermodynamic modeling of liquid Fe-Ni-Cu-Co-S mattes. *Metall. Mater. Trans. B* 29, 591–601 (1998).
51	Calculation of molar volume, thermal expansivity and density.
52	This example shows how to change the excess models for binary/ternary interactions in a solution phase, either through direct interactive amendments of phase descriptions within the GIBBS_ENERGY_SYSTEM (GES) module, or enforced by specific type-definitions given in a database file retrieved by the TDB module. For binary excess model: from the default R-K model to Mixed-Excess-Model (Note the phase has to be a substitutional phase) For Ternary Extrapolation Model: from the default R-K-M model to Toop_Kohler model.
53	Pourbaix Diagram Calculations through the TDB-GES-POLY-POST routine. Using PAQ2 or PAQS2 database; for the Fe-X-H2O-NaCl heterogeneous interaction systems (X = Cr-Ni-Co).

The Thermo-Calc Modules and Activities

Although many calculations can be done in either Graphical or Console Mode, only data optimization and thermodynamic or kinetic assessments are available in Console Mode. Conversely, only the Property Model Calculator is available in Graphical Mode.

For the Console Mode only commands see *The Console Mode Interface* on page 111 and the *Thermo-Calc Console Mode Command Reference* included with this documentation set.

Modules and Activities briefly describes the available modules. The specific modules, using the names in the table, are accessed through Console Mode. For Graphical Mode, the underlying principles, and the final calculations, are incorporated into the projects and activities. Some modules require additional licenses.

Modules and Activities

The first and second columns are the Console Mode module names. The third column indicates whether there is an equivalent activity available in Graphical mode.

Full Module Name in Console Mode	Short Name	Graphical Mode Activity	Primary Functions
BINARY_ DIAGRAM_EASY	BIN	*Binary Calculator* on page 104	Calculate binary phase diagrams. Access to specific databases designed for BIN, such as TCBIN, is required.
DATABASE_ RETRIEVAL	DATA	*System Definer* on page 54	Select appropriate databases, define chemical systems and retrieve the data required for calculations.
DICTRA_ MONITOR	DICTRA	Console Mode only	Run simulations of diffusion controlled transformations in Thermo-Calc using the Diffusion Module (DICTRA). An additional license is required for this feature.
DIC_PARROT	DIC_PARROT	Console Mode only	This is the data optimization module for the Diffusion Module (DICTRA). It allows you to assess experimental data and use such data to optimize calculations. An additional license is required for this feature.
EDIT_ EXPERIMENTS	ED_EXP	Console Mode only	Part of the PARROT submodule, use it to edit experimental data points to make reliable optimizations.
GIBBS_ENERGY_ SYSTEM	GIBBS	Console Mode only	Handles thermodynamic models and quantities. Generally you do not need to use this module, but you can get information about models, and list, add or amend the thermodynamic parameters in use.
PARROT	PARROT	Console Mode only	A data optimization module to assess experimental data and use the data to optimize calculations.
not available	-	*Precipitation Calculator* on page 78	Set the conditions for, and perform, a precipitation calculation.

Full Module Name in Console Mode	Short Name	Graphical Mode Activity	Primary Functions
not available	-	*Property Model Calculator* on page 60	You can evaluate models simultaneously over a range of compositions and cross plot the results. You can then conduct an uncertainty analysis and plot the results as either a histogram or as a probability plot.
POLY-3	POLY	*Equilibrium Calculator* on page 72	Equilibrium calculation module. Specify conditions and make calculations.
POST_PROCESSOR	POST	*Plot Renderer* on page 83	For post-processing and plotting of calculation results. Modify property and phase diagrams, add labels, change the diagram colours, etc. Submodules to POLY-3 and DICTRA module for example.
POTENTIAL_DIAGRAM	POTENTIAL	*Ternary Calculator* on page 106	Calculate and plot gas potential diagrams in a ternary system. The potential of two gas species are used as the diagram axes.
POURBAIX_DIAGRAM	POURBAIX	Console Mode only	Calculate and plot pH-Eh diagrams (Pourbaix diagrams) and property diagrams for heterogeneous interaction systems involving aqueous solutions.
REACTOR_SIMULATOR_3	REACTOR	Console Mode only	Simulate chemical reaction processes in several feed-forward steady-state stages or in several dynamic stages.
SCHEIL_SIMULATION	SCHEIL	*Scheil Calculator* on page 108	Simulate and plot Scheil-Gulliver solidification processes with no diffusion in the solid phases and with/without consideration of back diffusion of interstitial components (such as C, N, O, S).
SYSTEM_UTILITIES	SYS	Console Mode only	In Console Mode, interact with the operating system and change environmental settings. Create and execute MACRO files.
TABULATE_REACTION	TAB	Console Mode only	Tabulate properties of chemical reactions and substances (stoichiometric or solution phases).

Full Module Name in Console Mode	Short Name	Graphical Mode Activity	Primary Functions
TERNARY_ DIAGRAM	TERN	*Ternary Calculator* on page 106	Calculate ternary phase diagrams.

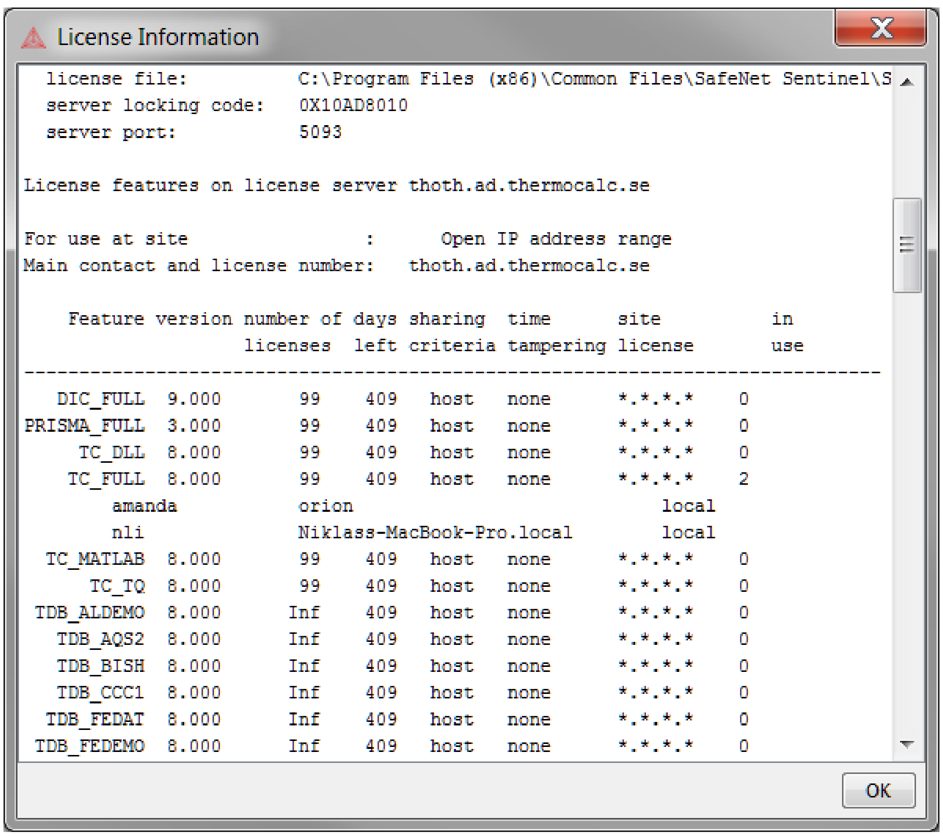
Displaying Thermo-Calc License Information

You can start (and install) the software without a valid Thermo-Calc license but you cannot do any calculations. To show information about the available and installed licenses, from the main menu select **Help → Show License Info**.

Also see the *Thermo-Calc Installation Guide*.

License Information Window Example

This is an example of part of a License Information window for a network installation.



```
License Information
  license file:         C:\Program Files (x86)\Common Files\SafeNet Sentinel\S
  server locking code:  0X10AD8010
  server port:          5093

License features on license server thoth.ad.thermocalc.se

For use at site                 :     Open IP address range
Main contact and license number:   thoth.ad.thermocalc.se

    Feature version number of days sharing  time      site          in
                    licenses  left criteria tampering license       use
-------------------------------------------------------------------------------
   DIC_FULL  9.000       99     409   host   none      *.*.*.*    0
PRISMA_FULL  3.000       99     409   host   none      *.*.*.*    0
     TC_DLL  8.000       99     409   host   none      *.*.*.*    0
    TC_FULL  8.000       99     409   host   none      *.*.*.*    2
        amanda          orion                               local
        nli             Niklass-MacBook-Pro.local           local
  TC_MATLAB  8.000       99     409   host   none      *.*.*.*    0
      TC_TQ  8.000       99     409   host   none      *.*.*.*    0
 TDB_ALDEMO  8.000      Inf     409   host   none      *.*.*.*    0
   TDB_AQS2  8.000      Inf     409   host   none      *.*.*.*    0
   TDB_BISH  8.000      Inf     409   host   none      *.*.*.*    0
   TDB_CCC1  8.000      Inf     409   host   none      *.*.*.*    0
  TDB_FEDAT  8.000      Inf     409   host   none      *.*.*.*    0
 TDB_FEDEMO  8.000      Inf     409   host   none      *.*.*.*    0
OK
```

Under each license type with instances checked out, there is a list of who checked out each license. In Graphical Mode, an instance is checked out when an Equilibrium Calculator activity is created, and checked back in when that Equilibrium Calculator is removed. In Console Mode, an instance is checked out when you enter the POLY module and is checked back in when you exit the POLY module. In the example, two users, amanda and nli, have licenses checked out.

- The **Features** column is a list of the specific software, databases and API licenses purchases.
- The **Version** is an internal version number for the feature.
- The **number of licenses** column is how many instances of each license type available to be checked out simultaneously.
- The **in use** column shows how many license instances are checked out.

Console Mode License Restrictions

- If you start Thermo-Calc without a valid Thermo-Calc license you cannot leave the SYS module.
- If all the licenses are checked out for a network client installation of Thermo-Calc, you may not be able to enter the POLY or DICTRA modules until a license is available.
- To enter the DICTRA and the DIC_PARROT modules, you need a valid Diffusion Module (DICTRA) license key.
- If you enter and use either the DIC_PARROT module or the PARROT module, then you cannot enter the other module in the same session. You must close down Thermo-Calc and launch the program again to enter the other data optimization module.

Network Computer License Restriction

If you have a network computer installation of Thermo-Calc, then you may not be able to do any calculations even if you have access to a valid network license file. This is because others who are part of your network installation may have checked out all allowed instances of the software.

If you are running a network installation of Thermo-Calc, you can see how many instances of the client licenses are currently checked out and how many of them are left. You can also see which client computers have checked out the licenses.

Using This Guide

If you are a new user, Graphical Mode is a good way to learn how to work with Thermo-Calc. Console Mode has more functionality but until you are comfortable with the concepts, it is recommended you start in Graphical Mode.

Additional documentation and training material is described in *Help Resources* on page 5.

Graphical Mode

The Graphical Mode Interface on page 26 introduces you to the GUI layout and the workflow. In *Graphical Mode Projects* on page 42, it goes into more detail about how to set up and run your projects (with the Quick Start Wizard), using templates, scheduling jobs and creating activities. *Graphical Mode Activities* on page 53 specifically describes how to set up each type of available activity in the tree structure.

For both Graphical and Console mode, *Changing Global Settings* on page 187 gives information about adjusting global user settings.

Console Mode

Console Mode has more functionality than Graphical Mode but the underlying concepts are the same. There are, however, command lines and extensive details applicable to this mode. This guide is only the beginning of the learning objectives. In particular, refer to the *Thermo-Calc Console Mode Command Reference* for details about all the commands available.

In this guide it is a basic introduction to the layout of the Command windows, as described in *The Console Mode Interface* on page 111.

Several topics describe and give an example of the calculation types:

- *Equilibrium Calculations* on page 130
- *Property Diagrams* on page 136
- *Phase Diagrams* on page 140
- *Scheil Simulations* on page 144
- *T0 Temperature Simulations* on page 150
- *Paraequilibrium* on page 153
- *Potential Diagrams* on page 157
- *Aqueous Solutions* on page 161
- *Console Mode TAB Module* on page 174

Then in *Console Mode POST Module* on page 179 there is an overview of plotting.

For both Graphical and Console mode, *Changing Global Settings* on page 187 gives information about adjusting global user settings.

The Graphical Mode Interface

These topics introduce you to the concepts and terminology of Graphical Mode and describe the basic workflow.

In this section:

The GUI Layout

Open Thermo-Calc to view the default GUI layout for Graphical Mode. You can easily reorganize the windows.

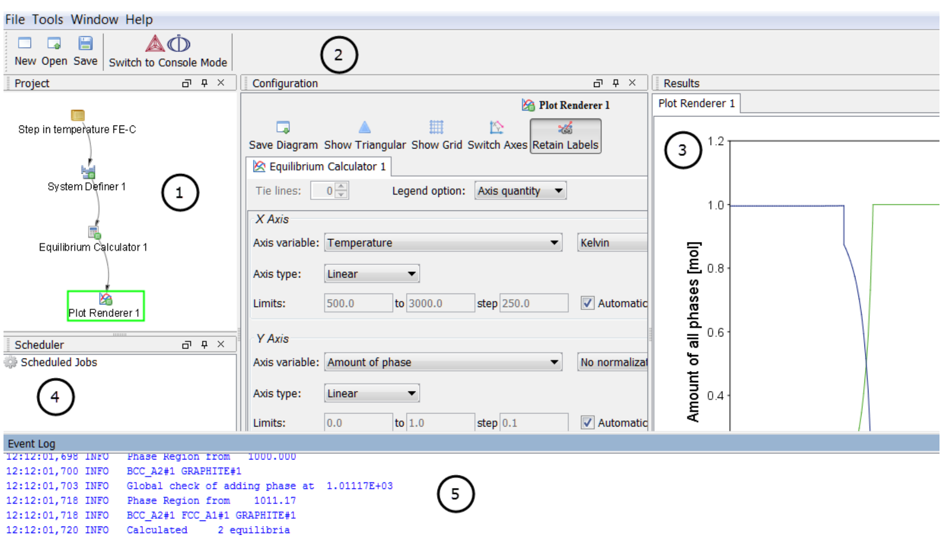
Also see *Moving Windows* on page 38.

The windows are:

1. **Project**: Create, manipulate and navigate between the activities that make up a project.
2. **Configuration**: Shows the settings that can be configured for the currently selected activity.
3. **Results**: Shows the results of a calculation, either plotted as a diagram or displayed in table format. In this example, there is a Plot Renderer *tab* in the Results window.
4. **Scheduler**: Displays information about jobs, such as calculations, that are being performed or are scheduled to be performed. You can cancel scheduled jobs and if a job has failed, then you can view information about the error.
5. **Event Log**: By default this window is closed but it displays during calculation processes. You can always open it to view progress of calculations and to troubleshoot.

Graphical Mode Windows

In this example, the Results window also has a Plot Renderer tab. You can also have a Table Renderer tab.



Project Activities and the Tree Structure

In a *project*, a set of linked activities is called a *tree*. A result calculated within a tree is fed as input into the next *activity* in the tree. Consequently, if you have an Equilibrium Calculator with another Equilibrium Calculator as its *successor*, the successor takes the calculation results of the *predecessor* as the starting values for its calculation. When working with activities in the Project window these are also referred to as *nodes* (or *activity nodes*) in the tree structure.

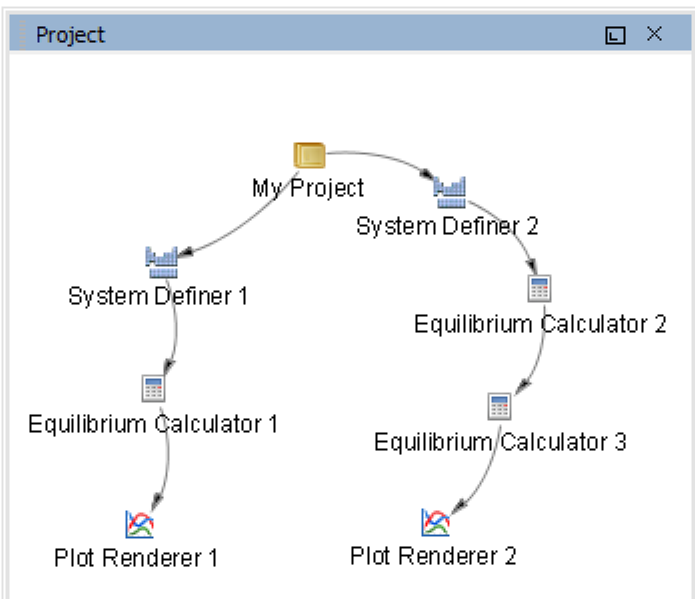
Also see *Activity Successor and Predecessor Nodes* below for definitions.

The **Project Tree Example** shows a My Project node with two trees. In each tree, settings and calculation results are propagated downward until the calculation and the visualisation of the results are completed in the Plot Renderer nodes.

The system definitions, settings and calculation results of the two trees are independent of each other.

Project Tree Example

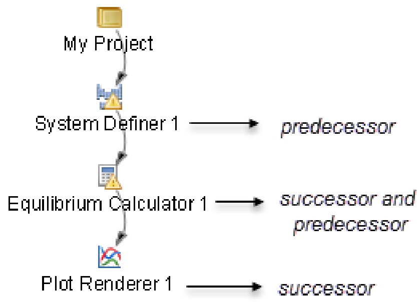
Project window with a My Project node. There are two trees with successor and predecessor activity nodes.



Activity Successor and Predecessor Nodes

An activity node located below another activity node in a tree is referred to as that activity's *successor*. An activity located above another activity is called that activity's *predecessor*. A predecessor is performed before the predecessor's successors and its result is fed forward to any successor activities.

Example of the Tree Structure



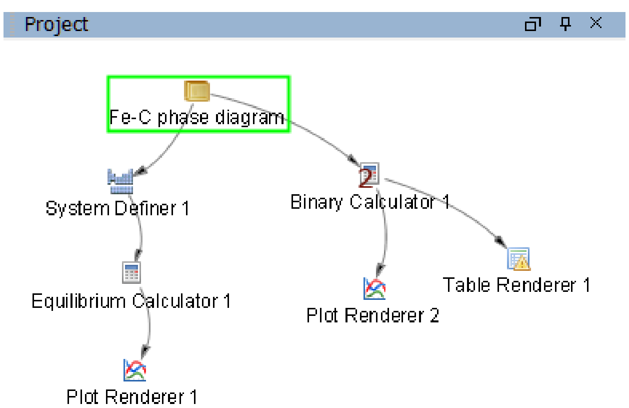
For example, to calculate and display a phase diagram, create a branch with three linked activities: A System Definer activity linked to an Equilibrium Calculator activity, which in turn is connected to a Plot Renderer activity.

See *Successor and Predecessor Example* below, which shows that Binary Calculator 1 is a predecessor to the successors, Plot Renderer 2 and Table Render 1.

You can determine the available successors by right-clicking a node in the Project window and choosing options from the menus.

Also see *Creating Activities and Successors* below.

Successor and Predecessor Example



Creating Activities and Successors

You can create a new activity in these ways:

- In the **Project** window, right-click an activity and choose **Create New Activity** (**My Project** node only) or **Create New Successor** (all other nodes).
- Select the activity you want to create from the submenu.

- At the bottom of the **Configuration** window, click **Create New Activity** or **Create New Successor** and select the activity to create.

Predecessors and Successors by Activity

Activity	Possible Predecessors	Possible Successors
My Project	None; this is the first node of the tree structure	System Definer, Binary Calculator, Ternary Calculator, Experimental File Reader. Also choose templates from this level.
System Definer on page 54	My Project, System Definer	Property Model Calculator, Equilibrium Calculator, Precipitation Calculator, Scheil Calculator
Property Model Calculator on page 60	System Definer	Plot Renderer
Equilibrium Calculator on page 72	System Definer, Equilibrium Calculator	Equilibrium Calculator, Plot Renderer, Table Renderer
Precipitation Calculator on page 78	System Definer	Plot Renderer, Table Renderer
Plot Renderer on page 83	Property Model Calculator, Equilibrium Calculator, Precipitation Calculator, Binary Calculator, Ternary Calculator, Scheil Calculator, Experimental File Reader	None
Table Renderer on page 101	Equilibrium Calculator, Precipitation Calculator, Binary Calculator, Scheil Calculator	None
Experimental File Reader on page 104	My Project	Plot Renderer
Binary Calculator on page 104	My Project	Plot Renderer, Table Renderer
Ternary Calculator on page 106	My Project	Plot Renderer
Scheil Calculator on page 108	System Definer	Plot Renderer, Table Renderer

Graphical Mode Workflow

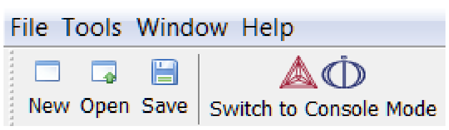
You can set up a tree in the Project window (starting with the My Project node) and then perform all the activities at once, or create and perform one activity at a time.

The typical workflow is to:

1. *Define a System*: Create a *System Definer* on page 54 activity (in the **Project** window) where you select a database and the elements to have as system components (in the **Configuration** window).
2. *Set up and run a calculation*: Create an *Equilibrium Calculator* on page 72 activity (a successor to the **System Definer**), where calculation conditions are set (temperature, pressure, etc.). This is where axis variables are set to create a property or phase diagram. These settings are also made in the **Configuration** window. You can also create a *Property Model Calculator* on page 60, *Precipitation Calculator* on page 78 or *Scheil Calculator* on page 108.
3. *Visualize the results*: Create either a *Plot Renderer* on page 83 or a *Table Renderer* on page 101 activity in the **Project** window. When calculated, this creates a plot or table in the **Results** window. The results are shown in a **Plot Renderer** or a **Table Renderer** tab.

You need to Perform a Plot Renderer or a Table Renderer activity to visualize the calculation results from an Equilibrium Calculator.

Main Menu and Toolbar



Graphical Mode Menu and Toolbar Options lists the main menu and toolbar options. The **File** menu and toolbar are reserved for Graphical Mode but in Console Mode the **Tools**, **Window** and **Help** menus have the same options.

In Console Mode you use commands to access the other options instead. These are discussed in *The Console Mode Interface* on page 111.

Graphical Mode Menu and Toolbar Options

Option	Description	Action (s)
New	Create a new project.	Click **New** on the toolbar Select **File → New Project** Press <Ctrl+N>
Open	Open an existing project. See *Saving and Opening Thermo-Calc Project Files* on page 34.	Click **Open** on the toolbar Select **File → Open Project**

Option	**Description**	**Action (s)**
		Press <Ctrl+O>.
Save	Save a project. See *Saving and Opening Thermo-Calc Project Files* on the next page.	Click **Save** on the toolbar Select **File → Save Project** Press <Ctrl+S>.
Switch to Console Mode	Launch Console Mode and the command prompt.	Click the button to exit Graphical Mode and open Console Mode.
Switch to Graphical Mode	Launch Graphical Mode to work with the GUI version.	Click the button to exit Console Mode and open Graphical Mode.
Append Project	Combine two projects into one project file. See *Saving and Opening Thermo-Calc Project Files* on the next page.	Select **File → Append Project**
Save Project As...	Save an existing project with a new name.	Select **File → Save Project As...**
Exit	Exit the program.	In the upper right-corner of the program, click the X. Select **File → Exit** Press <Ctrl+Q>.
Options	Change the global defaults for a variety of settings. See *Changing Global Settings* on page 187.	Select **Tools → Options**
Database Checker	Open the Database Checker, a program to check that the syntax of Thermo-Calc database files is correct. This is for advanced users who develop and manage databases.	Select **Tools → Database Checker**
Window menu	Highlight a specific window on the GUI	From the **Window** menu, select an option to refocus on that specific window.
Online Help	Open the online help system where you can browse or search all the Thermo-Calc documentation.	Select **Help → Online Help**
Show License Info	Open the License Information window.	Select **Help → Show License Info**

Option	Description	Action (s)
	Also see *Displaying Thermo-Calc License Information* on page 23 and in the *Thermo-Calc Installation Guide* .	
Check for update	Check the Thermo-Calc Software website for updates to the software.	Select **Help → Check for update**

Saving and Opening Thermo-Calc Project Files

Thermo-Calc Graphical Mode uses project files with the file name suffix `.TCU`.

To save your project and all its settings and results, on the toolbar click **Save**. Select to **Include calculated results in the project file** (the default) as required.

To open a Thermo-Calc project file, on the toolbar click **Open** and select the project file in the **Open file** window.

You can only have one project file open at a time. However, you can attach (*append*) the trees from additional project files to the topmost My Project node. To append an additional project file in this way, from the main menu select **File → Append Project** and open a project file.

Also see *Saving Plots* on page 95 and *Saving Tabulated Data* for the *Table Renderer* on page 101.

Project Activity Nodes

Project Activity Node Descriptions is a brief description of the project activity types. When working with activities in the Project window these are also referred to as nodes.

Project Activity Node Descriptions

These are the project activity nodes available in Graphical Mode.

Activity type	**Description**
System Definer on page 54	Define a certain thermodynamic system and read it from file into memory.
Property Model Calculator on page 60	Choose one or more general models to predict and optimize material properties. You can evaluate models simultaneously over a range of compositions and cross plot the results. You can then conduct an uncertainty analysis and plot the results as either a histogram or as a probability plot.
Equilibrium Calculator on page 72	Set thermodynamic conditions and define axis variables when a series of equilibrium calculations are to be performed in one or more dimensions.
Precipitation Calculator on page 78	Set the conditions for, and perform, a precipitation calculation.
Plot Renderer on page 83	Determine the layout of non-text based output.
Table Renderer on page 101	Use it for text-based output.
Binary Calculator on page 104	Use it for some calculations involving two components only. It is like a combination of System Definer and Equilibrium Calculator activities with adaptations to simplify the configuration of calculations on binary systems. To perform this activity, you need a database designed for the Binary Calculator, such as the TCBIN database.
Ternary Calculator on page 106	Use it for some calculations involving three components. It is like a combination of System Definer and Equilibrium Calculator activities with adaptations to simplify the configuration of calculations on ternary systems. To get a reliable result you need a database that fully assesses binary and ternary systems.
Scheil Calculator on page 108	Perform Scheil-Gulliver calculations (also known as Scheil calculations). A default Scheil calculation is used to estimate the solidification range of an alloy assuming that i) the liquid phase is homogeneous at all times and ii) the diffusivity is zero in the solid. However, it is possible to disregard the second assumption for selected components.
Experimental File Reader on page 104	Read experimental data files (*.EXP). This type of file contains information specifying a plotted diagram, written in the DATAPLOT graphical language.

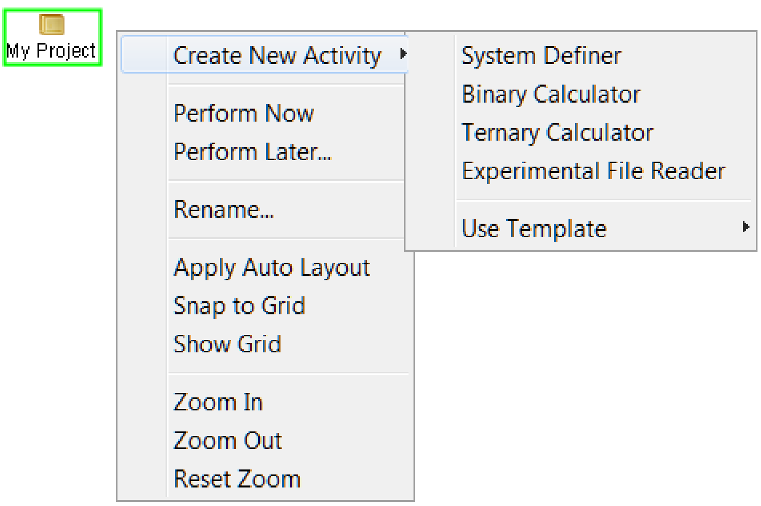
Activity type	Description
	Also see the *DATPLOT User Guide*.

Project and Activity Menus

When working in the Project window you can right-click nodes to open menus with the available actions. For example, click the **My Project** node to open the menus.

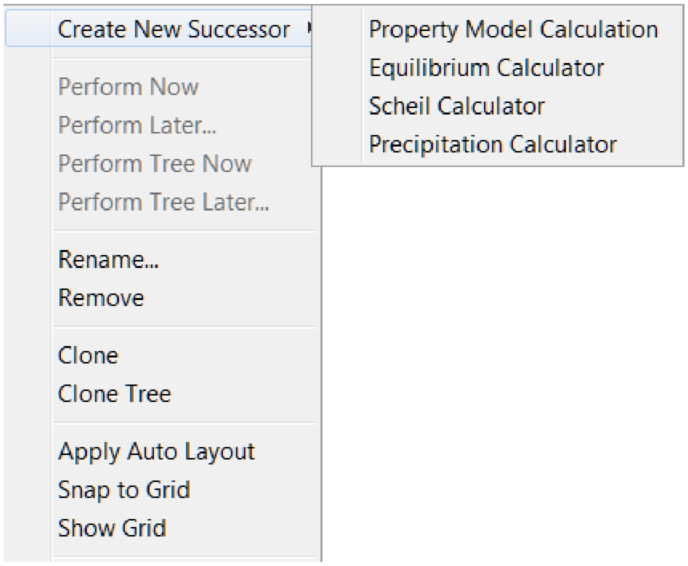
My Project Node

Menu options available for My Project node.



Activity Node Menu Example

Most nodes in the tree have a common menu with differences due to placement in the tree and the activity type.



Menu Options for Activities

Option	Description	Action (s)
Create New Activity	Add System Definer, Binary Calculator, Ternary Calculator, and Experimental File Reader nodes. See *Creating Activities and Successors* on page 29.	Right-click **My Project** and from the **Create New Activity** menu choose an option.
Create New Activity → Use Template	Choose from the available templates to use the Wizard or add predefined nodes. See *Creating a Project from a Template* on page 46.	Right-click **My Project** and from the **Create New Activity→ Use Templates** menu choose an option. The templates are also available on the **Configuration** window, which you can access by clicking the **My Project** node, or from the **Window** menu.
Create New Successor	Add a successor to the selected node. See *Creating Activities and Successors* on page 29.	Right-click one of these nodes **System Definer**, **Property Model Calculator**, **Equilibrium Calculator**, **PRISM Calculator**, **Binary Calculator**, **Ternary Calculator** or **Experimental File Reader**, and from the **Create New Successor** menu choose an option.

Option	Description	Action (s)
Add a Predecessor	Add a predecessor to the selected node.	Right-click a node to add a predecessor to it (when available).
Perform Now or Perform Later	Perform a calculation or create a plot or table immediately, or schedule it for a time in the future. See *Using the Scheduler* on page 49.	Right-click any node.
Rename	Change the name of the node.	Right-click any node.
Remove	Delete the selected node from the tree.	Right-click any node except My Project.
Clone	Duplicate the selected node and add it to the same tree. See *Cloning Activities and Trees* on page 47.	Right-click any node except My Project.
Clone tree	Duplicate a tree and add it to My project.	Right-click any node except My Project.
Apply Auto Layout	Apply an automatic layout to the nodes in the tree.	Right-click any node.
Snap to Grid	Snap a node to align it with the grid. See *Moving Nodes in the Tree and Using the Grid* on the next page.	Right-click any node.
Show Grid	Turn the grid on or off in the Project window.	Right-click any node.
Zoom In, Zoom Out, Reset Zoom	Zoom in and out of the Project window or reset the zoom to the default.	Right-click any node.

Moving Windows

The windows in Thermo-Calc can be fixed or free-floating. When fixed, it can also be set to auto-hide, where the window is minimized if you select another window. The minimized window is then shown either along the bottom or side of the Thermo-Calc window. A free-floating window is shown on top of the other windows and can be moved outside the frame of the Thermo-Calc window.

By default, the windows are fixed and open, except the **Event Log** window, which is set to auto-hide by default.

To return to the standard desktop layout, from the main menu, select **Windows → Reset Windows**.

Floating, Hiding and Minimizing Windows in the GUI

For each window you can rearrange, resize, minimize and close it. At the top right corner are the buttons listed in the table.

Button(s)	Description
	Click to toggle on/off free-floating
	Click to toggle on/off auto-hide.
	Click to minimize an open window that has auto-hide turned on. The window automatically minimizes if another window is selected.
	Click to close the window. You can open the window again from the Window main menu.

Moving Nodes in the Tree and Using the Grid

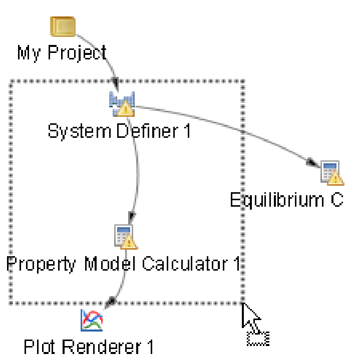
In the Project window, you can work with the activity nodes in these ways.

Select and Move Individual Nodes

- Click an activity node to *select* it.
- To move activities, click and hold the activity node and move the cursor.

Select Several Nodes at a Time

- To select several activity nodes, hold down <Ctrl> while clicking on each node you want to select.
- Click and hold the mouse button and draw a square around the activity nodes to select. Both the name and the icon that represents an activity must be within the square.



Zoom In and Out

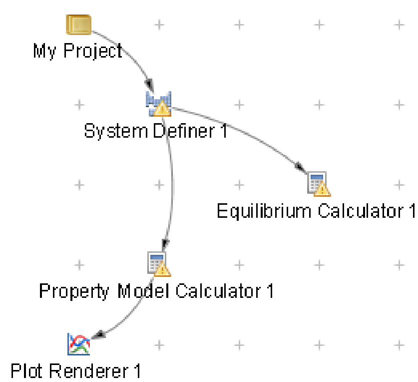
- Zoom in and out with the mouse scroll wheel.
- In the **Project** window right-click and select **Zoom In** or **Zoom Out**. To go back to the default zoom, select **Reset Zoom**.

Show Grid and Snap to Grid

To show a grid of light-grey cross-hairs overlaid on a project, click **Show Grid** in the toolbar. Click the button again to turn the grid off. Or right-click in the **Project** window and select **Show Grid**.

Click **Snap to Grid** in the toolbar or right-click in the **Project** window and select **Snap to Grid** to automatically position each activity in the Project window at a cross-hairs point in the grid.

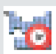
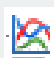
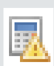
The Project window with the grid turned on and activity nodes moved from the default location. These nodes are snapped to the grid.



Node Status Markers

Status markers indicate whether an activity is ready to be performed, is being performed, or has been performed. The markers are overlaid on the activity icons in the **Project** window. The status markers are displayed on a System Definer icon.

Node Status Markers for the Project and Scheduler Windows

Window	Status marker	Description
Project window		No status marker means the activity is ready to be performed (it has not been performed).
		A yellow triangle means the activity cannot be run. The necessary configurations for the activity may not be set or other prerequisites are not met.
		A green disc means the activity is performed.
		A red circle with a dial (a clock face) means the activity is currently being performed.
Scheduler window		No status marker means the activity (the job) is scheduled to be performed (the job has not started).
		A yellow triangle means there an error happened during the activity job.
		A green disc means the activity (the job) performed successfully.
		A green right-facing arrow means the activity (the job) is being performed.

Graphical Mode Projects

These topics describes how you create, perform and link activities.

In this section:

The Quick Start Wizard

Even before you understand how to work with Thermo-Calc projects, the Quick Start wizard helps you set up a project and run calculations in Graphical Mode. You can set up, calculate and visualize any of the following:

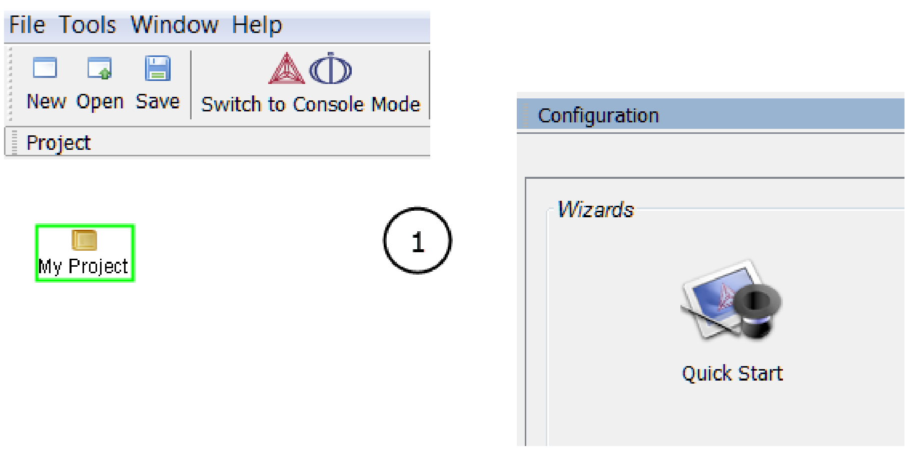
- Single point equilibrium
- Property diagram
- Phase diagram
- Scheil solidification simulation

See *Creating a Project with the Quick Start Wizard* below

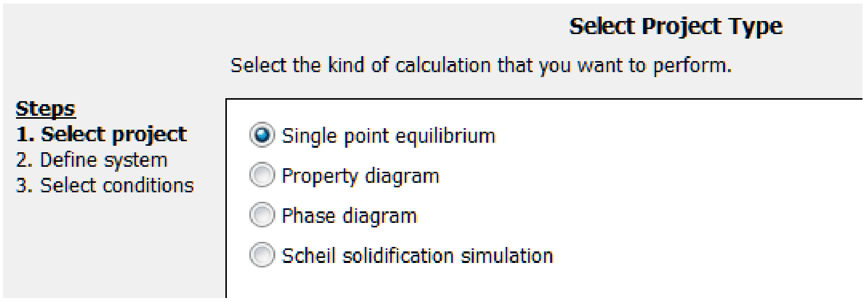
Creating a Project with the Quick Start Wizard

To set up, perform and visualize a calculation using the Quick Start wizard:

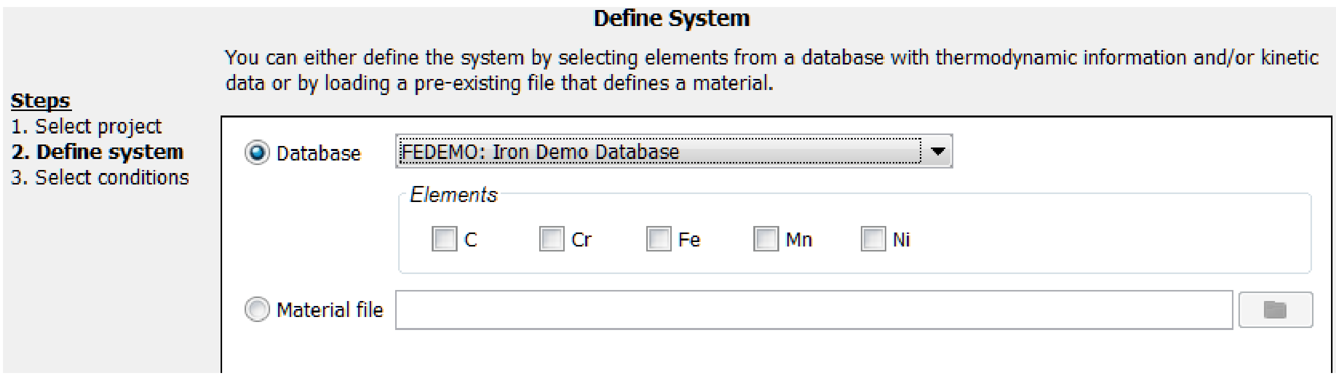
1. In the **Configuration** window click **Quick Start**. If you cannot see the **Quick Start** button, first click **My Project** in the **Project** window.



2. In the **Select Project Type** window click to select the type of calculation to perform. **Single point equilibrium**, **Property diagram**, **Phase diagram** or **Scheil solidification simulation**. Click **Next**.



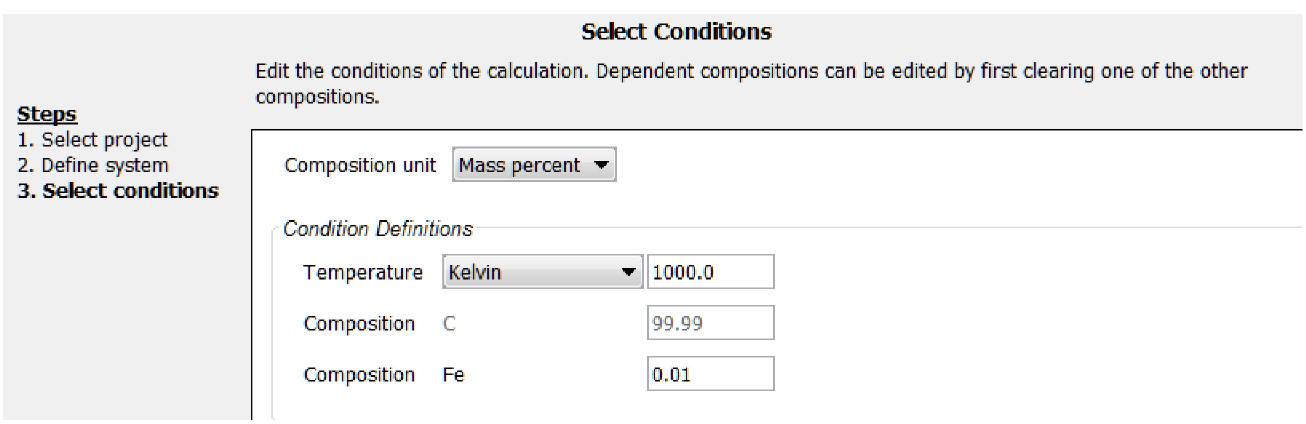
3. In the **Define Systems** window, choose an available **Database**.



4. Click to select **Elements** check boxes to include as system components. To add elements from an existing material file, click the **Material file** button to enter the file path to a material file or click the **Select Material File** button (to the right of the field) to navigate to a file on the computer.
5. Click **Next**.

The **Select Conditions** window has different options depending on the Project type and the elements selected.

In this example figure, using a **Single Point Equilibrium**, choose the **Composition unit** and under **Condition Definitions** specify the values of the state variables in the system.



For a **Property diagram** or **Phase diagram**, specify the **Axis Definitions** of the stepping or mapping operation. Choose minimum and maximum values for the variable(s) (select **Linear – min no. of steps**, **Linear – max step size** or **Logarithmic –scale factors**) and whether it is **Normal** or **Separate phases**.

For a **Scheil solidification simulation**, and where applicable, select **Fast diffuser** to account for back-diffusion of any fast-diffusing elements.

6. Click **Finish**.

7. Click to expand the **Event Log** window to see information about the progress of the calculation.

Event Log

```
12:12:02,234 INFO   Global test at  5.20000E+02
12:12:02,249 INFO   Terminating at    500.000
12:12:02,252 INFO   *** Buffer saved on file: C:\Windows\TEMP\TH35C8~1\STATE_~1.POL
12:41:16,752 INFO  Opened project file: Ex_05_Fe-C_stable_and_metastable_phase_diagram.tcu
12:42:17,239 INFO  Opened project file: Ex_03_Fe-C_phase_diagram.tcu
14:11:35,181 INFO  Define component C
14:11:35,181 INFO  Define component Fe
14:11:35,211 INFO  Performing general equilibrium calculation
```

8. The diagram is plotted in a **Plot Renderer** tab in the **Results** window. If you have calculated a single-point equilibrium, then a **Table Renderer** tab shows information about the equilibrium.

Results

Table Renderer 1

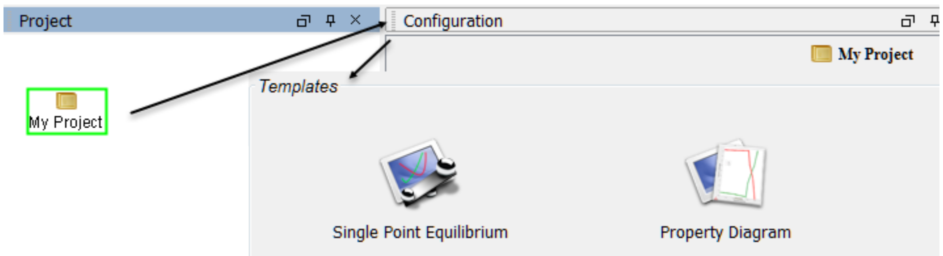
System				
Moles	1.00000			
Mass	12.01194	[g]		
Temperature	1000.00000	[K]		
Total Gibbs Energy	-12659.53136	[J]		
Enthalpy	11784.43824	[J]		
Volume	5.39784E-6	[m3]		
Component	*Mole Fraction*	*Mass Fraction*	*Activity*	*Potential*
Fe	0.00002	0.00010	0.00619	-42278.5054
C	0.99998	0.99990	0.21816	-12658.8942
Stable Phases				
	Moles	*Mass*	*Volume Fraction*	
BCC_A2#1	0.00002	0.00120	0.00003	Composition

9. To save the project, click **Save** on the main toolbar. To save a diagram or table, right-click the diagram or table and select **Save As**.

Creating a Project from a Template

Use a *Template* to create all the activities for a certain type of calculation. In the **Project** window click the **My Project** node to display the templates in the **Configuration** window as in

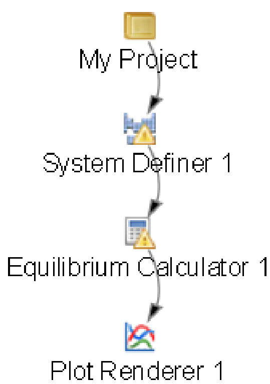
Configuration Window Templates



When you click a template, an activity tree is added to the Project window. You can then configure and perform the activities or the project.

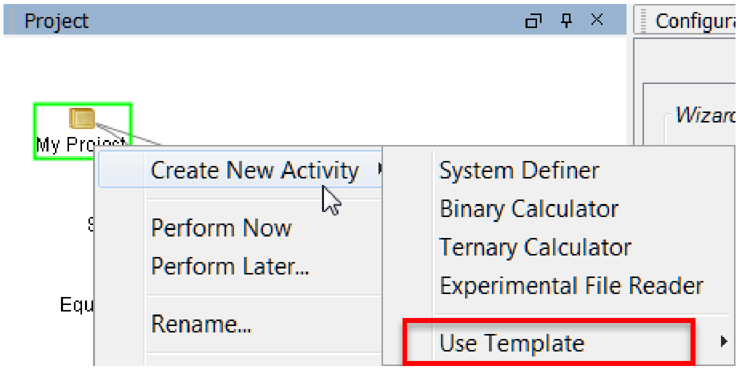
For example, if you click **Property Diagram**, a tree is added to the **My Project** node with **System Definer**, **Equilibrium Calculator** and **Plot Renderer** nodes.

Property Diagram Template



You can also select templates from the Project window. Right-click the **My Project** node and choose templates from the **Create New Activity → Use Template** submenu.

Use Template Submenu



Cloning Activities and Trees

As described in *Creating Activities and Successors* on page 29, you can clone a single activity or the selected activity and all the activities that come after it (successors). The predecessor of the selected activity also is the predecessor of the clone.

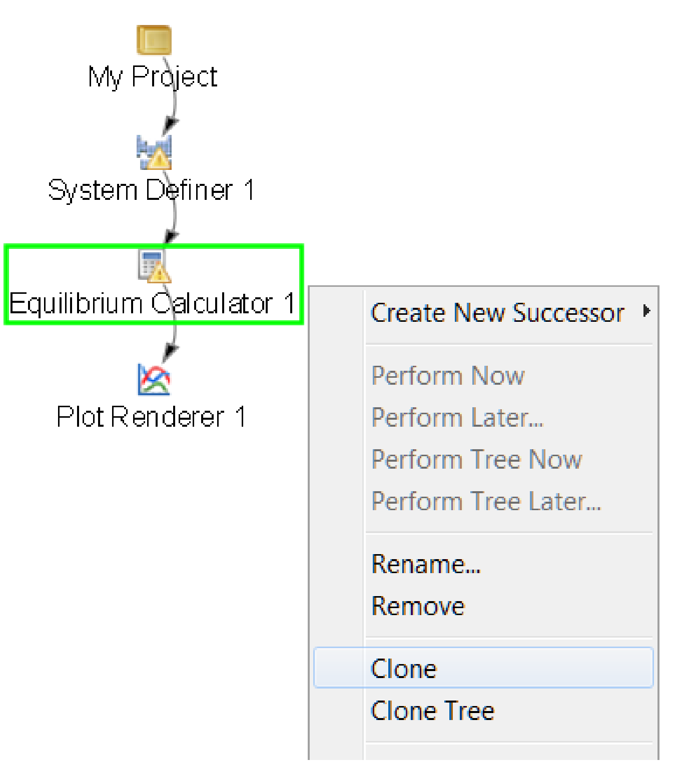
Cloned activities are configured exactly like the activities that they were cloned from, and any results of calculations or plots are also cloned. To clone a selected activity, right-click the activity and select **Clone**.

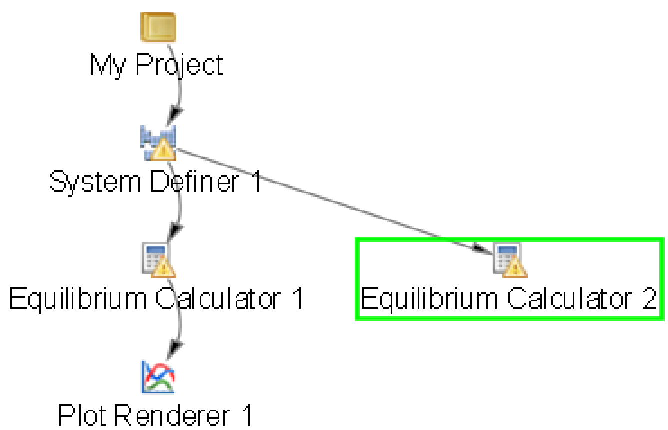
For example, if you clone the Equilibrium Calculator in this project the result is an **Equilibrium Calculator 2** created as a successor to **System Definer 1**.

The following applies to clones.

- The new Equilibrium Calculator has the same settings for conditions, functions and options as its Equilibrium Calculator 1 clone.
- If the Equilibrium Calculator is already performed, then the calculation results are cloned.
- If the activity is a Plot Renderer or a Table Renderer, then the plot or table is cloned.
- If you want to clone all the successor activities that follow a selected activity, then right-click and select **Clone Tree** instead.

Cloning an Equilibrium Calculator Activity Node





Equilibrium Calculator Clones and Successors

If you create an Equilibrium Calculator as a successor to another Equilibrium Calculator, then the successor inherits all the settings for conditions, functions and options from its predecessor. In this respect, it is similar to a clone.

The Equilibrium Calculator successor is different from a clone in that it does not inherit any calculated results and it is a successor to the Equilibrium Calculator instead of a successor to its predecessor.

The calculated result from an Equilibrium Calculator that is the predecessor to another Equilibrium Calculator is the starting value for the latter's calculation. The clone of an Equilibrium Calculator on the other hand, does not receive any data as input from the Equilibrium Calculator that it is cloned from.

Performing Projects, Trees and Activities

You can trigger the performance of an activity (also called a *job*) and all the activities below it in the same tree in either the **Project** or **Configuration** window. In the Project window you can also perform a single activity without performing its successors or perform the whole project (all the activity trees).

In the **Configuration** window, click **Perform Tree** to perform the currently selected activity and all the activities below it. If there are no activities below the selected activity, then the button says **Perform** instead.

All activities that must be completed as prerequisites are automatically performed first. For example, if you Perform Tree for a Plot Renderer, then the predecessors, Equilibrium Calculator and System Definer, are automatically performed before the Plot Renderer.

In the Project window, you can do any of the following:

- *Perform an activity*: Right-click the activity and select **Perform Now**, or **Perform Later** and schedule a time to perform the activity.
- *Perform an activity and all activities below it in the tree*: Right-click any node in the tree and select **Perform Tree Now**. Or select **Perform Tree Later** and schedule a time for all activities in the tree to be performed.
- *Perform a project*: Right-click **My Project** and select **Perform Now** or **Perform Later** and schedule a time to perform all project activities in the tree.

If there is an error during the performance of an activity, icons display status markers, which are described in *Node Status Markers* on page 40.

Using the Scheduler

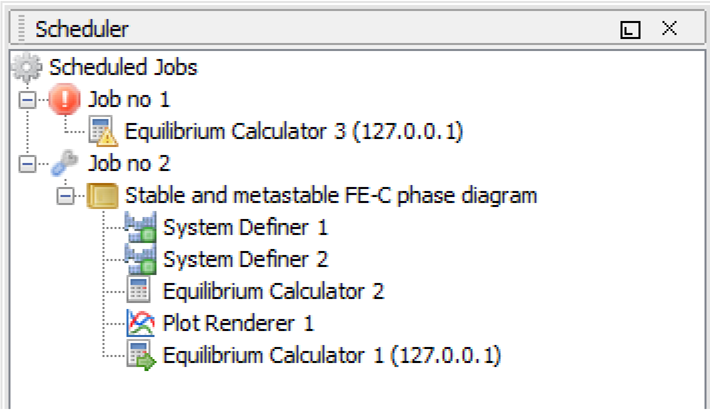
The **Scheduler** window has activity information about current or scheduled jobs. A *job* is any activity you have selected to be performed. The Scheduler also indicates if there is an error during a job and when a job has stopped. If you have set an activity, tree or project to be performed at a later time then this is also listed.

The following applies to jobs.

- Whenever you trigger the performance of a whole project, tree or individual activity, all the activities to be performed are shown under a job heading.
- When all the activities belonging to a job are completed the job and its activities disappear from the Scheduler window.
- If there is an error during the performance of an activity, icons for that job and activity remain in the Scheduler window until removed.

Also see *Node Status Markers* on page 40.

Scheduled Jobs Example



Managing the Schedule

In the Scheduler window, you can cancel scheduled jobs, remove errors and show information about errors.

- To cancel all jobs, right-click the **Scheduled Jobs** label or the cogwheel icon, and select **Cancel All Jobs**.
- To remove (clear) all failed activities and jobs, right-click the **Scheduled Jobs** label or the cogwheel icon, and select **Clear All Errors**.
- To cancel a specific job, right-click the job label (for example, **Job no 1**) and select **Cancel Job**.
- To remove (clear) a specific failed job, right-click the job label (for example, **Job no 1**) and select **Cancel Job**.
- To open a window with error details, right-click the label for the failed job (for example, **Job no 1**), and select **Show Error Log**.

The Event Log

The **Event Log** window is closed by default. Click the **Event Log** window once to expand it or select it from the **Window→Event Log** menu. Double-clicking the window maximizes the window.

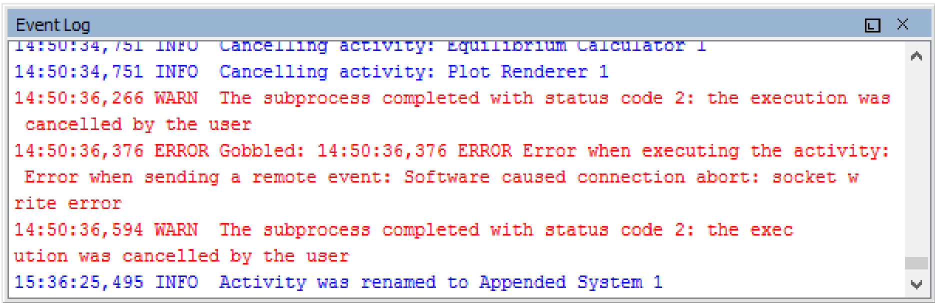
Information about what Thermo-Calc is processing or doing is in blue text and error or warning messages are in red text.

To specify the level of detail in the Event Log window, from the main menu select **Tools →Options**. In the **General** tab set the **Log level** slide bar to anything between **Debug** (most detailed) and **Error** (least detailed).

Also see *Changing Global Settings* on page 187.

Event Log Window Example

The example shows information, warnings and errors.



Graphical Mode Activities

These topics describe the activity types and has instructions for procedures tied to specific activities.

In this section:

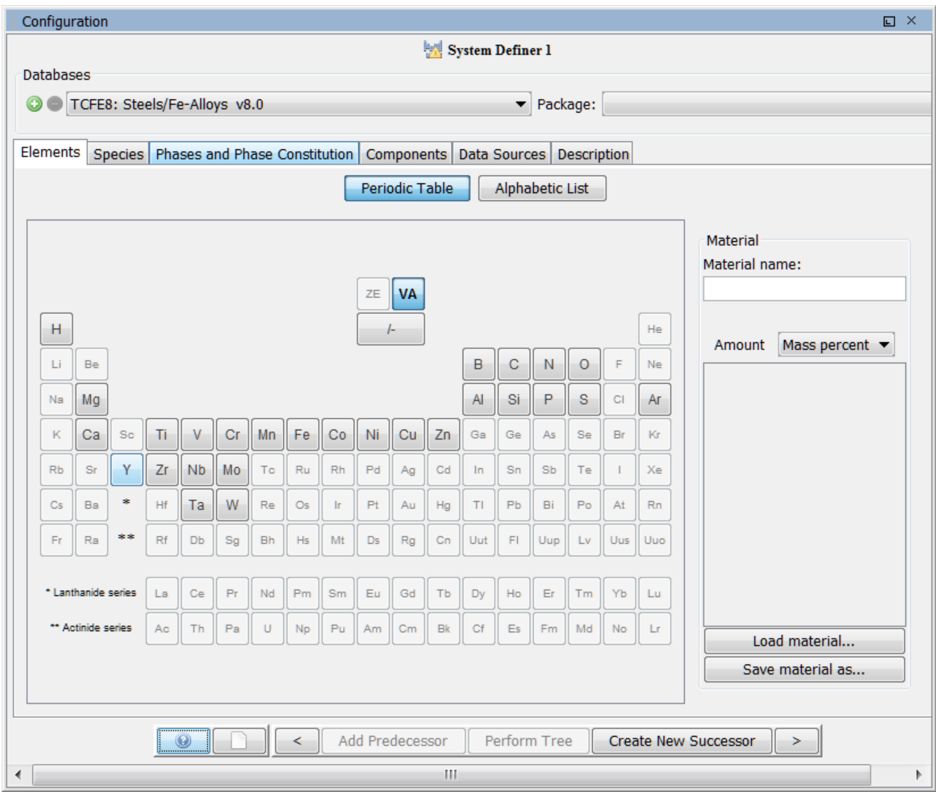
System Definer

In a System Definer activity, you select the database to use to retrieve thermodynamic data and define which elements the system has as components. You can also select which species to include as well as change the reference temperature and pressure for your components.

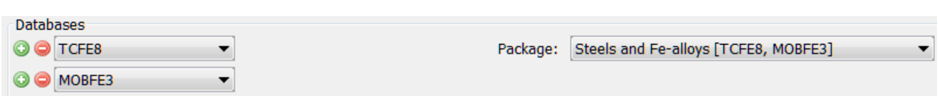
Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity.

The System Definer Settings

Configuration Window



Choose Database(s) or a Database Package



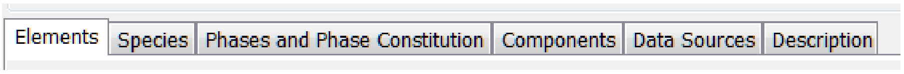
At the top of the **Configuration** window you can choose to add one or more databases by clicking the **Add a database** button. You can also choose a predefined or custom database package from the **Package** list. Depending on your license, there are predefined database packages available to select.

You can also click **Remove this database** button as required.

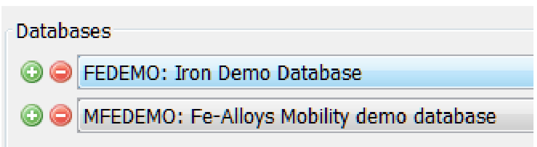
Also see *Global Settings: System Definition (System Definer)* on page 191 to learn how to create a database package.

If you have chosen a database package and then remove or change a database, a message displays *Selected databases and selected package may differ*. This means that the original package you selected, which added preset databases, does not match the databases you are using for the project. For information about how to clear the message, see *Global Settings: System Definition (System Definer)* on page 191.

There are several settings tabs available for the System Definer activity.



The following example uses the FEDEMO and MFEDEMO databases and Example 12 available with the *Graphical Mode Example Collection* on page 9.

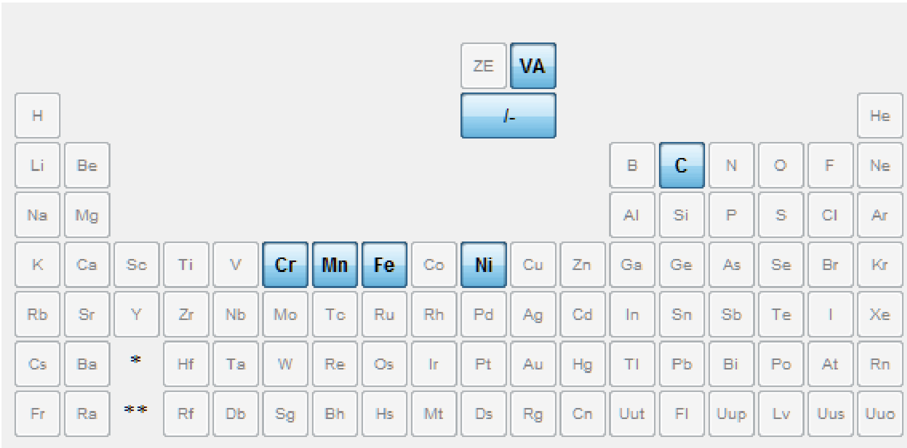


Elements

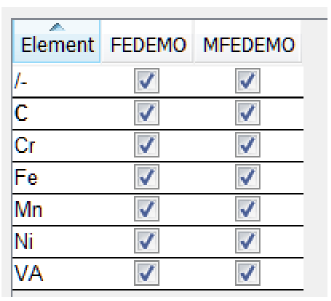
On the **Elements** tab, and based on the databases or database package selected, you can select and review elements using a **Periodic Table** or **Alphabetic List** format. Unavailable elements are greyed out on the Periodic Table view.

Periodic Table

In this example, using the demonstration databases FEDEMO and MFEDEMO, there are only a select number of elements to choose from.

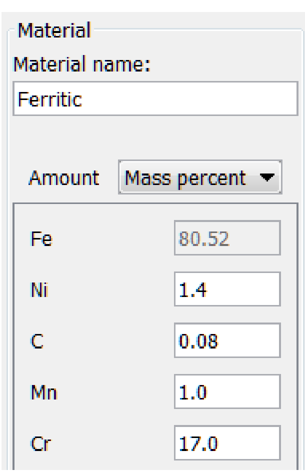


Alphabetic List



Material Composition

On the **Periodic Table** page, you can also define the material composition:



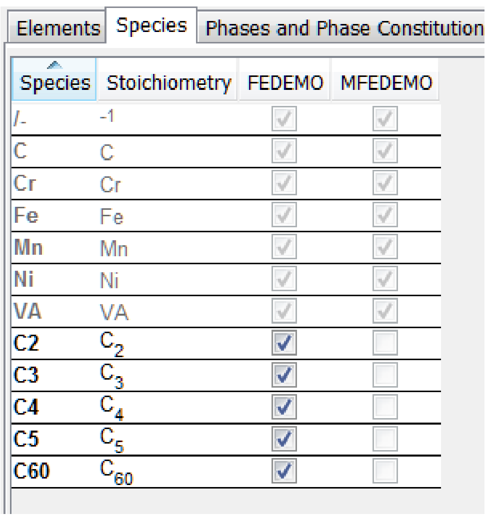
At the bottom of the window, click **Load material** to open a material file. Click **Save material as** to save the material you defined, for example **Ferritic**.

Also see *Global Settings: System Definition (System Definer)* on page 191 to learn how to create your own database packages.

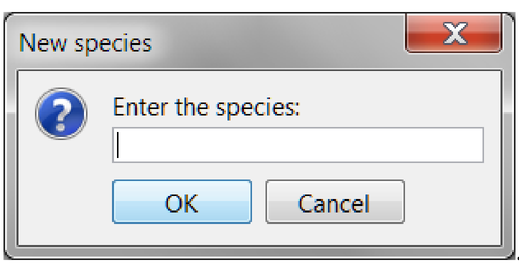
If you Load a material file and then change the composition, a message displays *The composition has changed*. This means that the original values/composition does not match the values you are using for the project. The message is cleared by either reloading the material file or editing the composition field to the original value.

Species

On the **Species** tab for each database you can select and edit the **Species**.

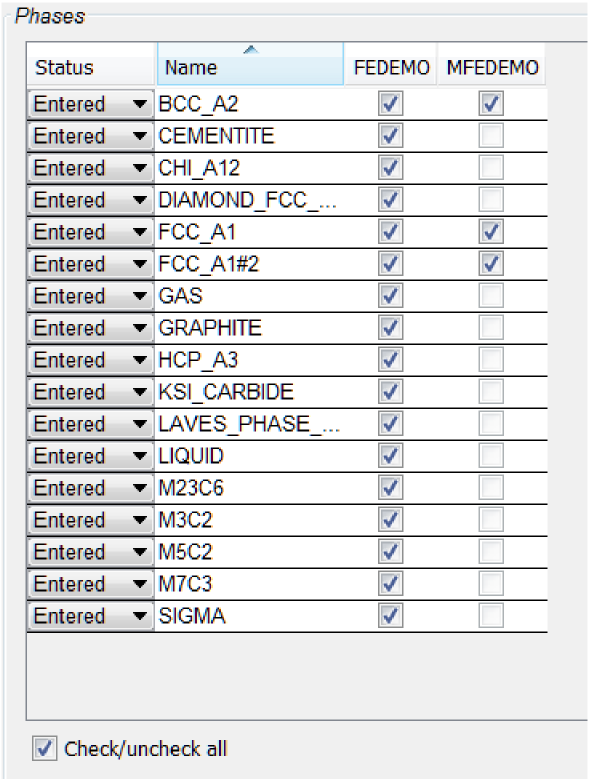


At the bottom of the window, click **Add new species** and enter the name of species in the field.



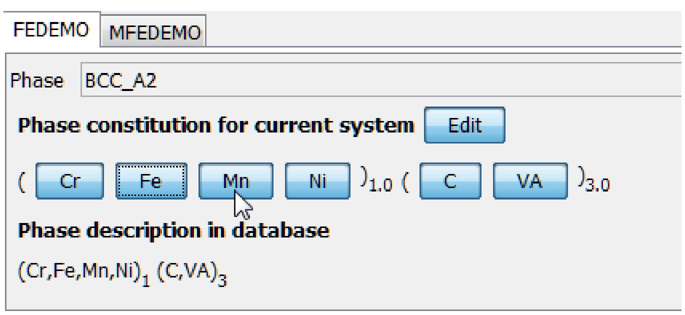
.

Phases and Phase Constitution



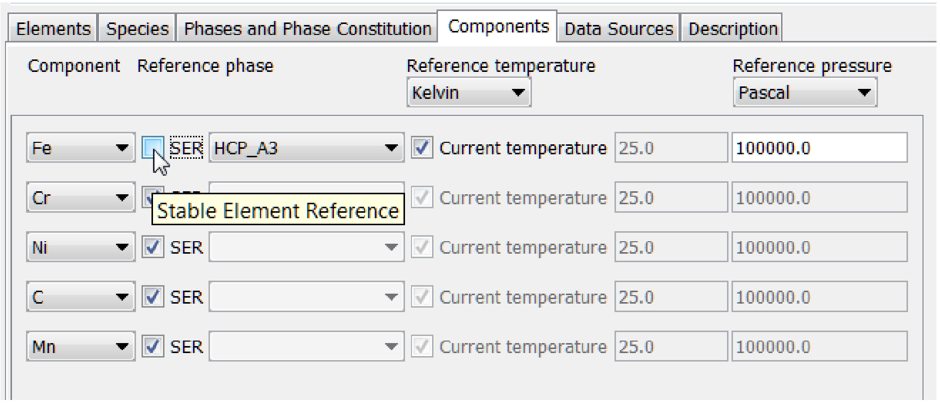
At the top of the **Phases and Phase Constitution** tab for each database, you can review and edit the **Status** (**Entered** or **Dormant**) for the selected**Phases**. Select the **Check/uncheck all** check box to toggle between the options.

To review the **Phase constitution** for a specific phase, click it in the **Phase** list (for example **BCC_A2**) and its details are displayed at the bottom of the window. Click **Edit** then click an element to select or reject a constituent. The last remaining constituent on a sublattice cannot be rejected.



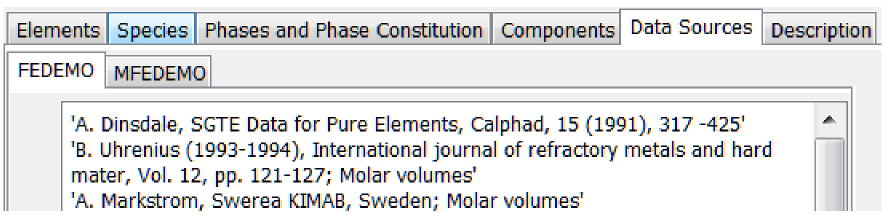
Components

The **Components** tab is where you can review and edit the **Component**, **Reference phase**, **Reference temperature**, and **Reference pressure**. Click to toggle the **SER** (Stable Element Reference) check box on and off. Then choose a **Reference phase** from the list and edit accordingly.



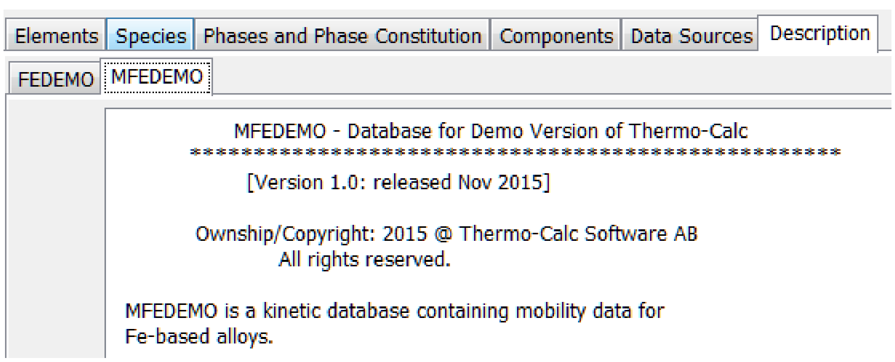
Data Sources

Data sources are not available until the activity has been performed. Click **Perform Tree** to display available data sources.



Description

The **Description** tab displays a description of the selected database(s). In this example, **MFEDEMO**.



Creating a System Definer Activity

This topic describes how to define a system; specifically how to select one or more databases and define the elements in the system.

To change the default database or create a database package, see *Global Settings: System Definition (System Definer)* on page 191.

Also see *About the Database Files* on page 184.

1. Right-click **My Project** and from the menu select **Create New Activity→ System Definer** .
2. In the **Project** window, click the *System Definer* on page 54 activity node.
3. In the **Configuration** window, from the **Databases** menu choose **Database(s)** or a **Database Package**.

   Options are to:

   - Select **<User database>** to define your own.
   - You can choose to add one or more databases by clicking the **Add a database** button.
   - From the **Package** list, choose a predefined database package to add multiple databases at a time. Depending on your license, there are predefined database packages available to select.

4. Select the **Elements** tab to choose the elements in the **Periodic Table** or use the **Alphabetic List** to choose elements for the system.
5. As required click the tabs to make edits: **Species**, **Phases and Phase Constitution**, and **Components**. The **Data Sources** and **Description** tabs provide information based on the database.
6. Click **Perform** to start the System Definer activity immediately and retrieve thermodynamic data about the system. Click the **Event Log** window to watch the progress of the job.

Property Model Calculator

A Property Model Calculator allows you to predict and optimize material properties. A set of general models is available to use: a *Phase Transition Model* on page 68, *Driving Force Model* on page 69, *Interfacial Energy Model* on page 70, *Coarsening Model* on page 70, and an *Equilibrium Model* on page 72.

You can evaluate models simultaneously over a range of compositions and cross plot the results. You can then conduct an uncertainty analysis and plot the results as either a histogram or as a probability plot. The **Configuration** window for a Property Model Calculator using the *Phase transition* model is shown and described in *Defining the Property Model Calculator* on page 64.

Property Model Calculator Workflow

When a project is first being designed, it is recommended you test it by first performing a **Single** calculation type. This verifies that the configuration of the model is valid, otherwise you will get a *Not a Number* result in *Event Log Window* on page 116 window indicating the calculation did not work correctly.

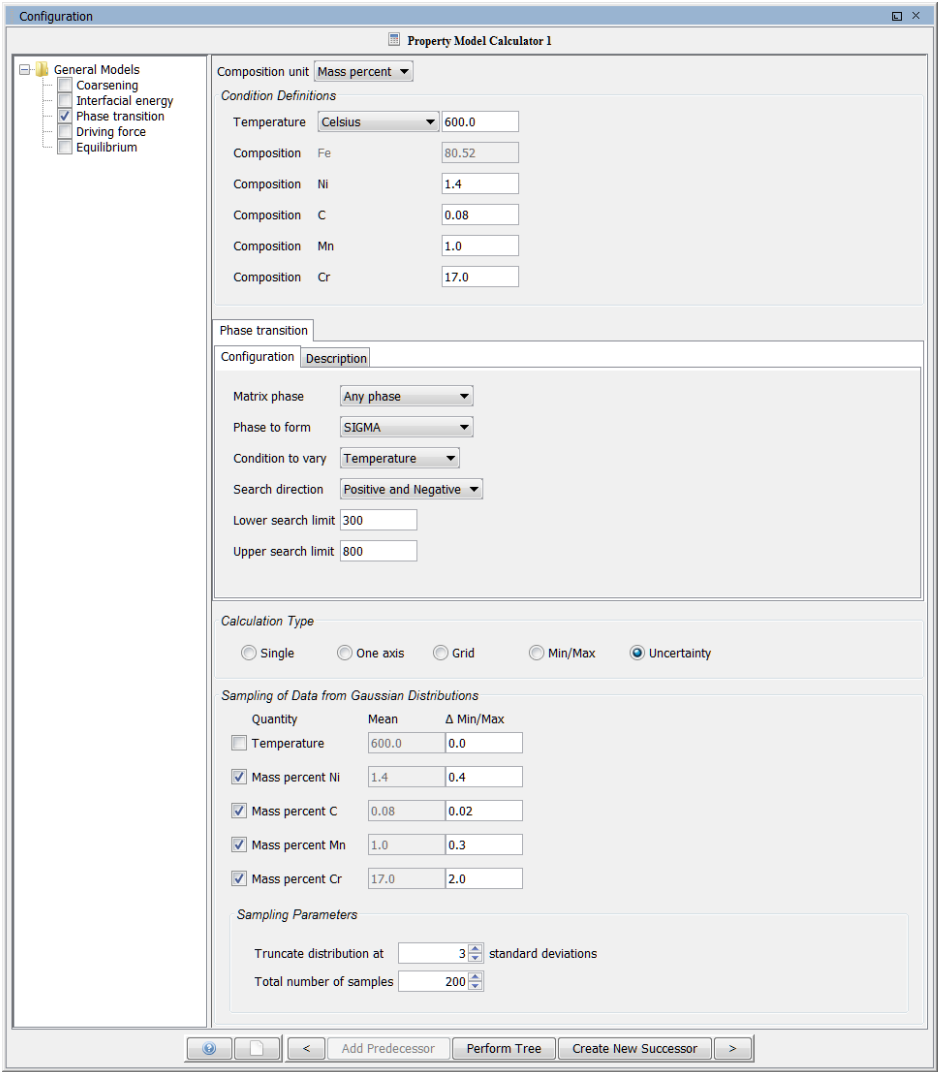
Once the Single calculation type performs successfully, you can experiment with other calculation types such as **Grid**, **Min/Max** or **Uncertainty**. Best practice for both Grid and Uncertainty is to start with a small number of steps (Grid) or samples (Uncertainty) to confirm that it works before increasing the number of steps or samples.

In general, the workflow for the different models will differ. A recommended start point is to do a Property Diagram calculation to determine what phases you get and where the transitions are before setting up the Property Model Calculator. You can create a Property Diagram using the Quick Start wizard or a template.

Also see *The Quick Start Wizard* on page 43 and *Creating a Project from a Template* on page 46.

For examples using the Property Model Calculator, see examples 10, 11, and 12 in *Graphical Mode Example Collection* on page 9.

Configuration Window



Configuration Settings

Setting/Button	Description
Composition unit	Choose a unit: **Mass percent**, **Mole percent**, **Mass fraction** or **Mole fraction**
Condition Definitions	Choose a unit (**Kelvin**, **Celsius** or **Fahrenheit**) and then enter a value for the **Temperature** and **Composition**. The values entered in Condition Definitions are used as the start values for the different calculation types.
Calculation Type	Choose a calculation type: **Single** (no axes), **One axis**, **Grid**, **Min/Max** or **Uncertainty**. Select **Single** to calculate a single point.

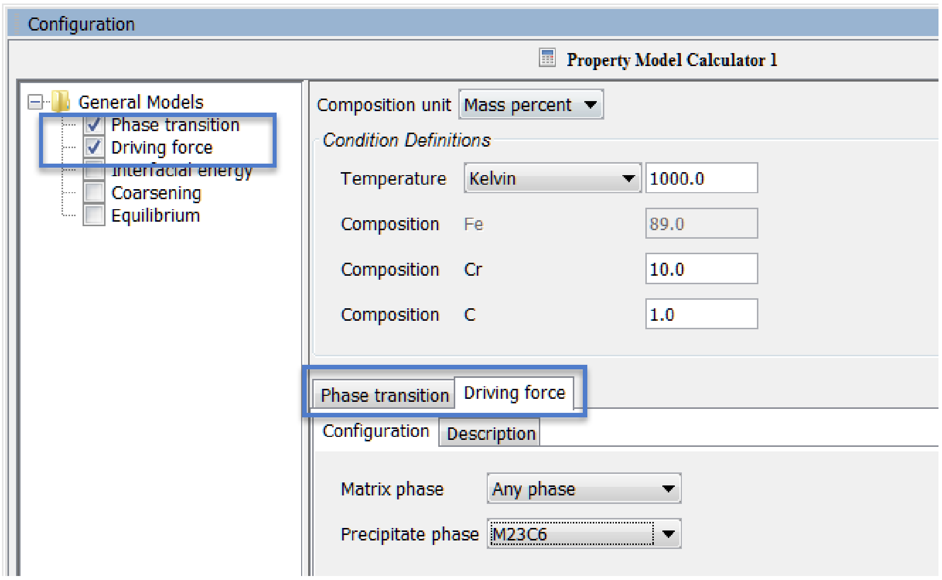
Setting/Button	Description
	Select **One axis** to vary a quantity on the X-axis. Select a **Grid** to evaluate two axis variables of the selected quantities in the specified range and number of steps. The **Min/Max** calculation evaluates the property model(s) for all variations of the selected quantities at the given limits. The **Mean** field is as defined under Condition Definitions for the respective quantity. The total minimum and maximum of the model(s) results are shown in the Event log. The **Uncertainty** calculation evaluates the property model(s) where the values of the quantities are sampled from Gaussian distributions. The **Mean** field is as defined under Condition Definitions for the respective quantity. The result is visualized as a histogram or normal probability plot by adding a Plot Render activity.
Single calculation	For the **Single** calculation type no further settings are required. The results from this calculation are displayed in the **Event Log**.
Axis Definition	For **One axis**, select a **Quantity** to vary along the X-axis, for example, **Temperature** then enter a **Min**, **Max** and **Number of steps**.
Grid Definitions	For the **Grid** calculation type, define the two axes variables using the fields and menus: **Quantity**, **Min**, **Max** and **Number of steps**. The number of steps along with the minimum and maximum values for the axes define a grid. For each grid point the selected models are evaluated.
Min/Max Definitions	For the **Min/Max** calculation type, click to select the applicable check boxes under **Quantity to enable or disable an axis**. The **Mean** field is as defined under Condition Definitions for the respective quantity. Enter numerical values in the Δ **Min/Max** fields to evaluate the change to the quantity. The highest and lowest value of each selected model results are shown in the Event log.
Sampling of Data from Gaussian Distributions	For the **Uncertainty** calculation type, click to select the applicable check boxes under **Quantity to vary this quantity**. The **Mean** field is for the distribution function as defined under Condition Definitions for the respective quantity. ΔMin/Max is the 95% confidence interval.
Sampling Parameters	For the **Uncertainty** calculation type, choose a number of **Standard deviations** where you want to **Truncate the distribution**. Choose the **Total number of samples**.

Also see *About the Uncertainty Calculations* on the next page for more detail.

Defining the Property Model Calculator

1. From the left side of the Configuration window, click to select one or more of the check boxes under the **General Models** folder.

   A tab with the same name as the property model opens in the right side of the window under *Condition Definitions*. When you select more than one model, a matching tab is added in the same order that the check box is selected.



2. Enter the settings described in the following table for the *Condition Definitions* and for each *Calculation Type* (Single, Grid, Min/Max, and Uncertainty). For each model on the **Configuration** tab, select or enter settings as required. Click the **Description** tab for more information about the model and refer to the individual sections for details.

   - *Phase Transition Model* on page 68
   - *Driving Force Model* on page 69
   - *Interfacial Energy Model* on page 70
   - *Coarsening Model* on page 70
   - *Equilibrium Model* on page 72

About the Uncertainty Calculations

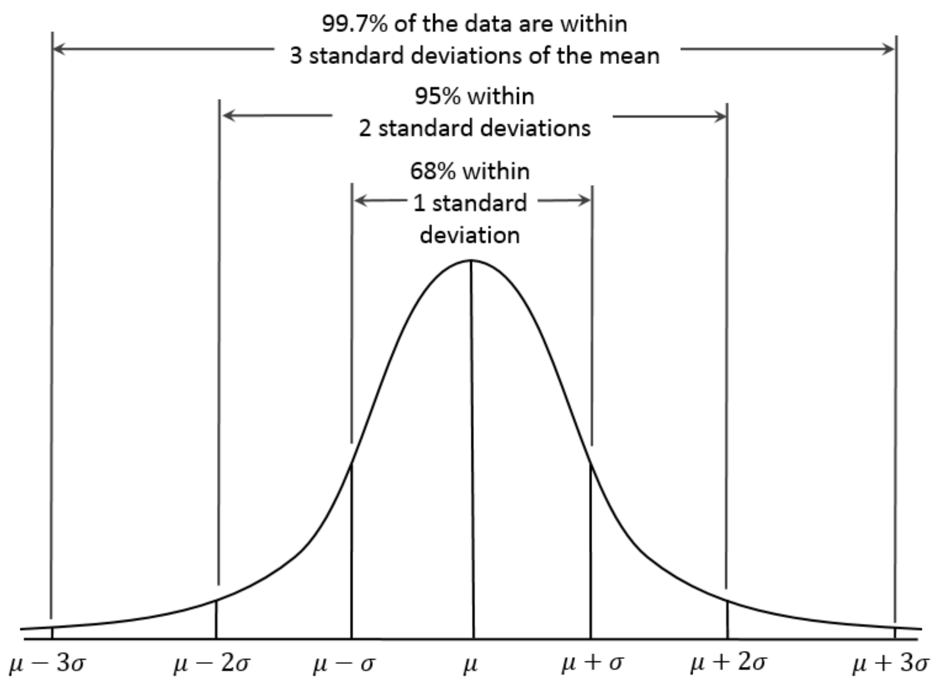
The following example uses the *Phase Transition Model* on page 68 and an **Uncertainty Calculation Type** to describe how the sampling of data is performed. The example (which is example 10 in *Graphical Mode Example Collection* on page 9) predicts the transition temperature to the unwanted brittle Sigma phase and shows how the temperature is influenced by changes to a steel alloy's composition.

The **Uncertainty** calculation in Thermo-Calc uses the values (the mean) entered for quantities in the Condition Definitions section as the starting point to estimate, within a certain probability, the true variation of the quantity.

An example of a Gaussian distribution is shown The *mean* is at the peak of the curve (labelled μ). The *standard deviation* (σ) is a plus and minus range, ±1σ, ±2σ or ±3σ, of the variance from the mean.

Gaussian Distribution, the 3-Sigma Rule

Image by Dan Kernler, https://en.wikipedia.org/wiki/Normal_distribution#/media/File:Empirical_Rule.PNG

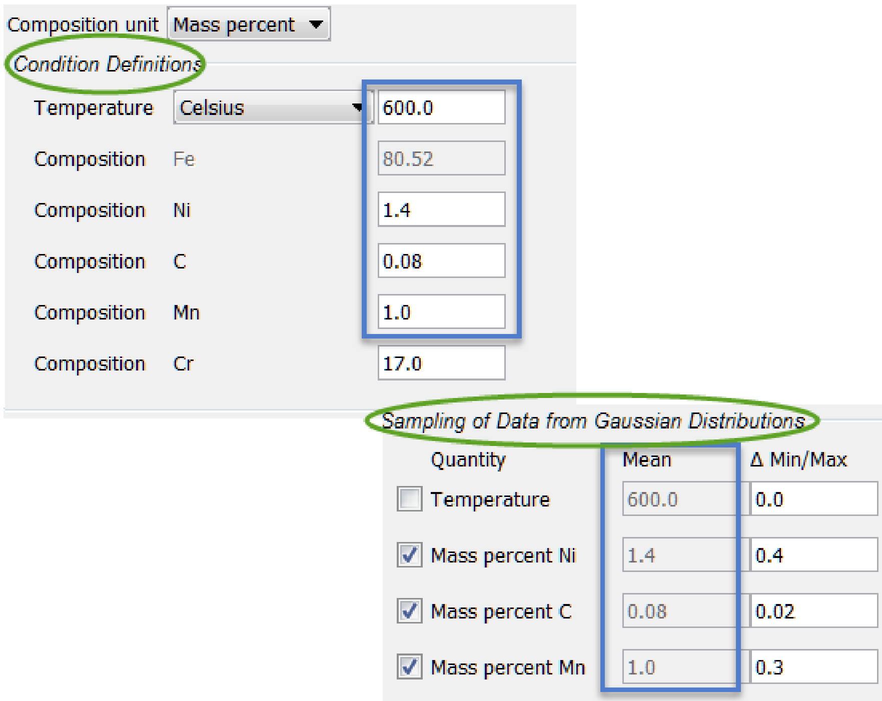


As long as the number of events (the sample size) is very large, then the Gaussian distribution function can be used to describe physical events. A Gaussian distribution is assumed to describe how each quantity varies in an uncertainty calculation.

In Thermo-Calc, the mean of the quantity's distribution is defined in the *Condition Definitions* section. The **Mean** is then used in the *Sampling of Data from Gaussian Distributions* section as the starting point for how much you want to deviate from this value.

Condition Definitions and Sampling Settings

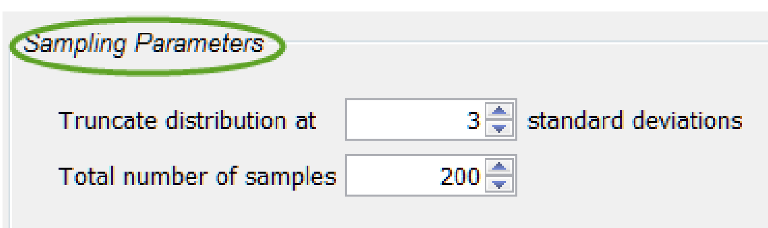
The mean defined in Condition Definitions is used during the sampling.



The values entered in the Δ **Min/Max** fields are the resulting width of the distribution corresponding to the 95% confidence interval (approximately $\mu \pm 2\sigma$) shown in *Gaussian distribution, the 3-sigma rule*.This confidence interval means that approximately 95% of the sampled values will fall within these limits.

The default value for truncating the distribution is set at **3 standard deviations** in the *Sampling Parameters* section. This is also where you set the **Total number of samples**. The default is 200. In order to maintain a Gaussian distribution for the varied quantities, this number should not be too low.

Sampling Parameter Settings

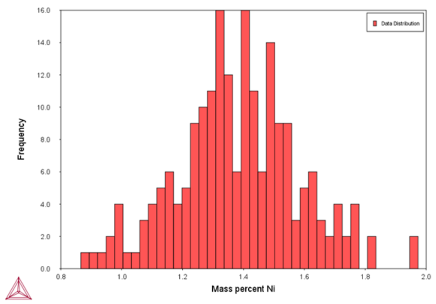


You can experiment by varying the conditions and the number of samples to see the distribution results smooth out as is to be expected based on the Central Limit Theorem. For example, on the **Plot Renderer** the variable is set to measure the **Composition** of **Ni**. The initial sample size uses the default, 200, and then, as in the example below where the sample size is increased to 300, 400, 600, 800, and 1000 to show how the plot follows the Gauss distribution.

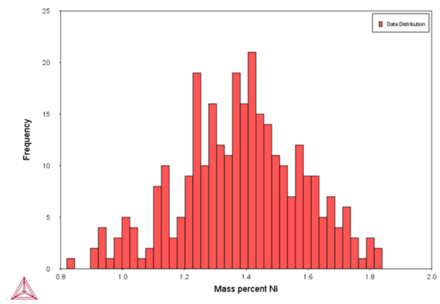
Histogram Plot

This is a series of histogram plots of the uncertainty calculations sampling data from Gaussian distributions. It compares the Ni composition where the total sample size is increased to show how the model follows the Gauss distribution.

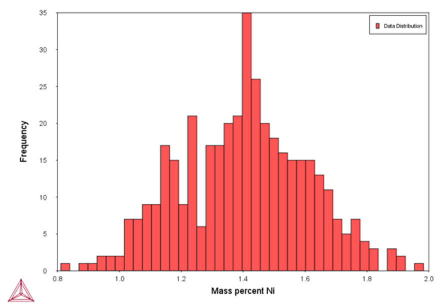
Total samples = 200



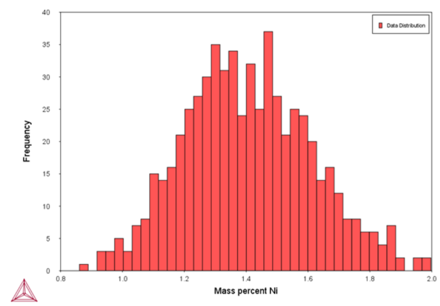
Total samples = 300



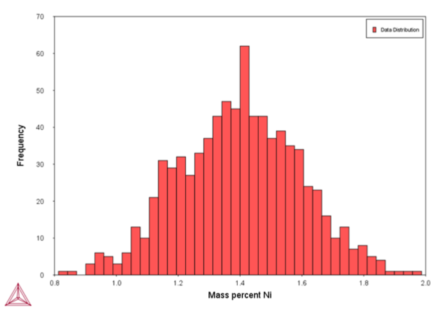
Total samples = 400



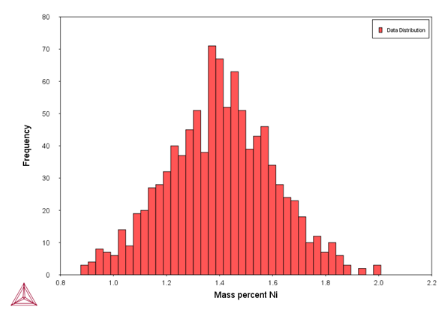
Total samples = 600



Total samples = 800

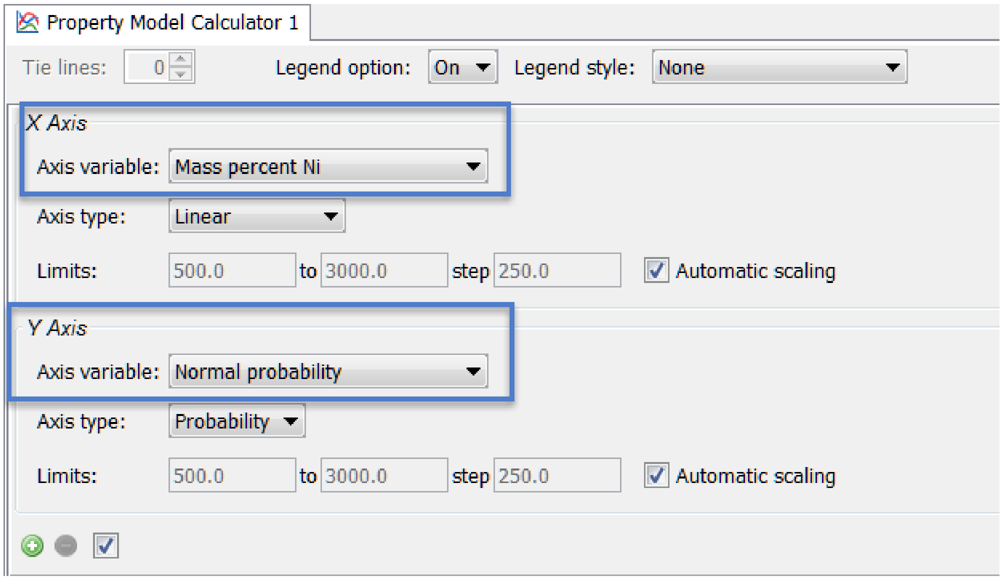


Total samples = 1000

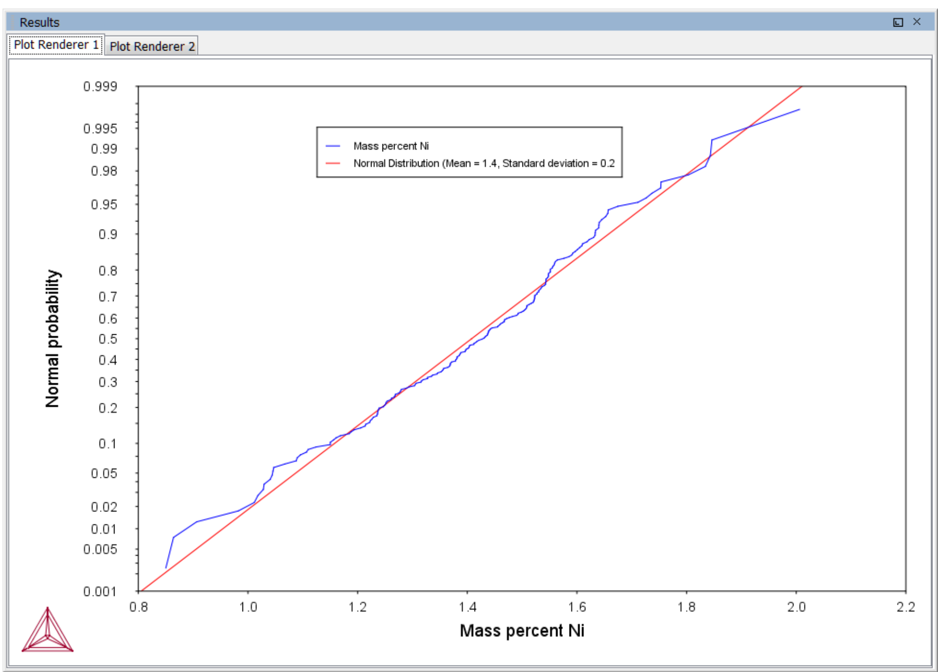


Normal Probability Plot

You can also plot the results as a Normal probability. Continuing with this example, on the Property Model Calculator Configuration window the X-Axis variable is plotted as Composition and the Y-Axis variable is plotted as the Normal probability.



This is a normal probability plot example of the uncertainty calculations sampling data from Gaussian distributions. It compares the Ni composition where the total sample size is 200. The sampled data (blue line) closely follows the ideal normal distribution (red line). Deviations are naturally larger at the tails (≈1%Ni and ≈2%Ni) since the number of samples are fewer there compared to the center of the distribution.



Phase Transition Model

By varying set conditions, the **Phase Transition** model calculates the point when a new phase may form. The model is useful to determine melting temperature, boiling temperature or solubility limits. It returns the phase transformation temperature, or composition, depending on the varied condition.

For an example of a phase transition model, see example 10 in *Graphical Mode Example Collection* on page 9.

Define the:

- **Matrix phase** - The single phase region to start in. Select 'Any phase' to use the phases that take part of the initial equilibrium or choose a specific phase.
- **Phase to form** - For the new phase to be formed, select 'Any phase' or choose a specific phase.
- **Condition to vary** – Choose whether to vary a composition or temperature condition when calculating the phase transition.
- **Search direction** - For the composition or temperature condition being varied, choose the Search direction to be in a **Positive**, **Negative**, or **Positive and Negative** direction.
- **Upper search limit** and **Lower search limit** - These values also relate to the search direction, where an estimated change to the released condition implies where a new phase is expected.

  The start value (given in Condition Definitions) and phase transition point need to be within these limits, otherwise the model returns *Not a Number* as result. For example, a start temperature is 1100 K then the limits need to be outside these, e.g. 500-1500.

Driving Force Model

The **Driving Force** model calculates the thermodynamic driving force for a phase.

- **Matrix phase**: Choose a **Matrix phase**, the single phase region to start in. Select 'Any phase' to use the phases that take part of the initial equilibrium or choose a specific phase.
- **Precipitate phase**: Choose a **Precipitate phase** to calculate its driving force.

  The calculation result is the driving force divided by RT.

For an example of a driving force model, see example 12 in *Graphical Mode Example Collection* on page 9.

Interfacial Energy Model

Use the **Interfacial Energy** model to estimate the interfacial energy between a matrix phase and a precipitate phase using thermodynamic data from a CALPHAD database. The approximation model is based on Becker's bond energy approach and is also available in Precipitation Module (TC-PRISMA). In the actual calculation, the coefficients taken distinguish between only bcc and fcc structure for the (110) and (111) lattice plane, respectively.

For other planes and/or other phases, the estimate is approximate. In most situations, it is recommended to perform calibrations against some experimental data because the interfacial energy is dependent on so many factors that are ignored in the estimation, for example entropy, incoherency, orientation, and curvature, etc. Uncertainty factors must also be considered as thermodynamic data in CALPHAD databases are not always accurate.

Select the **Matrix phase** and **Precipitate phase** to return the interfacial energy [J/m^2].

For an example of an interfacial energy model, see example 11 in *Graphical Mode Example Collection* on page 9.

Estimation of Coherent Interfacial Energy

Interfacial energy is an important parameter used in precipitation simulations to calculate the rates of nucleation, growth/dissolution, and coarsening. The value of interfacial energy can vary dramatically (usually between 0.01 to 2.0 J/m^2).

The extended Becker’s model functions to estimate coherent interfacial energy by using thermodynamic data from existing CALPHAD thermodynamic databases:

$$\sigma_c = \frac{n_s z_s}{N_A z_l} \Delta E_s$$

where σ_c is the coherent interfacial energy, n_s is the number of atoms per unit area at the interface, z_s is the number of cross bonds per atom at the interface, z_l is the coordination number of an atom within the bulk crystal lattice, and ΔE_s is the energy of solution in a multicomponent system involving the two phases being considered. (Reference[1]).

Coarsening Model

The **Coarsening** model calculates the coarsening rate coefficient K (m^3/s) of a spherical precipitate phase in a matrix phase.

- **Matrix phase**: Choose the **Matrix phase**.
- **Precipitate phase**: Choose a **Precipitate phase** to calculate its coarsening rate.

[1] Becker, R., Die Keimbildung bei der Ausscheidung in metallischen Mischkristallen, Ann. Phys. 424, 128–140 (1938).

Returns the coarsening rate coefficient [m^3/s] for the precipitate phase and the interfacial energy [J/m^2] between matrix and the precipitate phase.

This model requires that the thermodynamic and kinetic databases used include descriptions of molar volume of the matrix and precipitates as well as mobilities for the matrix phase.

For an example of a coarsening model, see example 11 in *Graphical Mode Example Collection* on page 9.

Particle Coarsening

In the coarsening regime where the driving force is capillarity, the change of mean particle radius evolves according to the equation:

$$\bar{r}^3 - \bar{r}_0^3 = Kt$$

Following Morral and Purdy, the rate constant of precipitating β phase in a multicomponent α phase is

$$K = \frac{8\sigma V_m^\beta}{9}\left[\left[c_i^\beta - c_i^\alpha\right]^T [M]^{-1}\left[c_k^\beta - c_k^\alpha\right]\right]^{-1} K = \frac{8\sigma V_m^\beta}{9}\left[\left[c_i^\beta - c_i^\alpha\right]^T [M]^{-1}\left[c_k^\beta - c_k^\alpha\right]\right]^{-1}$$

The mobility matrix M is not the same as the mobilites that can be obtained directly from the Diffusion Module (DICTRA). The M matrix instead corresponds to the L'' matrix as defined by Andersson and Ågren.

Using the u-fractions that is defined as

$$u_k = \frac{x_k}{\sum_{j \in S} x_j}$$

where the summation only is performed over the substitutional elements give the final equation for the coarsening rate coefficient.

$$K = \frac{8\sigma V_m^\beta}{9}\left[\left[u_i^\beta - u_i^\alpha\right]^T \left[L''^{\alpha}_{jk}\right]^{-1}\left[u_k^\beta - u_k^\alpha\right]\right]^{-1} K = \frac{8\sigma V_m^\beta}{9}\left[\left[u_i^\beta - u_i^\alpha\right]^T \left[L''^{\alpha}_{jk}\right]^{-1}\left[u_k^\beta - u_k^\alpha\right]\right]^{-1}$$

The interfacial energy σ is calculated using the extended Becker's model (that also is available as a separate Property Model Calculator model)

References

- Morral, J. E. & Purdy, G. R. "Particle coarsening in binary and multicomponent alloys," *Scr. Metall. Mater.* 30, 905–908 (1994).

- Andersson, J. & Ågren, J. Models for numerical treatment of multicomponent diffusion in simple phases. *J. Appl. Phys.* 72, 1350–1355 (1992).

Equilibrium Model

The Equilibrium model calculates the equilibrium for the given conditions. Optionally define additional **Function Definitions**. All quantities that normally can be obtained from an equilibrium calculation can also be obtained when plotting the results, as well as the defined functions.

Equilibrium Calculator

An Equilibrium Calculator allows you to set the conditions for, and perform, a calculation. The Configuration window for an Equilibrium Calculator has these tabs:

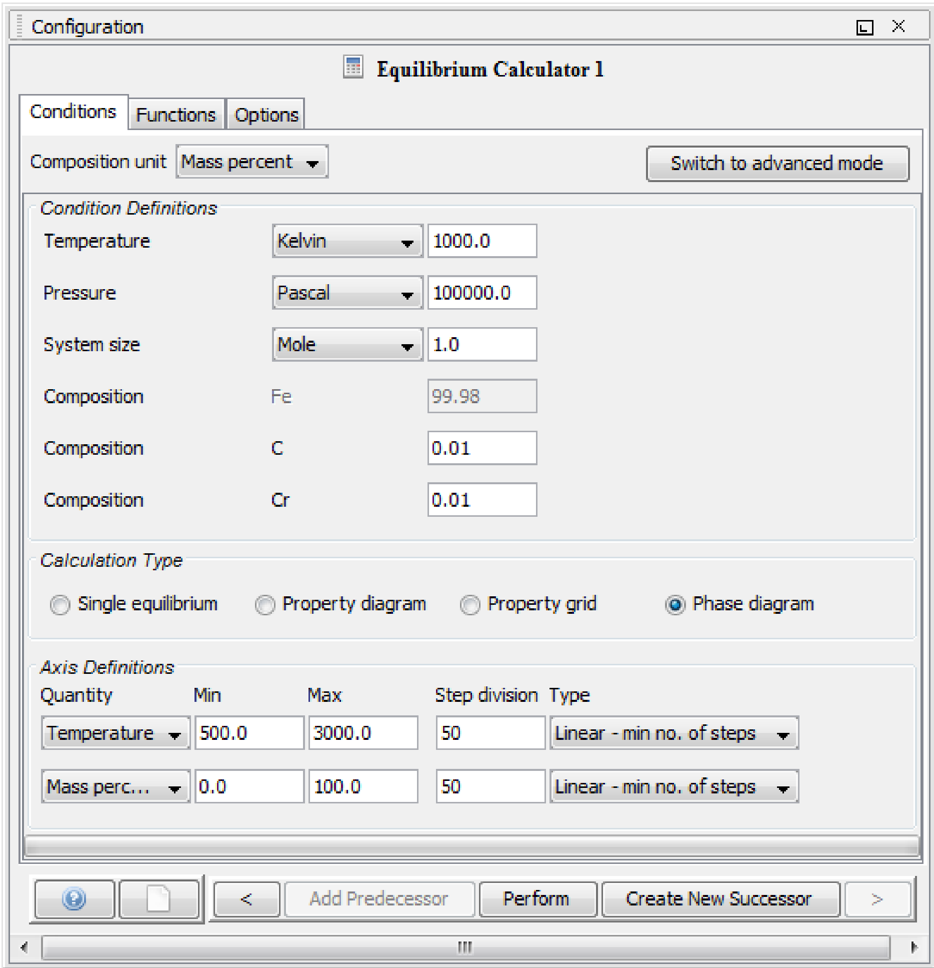
- **Conditions**: Set the conditions for your calculation that define the stepping or mapping axis variables. This tab can be viewed in a simplified mode and in an advanced mode. See *Setting Equilibrium Calculator Conditions* on the next page.
- **Functions**: Define your own quantities and functions, which you then can calculate and plot. See *Calculating and Plotting Functions* on page 76.
- **Options**: Modify numerical settings that determine how equilibria are calculated, as well as how stepping and mapping calculations are performed.

The **Options** tab is where you can locally define specific Equilibrium Calculator settings. These are the same settings described in *Global Settings: Calculation (Equilibrium Calculator)* on page 194.

See *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity. An Equilibrium Calculator that is the successor of another Equilibrium Calculator inherits all the Configuration window settings of the predecessor.

For an example of how serially coupled Equilibrium Calculators can be used, see example 6 in *Graphical Mode Example Collection* on page 9.

Configuration Window



Setting Equilibrium Calculator Conditions

Conditions and axis variables for the Equilibrium Calculator are set in the Conditions tab of the Configuration window. The tab can be viewed in two modes: simplified (the default) or advanced.

The conditions to set depend on the type of calculation.

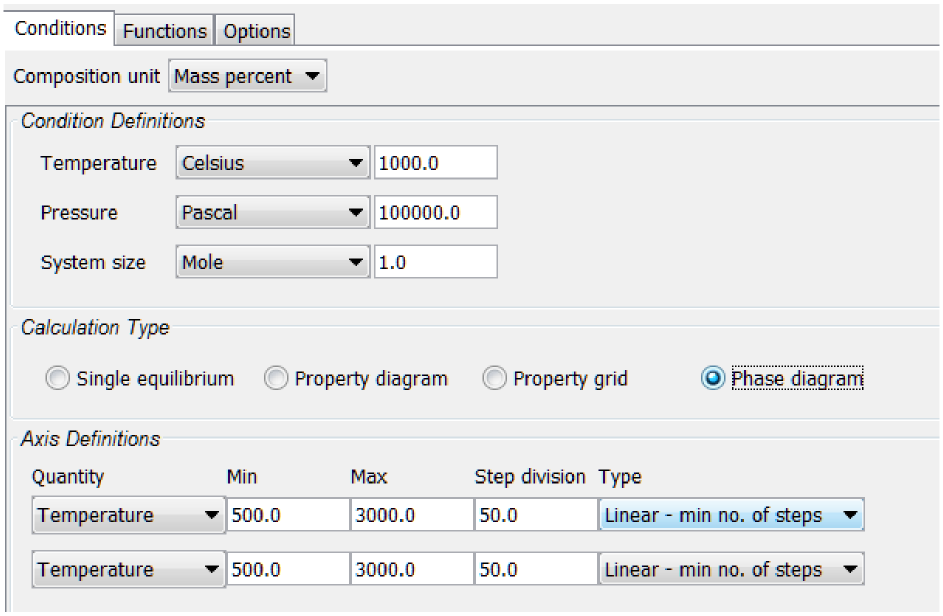
See *Graphical Mode Example Collection* on page 9: Example 1 is a single-point equilibrium calculation; Example 2 is a stepping calculation; Example 5 is a mapping calculation.

In advanced mode, you can add and remove conditions as well as set additional axis definitions. However, the number and types of conditions set must still conform to the Gibbs' phase rule. The **Number of missing conditions is** field, at the top of the tab shows how many conditions that you have left to set. If the number is negative, then that number of conditions need to be removed.

Setting Conditions for Simplified Mode

The following table briefly describes what you can set on the **Conditions** tab in simplified mode. Click **Switch to advanced mode** for more options.

The Conditions Tab



Conditions Tab, Simplified Mode Settings

These are the settings on the Equilibrium Calculator Conditions tab in Simplified Mode.

Setting or Condition	Options
Composition unit	Choose from **Mass percent**, **Mole percent**, **Mass fraction** or **Mole fraction**.
Condition Definitions	Specify the values (and units) of the state variables: **Temperature**, **Pressure**, and **System size**.
Calculation Type	**Single equilibrium** (no axes), **Property diagram**, **Property grid** or **Phase diagram**. For the **Single** calculation type no further settings are required. The results from this calculation are displayed in the **Event Log** .
Axis Definitions	**Property diagram**: Define the axis variable using these fields and menus: **Quantity**, **Min**, **Max**, **Step division** and **Type**. **Property grid**: Define the two axis variables using these fields and menus: **Quantity**, **Min**, **Max** and **Number of steps**. **Phase diagram**: Define the two axis variables using these fields and menus: **Quantity**, **Min**, **Max**, **Step division** and **Type**.

Setting or Condition	Options
	If the Equilibrium Calculator already has a Plot Renderer as a successor and you change the stepping/mapping quantities, then the quantities represented on the X- and Y-axis are automatically updated in the Plot Renderer.
Number of steps (Property grid)	Set a fixed **Number of steps**. The number of steps along with the minimum and maximum values for the axes define a grid. For each grid point an equilibrium is calculated.
Type and Step division (Property diagram and Phase diagram)	**Type** options: Linear - min no. of steps, Linear - max step size and Logarithmic - scale factor. For Linear - min no. of steps, the **Step division** value specifies a minimum number of steps that is calculated between the minimum and maximum value during the stepping/mapping operation. For Linear - max step size, the **Step division** sets the maximum size of each step.For Logarithmic - scale factor, **Step division** specifies the scale factor of a logarithmic function that distributes the steps from the initial equilibrium to the minimum and maximum values defined.

Calculating with a Fixed Phase (Advanced Mode)

This describes how to use the advanced mode to set a phase to be fixed at a certain amount. This allows you to, for example, find out at what temperature a material starts to melt. If you set the phase to be fixed to liquid phase at zero amount, and do not define the temperature state variable, then you can calculate at what temperature the material enters a state where the liquid phase is no longer zero (that is, when it starts to melt).

How to Calculate with a Fixed Phase

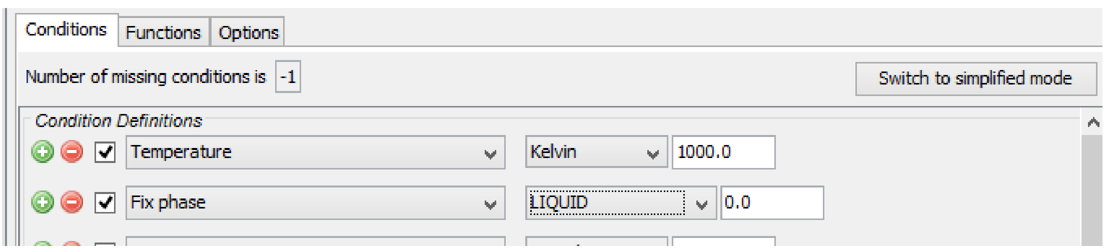
For a fixed phase equilibrium calculation, it is recommended to serially couple the Equilibrium Calculators. This is where one Equilibrium Calculator activity is the successor to another Equilibrium Calculator activity that performs an ordinary equilibrium calculation, and then, for the condition you are interested in, you make a rough estimate of the value. This gives the fixed phase calculation better starting values.

1. Right-click a node that can add an Equilibrium Calculator to your project. See *Creating Activities and Successors* on page 29 for a list.
2. In the **Project** window, click the **Equilibrium Calculator** activity node.

3. In the **Configuration** window, click **Switch to advanced mode**.

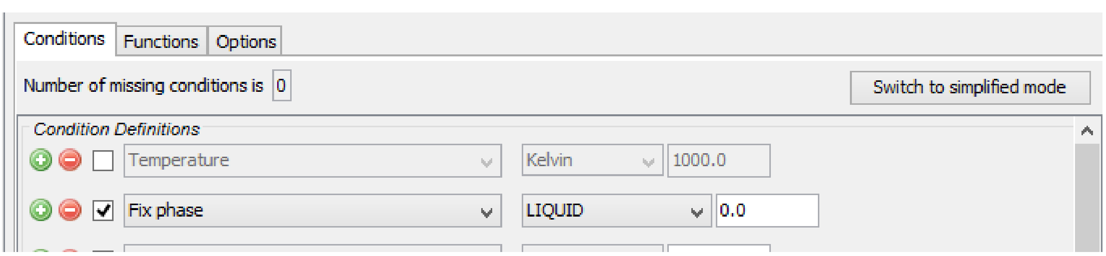
> If the **Default calculation mode** is globally set to **Advanced** then by default the Configuration window is already in advanced mode. See *Changing Global Settings* on page 187.

4. Under **Condition Definitions**, click the **Add quantity** button.
5. Select **Fix phase** from the first menu.
6. Select the phase that you want to fix in the second menu, and set the amount of that phase you want to fix the phase to. For example, if you want to know when your material starts to melt, set the phase to **LIQUID** and the value to **0.0**:



Above the condition definitions, the **Number of missing conditions is** field is probably **-1**. This means that you must remove one condition.

7. To remove a condition, either click the remove button, or click to clear the check-box. For example, if you want to know at what temperature your material starts to melt, then you cannot have a fixed temperature, in which case the temperature condition is not selected:



The **Number of missing conditions is** field displays **0**. When you have set all your conditions, you can perform the Equilibrium Calculator (now or later).

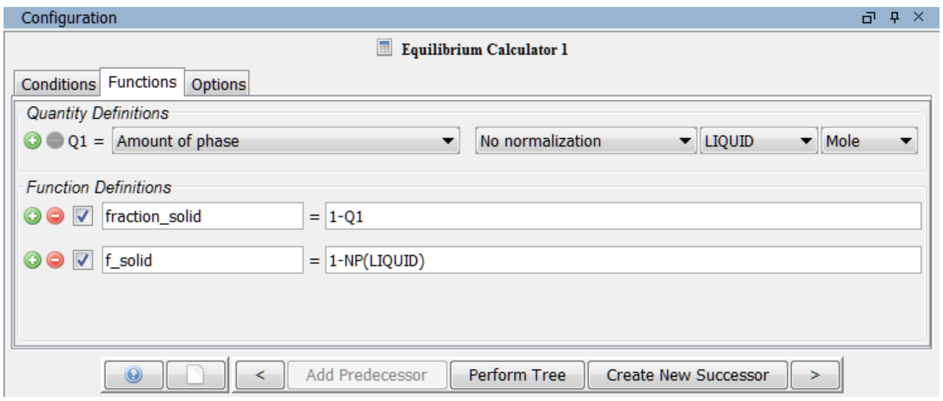
Calculating and Plotting Functions

The **Functions** tab allows you to define your own functions that can be plotted in a Plot Renderer.

For an example of how you can use functions that you yourself define, see example 7 in *Graphical Mode Example Collection* on page 9.

How to Calculate and Plot Functions

1. Create an **Equilibrium Calculator** node and click it in the **Project** window.
2. In the **Configuration** window click the **Functions** tab.
3. Under **Quantity Definitions**, define **Q1**. Click the add button to **Add a new quantity**. Then from the lists, choose the variable for each quantity to define. The quantities are called **Q1**, **Q2**, **Q3** and so on. In this example, the quantity **Q1** is defined as the **Amount of phase**, with **No normalization**, **LIQUID** in **Mole** units.

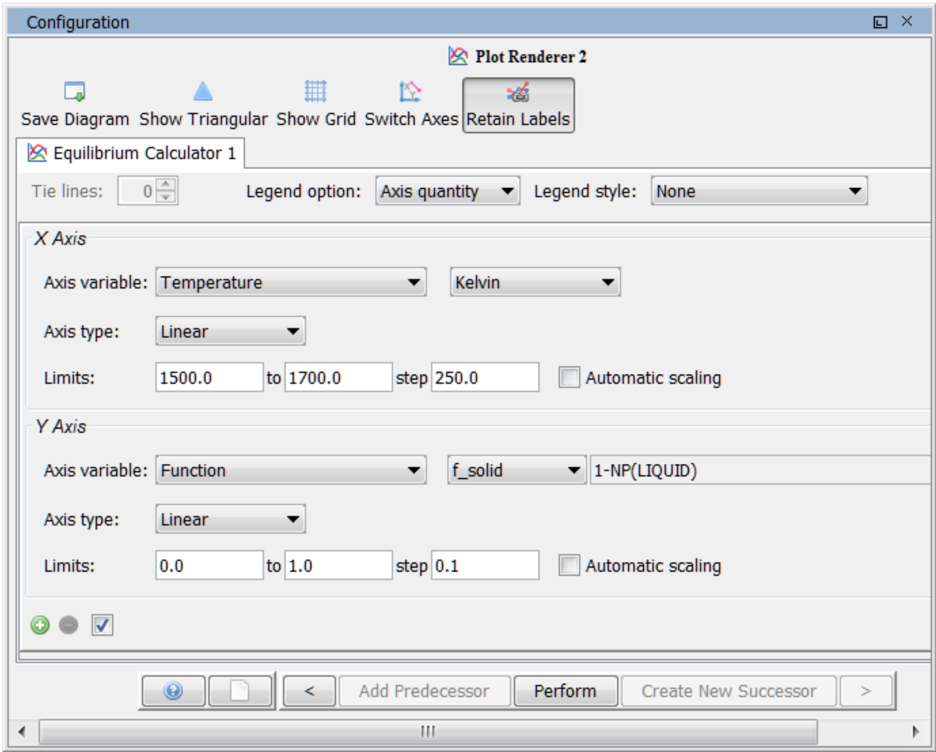


4. Under **Function Definitions**, click the add button to add a new function. In the left field edit the default name of each function and enter the function itself in the right field after the equal sign. In this example, the quantity defined as Q1 is renamed to **fraction_solid**, which gives as its output the fraction of solid phase (**1-Q1**).

You can also use Console Mode syntax when entering the function. For example, the function above can be entered as `1-NP(LIQUID)`.

5. To plot the defined function, right-click **Equilibrium Calculator** and choose **Create new successor→Plot Renderer**.
6. Click the **Plot Renderer** node. In the tab for the Equilibrium Calculator that contains the function, set an **Axis variable** to **Function**, then from the menu select the defined function. In this example the Y-axis is set to plot the value of the function **fraction_solid**.

   When run, the Plot Renderer plots the value of the function.



Precipitation Calculator

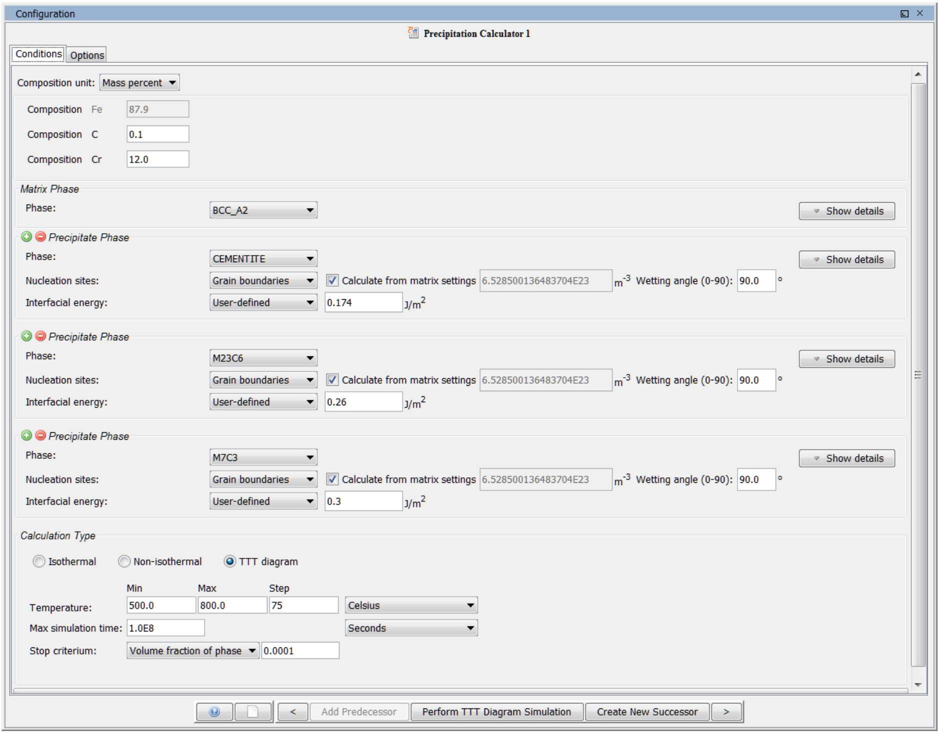
> The Precipitation Calculator is available with two components if you do not have the additional Precipitation Module (TC-PRISMA) license. With the add-on module you can use all available components.

A Precipitation Calculator allows you to set the conditions for, and perform, a precipitation calculation. The Configuration window for a Precipitation Calculator has these tabs:

- **Conditions**: Set the conditions for your calculation that define the Matrix and Precipitate phases. Choose the Calculation Type.
- **Options**: Modify Numerical Parameters that determine how the conditions are calculated. The Growth rate model can be viewed in Simplified or Advanced mode.

Configuration Window

Example



Conditions Tab Settings

These are the settings on the Precipitation Calculator Conditions tab.

Setting or Condition	Options
Composition unit	Choose from **Mass percent**, **Mole percent**, **Mass fraction** or **Mole fraction**.
Matrix Phase	Specify the **Phase**. Click **Show Details** for more settings.
Precipitate Phase	Specify the **Phase**, **Nucleation Sites**, and **Interfacial energy**. Click **Show Details** for more settings.
Calculation Type	Choose **Isothermal**, **Non-isothermal** or **TTT diagram**.

Matrix Phase

Phase: Choose a phase from the list. The list is based on the settings for the System Definer. When setting up your system, choose a matrix phase with kinetic data available in the database.

Only phases with kinetic data can be selected as the matrix phase. If the list is empty, go to the System Definer to make sure that both types of databases are selected and defined. See "System Definer" in the *Thermo-Calc User Guide* or search the Online Help.

Click **Show details** to view these additional settings:

- **Molar volume**: Choose the default to take the value from the **Database** or **User-defined** to enter another value. **Database** is available in the list when the molar volume for the phase is defined in the thermodynamic database.
- **Grain radius**: Enter a numerical value and choose a unit from the list.
- **Grain aspect ratio**: Enter a numerical value.
- **Dislocation density**: Enter a numerical value.
- **Mobility enhancement prefactor**: Enter a numerical value.
- **Mobility enhancement activation energy**: Enter a numerical value.

Precipitate Phase

Phase: Choose a phase from the list. The list is based on the settings for the System Definer.

The phases available to choose have both thermodynamic and kinetic data. If the list is empty, go to the System Definer to make sure that both types of databases are selected and defined. See "System Definer" in the *Thermo-Calc User Guide* or search the Online Help.

Nucleation sites: Choose one of the following from the list.

- **Bulk**, **Grain boundaries**, **Grain edges**, **Grain corners**, or **Dislocations**. Click to select the **Calculate from matrix settings** check box if you want to calculate the number density of sites.
- For Grain boundaries, Grain edges and Grain corners, enter the **Wetting angle** in addition to the matrix settings.
- To enter a specific value for the number of nucleation sites, deselect the check box.

Interfacial energy: Choose **Calculated** to use the estimated value and then enter a different **prefactor** value if you want to adjust the estimated value. You can also choose **User defined** to enter a value in J/m^2.

Click **Show details** to view these additional settings:

- **Molar volume**: Choose the default to take the value from the **Database** or **User-defined** to enter another value. **Database** is available in the list when the molar volume for the phase is defined in the thermodynamic database.
- **Phase boundary mobility**: Enter a numerical value.
- **Phase energy addition**: Enter a numerical value.

- **Approximate driving force**: Select the check box to include this if simulations with several compositions sets of the same phase create problems.

Calculation Type

Isothermal

Use an **Isothermal** calculation type to do a precipitation simulation at constant temperature.

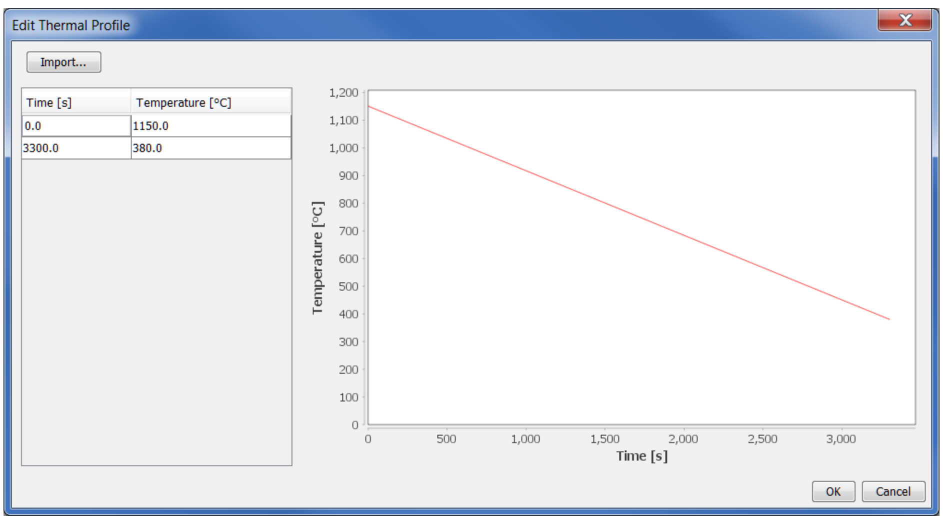
For the **Isothermal** calculation type, enter a **Temperature** and **Simulation time**.

Non-Isothermal

For the **Non-isothermal** calculation type, select a **Temperature unit** and **Time unit** from the lists. Enter a value for the **Simulation time**.

Click **Thermal Profile** to define the heat treatment schedule. Here the **Temperature** and **Time** coordinates of thermal profile points are entered. A minimum of two points is required. You can also click **Import** to add your own thermal profile from an Excel spreadsheet.

Edit Thermal Profile Window



TTT-Diagram

Use a **TTT-diagram** calculation type to do a a precipitation simulation of the time-temperature-transformation (TTT).

For the **TTT diagram** calculation type , enter **Min**, **Max**, and **Step** values for the **Temperature** and choose a **Unit** from the list.

Max simulation time: Enter a numerical value and choose a **Unit** from the list.

Stop criterium: Choose **Volume fraction of phase** or **% of equilibrium fraction** and then enter a numerical value in the field. For each temperature, the simulation stops when the stop criterium is fulfilled or if the maximum simulation time is reached, whichever happens first.

Options Tab Settings

These are the settings on the Precipitation Calculator Options tab.

Settings

Growth Rate Model

Choose from **Simplified** (the default) or **Advanced**.

Numerical Parameters

Enter numerical values for the following as required.

- **Max time step fraction**: The maximum time step allowed for time integration as fraction of the simulation time.
- **No. of grid points over one order of magnitude in radius**: Default number of grid points for every order of magnitude in size space.
- **Max no. of grid points over one order of magnitude in radius**: The maximum allowed number of grid points in size space.
- **Min no. of grid points over one order of magnitude in radius**: The minimum allowed number of grid points in size space.
- **Max relative volume fraction of subcritical particles allowed to dissolve in one time step**: The portion of the volume fraction that can be ignored when determining the time step.
- **Max relative radius change**: The maximum value allowed for relative radius change in one time step.
- **Relative radius change for avoiding class collision**: Set a limit on the time step.
- **Max overall volume change**: This defines the maximum absolute (not ratio) change of the volume fraction allowed during one time step.
- **Max relative change of nucleation rate in logarithmic scale**: This parameter ensures accuracy for the evolution of effective nucleation rate.
- **Max relative change of critical radius**: Used to place a constraint on how fast the critical radium can vary, and thus put a limit on time step
- **Min radius for a nucleus to be considered as a particle**: The cut-off lower limit of precipitate radius.
- **Max time step during heating stages**: The upper limit of the time step that has been enforced in the heating stages.

Plot Renderer

The Plot Renderer displays results from the various calculator activities .

The available Plot Renderer functionality varies by calculation type. For example:

- If you choose a *Property grid* (Equilibrium Calculator) or *Grid* (Property Model Calculator) calculation type, select a **Plot type** and whether to **Interpolate** the data or not.
- For a Property Model Calculator *Uncertainty* calculation type, you choose an **Axis type**: **Frequency**, **Histogram** or **Normal probability**. For other calculators (e.g. Equilibrium, Binary, and Scheil) there are different axis types available such as **Linear**, **Logarithmic**, **Logarithmic 10**, and **Inverse**.

The Plot Renderer is used to set the plotting axis variables for a property diagram and phase diagram.

If you are plotting a Property grid (Equilibrium Calculator) or Grid (Property Model Calculator) then the X- and Y-axes always represent the quantities that define the stepping axes, but you need to set a variable for the Z-axis (the axis perpendicular to the plane defined by the X- and Y-axes).

Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity.

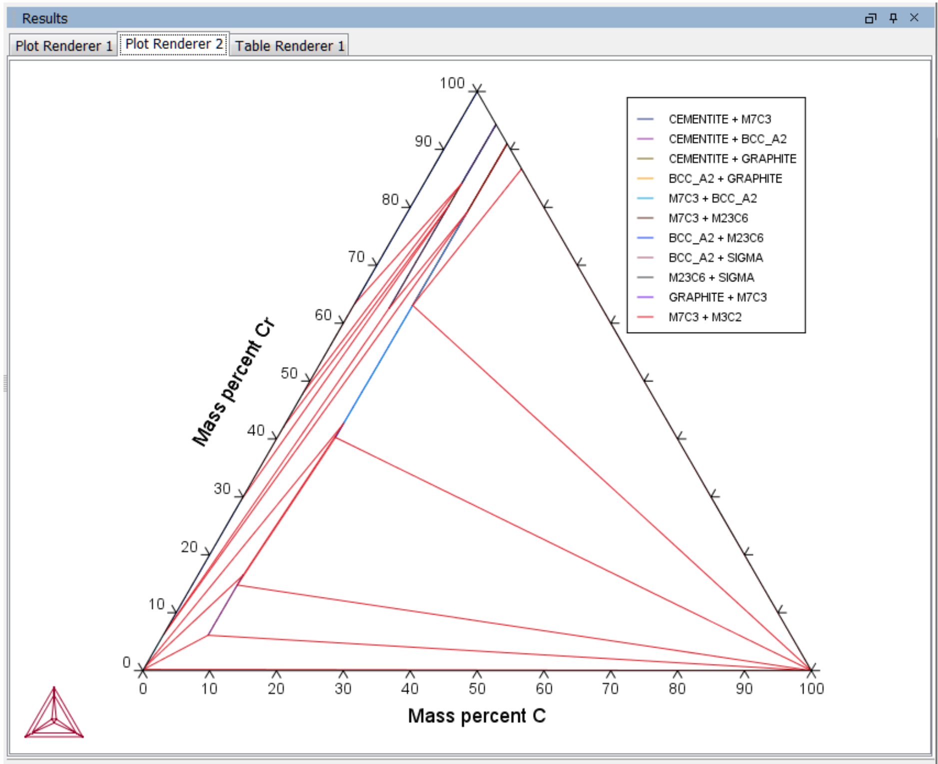
The configurations that can be set in the Configuration window for a Plot Renderer vary depending on the kind of predecessor it is created from.

Plot Several Calculations in One Diagram

A Plot Renderer can have several predecessors (if you want to plot several calculations in the same diagram), in which case there are several tabs in the Configuration window, one for each predecessor.

Several Plot Renderer activities (as well as Table Renderer activities) can be shown in the Results window as different tabs. The figure is from Graphical Mode example 4 .

Also see *Plotting Several Calculations in One Diagram* on page 96 for details.

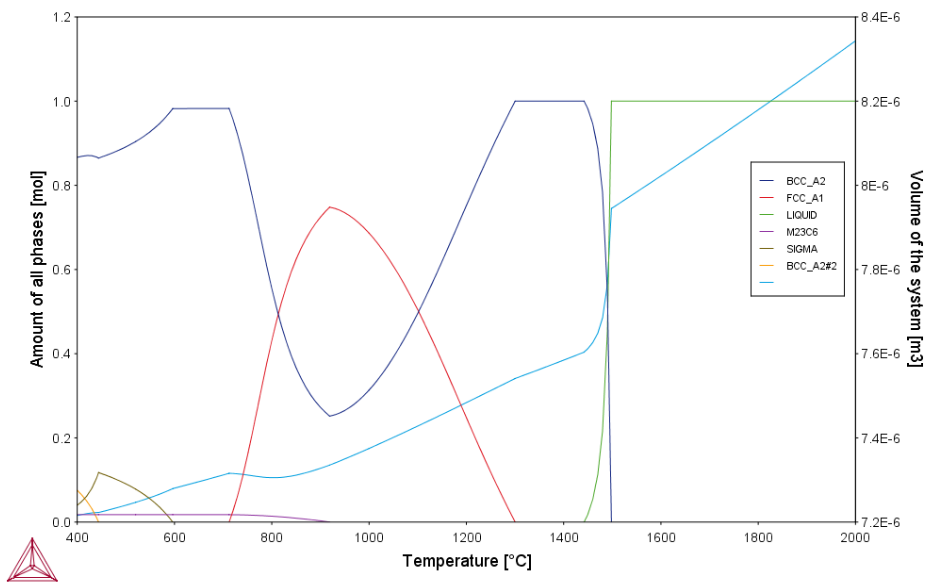


Add Axes to Diagrams

You can add additional X- and/or Y-axes to a plot to compare two or more variables at the same time as in the figure.

Also see *Adding Axes to a Diagram* on page 97 for details.

Two variables plotted on the Y-axis.



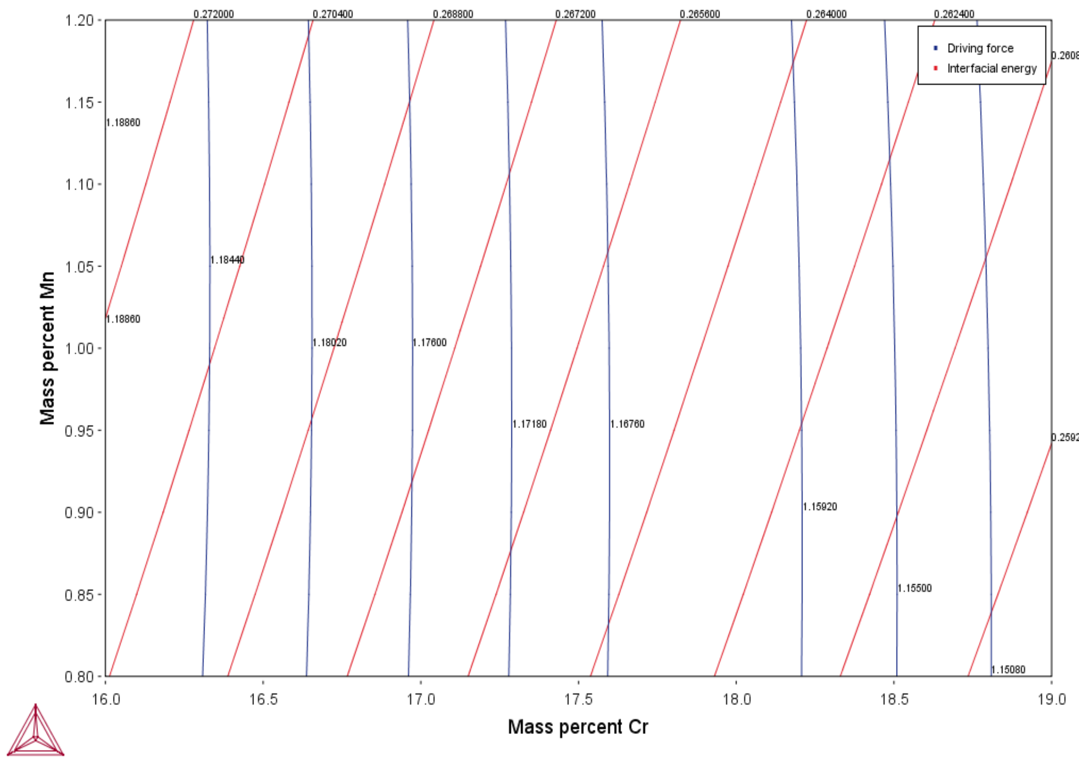
Merge Results in One Diagram

You can merge (or overlay) calculated results for several variables into one diagram. This is similar to adding an axis, but instead of displaying different variables along the X- and/or Y-axes, the same X and Y variables are used to compare the third variable (defined in the Plot Renderer as the Z-axis).

As in the figure (from example 12 in Graphical Mode Examples), the driving force and interfacial energy results are combined into one diagram. The X and Y axis is the same for both (Mass percent Cr/Mass percent Mn).

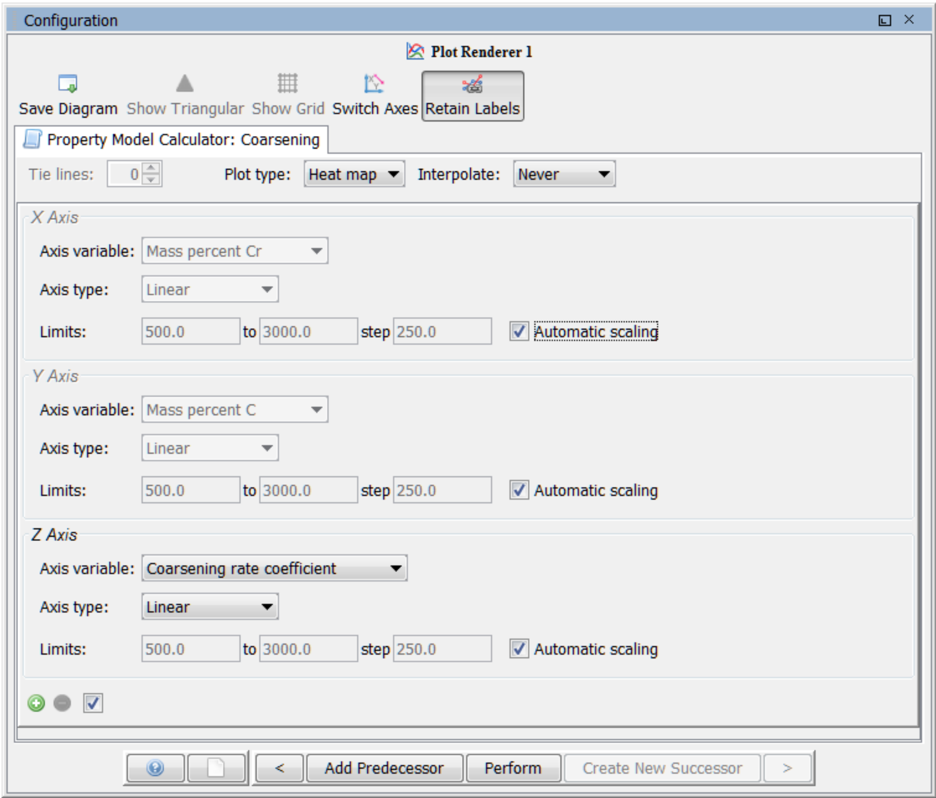
Also see *Merging Plots into One Diagram* on page 99 for details of how to combine diagrams.

A contour plot showing both the driving force and interfacial energy calculation results comparing Mass percent Cr to Mass percent Mn.



Configuration Window

The Configuration window has different options available based on its predecessor. The following is an example of the Configuration window for a Plot Renderer that is the successor to an Equilibrium Calculator. The options along the top are available for most of the calculation types (e.g. phase diagrams, property diagrams, min/max, uncertainty and so forth). For an Equilibrium Property Grid calculation type and a Property Model Calculator Grid calculation you can choose a **Plot type**.



Configuration Settings

Setting/Button	Description
Save Diagram	Save the plot diagram.
Show Triangular	Display the diagram in triangular form, with the X-axis along the base and the Y-axis along the left side. Such a diagram is often useful if you want to plot the fractions of two components in a ternary diagram.
Show Grid	Overlay a grid on the diagram.
Switch Axes	Show the plotting X axis variable on the diagram's Y-axis, and the Y axis variable on the diagram's X-axis.
Retain Labels	Toggle the default to keep labels displaying on the plot. By default labels are retained.

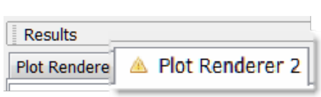
Setting/Button	Description
Tie lines	Select how many tie lines to be plotted in the diagram. Every nth number of tie line is plotted, where n is the number set here. Consequently, the higher the number, the fewer the number of plotted tie lines in the diagram. This setting is not available if the Plot Renderer is a successor to a Precipitation Calculator, Scheil Calculator or to an Equilibrium Calculator where only one axis is defined and varied (that is, an Equilibrium Calculator that has performed a stepping calculation).
Legend option	Select whether the diagram's legend displays the Stable phases, the Axis quantity, or choose Off for no legend.
Legend style	You can select a Legend style for the **Stable phases** or **Axis quantity** selected for **Legend option**. Choose **None**, **All**, **Constitution description**, or **Ordering description**. Also see *About Legend Styles* on page 91.
Plot type (property grid or grid)	This menu is available if the Plot Renderer is a successor to either an Equilibrium Calculator that is set up to calculate a *Property grid* or a Property Model Calculator that is calculating a *Grid*. On a **Heat map** diagram, each equilibrium calculation in the grid is represented by a colour-coded pixel, where dark red represents the highest Z-axis variable value and dark blue represents the lowest Z-axis variable value. On a **Contour** diagram, Z-axis variable values are traced with curves in different colours (in the same way that height values are traced with curves on a topographical map). From the **Interpolate** list, choose to interpolate data **Never** (the default), **Once**, **Twice** or **3 times**. This means points are put in between calculated points in the grid and interpolated instead of calculated in the calculation engine.
Interpolate (property grid or grid)	Choose how many points of data to interpolate on a grid or property grid to improve curve smoothness – **Never**, **20x20**, **50x50**, or **100x100**. Try different settings for your own set of data.

Setting/Button	Description
Axis variable	Set the state variable you want plotted along the X-axis and the Y-axis. If the stepping/mapping variables are changed in the Equilibrium Calculator that precedes the Plot Renderer, then the variables to be plotted along the diagram’s X- and Y-axis are automatically updated to the appropriate quantities. For a Precipitation Calculator: • Select the **Separate multimodal PSD** check box to enter settings for **Points**, **Excess kurtosis**, **Valley depth ratio** and **Minimum peak**.
Axis type	Select what type of scale to use on the axis. Choose between **Linear**, **Logarithmic**, **Logarithmic 10** or **Inverse**.
Limits	Specify the range along the axis that is shown in the plot by setting the minimum and maximum values of the axis variable. You can also determine the step size between the tick marks along each axis. Alternatively, select Automatic scaling.
Unit (*Time X Axis*)	For a Precipitation Calculator choose a **Unit** - **Seconds**, **Hours**, or **Days**.
Add an axis and Remove this axis buttons.	Use these buttons to add additional X- and/or Y-axes to a plot or to merge plots (to overlay one plot with another in the same diagram). When merging plots, you define this variable as the Z-axis. See Adding Axes to a Diagram and Merging Plots into One Diagram for details.

Plot Status Marker

If you have plotted a diagram and then modify something in the tree that the Plot Renderer is associated with, then a notification sign—an exclamation mark enclosed in a yellow triangle—is shown in the Plot Renderer’s tab.

Also see *Node Status Markers* on page 40.



About Legend Styles

There are a variety of legend styles you can use to display detail about the phases.

The same legend styles are available for both Graphical Mode and Console Mode plots. For Console Mode, use the command CHANGE_LEGEND in the *Thermo-Calc Console Mode Command Reference Guide*.

In Graphical Mode the label styles are defined when using the *Plot Renderer* on page 83. You can select a **Legend style** for the **Stable phases** or **Axis quantity** selected for **Legend option**.

When there is a legend in the diagram you can append constitution and/or ordering descriptions to the phase name.

If constituent description is included in the legend, then for each equilibria along a line in the diagram, the largest constituent fractions are summed until they are equal or greater than 0.6. This is done separately for each sublattice. If the equilibria along the line gets different sets of constituents these are all appended to the list for that sublattice.

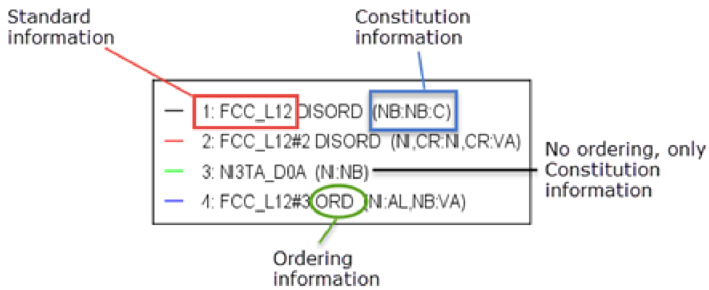
The ordering description is only printed for partitioned phases (phases that have a disordered contribution) that begins with at least two identical sublattices (sublattices where the set of constituents are identical). `DISORD` (disordered) is shown if site fractions are identical between identical constituents on all identical sub lattices (with an error margin) otherwise `ORD` (ordered) is shown in the legend.

If ordering between equilibria in a line are different, then `ORD + DISORD` is shown in the legend.

Legend Examples

To show the differences between labels, the following uses the same example file that creates Figure 6 in TCNi8 database extended information document (found on the Thermo-Calc website).

When **All** is selected for this specific example the label includes the following:



Specifically:

- There are two disordered FCC phases (matrix and carbide), one ordered FCC (gamma prime), plus a phase NI3TA_D0A (called Delta in Ni-alloys).
- ORDERING shows the different FCC phases.

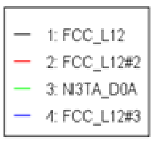
- CONSTITUTION shows that FCC_L12#1 is NbC, FCC_L12#2 is matrix with Ni and Cr as major constituent, FCC_L12#3 is Ni3Al with Nb addition and NI3TA_D0A is Nb-rich (eg. Ni3Nb).

Label Style Choice Examples

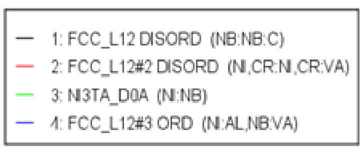
These examples are for each label style choice - None, All, Constitution or Ordering

The instructions are for when you are setting the options using the Plot Renderer. In Console Mode, you choose the options (`ALL`, `NONE`, `CONSTITUTION`, or `ORDERING`) after the command CHANGE_LEGEND.

When **None** is selected the legend includes the standard information about the phases.

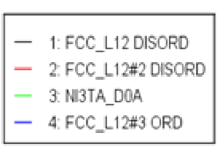


When you select **All** the **Constitution description** and the **Ordering description** (when there is ordering) are included.



When you select **Constitution description**, this is added to the standard description.

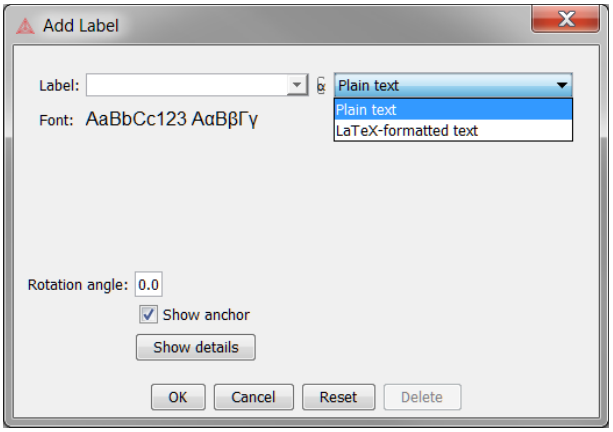
When you select **Ordering description** and when there is ordering, this is added to the standard description.



Adding and Editing Plot Labels

To add a label text at a certain point in a plot, right click the point, and select **Add Label**. For phase diagrams, the program suggests a label that you can keep or edit. The suggested label specifies the names of the stable phases at the coordinate where you clicked. For all other plot types, you enter your own text, which can either be Plain text or LaTeX formatted text.

Add Label WIndow



Label Formatting

You can enter the label as **Plain text** or in **LaTeX-formatted text**. For **LaTeX-formatted text** enter or edit the text as required. For example, underscores (_) are added before the numbers to make subscripts in the label.

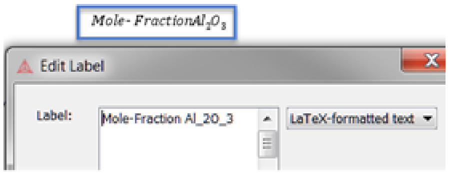
For a list of supported LaTeX commands, see: http://www2.ph.ed.ac.uk/snuggletex/documentation/math-mode.html.

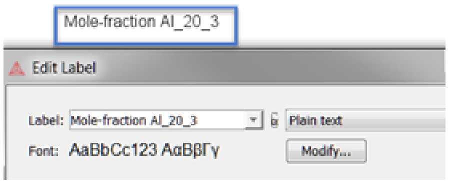
By default when using LaTeX, you are in math mode and Thermo-Calc automatically interprets the entered text as math. The above link uses the dollar sign ($) in all the examples but this is NOT required. For example, if you enter `$E=mc^2$` this is displayed including the $ instead of the equation.

Entering a Mixture of Plain Text and Equations

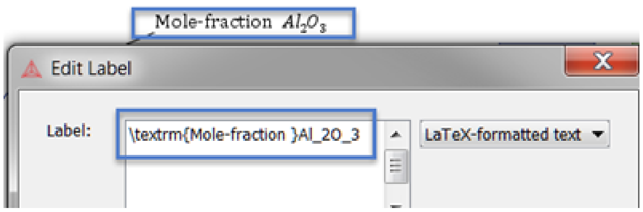
When you enter text using LaTaX formatted text it displays by default on the plot in italics and treats the text as an equation.

As in the following example, the top label does not display correctly because the label is a mixture of text (Mole-Fraction) and an equation (Al_2O_3). In the bottom image, when the same information is added as Plain text to the plot it now does not include the subscripts.





To add a mixture of plain text and equation text, you can exit math mode using the string `\textrm` followed by the text to display enclosed in curly brackets. For example `\textrm{Mole-fraction }Al_20_3` results in the label as shown:



In this example, you need to add a space at the end of fraction in order to separate the text from the equation part of the label.

When working in Console Mode, you can also use the above windows on the Console Results window to add and edit labels. In addition, you can use the ADD_LABEL_TEXT command described in the *Thermo-Calc Console Mode Command Reference*.

Rotation Angle and Show Anchor

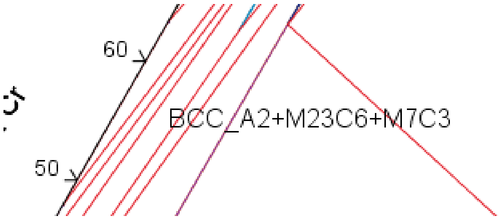
By default the **Show anchor** check box is selected, which includes an arrow from the label point to the coordinates.

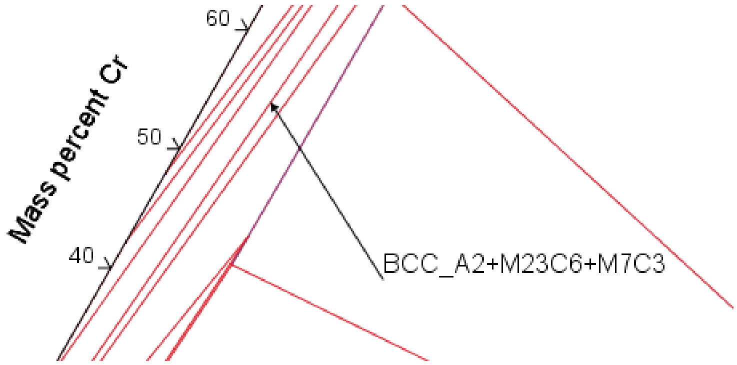
If you do not want the label oriented horizontally in the Plot Renderer tab, then specify the number of degrees of rotation in the **Rotation angle** field.

Click **OK** to add the label to the plot. You can also move a label by clicking and dragging it. The default anchor arrow is added to indicate the point that belongs to the label.

Moving Labels in a Plot

Drag and drop labels anywhere on the plot. The anchor arrow points to the point of origin.



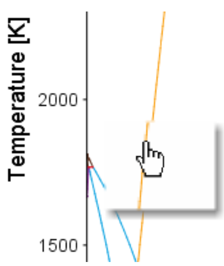


Changing the Plot Properties

For global changes to the default settings plots see *Global Settings: Graphical and Console Mode-Plotting* on page 200. The settings are the same but applied to all new plots.

To make local changes to the appearance of a specific plot, in the **Results** window right-click a plot and select **Properties**. You can also edit some properties for individual plot lines (the color, the line width and type, and whether data points are included).

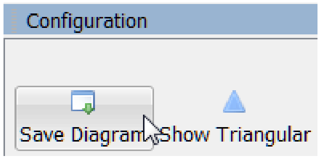
In the **Results** window, hover the mouse over a plot line. When it is over a line that can clicked the crosshair cursor turns into a pointing finger.



To prevent unintended edits, hold down <Ctrl> while moving the cursor around the plot to only display it as a crosshair.

Saving Plots

1. To save a diagram, at the top of the **Configuration** window click **Save Diagram**.



Or right-click the diagram in the **Results** window and select **Save As**.

2. In the **Save** window, navigate to where you want to save the plot, enter a **File name**, and choose the **File type** to save it in. Click **Save**.

Plotting Several Calculations in One Diagram

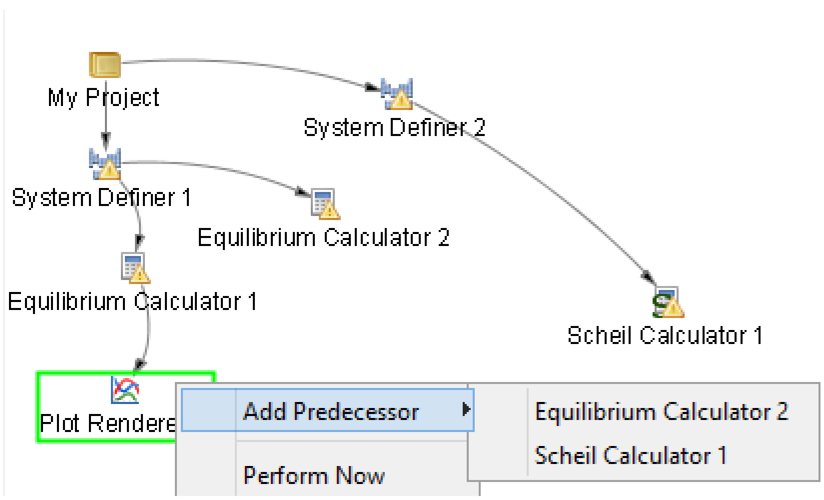
You can plot several calculations in the same diagram by adding extra predecessors to the Plot Renderer activity.

For an example of a project in which several calculations are plotted in one diagram, see example 5 in *Graphical Mode Example Collection* on page 9.

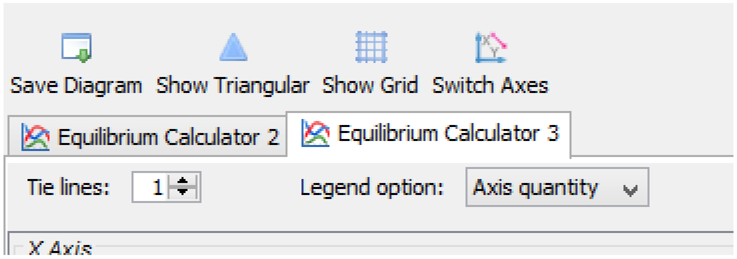
To plot several calculations in one diagram, you must have, in addition to the Plot Renderer's original predecessor, at least one available Plot Renderer predecessor in your project. The extra predecessor can be any of these: Equilibrium Calculator, Binary Calculator, Ternary Calculator, Scheil Calculator or Experimental File Reader.

How to Plot Several Calculations in One Diagram

1. In the Project window click the **Plot Renderer** node.
2. In the **Configuration** window click **Add Predecessor**. Or right-click the **Plot Renderer** and choose **Add Predecessor**.

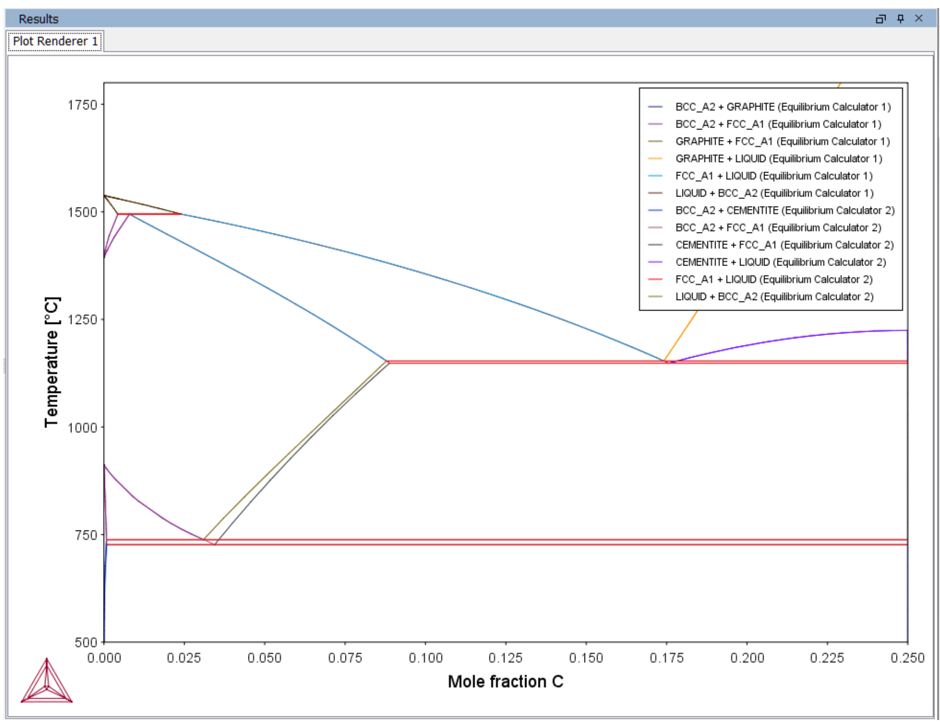


3. Select a predecessor. When added, the extra predecessor is linked to the Plot Renderer. On the Plot Renderer's Configuration window, the predecessors are shown as two tabs:



4. Configure how you want each calculation plotted on the tabs in the **Configuration** window, then click to **Perform** the Plot Renderer activity.

   The calculations are plotted in the same diagram in a tab in the **Results** window. If any of the axis variables of the plots differ, then the label and scale of all calculations are shown parallel to each other. The following screen shot shows a diagram in which two calculations with different plotting axes variables have been plotted.



Adding Axes to a Diagram

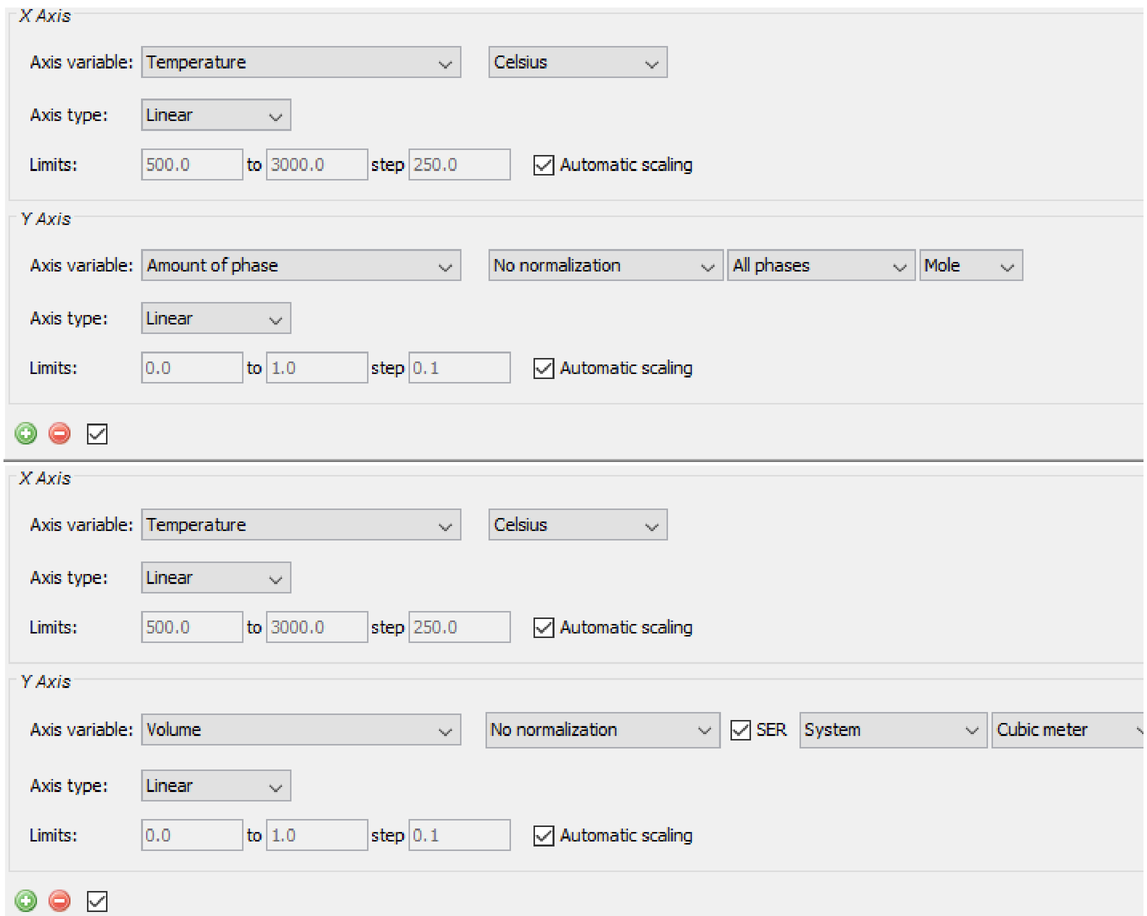
You can add additional *X*- and/or *Y*-axes to a plot to compare two or more variables at the same time.

1. In the **Project** window right-click the calculator node (e.g. an Equilibrium Calculator) and add a **Plot Renderer** node from the **Create New Successor** menu.
2. Click the **Plot Renderer** node.
3. In the **Configuration** window, define the plot. Available settings are based on the calculation type.
4. Click the **Add an axis** button. A new settings section is added underneath.
5. From the *X Axis* and/or *Y Axis* sections define another **Axis variable** and choose an **Axis type**.

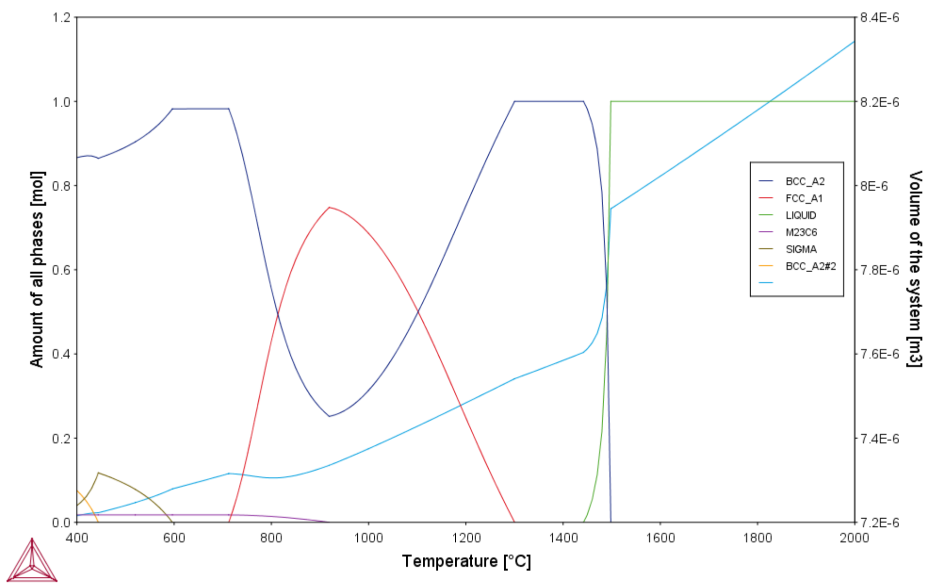
6. Click **Perform** to plot both variables on the selected axis, in this example, the Y-axis.

Axis Example

In this example **Amount of phase** and **Volume** are selected as the Y-axis variables and **Temperature** as the X-axis variable.



Plot Example



Merging Plots into One Diagram

You can merge (or overlay) calculated results for several variables into one diagram. The same X and Y variables are used to compare the third variable (defined in the Plot Renderer as the Z-axis).

You cannot overlay heat map or frequency plots.

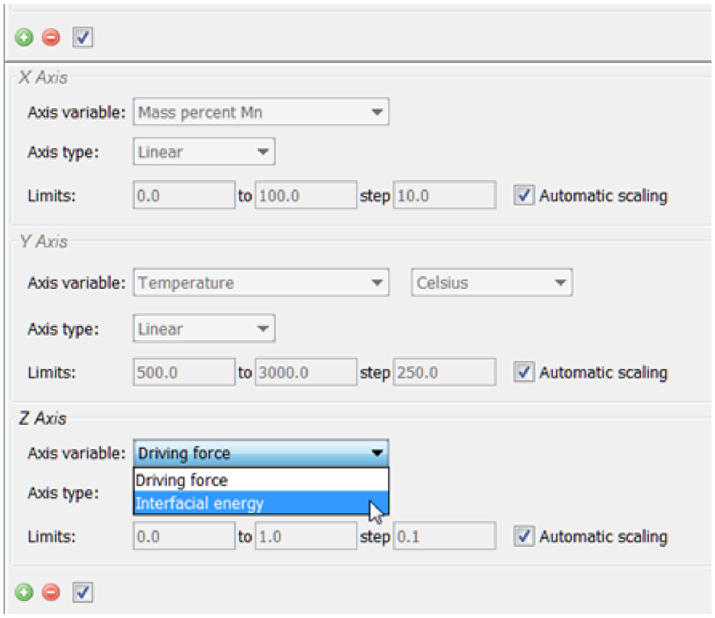
The following uses example 12 in Graphical Mode Examples to describe the steps to overlay two plots: the driving force and interfacial energy results are combined into one diagram. The X and Y axis is the same for both (Mass percent Cr/Mass percent Mn).

1. In the **Project** window right-click the calculator node (e.g. Property Model Calculator) and add a **Plot Renderer** node from the **Create New Successor** menu.
2. Click the **Plot Renderer** node.
3. In the **Configuration** window, define the plot. Available settings are based on the calculation type.

   Example

   In this example the Property Model Calculator is set to a *Grid* calculation type.

   Choose **On** for the **Legend option** and **Contour** as the **Plot type**.



4. Click the **Add an axis** button. A new axes settings section is added underneath.
5. From the *Z Axis* section choose the **Axis variable**. In this example, **Interfacial energy**.
6. Click **Perform** to plot both on the same diagram.

Plot Example

In this case, a contour plot showing both the driving force and interfacial energy calculation results comparing Mass percent Cr to Mass percent Mn.

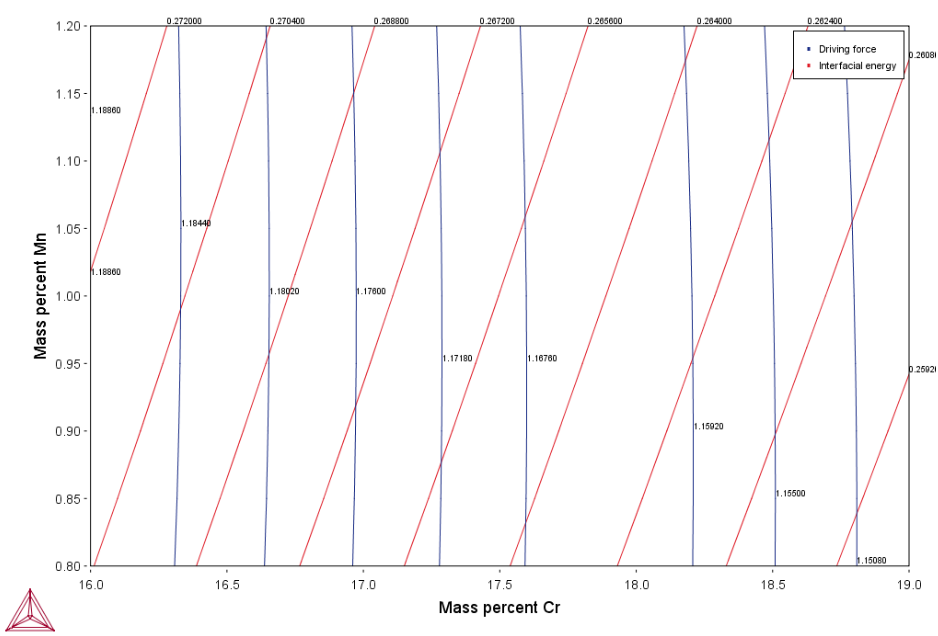


Table Renderer

A Table Renderer visualizes calculation results in a table. Several Table Renderer activities (as well as Plot Renderer activities) can be shown in the Results window as different tabs.

You can only successfully tabulate the results of a single-equilibrium calculation or a stepping operation. The tabulated data from an equilibrium calculation is different from the tabulated data from a stepping calculation.

Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity.

Tabulated Results from an Equilibrium Calculation

The following tab shows the result of an equilibrium calculation (from example 1 in *Graphical Mode Example Collection* on page 9).

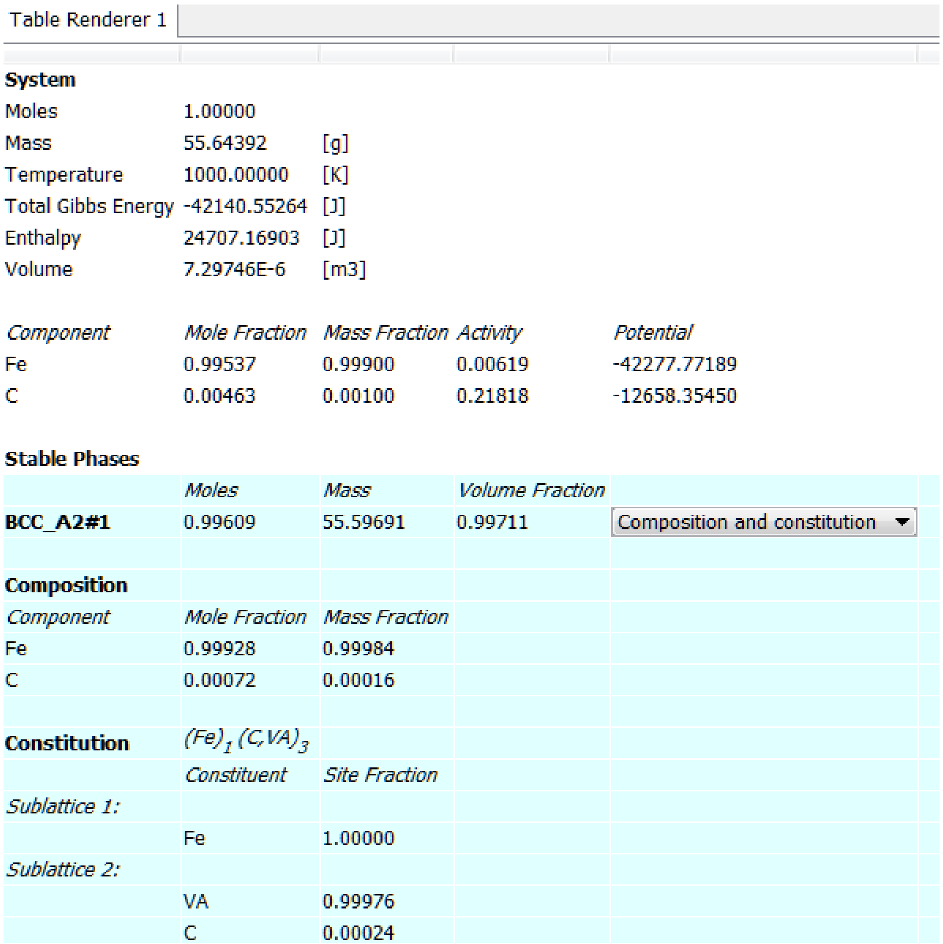


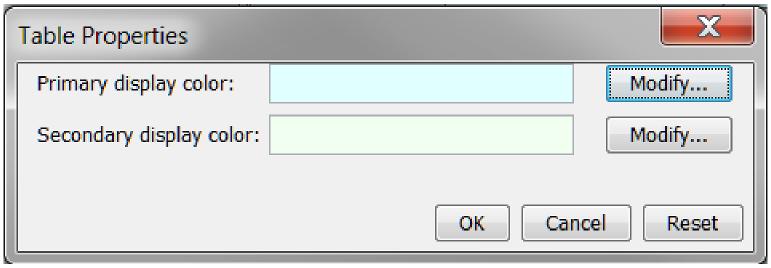
Table Renderer 1

System				
Moles	1.00000			
Mass	55.64392	[g]		
Temperature	1000.00000	[K]		
Total Gibbs Energy	-42140.55264	[J]		
Enthalpy	24707.16903	[J]		
Volume	7.29746E-6	[m3]		
Component	*Mole Fraction*	*Mass Fraction*	*Activity*	*Potential*
Fe	0.99537	0.99900	0.00619	-42277.77189
C	0.00463	0.00100	0.21818	-12658.35450
Stable Phases				
	Moles	*Mass*	*Volume Fraction*	
BCC_A2#1	0.99609	55.59691	0.99711	Composition and constitution
Composition				
Component	*Mole Fraction*	*Mass Fraction*		
Fe	0.99928	0.99984		
C	0.00072	0.00016		
Constitution	*$(Fe)_1 (C,VA)_3$*			
	Constituent	*Site Fraction*		
Sublattice 1:				
	Fe	1.00000		
Sublattice 2:				
	VA	0.99976		
	C	0.00024		

For each stable phase listed, from the **Composition and constitution** menu you can select the information to view. In this example, the maximum amount of information is shown and includes details about both composition and constitution for **BCC_A2#1**.

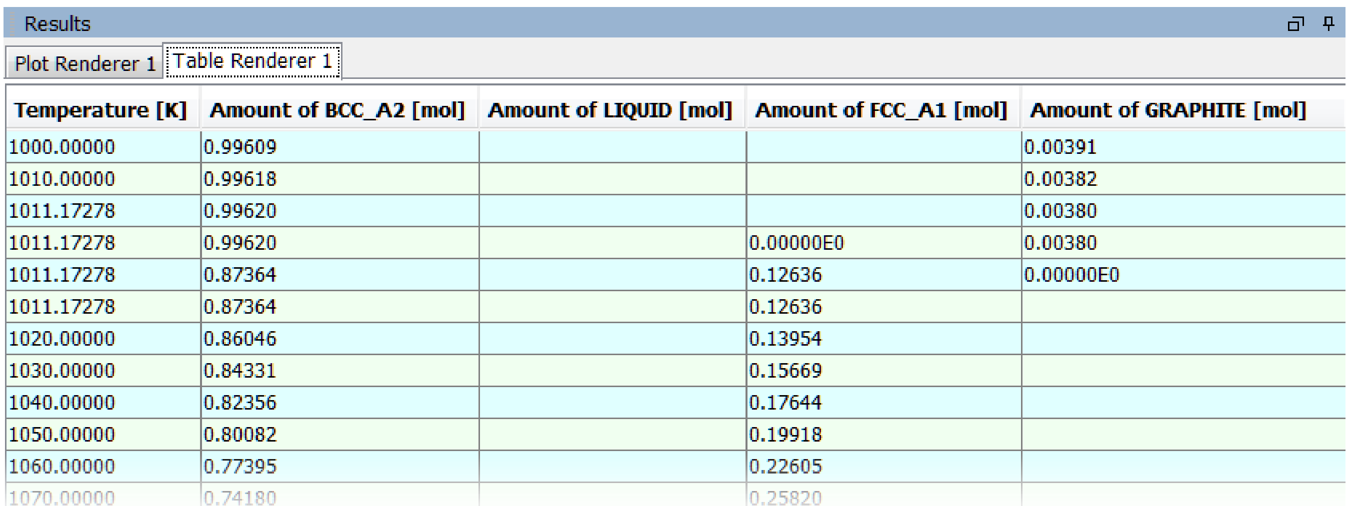
Setting Background Colours

To change the background colours of the table, in the **Results** window right-click a Table Renderer tab, and select **Properties**. Modify as required.



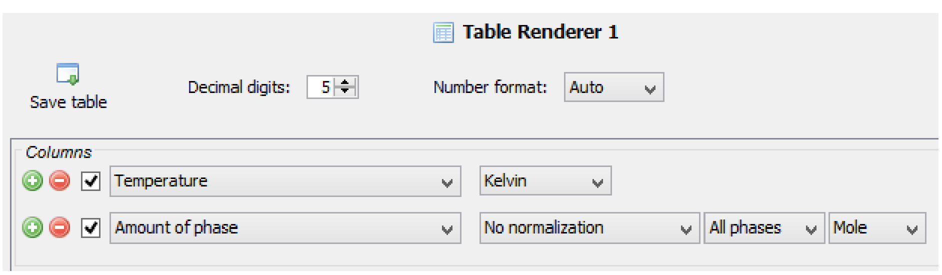
Tabulated Results from a Stepping Operation

The **Table Render 1** tab in the example shows the results of a stepping calculation (from example 2 in *Graphical Mode Example Collection* on page 9).



Temperature [K]	Amount of BCC_A2 [mol]	Amount of LIQUID [mol]	Amount of FCC_A1 [mol]	Amount of GRAPHITE [mol]
1000.00000	0.99609			0.00391
1010.00000	0.99618			0.00382
1011.17278	0.99620			0.00380
1011.17278	0.99620		0.00000E0	0.00380
1011.17278	0.87364		0.12636	0.00000E0
1011.17278	0.87364		0.12636	
1020.00000	0.86046		0.13954	
1030.00000	0.84331		0.15669	
1040.00000	0.82356		0.17644	
1050.00000	0.80082		0.19918	
1060.00000	0.77395		0.22605	
1070.00000	0.74180		0.25820	

Configuration Window Settings



Defining the Table Columns

In the **Configuration** window for the Table Renderer 1 node, set the level of detail for the Columns:

Setting Number Format and Number of Decimal Digits

The table results can be presented in ordinary decimal number format or with normalized scientific notion. The **Number format** menu defaults to **Auto** to use the most appropriate format based on the tabulated data. Or choose **Scientific** or **Decimal** to increase or decrease the number of **Decimal digits** used in the table.

Saving Tabulated Data

To save all the data in a table as a plain text file, in the **Configuration** window click **Save table** or in the **Results** window right-click in the table and select Save As.

To copy the data from a single cell of a table to the clipboard, right-click the cell and select **Copy**. To copy the whole table to the clipboard, select **Copy all**.

The data saved is what is shown in the **Table Renderer** tab. If data is calculated but not displayed it is not saved.

Experimental File Reader

The Experimental File Reader activity can only be created from the top My Project node. This activity allows you to read an experimental data file (*.EXP). This type of file has information specifying a plotted diagram, written in the DATAPLOT graphical language.

Also see *Creating Activities and Successors* on page 29 and the *DATPLOT User Guide* included with this documentation set.

How to Plot an Experimental Data File

1. Right-click **My Project** and from the **Create New Activity** submenu choose **Experimental File Reader**.
2. In the **Configuration** window in the **EXP file** field, enter the name of the EXP-file to plot.
3. Click **Create New Successor** and select **Plot Renderer**. Or right-click the **Experimental File Reader** node and from the **Create New Successor** submenu choose **Plot Renderer**.
4. In the **Configuration** window click **Perform** or right-click the **Plot Renderer** node and select **Perform**.

Binary Calculator

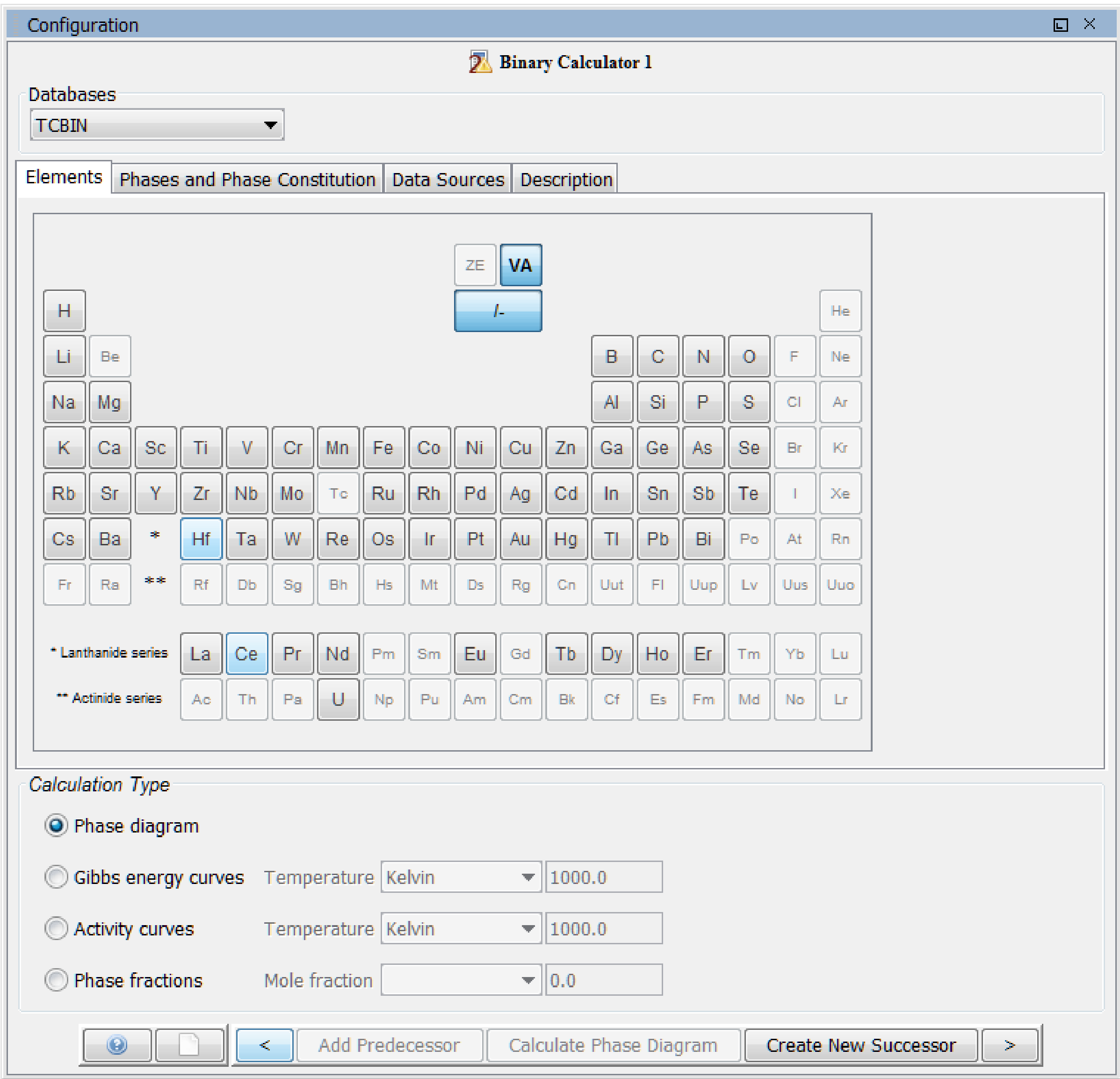
A Binary Calculator can be used for some calculations involving two components. You can think of this activity as a combination of a System Definer and an Equilibrium Calculator, but designed to simplify setting up and performing calculations on binary systems.

The Binary Calculator relies on some specifications that are not supported by all databases. You need a specially designed database for the Binary Calculator such as the TCBIN database.

For an example of how a Binary Calculator can be used in a project, see example 3 in *Graphical Mode Example Collection* on page 9.

Configuration Window

The following shows the **Configuration** window of a **Binary Calculator** activity.



Configuration Settings

The table lists the available settings to configure before performing the calculation.

Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity.

Setting	Description
Select database	Select a database that contains specifications dedicated to the Binary Calculator.
Elements (periodic table)	In the periodic table, select the two elements that make up the components of the system.
Phases and Phase Constitution	This tab is the same as for a *System Definer* on page 54. See *Phases and Phase Constitution* for an example.
Data Sources	This tab is the same as for a *System Definer* on page 54. See *Data Sources* for an example.
Description	This tab is the same as for a *System Definer* on page 54. See *Description* for an example.
Calculation Type	
Phase diagram	Select to perform a mapping calculation. Click **Calculate Phase Diagram** to plot the results.
Gibbs energy curves	Select the **Temperature** and click **Calculate Gibbs Energy Curves** to calculate at a constant temperature over the whole composition range.
Activity curves	Select the **Temperature** and click **Calculate Activity Curves** to calculate at a constant temperature over the whole composition range.
Phase fractions	Select the **Mole fraction** and click **Calculate Phase Fractions** to calculate as a function of temperature at a constant composition.

Ternary Calculator

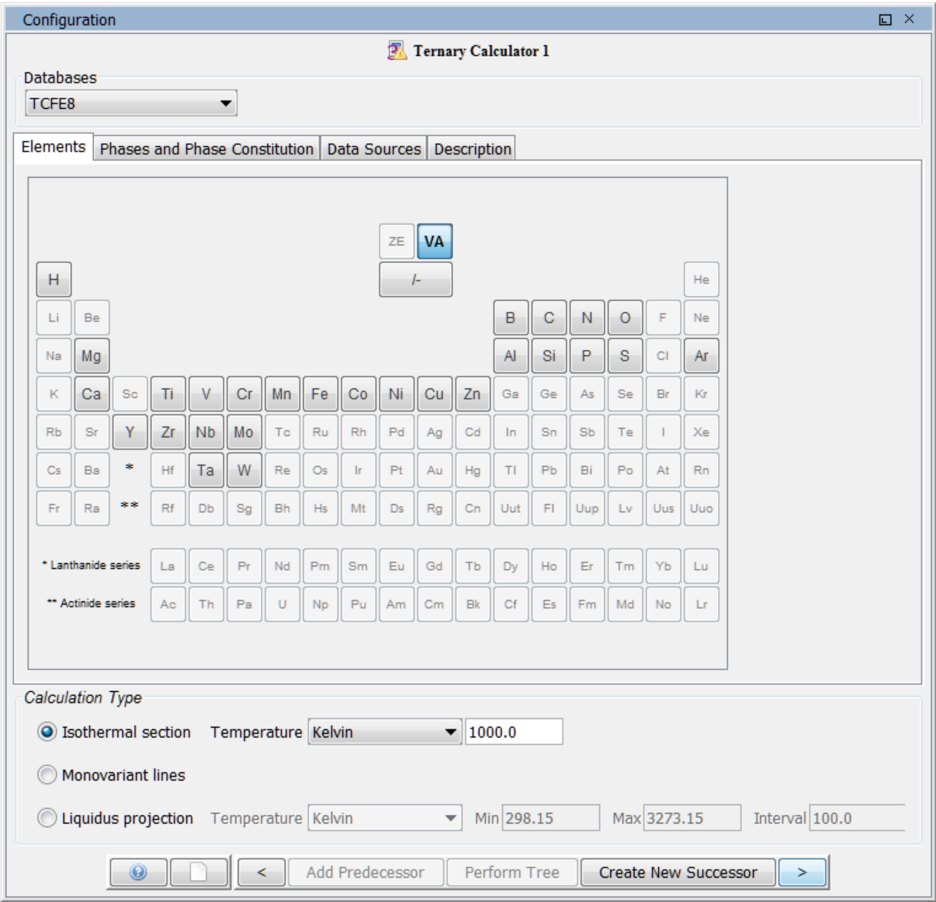
A Ternary Calculator is used for some calculations involving three components. You can think of this activity as a combination of a System Definer and an Equilibrium Calculator, but designed to simplify setting up and performing calculations on ternary systems.

For an example of how a Ternary Calculator is used in a project, see example 4 in *Graphical Mode Example Collection* on page 9.

Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity.

Configuration Window

The following shows the Configuration window of a Ternary Calculator activity:



Configuration Settings

The table lists the available settings that must be configured before performing the calculation.

Setting	Description
Select database	Select a database that contains specifications dedicated to the Ternary Calculator.
Elements (periodic table)	In the periodic table, select the three elements that make up the components of the system.

Setting	Description
Phases and Phase Constitution	This tab is the same as for a *System Definer* on page 54. See *Phases and Phase Constitution* for an example.
Data Sources	This tab is the same as for a *System Definer* on page 54. See *Data Sources* for an example.
Description	This tab is the same as for a *System Definer* on page 54. See *Description* for an example.
Calculation Types	
Isothermal section	Select the **Temperature** at which the ternary system is calculated for the whole composition range.
Monovariant lines	The variation of the liquid compositions with temperature is calculated.
Liquidus projection	Select the **Temperature** range (**Min** and **Max**) and an **Interval**. The projection liquid surface and the monovariant lines are calculated over the given temperature range.

Scheil Calculator

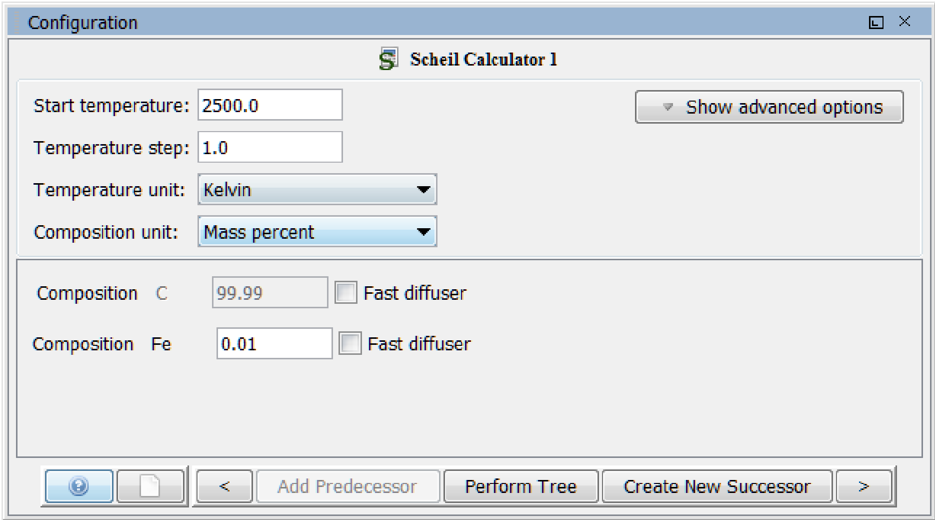
The Scheil Calculator activity is used to perform simulations of *Scheil (Scheil-Gulliver) solidification processes*. By default, a Scheil calculation results in an estimation of the solidification range of an alloy. The calculation is based on the assumption that the liquid phase is homogeneous at all times and that the diffusivity is zero in the solid. However, it is possible to disregard the latter assumption for selected components.

By default, a Plot Renderer successor has the default plot axes **Mole fraction of solid** (X-axis) and **Temperature Celsius** (Y-axis). When these default axes are used in the Plot Renderer, the results of normal equilibria calculations are automatically plotted in the same diagram as the Scheil calculations.

See example 8 in *Graphical Mode Example Collection* on page 9.

Configuration Window

The main **Configuration** window of a **Scheil Calculator** activity is shown below and listed in the table. These settings must be configured before performing the calculation.



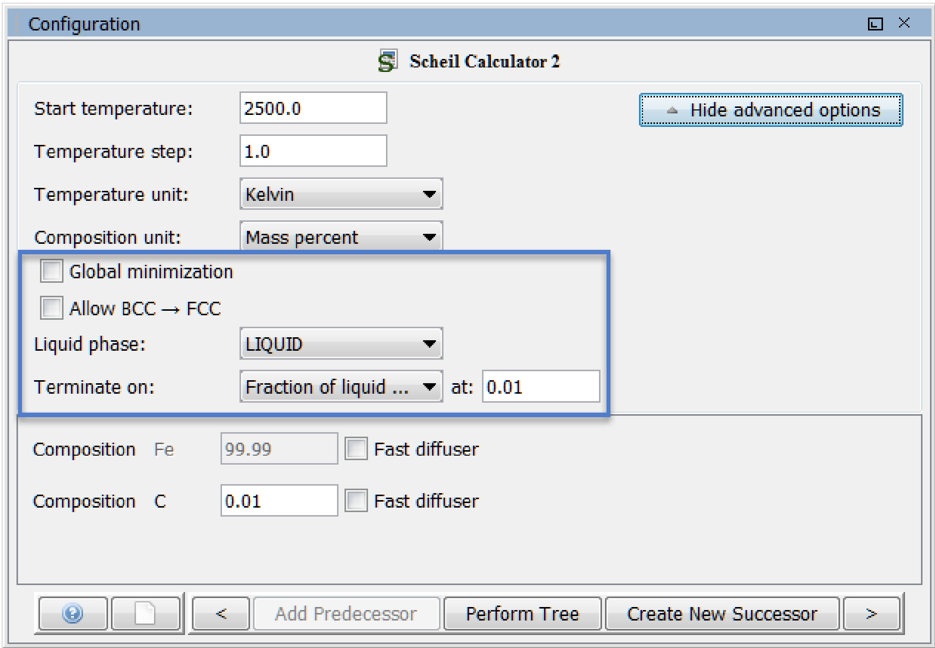
Configuration Settings

Setting	Description
Start temperature	Enter a **Start temperature** that is higher than the liquidus temperature of the alloy, in other words, the temperature at which the alloy is completely melted.
Temperature step	Enter a **Temperature step**. Decreasing the temperature step increases the accuracy, but the default value is usually adequate.
Temperature unit	Select the **Temperature unit**: **Kelvin**, **Celsius** or **Fahrenheit**.
Composition unit	Select the **Composition unit**: **Mass percent**, **Mole percent**, **Mass fraction** or **Mole fraction**.
Composition	This is based on what is entered in the *System Definer* on page 54.
Fast diffuser	Click to select the **Fast diffuser** check box to allow redistribution of this component in both the solid and liquid parts of the alloy.

Scheil Calculator Advanced Options and Settings

Additional advanced settings are available as shown below and in the table:

Click **Show advanced options** to open this configuration window:



Scheil Calculator Advanced Settings

Setting	Description
Global minimization	By default the **Global minimization** check box is not selected. Click to select it to perform a global minimization test when an equilibrium is reached. This costs more computer time but the calculations are more robust.
Allow BCC→FCC	By default the **Allow BCC → FCC** check box is not selected. Select the check box to allow transformations in the solidified part of the alloy caused by each of the components specified to be a *fast diffuser*. It is recommended that you only select this for steels.
Liquid phase	The default in the list is the **Liquid phase** already defined in the database.
Terminate on	Enter a value to terminate the calculation on the **Fraction of liquid phase** or at a specific **Temperature**.

Also see *Creating Activities and Successors* on page 29 for a list of possible successors and predecessors for this activity and *Scheil Simulations* on page 144 in the Console Mode section.

The Console Mode Interface

The following is an introduction to the Console Mode's user interface and the command line prompt.

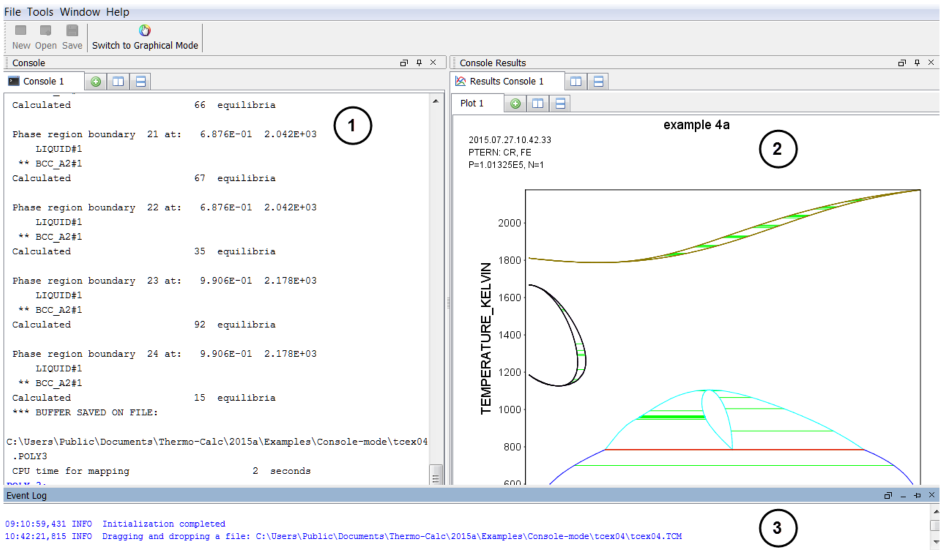
Also see *The Graphical Mode Interface* on page 26 for information about that mode.

In this section:

The Console Mode Layout

By default the first time you launch Thermo-Calc it opens in Graphical Mode. After this, the mode defaults to the most recent one. You can switch back and forth between modes. To open the Console Mode interface, click **Switch to Console Mode** on the toolbar.

The Console Mode User Interface

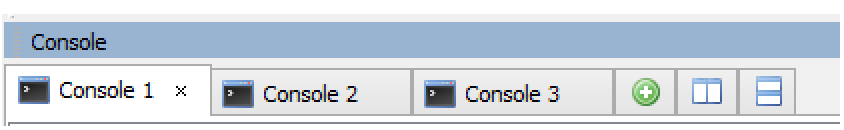


The currently selected window is called the *active* window. This has a darker title bar than the other windows. The windows are:

1. **Console**, with a **Console 1** tab.
2. **Console Results**, with a **Results Console 1** tab as well as a sub-tab (**Plot 1**), which is what is displayed in the window in the figure.
3. **Event Log**, which is the active window in this example.

Console Tabs

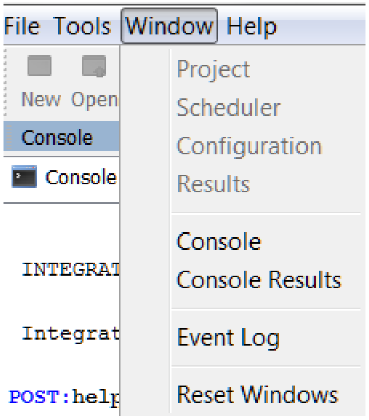
By default, the tabs are side-by-side:



You can toggle to view the tabs top-to-bottom/stacked (click the button) and back to side-by-side (click the button) or the button.

Open Console Windows

Open a specific window from the **Window** main menu, or select **Reset Windows** to return to the default window layout.



Console Window

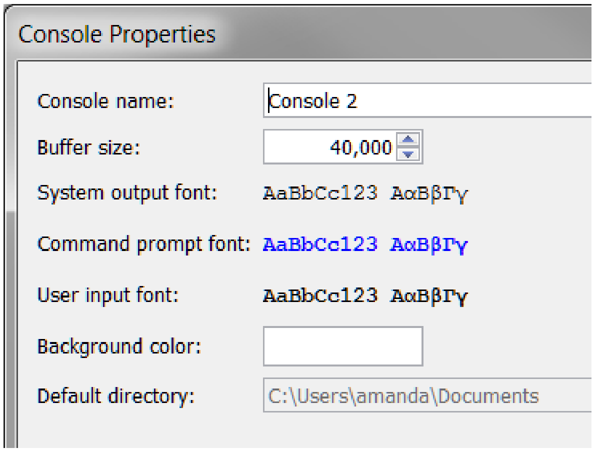
By default the Console window is on the left-hand side of the console. This is where you enter commands. The window can have several Console tabs.

To open and close Console tabs:

- To open (or add), click the button, press <Ctrl+T> when the Console window is active, or right-click a tab header and select **New Console**.
- To close, click the button, press <Ctrl+W> or <Ctrl+F4>, or right-click the tab header and select **Close**.
- To close all tabs except one, right-click the header of the tab you want to keep and select **Close Other**.
- To close all tabs and open a new one, right-click the header of any tab and select **Close All**.

You can change the tab name, font and colours for an individual Console tab by right-clicking its header and selecting **Properties**.

Console Properties Window



Editing the Default Console Directory

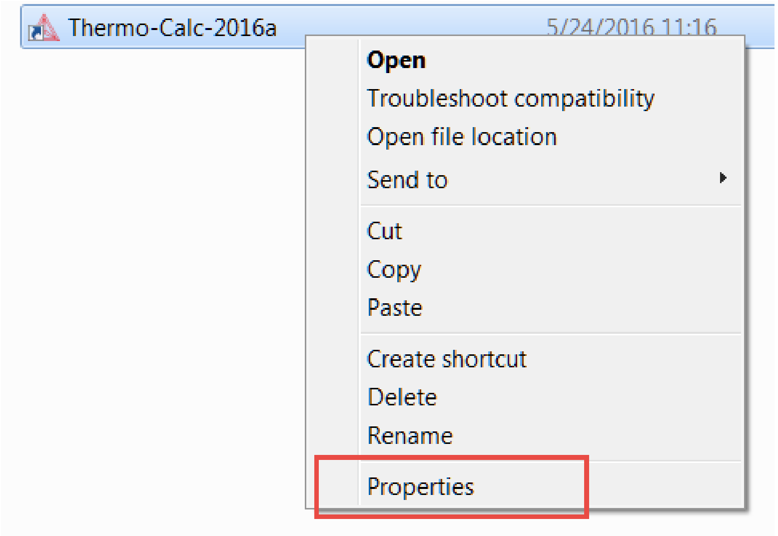
You can add a startup parameter to a shortcut when launching Thermo-Calc on any of the platforms. When you launch the revised shortcut, the file path is set to a user-defined default directory. An example of why you might want to have a custom directory path is if you use batch projects or if you have macro files for batch jobs. The custom directory file path can also be viewed from within Console Mode.

> The shortcut is only valid throughout the existing session; it is not saved as a default. Each time you want to use this file path you must launch Thermo-Calc from the customized shortcut.

The following uses Windows for the example.

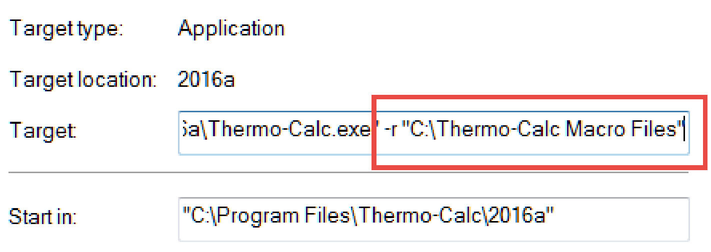
Setting up the Default Console Directory

1. Right-click a shortcut to Thermo-Calc and click **Properties**.



2. In the **Thermo-Calc 2016a Properties** dialog, **Target** field, add the following to the end of the string: **–r** plus the *"path to the default directory"*. Include a space between the –r and path, and use quotation marks around the full file path. A simple example:

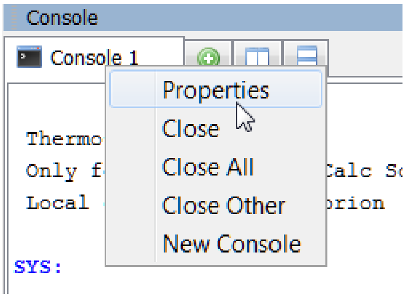
    -r "C:\Thermo-Calc_Macro_Files"



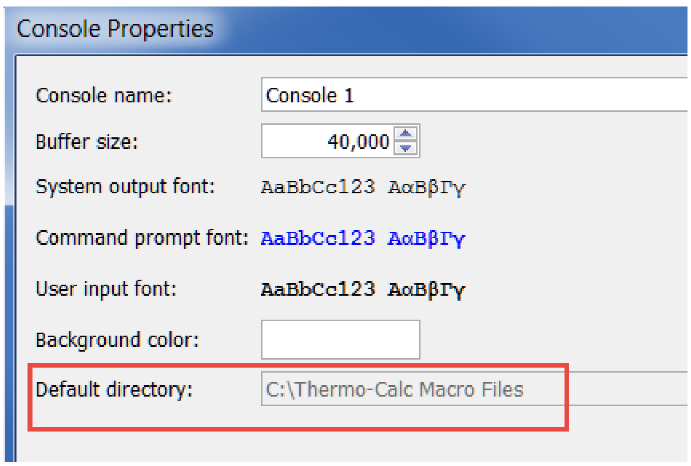
3. Click **Apply** and **OK**.
4. Launch Thermo-Calc from the shortcut where you added the parameter. You can confirm that it is set correctly from within Thermo-Calc.

Viewing the File Path in Thermo-Calc

1. Launch Thermo-Calc from the short cut where you added the parameter.
2. In Console Mode, right click the **Console** window and choose **Properties**.



3. In the **Console Properties** window the **Default directory** section displays the custom file path.



Console Mode Results Window

By default, the Console Results window is on the right-hand side of the screen. This is where plotted diagrams display. Each Console has its own Results tab in the Results window. All the plots and tables generated are presented in the Results tab.

- To open new **Plot** tabs press <Ctrl+Shift+T> or click the add button.
- To open a new Plot tab for the currently active Console tab, press <Ctrl+Shift+T>. To close the Plot tab, press <Ctrl+Shift+W> or <Ctrl+Shift+F4>.

Event Log Window

By default the Event Log window is collapsed (closed). It includes important information about the commands that run and are completed, including error messages. Click once to toggle it open and closed.

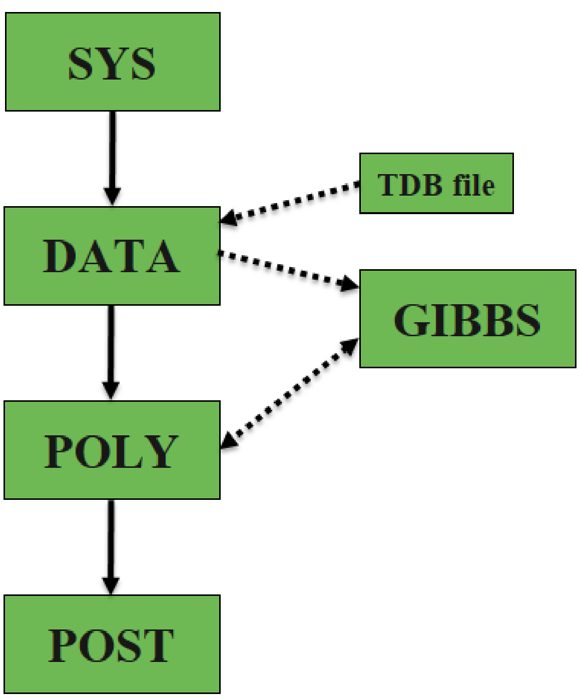
Console Mode Workflow

Working with Thermo-Calc involves moving between different modules. The workflow differs depending on the type of calculation:

- If you want to perform a calculation using the POLY module, you typically first retrieve thermodynamic data in the DATA module, perform the calculation in POLY, and visualize the results in the POST module. It is possible to define a system directly in POLY using *DEFINE_MATERIAL* on page 147 and *DEFINE_DIAGRAM* on page 144.
- If you want to calculate and plot a diagram using a response-driven module, such as POTENTIAL or SCHEIL, then you can go directly to that module. Response-driven modules prompt you to go through the steps to do the calculation and the required post-processing. These modules include BIN, TERN, POTENTIAL, POURBAIX, and SCHEIL. You typically end up in the POST module after having used a response-driven module. In the POST module you can modify the plots and save the diagram.
- If you want to tabulate a chemical substance, phase, or reaction, go directly to the TAB module.

Typical POLY Module Workflow

The following represents the typical workflow when you perform calculations in the POLY module. The solid arrows represent your typical movements between the modules. The dashed arrows represent the movement of data within Thermo-Calc.



The basic workflow is:

1. You are in the **SYS** module when you start Thermo-Calc in Console Mode.
2. In the **DATA** module define the system. Before performing a calculation, you must define the system and retrieve thermodynamic data. The data is retrieved from a database file (*.TDB or *.TDC).
3. The data needed to describe a system is then sent to the **GIBBS** module. This data is accessed by the **POLY** module when you instruct it to perform calculations.
4. In the **POLY** module, perform the calculations.

- In POLY, the conditions are set for an equilibrium calculation (temperature, pressure, system composition, etc.), then an equilibrium calculation is done.
- If you want to make a property or phase diagram, you set the conditions for a stepping or mapping operation and perform that operation.
- This generates data that can be used to plot a property diagram (if a stepping operation is performed) or a phase diagram (if a mapping operation is performed).

5. In the **POST** module plot and visualize your data.

- If a stepping or mapping operation is performed, then you can go to the POST module and plot a property or phase diagram.
- The diagram can be plotted quickly using default settings, but you can also modify which variables to plot of the diagram axes and change the appearance of the diagram.
- You can save the diagram either as a plain text file with data about all the coordinates or as an image file.

Performing Calculations in Console Mode

Calculations can be performed either in the POLY module, or using some of the response-driven modules designed to perform specific types of calculations.

About Equilibrium Calculations on page 131 describes generally how to calculate and plot *Property Diagrams* and *Phase Diagrams*. To calculate and plot a property diagram or a phase diagram, you must first calculate an equilibrium.

Additional sections of use include *About Scheil Simulations* on page 145, *T0 Temperature Simulations* on page 151, *About Paraequilibrium* on page 154, *About Potential Diagrams* on page 158, and *About Aqueous Solutions* on page 162. Also search the online help or see *TABULATION_REACTION Commands* on page 245 and *POLY_3 Commands* on page 117 in the *Thermo-Calc Console Mode Command Reference*.

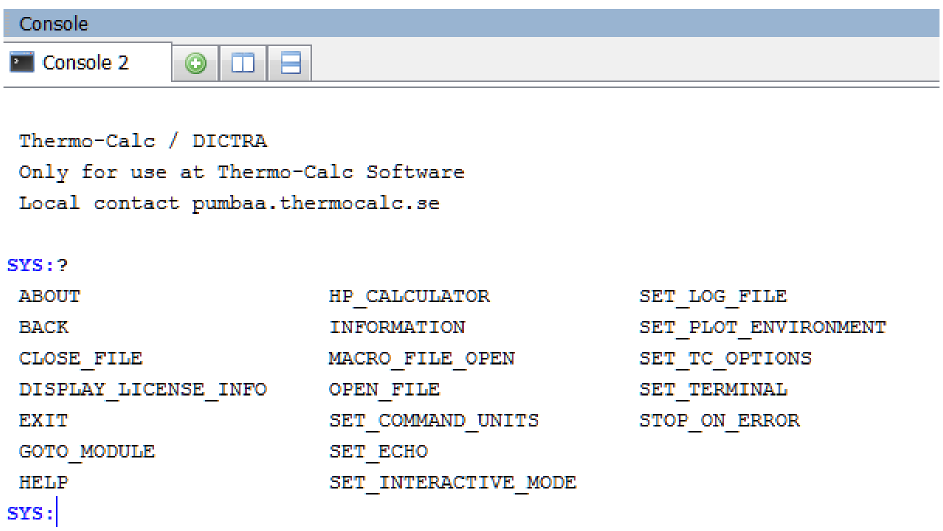
Using the Command Line Prompt

The command line prompt in the Console window tells you which module you are currently in. When Thermo-Calc is first opened in Console Mode, you are in the SYS module, which is indicated with the command line prompt `SYS`.

To use Console Mode, you type in commands at the command line prompt. The available commands depend on which module you are in. You can list the commands by typing `?` at the prompt and then press <Enter>.

- *Moving Between Modules and Submodules* below
- *Entering Command Names or Abbreviations* on the next page
- *Specifying Parameters* on page 121
- *Wild Card Characters* on page 122
- *Controlling Console Output* on page 123
- *Command History* on page 123

Available SYS Module Commands



Moving Between Modules and Submodules

To go to a specific module, you typically use GOTO_MODULE (or just GOTO) followed by the name of the module. For example, to go to the DATA module, type `GOTO DATA`.

The exceptions are the submodules POST and ED-EXP:

- To go to the POST module, you must enter the POST command from within the POLY module or DICTRA module (if you have the add-on license).

- To go to *Experimental Data Files* on page 36, you must enter the ED-EXP command from within the PARROT module.

To go back to the previous module type `Back`. For example, if you are in the DATA module and entered it from the SYS module, when you type `Back`, you return to the SYS prompt. This command is also available to exit from the submodules POST and ED-EXP.

Without a valid Thermo-Calc license, you cannot leave the SYS module. To enter the DICTRA and DIC_PARROT modules, you also need a valid Diffusion Module (DICTRA) license key.

Entering Command Names or Abbreviations

The name of a command typically consists of terms linked with underscores, for example, *LIST_EQUILIBRIUM* on page 156. Note the following:

- UPPER or lower case letters are OK.
- Hyphens (-) instead of underscores (_) are OK, e.g. LIST-EQUILIBRIUM.
- You can abbreviate commands as long as it is unambiguous, e.g. when more than one command begins with LIST_, you need to type the next letter or word to distinguish between the commands.
- Each word can be abbreviated e.g., L_E for LIST_EQUILIBRIUM.

Command Names and Abbreviations Examples

These are examples of how you can abbreviate commands at the command line prompt.

Command name	**Abbreviation**
CALCULATE_EQUILIBRIUM	C-E
CALCULATE_ALL_EQUILIBRIA	C-A
LIST_EQUILIBRIUM	L-E
LIST_INITIAL_EQUILBRIUM	LI-I-E
LOAD_INITIAL_EQUILBRIUM	LO-I-E
LIST_PHASE_DATA CBCC	L-P-D CBC
LIST_PHASE_DATA CEMENTITE	L-P-D CEM
SET_ALL_START_VALUES	S-A-SS-AL
SET_AXIS_VARIABLE 1 X(FCC,FE) 0 0.89 0.025	S-A-V 1 X(F,FE) 0 .89 .025

Command name	Abbreviation
SET_START_CONSTITUENT	S-S-C
SET_START_VALUE	S-S-V
SET_AXIS_PLOT_STATUS	S-A-P
SET_AXIS_TEXT_STATUS	S-A-T-SS-A-TE
SET_AXIS_TYPE	S-A-TY
SET_OPTIMIZING_CONDITION	S-O-C
SET_OPTIMIZING_VARIABLE	S-O-V
SET_OUTPUT_LEVEL	S-O-LS-OU

Specifying Parameters

Running a command typically requires that several parameter values are specified, which can be done directly in the command line after the command name. For example, to set an axis variable for a stepping or mapping operation, enter:

```
SET_AXIS_VARIABLE 1 X(FCC,FE) 0 0.89 0.025
```

You can also use SET_AXIS_VARIABLE and press <Enter>. At the command line prompt, you then specify the parameters one at a time.

On all command lines, you can enter a question mark (`?`) for help. For some parameters, enter two question marks (`??`) for more detailed information.

Default Parameters Values

When you are requested to specify a parameter, Thermo-Calc usually suggests an appropriate default value, which is displayed at the prompt between two slash signs (//). For example, this prompt means the default answer is `TERMINAL`:

```
OUTPUT FILE /TERMINAL/:
```

Press <Enter> to accept the default.

If you specify some parameters directly after the command name, then you can still choose to accept default values for some parameters.

To accept the default value for a parameter, type a comma (,) instead of a value for that parameter. For the comma to be interpreted correctly, type a blank space on each side of the comma. However, if you accept the default values for several parameters that follow each other, then the commas can be typed in without blank spaces separating them: (, , ,) is interpreted in the same way as (,   ,   ,).

For some commands and parameters, Thermo-Calc treats the values previously assigned as default values if the command is used again. For example, if the first stepping or mapping axis variable is specified as:

```
SET_AXIS_VARIABLE 1 X(FCC,FE) 0 0.89 0.025
```

Then if you want to change a parameter value for this axis variable, type:

```
SET_AXIS_VARIABLE 1 , 0.1 ,,,
```

This changes the minimum value from 0 to 0.1 at which a stepping or mapping operation halts. The other parameter values remain the same. Consequently, what is typed is equivalent to:

```
SET_AXIS_VARIABLE 1 X(FCC,FE) 0.1 0.89 0.025
```

Wild Card Characters

You can sometimes use the asterisk (*) as a wild card character to, for example, refer to all components, phases, or species, when showing calculated properties in the POLY module, or when you set axis variables for plotting diagrams in the POST module.

When showing calculated properties in the POLY module, you can also use the dollar sign ($) to refer to all stable phases.

For activity and/or chemical potential properties, the *and $ normally do not work properly as wild card characters.

Wild Card Character Examples

These are examples of the wild card characters you can use to search commands at the command line prompt.

Command	Action
SHOW_VALUE B(*)	Lists mass (gram) of all components in the system.
SET_AXIS_VAR Y B(*) *	Sets mass (gram) of all components in the system as Y-axis variable.
SHOW_VALUE MUR(*)	Lists chemical potentials of all components in the system.
SET_AXIS_VAR Y ACR(*) *	Sets activities of all components in the system as Y-axis variable.
SHOW_VALUE HM(*).T	Lists heat capacities of all phases.
SET_AXIS_VAR Y GM(*) *	Sets molar Gibbs free energies of all phases as Y-axis variable.
SHOW_VALUE TC($)	Lists curie temperature of all stable phases.
SHOW_VALUE W(*,*)	Lists mass fractions of all components in all phases.
SHOW_VALUE W($,*)	Lists mass fractions of all components in all stable phases.

Command	Action
SHOW_VALUE W(FCC,*)	Lists mass fractions of all components in the FCC phase.
SHOW_VALUE W(*,FE)	Lists mass fractions of the Fe component in all phases.
SHOW_VALUE W($,FE)	Lists mass fractions of the Fe component in all stable phases.
SHOW_VALUE Y(*,*)	Lists site fractions of all species in all sublattices of all phases.
SHOW_VALUE Y($,*)	Lists site fractions of all species in all sublattices of all stable phases.
SHOW_VALUE Y(*,*)	Lists site fractions of all species in all sublattices of all phases.

Controlling Console Output

In response to a command Thermo-Calc may display a lot of text in the Console window.

- To pause the text on printing press <Ctrl+S>.
- To resume the printing on screen, press <Ctrl+Q>.
- When Thermo-Calc is performing a mapping operation, the results are continuously printed in the Console window. To terminate the calculation of the current region of the mapping and stop the output, press <Ctrl+C>.

Command History

- To scroll through the last twenty commands used, type two exclamation marks (`!!`) and press <Enter>.
- To repeat the last n commands, type `!` followed by the number of previous commands to be executed again.
- To scroll through previous performed commands, use the Up (↑) or Down arrow (↓) keys.
- To read more information about the command history functionality in Thermo-Calc, type `! ?` at the command line prompt.

Log, Macro and Workspace Files

In Console Mode, Thermo-Calc uses different kinds of files, including:

- *Log Files* on the next page (*.LOG)
- *Macro Files* on the next page (*.TCM or *.LOG)

- POLY *Workspace Files* on page 126 (*.POLY3),
- *Experimental Data Files* on page 36 (*.EXP)

Also see the *Data Optimization User Guide* on page 1.

- The GIBBS module uses a workspace file (*.GES5). See *GIBBS_ENERGY_SYSTEM Commands* on page 42
- The PARROT module uses files with the suffixes *.PAR and *.POP. See *PARROT Commands* on page 86

Log Files

Log files are plain text files used to save a sequence of commands. Log files can be edited in a text editor.

- To start saving your input into such a file, use SET_LOG_FILE in the SYS module, followed by the name of the file that you want to save your command sequence to.
- If you want to save the output in the log file as well, use SET_ECHO before the SET_LOG_FILE command. Doing this is useful if you want use the log file later as a macro file.

See *SYSTEM_UTILITIES Commands* on page 239 in the *Thermo-Calc Console Mode Command Reference*.

Macro Files

Macro files are plain text files used to save a sequence of commands that can be loaded and executed. Macro files can be edited in a text editor.

Macro File Commands

Also see Example 12 in *Console Mode Example Collection* on page 14.

Action	Command	Description and Comments
To load a macro file	In the SYS module, *MACRO_FILE_OPEN* on page 5 followed by the name of the macro file.	Thermo-Calc starts the command sequence that the file contains.
Regain control of the console	*SET_INTERACTIVE* on page 8	End a macro file with this command to have control of the Console returned to you.

Action	Command	Description and Comments
Add comments to a macro file	Start a line with @@, or enclose it between an @ (-line and a @)-line.	When lines are enclosed the software ignores them when running the macro.
Open a macro file	MACRO_FILE_OPEN	When creating the log file, you can make a macro file load up to five other macro files.
Nest macro files inside of each other	MACRO_FILE_OPEN SET_INTERACTIVE	If a nested macro file ends with SET_INTERACTIVE, then Thermo-Calc resumes with the higher-level macro file at the command immediately following the MACRO_FILE_OPEN command that loaded and executed macro file that has just been terminated. If a nested macro file doesn't end with SET_INTERACTIVE (but with an end-of-file character), then the console shuts down and the macro is stopped.

Macro File Control Characters

The `@?` character allows you to make a macro interactive by allowing input. At the @? character, which is placed where a parameter value or argument is normally put, Thermo-Calc prompts to input the value of a parameter or argument. You can enter a string immediately following the @? character. This string is presented to the user when prompted to enter the parameter value or argument. The entered value is used by Thermo-Calc as input to the command in question. For example, you can request the user to specify the temperature range of a stepping calculation by entering the following in a macro file:

```
SET_AXIS_VARIABLE 1 T @?Low-temperature-limit: @?High-temperature-
limit:
```

You can also use up to nine variables in your macro file and prompt the user to enter values that can be assigned to these variables. Use the `@#n` character when you want to prompt the user to provide a value to the variable, where the n is a digit between `1` and `9`. You can then use this value by with the `##n` character. For example, you can request the user to provide the first element of a system by entering the following in the macro file:

```
@#3First-element?
```

You can then use this variable with the entering the character ##3 later in the macro file. For example, you can write:

```
SET_AXIS_VARIABLE 1 x(##3) 0 1,,,
```

Finally, there is the `@&` pause character. Thermo-Calc pauses and waits for input from the user when this pause character is read from a macro file. Inserting pause characters is useful if you want to allow the user to monitor what is happening when Thermo-Calc is running the macro.

Response-Driven Modules and Macros

Macro files that are created while you use a *response-driven module* begin with the module-entering command (for example, `Goto_Module Scheil)` followed by a number of lines with responses to the module's questions and requests. The file terminates with the commands *POST* on page 160 or *SET_INTERACTIVE* on page 8 (this command gives you back control of the Console). An empty line cannot be edited with any input rather than the default answer (to a specific question). Comment lines, commands to open other macro files, pause characters or input-controlling characters cannot be inserted between these empty lines (otherwise the response-driven module cannot be executed properly).

Workspace Files

You can save all the data in your current workspace by first using SAVE_WORKSPACES . This command is available in the POLY, GIBBS and PARROT modules. The workspace has all the settings specified and the result of any stepping or mapping operations performed after using SAVE_WORKSPACES. Consequently, the command opens a workspace in which all data is saved which is generated after the SAVE_WORKSPACE command is executed. The saved data includes original and modified thermodynamic data, the last set of conditions and options, and calculation results.

To load the data and calculation results of a workspace file, use the READ_WORKSPACES command available in the POLY module. The file can be used for calculation in POLY, visualization in POST, or data manipulation in GIBBS. You are also given the option in the POURBAIX and SCHEIL modules to open a previously saved workspace file. This allows you to make new POURBAIX or SCHEIL calculations on the same chemical system as before but with different temperature, pressure, and composition conditions, or to plot new property diagrams based on the previous calculations.

Main Menu Commands

The **File** menu and toolbar are reserved for Graphical Mode. In Console Mode you use commands to access these menu options. The **Tools**, **Window** and **Help** menus are described in *Main Menu and Toolbar* on page 32. If there is an equivalent command, it is listed in the table below.

Console Mode Main Menu Commands

Option	Command
Open files	In Thermo-Calc, drag and drop the file into the Console window. In the SYS module: *MACRO_FILE_OPEN* on page 5 In the SYS module: *OPEN_FILE* on page 241
Save (workspace)	This command depends on the module you are in. Type ? at the prompt to look for a command such as *SAVE_GES_WORKSPACE* on page 84 (GIBBS module)
Exit	In any module: EXIT
Close	In the SYS module: *CLOSE_FILE* on page 240
Display license info	In the SYS module: *DISPLAY_LICENSE_INFO* on page 240

Defining a System in Console Mode

Defining a system means to select the chemical components and to retrieve thermodynamic data about those components from an appropriate database in order to do the calculations. You use database(s) with thermodynamic data and then define what system elements as components. Once the system is defined, you retrieve the system's thermodynamic data from the database(s).

- If you use a *response-driven module* to perform your calculation, such as BIN or POURBAIX, then the module prompts you to select database(s) and define the system.
- When doing calculations in the POLY module, you typically have to manually select databases and define the system in the DATA module. However, it is also possible to define the system directly in POLY using *DEFINE_MATERIAL* on page 147 and *DEFINE_DIAGRAM* on page 144.

Also see *DATABASE_RETRIEVAL Commands* on page 11 in the *Thermo-Calc Console Mode Command Reference*.

How to Define a System in Console Mode

1. Go to the DATA module. At the `SYS` prompt, use GOTO_MODULE and enter `Data`.
2. Change the default database unless you directly specify the name of the database as a parameter to the command. Type `SWITCH_DATABASE` and follow the prompts.

   The second part of the command prompt indicates the current default database. For example, if the prompt is `TDB_TCFE8`, then the current database is TCFE8.

3. Use DEFINE_ELEMENTS followed by a list of the elements that you want in your system. (To list the elements that are available in your current database, use LIST_DATABASE and choose `Elements`). For example, if you want to have Fe and C in your system, type:

   ```
   DEFINE_ELEMENTS Fe C
   ```

4. Use REJECT and choose `Phases` if you want to avoid retrieving any phases from the database. (To list the phases that are available in your current database, use LIST_DATABASE and choose `Phases`. To list the phases that can form in the defined system, use LIST_SYSTEM and choose `Constituent`.) For example, if you do not want the graphite phase to be retrieved, you type:

   ```
   REJECT PHASES GRAPHITE
   ```

   If the number of phases to include is much lower than the total number of phases, then it can be convenient to first `Reject Phases *` and then restore the phases to include using `Restore Phases`.

5. Use GET_DATA to search the database and send the thermodynamic data about your system to the GIBBS workspace.

   At this point you can proceed to the POLY module (with GOTO_MODULE POLY). However, you may want to add elements, phases and species to the GIBBS workspace from other databases. If so, then proceed to the next step.

6. Use APPEND_DATABASE to select the (additional) database from which you want to retrieve data. This command works exactly in the same way as the SWITCH_DATABASE command, with the exception that it does not reinitialize the DATA module and the GIBBS workspace, but instead appends or replaces the data that has already been retrieved with new data from the additional database.
7. Define your elements and specify whether to reject and restore any phases. Do this in

exactly the same way when using the SWITCH_DATABASE command (see steps 2 and 3).

8. Use GET_DATA to search the database and add the thermodynamic data to the data that already exists in the GIBBS workspace.
9. Use APPEND_DATABASE again if you want to add data from yet another database. When you have retrieved all the data you need, you can proceed to the POLY module.

Equilibrium Calculations

In this section:

About Equilibrium Calculations

An equilibrium describes what the composition of the end state of a system is, given a full specification of state variables such as temperature, pressure, initial composition, system size, etc. An equilibrium calculation is normally done in POLY-3 according to the *Global Minimization Technique*, which ensures that the most stable minimum under the specified conditions is computed.

For an equilibrium calculation to be performed, the state variables must all be set as conditions for the calculations. Such conditions include, for example, temperature, pressure, and system composition. When you calculate an equilibrium in the POLY module, you have to set these conditions manually.

Setting Conditions

Setting a condition normally involves giving a single state variable a specific value. For example, you can set the temperature to 1273.5 Kelvin (T=1273.5). Alternatively, setting a condition can involve giving a linear expression with more than one state variable a specific value. For example, you can set the mole fraction of the S component to be the same in the liquid and the pyrrhotite phases (`X(LIQ,S)-X(PYRR,S)=0`).

The number of state variables that need to set is determined by the *Gibbs Phase Rule*. Typically, the state variables to give values to are:

- temperature (in K)
- pressure (in Pascal)
- system size in number of moles (in mole) or mass (in grams)
- the fraction of each component (in number of moles or mass)

If you fix the phase of the equilibrium and all but one of the state variables, you can discover the value of the other state variable at equilibrium.

It is possible to specify a set of conditions that does not have any equilibrium and the program detects this by failing to reach equilibrium during the calculation.

State Variables for Setting Conditions

The following table lists some of the available state variables you can use to set conditions:

State Variable	SET_CONDITION Parameter
temperature in the system (in K)	T
pressure in the system (in Pascal)	P
system size (mole number in moles)	N
system size (mass in grams)	B

State Variable	SET_CONDITION Parameter
number of moles of a component in the system	N(<component>)
mole fraction of a component in the system	X(<component>)
mass fraction of a component in the system	W(<component>)
activity of a component in the system	ACR(<component>)
chemical potential of a component in the system	MUR(<component>)
mole fraction of a component in a phase	X(<phase>,<component>)
mass fraction of a component in a phase	W(<phase>,<component>)
activity of a species referred to a phase at ambient temperature and pressure	ACR(<species>,<phase>)
chemical potential of a species referred to a phase at ambient temperature and pressure	MUR(<species>,<phase>)
enthalpy in the system (in J)	H
enthalpy of a phase (in J/mol)	HM(<phase>)

Equilibrium Information Example

```
POLY_3:l-e
OUTPUT TO SCREEN OR FILE /SCREEN/:
Options /VWCS/:
 Output from POLY-3, equilibrium =     1, label A0  , database: TCFE7

 Conditions:
 T=1000, P=1E5, X(C)=5E-2, N=1
 DEGREES OF FREEDOM 0

 Temperature   1000.00 K (   726.85 C),  Pressure  1.000000E+05
 Number of moles of components  1.00000E+00,  Mass in grams  5.36552E+01
 Total Gibbs energy -4.07975E+04,  Enthalpy  2.41176E+04,  Volume  7.21740E-06

 Component               Moles       W-Fraction  Activity   Potential  Ref.stat
 C                       5.0000E-02  1.1193E-02 2.1816E-01 -1.2659E+04 SER
 FE                      9.5000E-01  9.8881E-01 6.1895E-03 -4.2279E+04 SER

 BCC_A2                      Status ENTERED     Driving force  0.0000E+00
 Moles 9.5069E-01, Mass 5.3063E+01, Volume fraction 9.6312E-01  Mass fractions:
 FE  9.99844E-01  C   1.55514E-04

 GRAPHITE                    Status ENTERED     Driving force  0.0000E+00
 Moles 4.9313E-02, Mass 5.9230E-01, Volume fraction 3.6881E-02  Mass fractions:
 C   1.00000E+00  FE  0.00000E+00
```

Calculating an Equilibrium

This topic explains how to calculate an equilibrium in POLY. To calculate an equilibrium means to calculate the equilibrium composition of your system, given a full specification of conditions that reduce the degrees of freedom of the calculation to zero.

If you are running a network client installation of Thermo-Calc and cannot enter the POLY module, this may be because all the license instances on that the license server makes available have been checked out.

Also see *POLY_3 Commands* on page 117 in the *Thermo-Calc Console Mode Command Reference*.

1. Define your system (see *Defining a System in Console Mode* on page 127).
2. Go to the POLY module. At the SYS prompt, type `GOTO_MODULE` and enter `POLY`.

3. Use SET_CONDITION followed by conditions and value assignments to set the conditions of your calculation. For example, to set temperature, pressure, initial composition and system size for a Fe-Cr-C system, you might enter the following:

   ```
   SET_CONDITION T=1200 P=1E5 W(CR)=0.18 W(C)=0.0013 N=1
   ```

   This sets the temperature (T) to 1200 K, the pressure to 1 bar (100,000 Pascal), the mass fraction of Cr to 18 mole percent, the mass fraction of C to 0.13 weight percent, and the total amount of material to 1 mole. The fraction of Fe in the system is calculated from the fractions of Cr and C. You have to set the fraction of all the components in your system except one.

4. Use COMPUTE_EQUILIBRIUM to run the calculation.
5. Use LIST_EQUILIBRIUM to see the results of the calculation.

Calculating an Equilibrium with a Fixed Phase

There are many ways to calculate equilibriums in Thermo-Calc. The following is one example.

You can calculate an equilibrium that has a certain amount of a certain stable phase. Use *CHANGE_STATUS* on page 59 and choose `Phase` to specify the phase and the amount of that phase (in normalized mole number) that you want to set as fixed. For example, if you want to find out at what temperature a system starts to melt, enter the following:

```
CHANGE_STATUS PHASE LIQUID=FIX 0
```

You must leave the state variable whose equilibrium value you are interested in unspecified. However, if you have already specified that state variable, you can make it unspecified again by using *SET_CONDITION* on page 165 and set that state variable to `NONE`. For example, if you have given temperature a value, you can type:

```
SET_CONDITION T=NONE
```

The calculated equilibrium includes the value of the unspecified variable at which the equilibrium enters the phase that is fixed.

For an example where an equilibrium is calculated with a fixed phase, see Example 7 in *Console Mode Example Collection* on page 14.

Calculating an Equilibrium with Suspended or Dormant Phases

You can calculate an equilibrium under the assumption that one or several phases are `SUSPENDED` or `DORMANT` using CHANGE_STATUS and choose `Phase`. For example, to specify that all phases except one should be suspended, you can first suspend all phases and then enter a single phase in the following way:

```
CHANGE_STATUS PHASE *=SUSPENDED
CHANGE_STATUS PHASE FE_LIQUID=ENTERED
```

For an example where this is done, as well as where the status of phases is set to be dormant, see Example 10 in *Console Mode Example Collection* on page 14.

Property Diagrams

In this section:

About Property Diagrams

When you calculate and plot a property diagram, there is only one independent state variable. Many different properties can be plotted as a function of this independent variable. For example, if the independent state variable is temperature, then the mole fractions of all phases can be plotted as a function of temperature. Or the composition of a specific phase may be plotted relative to temperature, or the activity of a component in the system as a whole or in a specific phase may be plotted relative to it.

A property diagram is plotted based on a series of equilibria that is computed while the value of one state variable is varied between a minimum and a maximum value. This variable is referred to as the stepping axis variable. In Thermo-Calc, first calculate one initial equilibrium, and then new equilibria are calculated at incremental steps in both directions on the stepping axis from the initial equilibrium. This continues until the stepping operation has covered the length of the axis between a minimum and a maximum value that is specified.

For an example of the calculation of a property diagram, see Example 8 in *Console Mode Example Collection* on page 14.

Calculating and Plotting a Property Diagram

This topic explains how to calculate a property diagram in POLY. To calculate a property diagram means to calculate a series of equilibria while the value of the stepping axis variable varies between a minimum and a maximum value. With the exception of the stepping axis variable, all the state variables that you set when you calculate the initial equilibrium retain the values when the new equilibria are calculated.

POLY_3 Commands on page 117 in the *Console Mode Command Reference*.

How to Calculate and Plot a Property Diagram

1. Calculate an initial equilibrium in the POLY module (see *Equilibrium Calculations* on page 130).
2. Use SET_AXIS_VARIABLE to set the axis variable, the minimum and the maximum stepping variable values and the step length. The first parameter of SET_AXIS_VARIABLE is the axis number. Since a property diagram only has one axis variable, this is set to 1. For example, if you want to set the axis variable to temperature, and calculate an equilibrium at every 50 K between a minimum temperature of 100 K and a maximum temperature of 2000 K, enter

```
SET_AXIS_VARIABLE 1 T 100 2000 50
```

An axis variable must be a state variable that is set when you calculated the initial equilibrium. For example, if you set the fraction of a component in number of moles, then you cannot set the mass fraction of this component as an axis variable. Also, the minimum stepping variable value must be smaller than, and the maximum value larger than, the value that you set the state variable to when calculating the initial equilibrium.

3. Use STEP_WITH_OPTIONS and choose `Normal` to perform the stepping operation.

   Thermo-Calc lists the phase regions along the stepping axis, the phases contained in each region, and the number of equilibria calculated in that region. Each region is defined by the phases it contains.

The number of equilibria calculated is not just the number of steps that are performed according to the step length and the minimum and maximum values on the axis variable. This is because Thermo-Calc calculates extra equilibria when a new phase is discovered in order to determine the phase region boundary more precisely.

Step with Options Example

```
POLY_3:s-a-v 1 T 1000 2000 100
POLY_3:step
Option? /NORMAL/:
 No initial equilibrium, using default
 Step will start from axis value    1000.00
...OK

 Phase Region from    1000.00     for:
     BCC_A2
     GRAPHITE
 Global check of adding phase at  1.01118E+03
 Calculated     4 equilibria

 Phase Region from    1011.18     for:
     BCC_A2
     FCC_A1
     GRAPHITE
 Calculated     2 equilibria

 Phase Region from    1011.18     for:
     FCC_A1
     GRAPHITE
 Global test at  1.09000E+03 .... OK
 Global check of removing phase at  1.15727E+03
 Calculated    17 equilibria
```

4. To plot the diagram, use POST to open the POST module and then use PLOT_DIAGRAM. By default, the X-axis represents the stepping axis variable and Y-axis represents the sum of the mole fractions of all phases (NP(*)).

Calculating a Property Diagram One Phase at a Time

There are many ways to calculate property diagrams in Thermo-Calc. One example is where you can calculate a property diagram with a separate stepping operation being performed for each phase at a time in its default most stable composition (the major constitution). This is useful if you want to create a property diagram for a heterogeneous system with both ordered phases and their disordered pairs.

To calculate a property diagram one phase at a time, use *STEP_WITH_OPTIONS* on page 172 and when prompted choose `One_phase_at_time` to perform the stepping operation.

Calculating Several Properties in the Same Diagram

There are many ways to calculate property diagrams in Thermo-Calc. The following is another example. If you perform several stepping calculations after each other, the results of these calculations are all saved in your workspace file. This allows you to the following:

- Calculate (and then plot) missing parts of a specific property inside the first property diagram. These parts are calculated inside the range of the stepping variable with a different control condition.
- Calculate (and then plot) two or more sets of a specific property on the same property diagram for the same system. These can be calculated under different control conditions with stepping operations being performed across the same stepping axis variable range.
- Calculate phase boundary lines and then plot them in a corresponding phase diagram for the same system. This can be especially useful for some defined secondary phase-transformations. For example, if you want to find the phase boundary between BCC_A1 and BCC_B2, or the equal-Gm for two specific phases, or the equal-fraction or equal-activity for two specific phases of a certain species, then this can be useful.

Unless you have opened a new workspace file previously using SAVE_WORKSPACES , the results of the stepping calculations are saved in a RESULT.POLY3 file. To save the results of several stepping calculations in a file different from the workspace file that your results are currently saved to, then use SAVE_WORKSPACES before you perform the first stepping calculation.

SAVE_WORKSPACES overwrites and deletes the results of all previous stepping calculations.

Phase Diagrams

In this section:

About Phase Diagrams

Phase diagrams have two or more independent axis variables. Any state variable that has already been set can be used as the mapping variable for a mapping calculation and then as the axis variable for a phase diagram. From a mapping calculation, many types of phase diagrams can be plotted, with one of the mapped variables as one axis variable, and with other mapped variables or any varied property (state or derived variables) or entered symbol (variables, functions or table values) as the other axis variables.

All phase diagrams consist of zero phase fraction lines. There are two distinct types of phase diagrams: those with the tie-lines in the plane of the diagram and those where the tie-lines are not in the plane. The former includes binary phase diagrams and ternary isotherms. The latter includes more general isopleth diagrams with one or more fixed extensive variables (normally, this is a composition). The BIN and TERN modules calculate binary and ternary phase diagrams. The POTENTIAL and POURABIX modules calculate the more general isopleth diagrams and other related diagrams.

The following topics describe how to calculate the types of phase diagrams in the *POLY_3 Commands* on page 117 module. In POLY, you can calculate phase diagram for systems with up to 40 components and with thousands of phases. You can combine activity conditions and fixed phase status and fraction conditions in any way. Thermo-Calc can calculate any arbitrary 2D section through composition space.

There is no guarantee that the conditions set will result in a calculation that reaches an equilibrium.

For an example of the calculation and plotting of a phase diagram, see Example 4 in *Console Mode Example Collection* on page 14.

Calculating and Plotting a Phase Diagram

This topic describes how to calculate a phase diagram in the POLY module. The commands are described in *POLY_3 Commands* on page 117 and *POST Commands* on page 188.

1. Calculate an initial equilibrium in the POLY module (see *Equilibrium Calculations* on page 130).
2. Use SET_AXIS_VARIABLE to set the first axis variable, its minimum and maximum mapping value and the length of each incremental step on this axis. The first parameter of SET_AXIS_VARIABLE is the axis number and is set to 1. For example, suppose you want the mass fraction of C as your first axis variable. You want it to vary between 0 and 0.1, with the length of each incremental step being no more than 0.002. You then enter:

```
SET_AXIS_VARIABLE 1 W(C) 0 .1 0.002
```

An axis variable must be a state variable that is set when you calculated the initial equilibrium. For example, if you set the fraction of a component in number of moles, then you cannot set the mass fraction of this component as an axis variable.

3. Use SET_AXIS_VARIABLE to set the second axis variable and specify its minimum, maximum, and step length values. The axis number is set to 2. For example, suppose you want temperature as your second axis variable and you want it to vary between 900 K and 1900 K, with an incremental step of 25 K. You then enter:

   ```
   SET_AXIS_VARIABLE 2 T 900 1900 25
   ```

4. If you want more than two axis variables, then use SET_AXIS_VARIABLE until you have set all the axis variables. You can set up to five axis variables. Axis variables 3, 4 and 5 must be set temperature, pressure or to the chemical potentials of components.

   You may want to save your workspace with SAVE_WORKSPACES before you perform the mapping calculation. This saves the axis variables you have set. However, it overwrites the results of any previous stepping or mapping calculations done.

5. Use MAP to perform the mapping calculation. The console lists the phase region boundaries and the phases contained on one side of each boundary.

   ```
   Phase region boundary  30 at:   1.363E+03  2.012E-01
       FCC_A1
    ** GRAPHITE
   Calculated.                      3  equilibria
   Terminating at known equilibrium

   Phase region boundary  31 at:   1.990E+03  2.526E-01
       LIQUID
    ** GRAPHITE
   Calculated.                     13  equilibria
   Terminating at known equilibrium
   ```

6. To plot the diagram, use POST to open the POST module and then use PLOT_DIAGRAM. By default, the X-axis represents mapping axis variable 1 and the Y-axis represents mapping axis variable 2.

Calculating a Quasi-Binary Phase Diagram

There are many ways to calculate phase diagrams in Thermo-Calc. The following is an example where a quasi-binary phase diagram is used for calculations on a ternary system in which one component has an activity or chemical potential that is fixed (although if you have fixed a phase and the phase composition varies, then the activity or chemical potential may also vary). The tie-lines in a quasi-binary diagram are in the diagram's plane of the diagram. This means that the calculation follows the lever rule as well as other rules.

For an example of a calculation of a quasi-binary phase diagram, see Example 17 in *Console Mode Example Collection* on page 14.

How to Calculate a Quasi-Binary Phase Diagram

Use CHANGE_STATUS to set the component that has a fixed activity or chemical potential. For example, suppose you want to compute a phase diagram for a Ca-Fe-O system in which the liquid oxide (`FE-LIQ`) is in equilibrium with liquid Fe. You then enter:

```
CHANGE-STATUS FE-LIQ=FIX 0
```

Calculating a Quasi-Ternary Phase Diagram

There are many ways to calculate phase diagrams in Thermo-Calc. The following is an example of a quasi-ternary phase diagram used for calculations on a quaternary system where one component has a fixed activity or fixed chemical potential.

How to Calculate a Quasi-Ternary Phase Diagram

When calculating a quasi-ternary phase diagram it is necessary to set a condition on either activity or chemical potential of the fourth component. To specify a meaningful value it is recommended that you first change the component's reference state using SET_REFERENCE_STATE. For example, to calculate a phase diagram for a quasi-ternary Fe-Cr-Ni-C system with fixed carbon activity, you can enter the following:

```
SET_REFERENCE_STATE C GRAPH ,,
```

You must also use SET_CONDITION to set the activity of the component whose activity or chemical potential you have fixed. Do this by assigning a value to the ACR state variable. For example, if the component is carbon, you might enter

```
SET_CONDITION ACR(C)=0.002
```

Scheil Simulations

In this section:

About Scheil Simulations

Thermo-Calc is primarily a program for performing equilibrium calculations, but some non-equilibrium transformations or partial-equilibrium transformations can be simulated. One example of such a transformation is a Scheil-Gulliver solidification.

In a Scheil-Gulliver solidification, the diffusion in the solid phases is assumed to be so slow that it can be ignored, while the diffusion in the liquid phase is assumed to be very fast. With this approximation, the conditions at the liquid/solid interface can be described as a local equilibrium.

By making a stepping operation on the temperature variable (or enthalpy or amount liquid phase) with small decrementing steps, the new composition of the liquid can be determined. After each step, the amount of formed solid phase is removed and the overall composition is reset to the new liquid composition. In effect, the whole system is described as a non-equilibrium state regarding various parts of solidified phases at various solidification stages.

The Scheil Module

You can simulate Scheil-Guilliver solidification processes with the SCHEIL module. When you enter the module, you are prompted to answer a series of questions about which database to use, what the major element is and in which amount. Thermo-Calc then displays a property diagram showing how the fraction of solid phase varies with temperature (in Celsius). You can then plot diagrams using other variables you are interested in. Do this by starting a new simulation, by opening an old file with the results of a previous simulation that you can plot differently. For example, you might be interested in plotting the fraction of remaining liquid against temperature, the fraction of each solid phase or the total of solid phases against temperature, the microsegregation in each solid phase, or the latent heat evolution against temperature.

It is possible to use the SCHEIL module to run a modified Scheil-Gulliver simulation that also takes back-diffusion of some fast-diffusing interstitial elements such as C or N into account, as well as BCC to FCC phase transformation phenomena.

Scheil Module Examples

- See Examples 15 and 30 to simulate non-equilibrium transformations without considering fast-diffusion elements.
- See Example 48 to simulate partial-equilibrium transformation with fast-diffusion elements taken into consideration. This example also shows how you can perform lever-rule calculations (full-equilibrium transformations).

Scheil and the Diffusion Module (DICTRA)

If you also have a license for the Diffusion Module (DICTRA), you can import a previously calculated Scheil segregation profile into the software using the command INPUT_SCHEIL_PROFILE. Search the online help or see the *Diffusion Module (DICTRA) Command*

Reference.

Simulating a Scheil-Solidification

Simulating a Scheil-Gulliver solidification process involves calculating the liquid composition of a higher-order multicomponent system at each step of a cooling process, and resetting the liquid composition as the composition of the entire system (after having removed all amount of solid phase).

SCHEIL Commands on page 235 in the *Console Mode Command Reference*.

How to Simulate a Scheil-Solidification

1. Use GOTO_MODULE SCHEIL to enter the SCHEIL module.
2. Type START-WIZARD.
3. Specify which `Database` to use.
4. Specify the `Major element or alloy` in your system.

   For example, for a steel/Fe-alloy, enter `Fe`, for an Al-based alloy, enter `Al`, and for a Ni-based super alloy, enter `Ni`.

5. Set whether to specify the `Composition input in mass (weight) percent` (the default) or in `Mole percent`. Enter `Y` for mass percent and `N` for mole percent.
6. Specify the name of the `1st alloying element`. You can directly specify mass or mole percent after the name (for example, enter `Cr 5`). If this is not specified, then you are prompted to enter it separately.
7. Specify the other alloying elements in the same way as you specify the first (for example, `c 1`).
8. After you have specified your last alloying element, press <Enter> when requested to specify the next element. This ends the process of defining the bulk composition of the alloy system.

   You can also specify all the alloying elements and their corresponding compositions on the same line when you are prompted to specify your first alloying element in Step 6. For example, you can enter `Cr 5 Ni 1 Mo 0.5 C 0.01 N 0.02`.

9. Specify the starting `Temperature (C)` in Celsius. The default is `2000`.

   This value should be sufficiently high so that the solidification simulation starts with the alloy system in the liquid single-phase region.

10. Decide which phases (if any) to reject at the `Reject phase(s)` prompt. Enter the name of the phases to reject or press <Enter> for `NONE`.
11. Decide whether to `Restore phase(s)`. You may want to restore a phase that you rejected when you ran the simulation earlier, or you may want to restore a phase that is rejected by default in your database. Enter the name of the phases to restore or press <Enter> for `NONE`.
12. When prompted `OK?` type `N` to go back to Step 9 to reconsider which phases to reject or restore. Type `Y` to continue.

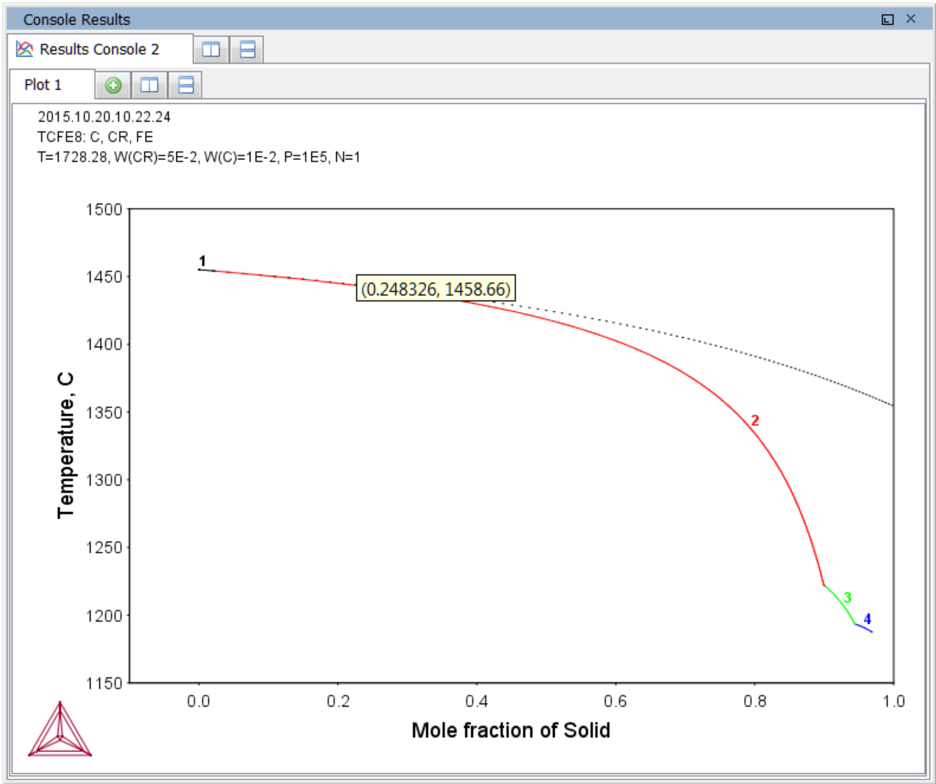
    Thermodynamic data about the alloy system you defined is retrieved from the database.

13. At the `Should any phase have a miscibility gap check` prompt, press <Enter> for `N` or answer `Y`.

    If you answer `Y`, you are prompted to specify the `Phase with miscibility gap`. Answer with a solution phase name(s) (in this example `FCC` is entered) as well as the `Major constituents for sublattice` (for each sublattice site) (for this example, `Cr` is entered for sublattice 1 and `C` for sublattice 2). In this example the `Phase with miscibility gap` is `LIQUID PHASE NAME: LIQUID`.

14. Press <Enter> to specify another phase.
15. Specify the names of any `Fast-diffusing components`. Press <Enter> for `NONE` or type all the names on the same line then press <Enter>. If you do specify any fast-diffusing components, then you are prompted to set whether BCC to FCC phase transformations should be simulated.

    The Scheil-Gulliver solidification simulation runs, and the default diagram (Temperature versus Mole Fraction of Solid) is automatically plotted in the Console Results window.



Plotting Additional Scheil Simulation Diagrams

Post processing is done the same way as POLY-3 (see *POLY_3 Commands* on page 117), specific variables for plotting in the Scheil Module are listed.

Abbreviations for Scheil State Variables

The variables can be plotted along the X- and Y-axes. In the table, *ph* is an abbreviation of *phase*, and *el* is an abbreviation of *element*. Use the abbreviations of the variables when you specify what the X- and Y-axes represent in a diagram.

Abbreviation	State variable
T	Temperature in Celsius
NL/BL	Mole/mass fraction of liquid
NS/BS	Mole/mass fraction of all solid phases
NS(ph)/BS(ph)	Mole/mass fraction of a solid phase
W(ph,el)	Weight fraction of an element in a phase
X(ph,el)	Mole fraction of an element in a phase

Abbreviation	State variable
Y(ph,el)	Site fraction of an element in a phase
NN(ph,el)	Distribution of an element in a phases
NH/BH	Heat release and Latent heat per mole/gram
CP/BCP	Apparent heat capacity per mole/gram
NV/NV(ph)	Molar volume of the system or a phase
DS/DS(ph)	Average density of the system or a phase
BT	Apparent volumetric TEC of the system

T0 Temperature Simulations

In this section:

T0 Temperature Simulations

The T0-temperature is the temperature at which two phases have the same Gibbs energy for a certain composition in a multicomponent system. The T_0-temperature is located within the two-phase field between the phases and it is the theoretical limit for a transformation without diffusion. T0-temperatures are often of interest if you want to understand how diffusionless transformations work. The T0 temperature in a multicomponent system with a fixed composition is temperature at which the tangent lines of the Gibbs energies of the two target phases are the same.

If the composition of one or two components varies, the common Gibbs energy for the two phases in partial equilibrium of a diffusionless transformation becomes a plane or surface, and the T0 temperature becomes a line or plane.

See Examples 23 and 41 in the *Console Mode Examples*.

Also see *Making a T0 Temperature Simulation* below for step-by-step instructions. as well as *Console Mode POST Module* on page 179.

Making a T0 Temperature Simulation

This describes how to perform a T0 temperature simulation and apply a stepping calculation to its initial result.

Before you perform a T0 temperature calculation, you must have defined your initial system and be in POLY. The initial overall composition must have appropriate settings for the desired T0 temperature calculation for the two target phases.

See *POLY_3 Commands* on page 117 in the *Thermo-Calc Console Mode Command Reference*.

How to Make a T0 Temperature Simulation

1. Use SET_CONDITION to set the conditions of your calculation in the same way as when you calculate an ordinary equilibrium.
2. Use COMPUTE_EQUILIBRIUM to run the calculation. The calculation does not need to reach an equilibrium in which any of the target phases is stable.
3. When you have calculated your initial equilibrium, use ADVANCED_OPTIONS and choose `T-zero`.
4. When prompted, specify the names of the two target phases for which the T0 temperature (where the phases' Gibbs energies are equal) is to be calculated.
5. If the T0 temperature between the two target phases is calculated, this temperature (in

Kelvin) is shown in the Console window.

6. If you want to do a stepping calculation based on this initial T0 temperature calculation, then use SET_AXIS_VARIABLE to specify which state variable to use as the stepping variable. This is done the same way as when you calculate a normal property diagram.

When a T0 temperature simulation is run, you cannot set temperature as the stepping variable.

7. Use STEP_WITH_OPTIONS and choose `T-zero` to initiate the stepping calculation. You are prompted to specify the names of the target phases for which the T0 temperature is calculated.

During the STEP T-ZERO calculation procedure, the calculated T0 values are presented next to the stepping variable.

For example, part of the output might look like this:

```
Phase Region from   1.000000E-01 for:
BCC_A2
FCC_A1
1.000000E-01         940.24
9.250000E-02         941.20
2.500000E-03         977.61
7.500000E-09         979.34
```

When the stepping calculation has finished, you can move to the POST module and plot the T0 line against the stepping composition variable or another varying axis value. Or you can impose the calculated T0 line onto a normal phase diagram.

Paraequilibrium

In this section:

About Paraequilibrium

A paraequilibrium is a partial equilibrium where one interstitial component (such as carbon C and nitrogen N) in a multicomponent alloy can diffuse much faster than the other components (the substitutional elements, including the matrix element and alloying elements). The chemical potential for the interstitial component in two partially equilibrated phases is equal in such a case, but this is not true for the other components. In such a paraequilibrium state, it is possible to have a transformation that is partly without partitions, where a new phase of the mobile component can form with different content but where the slowly diffusing components retain their compositions.

A paraequilibrium calculation is useful when, for example, you want to study phase transformations in systems with large differences in the diffusivities of different elements. Transformations occurring under paraequilibrium states can be much more rapid than if full local equilibrium holds at the phase interface.

See Example 42 in the *Console Mode Examples*.

Also see *Calculating a Paraequilibrium* below for step-by-step instructions.

Calculating a Paraequilibrium

This topic describes both how to calculate an initial paraequilibrium with two target phases, and how to base a stepping calculation on this initial calculation.

Before you do any paraequilibrium calculations, you must have defined your initial alloy system and be in POLY. The initial overall composition must have appropriate settings for the desired paraequilibrium calculation for the two target phases.

How to Calculate a Paraequilibrium

These commands are described in the *Thermo-Calc Console Mode Command Reference* included with this documentation set.

1. Use SET_CONDITION to set the conditions of your calculation just like when you calculate an ordinary equilibrium.
2. Use CHANGE_STATUS to set the status of the chosen interstitial components to SPECIAL. For example, if C is an interstitial component, enter:

```
CHANGE_STATUS COMPONENTS C=SPECIAL
```

   This gives you a clear picture on u-fractions of the substitutional and interstitial components, which are different from the overall composition in the system. The SPECIAL status means that the specified components are not included in summations for mole or mass fractions. Therefore, all the composition variables plotted from paraequilibrium calculations are u-fraction related quantities.

3. Use COMPUTE_EQUILIBRIUM to run the calculation.
4. When you have calculated your initial equilibrium, use ADVANCED_OPTIONS `Paraequilibrium` to perform the paraequilibrium calculation.
5. Specify the names of the target phases of the paraequilibrium state. For example, `FCC#1 BCC` and `FCC#2 M23C6`.

Both phases must have similar interstitial/vacancy sublattices that the fast-diffusion interstitial components occupy. Both target phases should have phase constitution definitions that cover all the defined substitutional and interstitial components of the current alloy system.

6. Specify the names of one or more fast-diffusing components. These components must be located on the interstitial/vacancy sublattices in both of the chosen phases.

Interstitial components (for instance C and N combined) may have significantly different behaviours depending on how they are partitioned in different structured phases.

If the paraequilibrium between the two specified phases is successfully calculated, then a message is displayed, for example:

```
NP(FCC) = 0.3586 with U-fractions C = 2.71821E-02 N = 4.1548129E-03
NP(BCC) = 0.6414 with U-fractions C = 7.10061E-04 N = 2.3781027E-04
All other compositions are the same in both phases
```

The first two lines show the phase amounts expressed in mole-percent [NP(phase)] and the contents of the interstitial components C and N in each phase. These contents are expressed in u-fractions. The third line states that the compositions of the matrix component and all the remaining compositions (regarding substitutional components) are equal in the two target phases at the paraequilibrium state.

7. If you want to do a stepping calculation based on this initial paraequilibrium, then use SET_AXIS_VARIABLE to specify which state variable to use as the stepping variable. This is done exactly in the same way as when you calculate a normal property diagram.
8. Use STEP_WITH_OPTIONS → `Paraequilibrium` to initiate the stepping calculation. You are prompted to specify the names of the target phases of the paraequilibrium states as well as the fast-diffusing components.

The stepping calculation is performed, and the different phase regions are listed with columns for the value of the stepping variable, the amounts of the two target phases, the u-fractions of interstitial(s) in each of the two phases, and the LNACR value(s) of the interstitial component(s).

Example

```
This command calculates a paraequilibrium between two phases.
You must calculate an equilibrium with the overall composition first.
Name of first phase: fcc_a1
Name of second phase: bcc_a2
Fast diffusing component: /C/: C
Fast diffusing component: /NONE/:
 Output during stepping is:
 axis value, phase amounts, u-fractions of interstitial(s) in phase 1 and 2,
 and LNACR value(s) of interstitial(s)

 Phase Region from    900.000     for:
     BCC_A2
     FCC_A1
   9.000000E+02    0.176    0.824   4.977501E-02   6.609390E-04  -2.303643E-01
   8.900000E+02    0.161    0.839   5.421135E-02   6.889794E-04  -1.689777E-02
   8.800000E+02    0.148    0.852   5.872152E-02   7.140931E-04   1.943989E-01
   8.700000E+02    0.137    0.863   6.329608E-02   7.362105E-04   4.040959E-01
   8.600000E+02    0.127    0.873   6.792661E-02   7.552871E-04   6.126989E-01
   8.500000E+02    0.119    0.881   7.260558E-02   7.713010E-04   8.206607E-01
   8.400000E+02    0.111    0.889   7.732629E-02   7.842505E-04   1.028392E+00
   8.300000E+02    0.105    0.895   8.208278E-02   7.941530E-04   1.236267E+00
   8.200000E+02    0.099    0.901   8.686971E-02   8.010431E-04   1.444633E+00
   8.100000E+02    0.093    0.907   9.168236E-02   8.049714E-04   1.653814E+00
   8.000000E+02    0.089    0.911   9.651648E-02   8.060032E-04   1.864114E+00
```

It is possible to now move to the POST module and plot some of the results of the calculation. For an example of a paraequilibrium calculation and how it can be plotted, see Example 23 in the *Console Mode Examples*.

Potential Diagrams

In this section:

About Potential Diagrams

If you have a metal-oxide/sulphide-gas interaction system, you might want to calculate a so-called potential phase diagram given certain temperature and pressure. A potential diagram uses the activities of two major species in the gaseous mixture phase as mapping variables. The activities (that is, the fugacities at 1 bar) of these two species are typically set as the X- and Y-axis when the diagram is plotted. The phase relations between the gaseous mixture and various metal forms, metal-oxides, metal-sulphides, or other metal-bearing solids, are represented by different fields controlled by these two potential quantities.

This topic describes how to calculate such potential diagrams with the POTENTIAL module, and how to later modify the pressure of such a calculation in POLY. The calculation done by the POTENTIAL module always assumes a pressure of 1 bar.

POTENTIAL_DIAGRAM Module on page 217 in the *Console Mode Command Reference*.

Calculating a Potential Diagram

You need access to a substance or solution database that contains a gaseous mixture phase, metals (or alloys), and solids such as metal-oxides, sulphides, sulphates, carbonates, silicates, nitrides or nitrates (stoichiometric or solution phases).

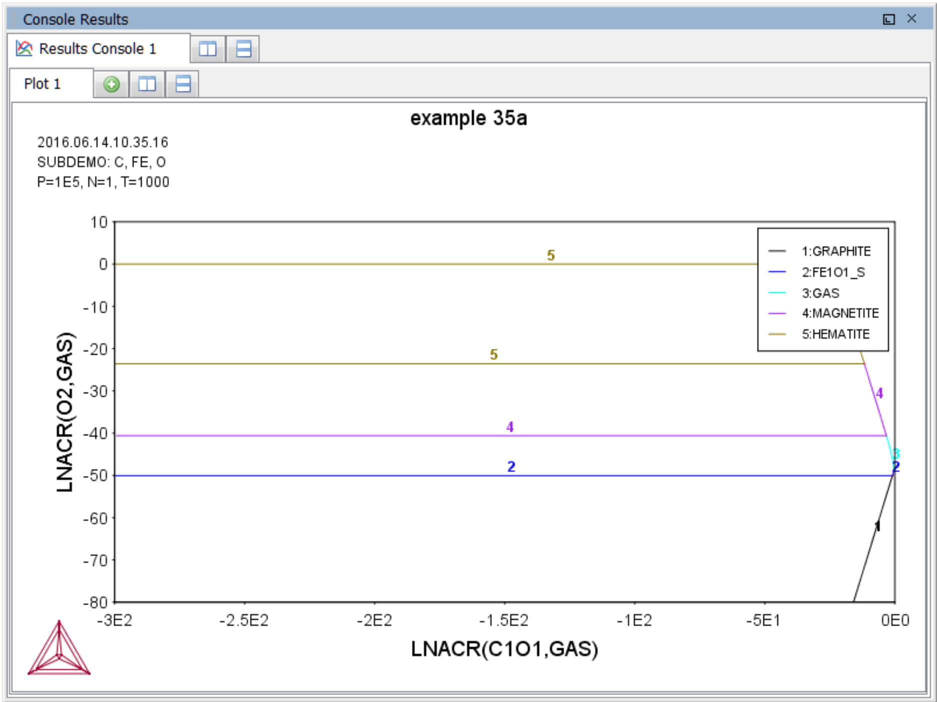
POTENTIAL_DIAGRAM Module on page 217 in the *Console Mode Command Reference*.

1. Use GOTO_MODULE and enter `POTENTIAL`.
2. Specify which substance or solution database to use. Any substance or solution database can be used that has a gaseous mixture phase, metals (or alloys), and solids such as metalxides, sulphides, sulphates, carbonates, silicates, nitrides, and nitrates (stoichiometric or solution phases).
3. Specify the matrix element (a metal) and the two gaseous species that define you metal-gas interaction system. The names of these components must be available in the database you are using.
4. Set the temperature (in Kelvin). (The pressure is always 1 bar.)

A potential diagram with the activities of the two gaseous species as mapping variables is calculated. These mapping variables are also set as the axis variables in the plotted diagram.

Example Plot

The following is an example diagram (from example 35 in the *Console Mode Examples*).



A POLY3-file is also automatically saved in the current working directory with the default name POT.POLY3.

5. When the diagram is plotted, it opens the POST module where you can modify the plotted diagram in any way you like.

Calculating a Potential Diagram with Different Pressure

It is possible to modify both the calculation and the plotted diagram that the POTENTIAL module generates in the POLY and POST modules. You can do this by opening a POLY3-file with the results from a POTENTIAL module calculation in either POLY or POST. You can modify the calculation in the same way that you can modify any ordinary mapping calculation or phase diagram. For example, this is useful for calculating a potential diagram at a pressure other than 1 bar.

POLY_3 Commands on page 117 in the *Console Mode Command Reference*.

You must have calculated a potential diagram already, and have the saved POLY3-file available.

1. Use GOTO_MODULE and enter `POLY`.
2. Use READ followed by the name of the POLY3-file that contains the results of the potential diagram calculation that you want to modify.

3. This opens the workspace file and loads the previous settings and results. The activities of the two gaseous species is now part of the conditions, and these conditions are set as the mapping variables.

4. Use SET_CONDITION to specify the pressure under which you want the potential diagram to be calculated. For example, to set the pressure to 1000 bar, enter:

```
SET_CONDITION P=1e8
```

5. Use COMPUTE_EQUILIBRIUM to compute the initial equilibrium.

   You may also want to use ADD_INITIAL_EQUILIBRIUM to calculate some additional equilibria that the mapping calculations are based on.

6. Use MAP to perform the mapping calculations.
7. Use GOTO_MODULE and enter `POST` to set conditions for visualization and to plot the diagram.

Aqueous Solutions

In this section:

About Aqueous Solutions

Heterogeneous equilibrium calculations involving aqueous solutions are not different from other equilibrium calculations, but the setup of the equilibrium conditions and the reference states of the system is more complex. For example, the pH and Eh values of the aqueous solution phase are normally complex functions of bulk composition, temperature and pressure conditions.

For some simple systems, thermodynamic approximations of pH-Eh calculations can be used. But Thermo-Calc also has comprehensive thermodynamic models and databases for various non-ideal solution phases. These let you accurately calculate pH and Eh property diagrams (and many other types of property diagrams) in complex aqueous solutions in a variety of conditions.

A Pourbaix diagram is a pH-Eh phase diagram where the tie-lines are not in the plane. The chemical potentials or activities of two system components (H+ and EA) are used as mapping variables and the pH and Eh quantities are normally plotted on the X- and Y-axes of the diagram. You can use many other properties as axis variables in various property diagrams for the interaction system at a specific temperature and pressure. The phase relations between the aqueous solution and various metal forms, oxides, hydroxides, sulphides, sulphates, nitrates, silicates, carbonates and other solids, or gaseous mixtures, are represented as different fields defined by acidity and electric potential.

You can calculate Pourbaix diagrams with the Pourbaix module. The compositional definition and variations and the basic solvent and electron components H2O, H+1 and ZE are automatically determined by the aqueous solution phase in the Pourbaix module.

Key Components

These key components are defined as follows:

Component	Component status	Reference state	Ref-T(K)	Ref-P(Pa)
H2O	ENTERED	AQUEOUS	*	100000
H+1	ENTERED	SER		
ZE	ENTERED	REFERENCE_ELECTRODE	*	100000

Besides the aqueous solution phase, your system may also contain a gaseous mixture phase. Other neutral and charged gaseous species may also be constituents. For multicomponent systems there are normally also some interacting solid phases (stoichiometric or solution) as the so-called secondary phases. These are typically the matrix and/or precipitate phases in the alloys of interest.

Some databases have been specially developed for the Pourbaix module’s diagram calculations. Such a database contains all necessary phases, i.e., an aqueous solution phase, a gaseous mixture phase, some alloy solid solution phases and some secondary solid phases. Databases developed for the Pourbaix module include *PAQ* (TCS Public Aqueous Solution Database) or *PAQS* (TGG Public Aqueous Solution Database).

If such a dedicated Pourbaix module is not used, then an aqueous solution database such as *TCAQ* (TC Aqueous Solution Database), *AQS* (TGG Aqueous Solution Database), or any user-specified aqueous databases can be used. This database contains at least the AQUEOUS and REFERENCE_ELECTRODE phases. The aqueous solution database needs to be appended with additional required data and phases, such as interacting alloy (solid) solution phases and secondary (solid and liquid) phases. Additional data is from one or more databases, such as SSUB/SSOL/TCFE or other substances/solutions databases.

Calculating a Pourbaix Diagram

The POURBAIX module lets you construct Pourbaix diagrams and other types of property diagrams for heterogeneous interaction systems with aqueous solution phases.

When calculating a Pourbaix diagram, you need to have a database with an aqueous solution phase (that is, a phase with water as the dissolving solvent and with aqueous cation/anion and complex species as the dissolved solutes). Due to restrictions of aqueous solution models (SIT, HKF, Pitzer) used in the Thermo-Calc software, the aqueous solution database must be designed in the same Thermo-Calc database format as that used in the PAQ, PAQS, TCAQ and AQS databases for the aqueous solution phase.

Pourbaix diagram calculations are done in the advanced Pourbaix module.

POURBAIX_DIAGRAM Commands on page 219 in the *Console Mode Command Reference*.

How to Calcuate a Pourbaix Diagram

1. At the `SYS` prompt, enter GOTO_MODULE POURBAIX.
2. At the `Need any instruction on the POURBAIX module?` prompt, press <Enter> to skip or enter `Y` to learn more.
3. Press <Enter> at the prompt `Enforce a PAUSE after plotting when running a MACRO?`.
4. Press <Enter> to accept the default `1` to `Start a completely new POURBAIX diagram calculation` or enter `3` to `Open an old file and make another POURBAIX calculation`.

   To choose `3`, you must have already done a Pourbaix calculation and have it saved in a POLY3-file. The new calculation uses the same system definition as the previous calculation, but the other conditions, such as the bulk composition, temperature and pressure, can be modified in the next step.

5. Type `Y` or `N` to set whether to `Consider the GAS phase` (gaseous mixture phase) in the calculation.

The gaseous mixture phase should at least contain H2O, O2 and H2. If a GAS phase is not considered, then only the interactions between the aqueous solution phase and various solid phases (alloy phases and secondary phases) are calculated. Such a calculation may not accurately present all the heterogeneous interactions. Some secondary phases are therefore usually needed. Such phases exist in the PAQ or PAQS Public Aqueous Solution Databases. They can also be appended from the SSUB/SSOL/TCFE or other appropriate substances/solutions databases.

6. Type `Y` or `N` to set whether to `Use single database` (`Y`) or a multiple database (`N`).

   If you choose to use a single database, at the `Combined Database: /PAQ2/` prompt specify a database that is specially developed for the Pourbaix module’s diagram calculations.

   If you choose to use a multiple database, then you are prompted to specify an aqueous solution database and to append any additional required data and select necessary phases, such as various interacting alloy (solid) solution phases, and secondary (solid and liquid) phases, for example.

7. Specify all your solutes and the molality of each solute when prompted to do so. This can be done in terms of elements (such as Fe, Ni, Na, Cl, C or S) or arbitrary complex species (such as NaCl, CO2, CoCl3, Fe0.93S, NaSO4-1, or H2SO4). You can enter the element/species and the molarity on one line (for an example, NaCl 2.5).

> The first letter of the element abbreviation must always be upper case. Any second letters must be lower case. When you have finished your specification, press <Enter> when asked to specify `Next solute`.

The definition of your system and its initial bulk composition is now presented.

8. Type `Y` to confirm the defined system and its bulk composition.

   The thermodynamic data for your system is now retrieved from the database(s) chosen.

9. Answer the questions that you are prompted regarding various parameters of your calculation. You are asked to do any of the following:

- `Rejectphase(s)` or `Restore phases(s)`
- Check for miscibility gaps on any phase (`Should any phase have a miscibility gap check?`)
- `Enforce the Global Minimization Technique in calculations?`
- Save all functions, variables and tables in POLY3 file?
- `Set numerical limits` on the calculation

- Confirm defined conditions?
- Accept the default settings for two mapping variables

8. The Pourbaix calculation starts. Answer the question `Any missing phase boundary you could possibly think of?` Type `Y` or `N` to set whether you want to add starting points to the calculation. If you suspect that the calculation has missed some phase boundaries, this may help the program to detect them.
9. Answer questions about how the diagram is plotted and how to save the results of your calculation. You are asked to do any of the following:

- Change the pH/Eh steps for smoother curves?
- `Zoom in` on a specific region of the diagram
- `Change the curve-label option for the diagram`
- `Add label-texts onto the Pourbaix diagram`
- `Change the subtitle of the diagram`
- `Change the axis-texts`
- `Further refine the diagram in the POST Module`
- Print the diagram (save a `Hard copy of the diagram`)
- `Save the X-Y coordinates of curve on a text file`
- `Modify the diagram`
- Calculate and plot another Pourbaix diagram (`Any more diagram?`)

Plotting Additional Aqueous Solution Diagrams

When you have calculated a Pourbaix diagram, you can plot additional diagrams based on the results of the same calculation, but with different axis variables than Ph and Eh.

The variables can be plotted along the X- and Y-axes. Use the abbreviations of the variables when you specify what the X- and Y-axes represent in a diagram. In the table, *AQsp* refers to the name of a specific aqueous species. The asterisk (*) can be used as a wild card character for all such species. If neither *AQsp* nor * is entered in parenthesis, then all species are searched.

You must have plotted a Pourbaix diagram to plot a diagram with other variables than Ph and Eh on the Y-axis and the X-axis. You can either plot such additional diagrams directly after having finished the plotting and saving/printing of a Poubaix Ph-Eh diagram or by choosing option `2` (Open an old file and plot other property diagrams) when you enter the POURBAIX module and opening a POLY3 workspace file.

Abbreviations for Aqueous Solution Variables

Abbreviation	Variable
pH	Acidity
Eh	Electronic Potential (V)
Ah	Electronic Affinity (kJ)
Pe	Electronic Activity (log10ACRe)
IS	Ionic Strength
TM	Total Concentration
Aw	Activity of Water
Oc	Osmotic Coefficient
MF(AQsp)	Mole Fractions of Aqueous Species
ML(AQsp)	Molalities of Aqueous Species
AI(AQsp)	Activities of Aqueous Species
RC(AQsp)	Activity Coefficients of Aqueous Species

How to Plot an Aqueous Solution Diagram

1. Type `Y` when prompted whether to plot any more diagrams, or select option 2 when you enter the POURBAIX module.
2. Specify which variable you want the X-axis to represent. Use the abbreviation specified in the preceding table. When you specify the name of an aqueous species, you can use the asterisk (*) to select all species.
3. Specify which variable you want the Y-axis to represent. The diagram is then automatically plotted.
4. Answer questions about how the diagram is plotted and how to save it. You are asked to do any of the following:

- change the axis-type (linear, logarithmic or inverse) on the X-axis or Y-axis
- zoom in on a specific region of the diagram
- change the curve-label option for the diagram
- add label-texts onto the Pourbaix diagram
- change the subtitle of the diagram
- change the axis-texts

- refine the diagram in the POST Module
- print the diagram (save a hard copy)
- save the X-Y coordinates of curve on a text file
- modify the plotted diagram
- calculate and plot another aqueous solution diagram

Stepping Calculations in an Aqueous Solution

Performing the stepping calculation overwrites all the previous settings and stepping results on the POLY3-file. To avoid this, use *SAVE_WORKSPACE* on page 232 to create a new workspace.

When you have calculated a Pourbaix diagram, you can use the saved results of the calculation (the POLY3-file), and make a property diagram based on stepping calculation.

Since the program normally sets the calculated initial equilibrium point as the starting point, the minimum value for the stepping variable must be specified as smaller or equal to the specific initial condition, and the maximum value as larger or equal to that condition. If pH is set as the stepping variable, the program automatically calculates the equilibrium constant in logarithmic (log10Ka) of the electrolysis reaction H2O (water) = H+ + OH- for the real aqueous solution under the current temperature-pressure-composition conditions, and thus this constant log10Ka value is considered as the highest pH in specifying the maximum value along stepping. Subsequently, confirm or modify an increment factor with a default value; for pressure and mole of a component, specify the step in logarithmic (log10P or log10N).

State Variable Abbreviations

You can use any of these state variables as a stepping variable.

State variable	Abbreviation
Temperature (K)	T
Pressure (bar)	P
Acidity	pH
Electronic potential (V)	Eh
Mole number of a dissolving component (such as Fe, Na, Cl, or NaCl) in 1 kg of water.	N(Comp) (for example, `N(Fe)`)

How to Do a Stepping Calculation

You must have plotted a Pourbaix diagram before making a stepping calculation on an aqueous solution.

1. Use GOTO_MODULE POURBAIX to enter the POURBAIX module.
2. Type `4` to Open an old file and make another STEPPING calculation.
3. Choose which POLY3-file to load.
4. Specify the conditions of the calculation by answering the questions that follows.

   The system definition is the same as in the previous Pourbaix calculation, but you can modify conditions such as bulk composition, temperature and pressure, and change some component definitions if necessary.

5. Specify which variable to use as the stepping variable (see the preceding table), its minimum and maximum value, and the length of each incremental step.

   The settings and stepping results are written to the opened POLY3-file, and a first default property diagram is plotted.

Pourbaix Examples TCEX40, 40A to 40E, and TCEX53

NOTES about TCEX40A and TCEX40E

In order to be able to subsequently run the TCEX40B (Main Option 2), TCEX40C (Main Option 3) and TCEX40D (Main Option 4), which open the previously-saved POURBAIX-type calculation POLY3 file, the user should make four copies of the resulting POURBAIX.POLY3 as TCEX40A.POLY3, TCEX40B.POLY3, TCEX40C.POLY3 and TCEX40D.POLY3.

However, this has to be properly done outsides of the TCCS software after the TCEX40A is executed. The reason is that, if saving the POLY3 workspace of the POURBAIX calculation results in the POLY3-module monitor (through the command sequence "GO PLOY3" and "SAVE TCEX40A.POLY3 Y"), all the initial equilibrium points and the mapping/stepping calculation results will be lost, and parts of the POLY3 file structure (e.g., some definitions of the previously defined POURBAIX-type calculation system) will be destroyed. In principle, before executing any Main Option (for calculations and graphical processing), it is highly recommended that the user shall save/rename/copy the previously-generated POLY3 file as another POLY3 file. Only in this way, it can avoid destroying the previous-saved POURBAIX-type calculation results.

Under Windows DOS-session (outside TCCS):

- copy POURBAIX.POLY3 TCEX40A.POLY3 (or TCEX40E.POLY3)
- copy POURBAIX.POLY3 TCEX40B.POLY3
- copy POURBAIX.POLY3 TCEX40C.POLY3
- copy POURBAIX.POLY3 TCEX40D.POLY3

or inside TCCS (under Windows NT/2000/XP, but not Windows 95/98/ME):

- @copy POURBAIX.POLY3 TCEX40A.POLY3 (or TCEX40E.POLY3)
- @copy POURBAIX.POLY3 TCEX40B.POLY3
- @copy POURBAIX.POLY3 TCEX40C.POLY3
- @copy POURBAIX.POLY3 TCEX40D.POLY3

Under Linux/UNIX platforms (outside TCCS):

- cp POURBAIX.poly3 TCEX40A.poly3 (or TCEX40E.poly3)
- cp POURBAIX.poly3 TCEX40B.poly3
- cp POURBAIX.poly3 TCEX40C.poly3
- cp POURBAIX.poly3 TCEX40D.poly3

Example TCEX40

- Using PAQ2 or PAQS2 database;
- For the Fe-H2O-NaCl heterogeneous interaction systems
- With the Main Option 1
- Using the Single-Database Option, i.e., retrieving data from the PAQ2 [TCS Public Aqueous Solution (SIT) Database, v2.4; using the SIT aqueous solution model] or PAQS2 [TCS Public Aqueous Solution Database, v2.4; using the Complete Revised HKF aqueous solution model]; both of which contain an AQUEOUS solution phase and REF_ ELECTRODE phase (as a reference for electron in aqueous electrolyte systems), as well as some data for various solid phases (solution or stoichiometric) and gaseous mixture phase. The PAQ2 and PAQS2 are specially designed for demonstrations of calculations of the so-called Pourbaix diagrams (i.e., pH-Eh plots) and other types of phase diagrams or property diagrams in some aqueous-bearing multicomponent heterogeneous interaction systems within the limited chemical framework of Fe-Co-Cr-Ni-C-N-S-H2O-NaCl, via the Single-Database Option in the advanced POURBAIX module or through the normal TDB-GES-PLOY-POST routine.

To further run the following Main Options:

- Main Option 2 for plotting many other property diagrams of the calculated interaction system;
- Main Option 3 for making another POURBAIX calculation of the same chemical system but under a different P-T-X condition;
- Main Option 4 for making a POLY3 STEPPING calculation of the same chemical system but varied with only one independent variable;

** The Main Options 2, 3 and 4 are not illustrated in this TCEX40 example.

Five more examples (TCEX40A, TCEX40B, TCEX40C, TCEX40D and TCEX40E) are also provided to demonstrate various options and features in the POURBAIX module and to show more advanced applications of the TCAQ2 and AQS2 aqueous solution databases [in the Multiple-Database Option, i.e., with appended data from the SSOL5 and SSUB5 databases].

Example TCEX40A

- Option 1
- Using the Single-Database Option, i.e., retrieving data from the PAQ2 [TCS Public Aqueous Solution (SIT) Database, v2.4; using the SIT aqueous solution model] or PAQS2 [TCS Public Aqueous Solution Database, v2.4; using the Complete Revised HKF aqueous solution model]; both of which contain an AQUEOUS solution phase and REF_ELECTRODE phase (as a reference for electron in aqueous electrolyte systems), as well as some data for various solid phases (solution or stoichiometric) and gaseous mixture phase. The PAQ2 and PAQS2 are specially designed for calculations of the so-called Pourbaix diagrams (i.e., Eh-pH plots) and other types of diagrams in aqueous-bearing multicomponent heterogeneous interaction systems, via the Single-Database Option in the POURBAIX module or through the normal TDB-GES-PLOY-POST routine.
- A POLY3 file called POURBAIX.POLY3 is automatically saved.
- For the purposes of recalling this file in other Main Options (as in the TCEX40B, TCEX40C & TCEX40D), the POURBAIX.POLY3 has to be renamed or copied as another file name (in this particular example, as TCEX40A.POLY3, and TCEX40B.POLY3, TCEX40C.POLY3 & TCEX40D.POLY3). Note that the renaming or copying has to be done outside TCCS, otherwise the POLY3 file structure will be destroyed. However, under WindowsNT/2000/XP environments, one can use MS-DOS commands to do so, e.g., @copy POURBAIX.POLY3 TCEX40B.POLY3
- The Main Option 1 also provides the opportunity to plot many property diagrams along phase boundaries, after the default Pourbaix diagram is generated and refined (not necessarily going through the Main Option 2).

Example TCEX40B

- Demonstration of the Main Option 2 Graphical Processing (by plotting more diagrams); various options for refining a plotted diagram (including direct interactions with the POST Module); Graphical processing of related property diagrams.
- Reading a previously saved POURBAIX.POLY3 file (or renamed as another name, for instance, the TCEX40B.POLY3), and then plotting some property diagrams for the previously calculated system.

- The loaded POLY3 file will not be modified by this Main Option. It is thus not necessary to save the file in another name; however, it is still recommended that the user to copy the TCEX40A.POLY3 (or TCEX40E.POLY3) as TCEX40B.POLY3 before the Main Option 2 is executed in this particular example.

Example TCEX40C

- Demonstration of the Main Option 3 Calculation (by modifying T-P-X conditions); graphical processing of a Pourbaix diagram.
- Reading a previously saved POURBAIX.POLY3 file (or renamed as another name, for instance, the TCEX40C.POLY3), and then making another POURBAIX-type calculation and graphical processing of a new Pourbaix diagram for the same chemical system but under a different P-T-X condition.
- The loaded POLY3 file will be rewritten according to the T-P-X settings in the new POURBAIX-type calculation. It is thus recommended to save the file in a new name outside of TCCS before this Main Option is proceeded (in this particular example, the TCEX40A.POLY3 is copied to TCEX40C.POLY3).
- The Main Option 3 also provides the opportunity to plot many property diagrams along phase boundaries, after the default Pourbaix diagram is generated and refined (not necessarily going through the Main Option 2).

Example TCEX40D

- Demonstration of the Main Option 4 Calculation (by utilizing various stepping variables); graphical processing of related property diagrams.
- Reading a previously saved POURBAIX.POLY3 file (or renamed as another name, for instance, the TCEX40D.POLY3), and then making some normal POLY3 STEPPING CALCULATIONs.
- The loaded POLY3 file will be rewritten according to the T-P-X and stepping variable settings in the stepping calculation. It is thus recommended to save the file in a new file name outside of TCCS before this Main Option is proceeded (in this particular example, the TCEX40A.POLY3 is copied to TCEX40D.POLY3).
- The resulting POLY3 file (in this example, TCEX40D.POLY3) can be subsequently called by a Main Option 2 for plotting more property diagrams, or by another Main Option 4 for making other stepping calculations.
- The Main Option 4 also provides the opportunity to plot many property diagrams after the default Pourbaix diagram is generated and refined (not necessarily going through the Main Option 2).

- Exploration of the possibility of modifying the definitions for some desired system-components, through the direct and internal POURBAIX-POLY3 intersection, for a STEPPING (and/or MAPPING) calculation inside the POURBAIX Module.

Example TCEX40E

- Demonstration of the Main Option 1 Calculation with the Multiple-Database Option; graphical processing of a Pourbaix diagram.

  For a more complex application than in Example 40A.

  This example requires licenses to three types of commercial Thermo-Calc databases otherwise it cannot be run, i.e.,

  - A: TCAQ2 (TCS Aqueous Solution Database, v2.5; SIT model) or AQS2 (TGG Aqueous Solution Database, v2.5; HKF model);
  - B: SSUB4 (SGTE Pure Substances Database, v4) or TCMP2 (TCS Materials Processing Database, v2);
  - C: SSOL4 (SGTE Alloy Solutions Database, v4) or TCFE6 (TCS Steels/Fe-Alloys Database, v6) or ALIKE.

- This particular example uses the multiple-database combination of TCAQ2+SSUB4+SSOL4; among them, TCAQ2 database contains only an AQUEOUS solution phase (using the SIT model) and REF_ELECTRODE phase (as a reference state for electron in aqueous electrolyte systems), SSUB4 database is appended for the GASeous mixture phase and some pure solid phases (so-called secondary phases formed during aqueous-involving interaction processes), and SSOL database is also appended for various solid solution phases.
- A POLY3 file called POURBAIX.POLY3 is automatically saved. By reading this POURBAIX.POLY3 file (or renamed as another name), one can further run the following Main Options: Main Option 2 for plotting many other property diagrams of the calculated interaction system; Main Option 3 for making another POURBAIX calculation of the same chemical system but under a different P-T-X condition; Main Option 4 for making a POLY3 STEPPING calculation of the same chemical system but varied with only one independent variable; The Main Options 2, 3 and 4 are illustrated in the TCEX40B, TCEX40C and TCEX40D examples.
- For the purposes of recalling this file in other Main Options (as in the TCEX40B, TCEX40C & TCEX40D), the POURBAIX.POLY3 has to be renamed or copied as another file name (in this particular example, as TCEX40A.POLY3, and TCEX40B.POLY3, TCEX40C.POLY3 & TCEX40D.POLY3). Note that the renaming or copying has to be done outside TCCS, otherwise the POLY3 file structure will be destroyed. However, under Windows environments,

one can use MS-DOS commands to do so, e.g., @copy POURBAIX.POLY3 TCEX40B.POLY3

- The Main Option 1 also provides the opportunity to plot many property diagrams along phase boundaries, after the default Pourbaix diagram is generated and refined (not necessarily going through the Main Option 2).
- The POURBAIX module inside the TCCS software also accepts the AQS2 aqueous solution database (using the complete Revised HKF model) in the Multiple-Database Option, in addition to the TCAQ2 aqueous solution database (using the SIT model).

Example TCEX53

- Using PAQ2 or PAQS2 database; for the Fe-X-H2O-NaCl heterogeneous interaction systems (X = Cr-Ni-Co)

Note: The PAQ2 (TCS Public Aqueous Solution (SIT) Database, v2.4; using the SIT aqueous solution model) or PAQS2 (TCS Public Aqueous Solution Database, v2.4; using the Complete Revised HKF aqueous solution model) contains an AQUEOUS solution phase and REF_ELECTRODE phase (as a reference for electron in aqueous electrolyte systems), as well as some data for various solid phases (solution or stoichiometric) and gaseous mixture phase. Therefore, it can be used, via the Single-Database Option in the POURBAIX module or through the normal TDB-GES-PLOY-POST routine, for calculations of the so-called Pourbaix diagrams (i.e., pH-Eh plots) and other types of diagrams in aqueous-bearing multicomponent heterogeneous interaction systems.

Note: The initial bulk composition of Fe-based alloy in this demonstrating calculation is just preliminarily assigned, in which the BCC_A2 and/or FCC_A1 solution phase(s) are considered as of primarily interest. For practical calculations, one shall have more precise inputs for the initial bulk compositions of alloys.

Console Mode TAB Module

In this section:

Tabulation of Chemical Substances, Phases or Reactions

The TAB module can be used to tabulate thermodynamic functions of any type of substance, stoichiometric phase or solution phase, or reaction. The module can tabulate thermodynamic functions of pure stoichiometric phases or solution phases at fixed composition, as well as various kinds of partial derivatives for a pure stoichiometric phase or solution phase at a given composition.

The TAB module begins by evaluating the most stable species of a substance in either a homogeneous state or in a reaction. Results are presented in a table or plotted as a graph.

Also see *Tabulating a Reaction* on the next page.

A reaction tabulation presents you with the following properties, on rows representing different temperature levels:

- Heat capacity change / Delta-Cp (Joule/K) in column 2.
- Enthalpy change / Delta-H (Joule) in column 3.
- Entropy change / Delta-S (Joule/K) in column 4.
- Gibbs energy change / Delta-G (Joule) in column 5.

In the case of a tabulation of a substance or solution phase at a fixed composition, the properties presented are the heat capacity, enthalpy, entropy and Gibbs energy, rather than the degree of change in those properties.

Also see *Tabulating a Substance or Solution Phase at Fixed Composition* on page 177.

You can add a 6th column to the table that shows the values of a property or user-defined function of your choice.

Reaction Tabulation Example

```
*********************************************************************
   T       Delta-Cp       Delta-H        Delta-S        Delta-G
  (K)     (Joule/K)       (Joule)       (Joule/K)       (Joule)
*********************************************************************
 298.15  -4.44006E+01  -9.18800E+04  -1.98115E+02  -3.28120E+04
 300.00  -4.43267E+01  -9.19621E+04  -1.98389E+02  -3.24452E+04
 400.00  -3.92294E+01  -9.61533E+04  -2.10482E+02  -1.19604E+04
 500.00  -3.34122E+01  -9.97861E+04  -2.18613E+02   9.52022E+03
 600.00  -2.77768E+01  -1.02842E+05  -2.24200E+02   3.16779E+04
 700.00  -2.26324E+01  -1.05358E+05  -2.28088E+02   5.43040E+04
 800.00  -1.81080E+01  -1.07390E+05  -2.30808E+02   7.72568E+04
 900.00  -1.41889E+01  -1.09000E+05  -2.32710E+02   1.00438E+05
1000.00  -1.08095E+01  -1.10245E+05  -2.34025E+02   1.23779E+05
1100.00  -7.77802E+00  -1.11169E+05  -2.34908E+02   1.47229E+05
1200.00  -5.07556E+00  -1.11807E+05  -2.35464E+02   1.70750E+05
1300.00  -2.93467E+00  -1.12203E+05  -2.35782E+02   1.94314E+05
1400.00  -1.19414E+00  -1.12407E+05  -2.35934E+02   2.17901E+05
1500.00   2.55400E-01  -1.12452E+05  -2.35966E+02   2.41497E+05
1600.00   1.49022E+00  -1.12363E+05  -2.35909E+02   2.65091E+05
1700.00   2.56484E+00  -1.12159E+05  -2.35785E+02   2.88676E+05
1800.00   3.51909E+00  -1.11854E+05  -2.35611E+02   3.12246E+05
1900.00   4.38259E+00  -1.11458E+05  -2.35397E+02   3.35797E+05
2000.00   5.17775E+00  -1.10980E+05  -2.35152E+02   3.59325E+05
```

Tabulating a Reaction

A tabulation of a given chemical reaction provides you with data about the rate of change of the standard tabulation properties at various levels of temperature. You can also add a column for a property of your choice to the table.

TABULATION_REACTION Commands on page 245 in the *Console Mode Command Reference*.

How to Tabulate a Reaction

1. Use GOTO_MODULE TAB to enter the TAB module.
2. If you want to add one thermodynamic functions to the tabulation calculation, which is present in a sixth column in the table, use ENTER_FUNCTION.

   You are prompted to enter the name of the column that is given at its head, and the function itself. The following state variables can be used to define the function: G, H, S, T, P, V and H298.

3. Use TABULATE_REACTION followed by a formula that specifies of the reacting species and products to perform the tabulation calculation. The name of the elements must be entered in upper case letters only. Terminate the chemical reaction with a semi-colon. For example, you can enter:

    ```
    TABULATE_REACTION 3H2+N2=2N1H3;
    ```

    Elements that are designated with a single letter must be followed by a stoichiometry factor, even if this factor is 1. `CO` hence be interpreted as Cobalt rather than as carbon monoxide. Carbon monoxide is entered as `C1O1` or `O1C1`.

4. Set the pressure, the range of temperature that you want the tabulation to cover, and change in temperature between each row in the table. You can also specify if you want to output the calculation results to an EXP-file and whether you want any graphical output, and if so, which column you want to be plotted against temperature.

    The tabulation calculation is performed and you are given a table that shows the various properties at each temperature level within the range defined.

    If you have chosen to receive graphical output, then the Console Results window presents a graph that plots the column of you chose (on the Y-axis) against temperature (on the X-axis). When this graph is plotted, you are now in the POST module, where you can modify the graph in various ways.

Tabulating a Substance or Solution Phase at Fixed Composition

A tabulation of a substance or solution phase at fixed composition provides you with data about the standard tabulation properties at various levels of temperature for a given chemical reaction. You can also add one extra property column of your choice to the table.

TABULATION_REACTION Commands on page 245 in the *Console Mode Command Reference*.

How to Tabulate a Substance or Solution Phase at Fixed Composition

1. Use GOTO_MODULE TAB to enter the TAB module.
2. Use SWITCH_DATABASE to switch to a database that contains solution-based data, such as TCFE for example.

You cannot use the default SSUB database when tabulating substances or solution phases at a fixed composition.

3. If you want to add one thermodynamic function to the tabulation calculation, which is

present in a sixth column in the table, use ENTER_FUNCTION.

You are prompted to specify the column header and the function itself. The following state variables can be used to define the function: G, H, S, T, P, V and H298.

4. Use TABULATE_SUBSTANCE followed by the name of the substance or solution phase to perform the tabulation calculation.
5. Set the pressure, the range of temperature that you want the tabulation to cover, and change in temperature between each row in the table. You can also specify if you want to output the calculation results to an EXP-file and whether you want any graphical output, and if so, which column you want to be plotted against temperature.

The tabulation calculation is performed and a table shows the various properties at each temperature level within the defined range.

If you have chosen to receive graphical output, then the Console Results window presents a graph that plots the column you chose (on the Y-axis) against temperature (on the X-axis).). When this graph is plotted, you end up in the POST module, where you can modify the graph in other ways.

Console Mode POST Module

In this section:

Console Mode Visualization

The results of a calculation in Thermo-Calc can be plotted by the post-processor module, also called the POST module. The diagram is presented in the Console Results window.

In the POST module, you can define and plot various kinds of diagrams based on a calculation. You can modify diagram type, which variables to plot on each axis and between which minimum and maximum values, append experimental data from an EXP-file, change labels and symbols in the diagram, zoom in on (and out from) a certain region in a diagram, etc.

If you calculate a property or phase diagram in POLY, then you have to open the POST module yourself. If your calculation was performed by a response-driven module such as SCHEIL or POURBAIX, then the plotting of a diagram based your calculation is part of the dialogue of questions and answers you have in that module (but under the hood, the response-driven module is actually utilizing the POST module to get the plotting done).

Plotting Diagrams

Plotting a diagram is to graphically represent how several variables (typically two variables) co-vary. Each axis in a property diagram or a phase diagram represents the value of one variable. The default variables represented by the X- and Y-axes are the following:

- If you plot a diagram based on the results of a mapping operation (a phase diagram), then by default the X- and Y-axes represent the mapping axis variables 1 and 2 (set using *SET_AXIS_VARIABLE* on page 164 in POLY).
- If you plot a diagram based on the results of a stepping operation (a property diagram), then by default, the X-axis represents the stepping variable (set using SET_AXIS_VARIABLE) and the Y-axis represents the sum of the mole fractions of all phases (that is, NP(*)).

POLY_3 Commands on page 117 in the *Console Mode Command Reference*.

How to Plot a Diagram

To plot a diagram, you must have already made a stepping or mapping operation in POLY (or in some cases, made a calculation as part of some opening a response-driven module), and you must have entered the POST module.

The POST command is available inside POLY. Use POST to enter the POST module

1. If you want to directly plot your diagram with the default variables represented by the X- and Y-axes, type `Plot` and press <Enter>. This plots a property or phase diagram with the default variables represented at each axis. If you do not want to plot your diagram with these default variables, go to the next step.

2. Use SET_DIAGRAM_AXIS to set which variables to plot on the X-axis and the Y-axis. For example, if you want to plot a phase diagram with pressure on the Y-axis and temperature on the X-axis, type:

```
SET_DIAGRAM_AXIS X T
SET_DIAGRAM_AXIS Y P
```

3. Use PLOT_DIAGRAM to plot your diagram.

Modifying Diagrams

Modifying a diagram amounts to changing the settings that determine how the POST module presents the calculated results and plots a new diagram. Suppose that you have already plotted a binary phase diagram with the mapping axis variables temperature and mole fraction of one of your components (the Fe component, say) on the X- and Y-axes. If you then want to modify your diagram so that the Y-axis instead represents, say, the mass fraction of Fe rather than the mole fraction, then you use *SET_DIAGRAM_AXIS* on page 206 to set diagram axis `Y` to `w(Fe)`, and then use *PLOT_DIAGRAM* on page 200.

There are many ways in which you can modify your diagram and how it is presented. Any modification made in the plot settings is taken into account the next time you use PLOT.

Modification Examples

The following commands are described in the *Thermo-Calc Console Mode Command Reference*.

The following are some of the modifications you can make.

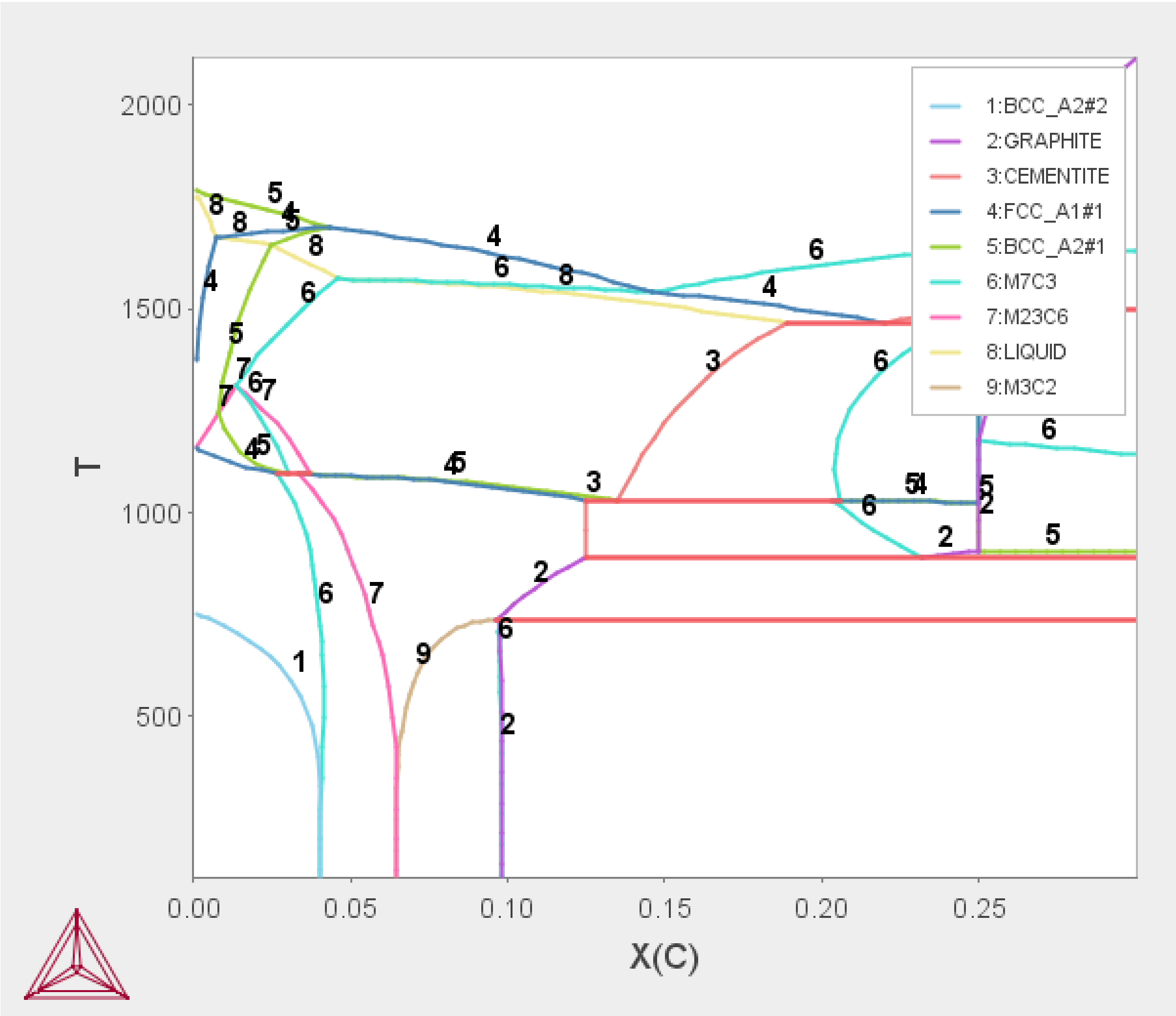
- Use *SET_TITLE* on page 215 to set add a title or to change an already existing title. The title is displayed above the plotted diagram in the Console Results window.
- Use *ADD_LABEL_TEXT* on page 190 to add a label text that starts at a certain X- and Y-coordinate. If you have plotted a phase diagram, then you can let Thermo-Calc set the label text that specifies the names of the stable phases at that coordinate.
- Use *SET_AXIS_TYPE* on page 204 to set whether the scale of an axis is linear, logarithmic or inverse.
- Use *SET_DIAGRAM_TYPE* on page 208 to plot a triangular diagram with the X-axis on the triangle’s base and the Y-axis on the triangle’s left side. Such a diagram is often useful if you want to plot the fractions of two components in a ternary diagram.
- Use *SET_SCALING_STATUS* on page 214 to modify the range of values that are displayed on an axis. In effect, you can use it to zoom in or out of the diagram. For example, to set the scale of the X-axis to range from 0 to 0 .3, type:

  ```
  SET_SCALING_STATUS X N 0 0.3
  ```

The `X` specifies which axis to change the scaling on, the `N` turns off Thermo-Calc's automatic scaling, and `0` and `0.3` specifies the minimum and maximum values on the scale. You can turn on automatic scaling again by setting the second parameter to `Y`.

SET_LABEL_CURVE_OPTION Example

Use *SET_LABEL_CURVE_OPTION* on page 209 to label each set of curves that have the same stable phases with a certain digit. A legend is also added, where you specify which phases each digit designates.



LTEXT vs LaTeX Commands for Labels

As of Thermo-Calc 2015a the LTEXT text formatting is replaced by LaTeX. For existing users who may have plots and files that use the LTEXT format, and especially when using Console Mode, the following examples show the command changes from LTEXT and its LaTeX equivalent:

Symbol or text format	LTEXT	LaTeX
Subscripts in an axis text	S-A-TEXT Y N Mole-fraction Al^DO2O^DO3	S-A-TEXT Y N \latex Mole-fraction Al_2O_3
Label with subscript	add .13 .15 n c-ZrO^DO2$	add .13 .15 n \latex c-ZrO_2
Greek symbol	ADD .05 .50 N ^GRd$	ADD .05 .50 N \latex \delta

Also see *Changing Global Settings* on page 187.

Saving Diagrams

When a diagram is plotted, there are several ways in which you can save it:

- Use *DUMP_DIAGRAM* on page 194 to save the diagram image to a PS, PDF, SVG, EMF, GIF, or PNG file. You are asked what format to save it in.
- To save the diagram image in a variety of image file formats, right click the diagram and select **Save Plot**.
- Use *PRINT_DIAGRAM* on page 201 to print a hard copy of the diagram.

Information about the underlying calculations that the diagram is based on is not saved in an EXP-file.

Also see the *DATAPLOT User Guide* included with this documentation set.

Loading Saved Diagrams

If you have previously saved a diagram in an EXP-file, then you can load and superimpose the diagram on another diagram. Both the following command requires you to specify which prologue and dataset(s) that are loaded from an experimental data file.

Use *APPEND_EXPERIMENTAL_DATA* on page 192 to plot selected data from an experimental data file (EXP-file). If you have set the axes of the diagram already, the labels on the diagram are not changed when you plot the additional data from the EXP-file, irrespectively of the data in that file.

Use *QUICK_EXPERIMENTAL_PLOT* on page 201 to plot selected data from an experimental data file (EXP-file). If you have set the axes of the diagram already, these settings are overwritten with the axes settings from the EXP-file.

About the Database Files

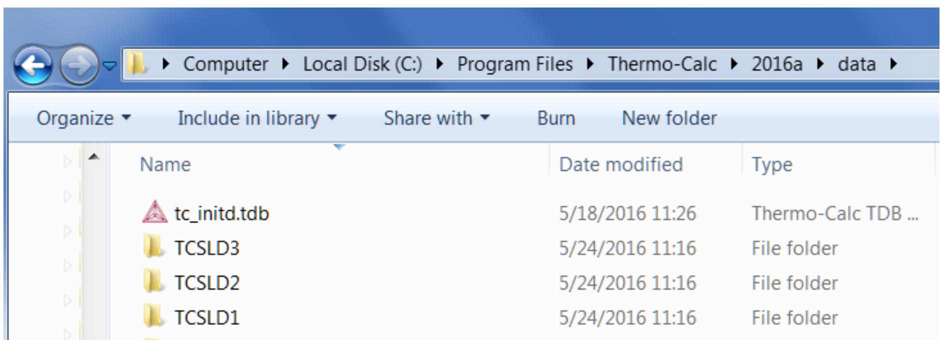
Encrypted vs Unencrypted Database Files

Any licensed databases purchased from Thermo-Calc Software are encrypted in files with the extension `.TDC`; these files cannot be edited. These are referred to as *encrypted databases*.

Unencrypted databases have the file extension `.TDB` and can be edited. For example, PURE5 or PG35 (the free databases) or a user-defined database file. If you plan to edit any of these files, see *Editing the Database Initialization File or Unencrypted Database File* on the next page for information.

Database Installation Folder

Depending on the type of license and operating system, the encrypted Thermo-Calc databases and the database initialization file are located in a subfolder to the Thermo-Calc installation. For example, in a Windows standalone installation, you can find these in a folder called **data**(file path is *C:\Program Files\Thermo-Calc\<version>\data*).



Database Initialization File

In Windows, the *database initialization file* is called `tc_initd.tdb`. In Linux and Mac it is called `initd.tdb`. The file contains a list of all the databases available from Thermo-Calc Software. Individual licenses are still required to use each database.

The file is used differently by the program depending on whether you are working with databases in Console Mode or Graphical Mode. For this reason, editing the file can be problematic if you don't understand the differences. See *About Cached Database Files (Graphical Mode only)* on the next page and *Editing the Database Initialization File or Unencrypted Database File* on the next page for more information.

User Database Files

A user-defined database is considered to be an unencrypted database because you can make changes to it.

In Graphical Mode you can add user databases via the **Options → Graphical Mode → System definer** window. Once you have added the user database it is available from the database list in the System Definer Configuration window. See *Global Settings: System Definition (System Definer)* on page 191 to add a user database.

In Console Mode, users may decide to add their own user-defined database to the database initiation file, which then makes it available in the Console window. If you plan to edit this file, see *Editing the Database Initialization File or Unencrypted Database File* below for important information.

About Cached Database Files (Graphical Mode only)

Also see Global Settings *General* on page 188 where you can set the default database directory path.

The first time you open a database in Graphical Mode it can take a little while for it to load, especially if it is one of the large databases such as TCFE or TCNI. However, the next time the database opens quickly because now the databases are cached, or stored, in Graphical Mode.

Think of a cache as an intermediate storage area. Once a database cache is established the file made easily accessible, even if you close the software program. It is the same concept as when you load a web page for the first time. The next time the page opens faster. This is also how it works for the databases in Thermo-Calc Graphical Mode.

Thermo-Calc caches the database files, both encrypted and unencrypted types, at two points –when you first open a database, sometimes called an *in-memory cache*, and then later as a *disk-based cache.*. The disk-based cache is what speeds up database loading and it is at this point that Thermo-Calc refers to the database initialization file for information. The disk-based cache uses a lookup key to find the abbreviated database name (e.g. TCFE8 or TCNI8) contained in this initialization file.

It is important to understand the way databases are cached in Graphical Mode to ensure best practices for those who also use Console Mode, especially if the database initialization file is edited. If you plan to edit this file, see Editing a Database Initialization File or Unencrypted Database File for important information.

Editing the Database Initialization File or Unencrypted Database File

Some advanced users may decide to edit the database initialization file, for example to reorder the list of encrypted databases displaying in the Console window or to add their own user-defined database for use in Console Mode. It is important to understand that changes to this file impact the database caching process used in Graphical Mode (see *About Cached Database Files (Graphical Mode only)* above).

The following recommendations are also applicable if you edit an unencrypted database file, which includes user databases and free databases such as PURE5 or PG35.

Best Practice Recommendations

If you edit the database initialization file (tc.initd.tdb in Windows or initd.tdb in Mac and Linux), then:

- If a user database is added to the file, it is subject to the same caching as the encrypted databases. This means that further changes (to the user database) will not be detected when you are working with databases in the System Definer in Graphical Mode.

If you have a user database, which is subject to regular updates, then:

- If you make changes to a user database, you then need to clear the cache.
- It is recommended that you create a separate directory to store unencrypted database files.

If you edit an unencrypted database file included with the installation (such as the PURE5 and PG35 files):

- Do not edit the file in the installation folder. Instead make a local copy and load it into the application as a user database.

Database Checker

The Thermo-Calc software package includes a program to check that the syntax of unencrypted database files is correct. The program applies the syntax rules set out in the *Database Manager User Guide* and reports errors and issues warnings. This program is intended for advanced users who develop and manage databases.

Thermo-Calc accepts deviations from these syntax rules. This means that a database can work even if the Database Checker reports errors and warnings. For example, an error is reported if an abbreviated phase name is found, but phase name abbreviations are accepted by Thermo-Calc and its add-on modules.

The executable **DatabaseChecker** file is found in the Thermo-Calc home directory. The program can also be launched by selecting **Tools→ Database Checker** from the main menu.

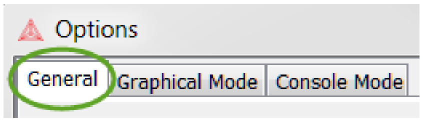
Also see the Database Manager User Guide.

Changing Global Settings

In the **Options** window (select **Tools → Options**) you can globally set general settings for Graphical Mode as well as default settings for any new activities for both Graphical Mode and Console Mode.

In this section:

General



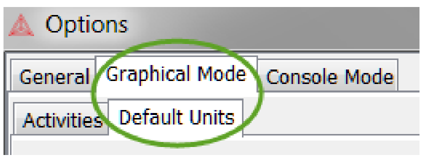
These settings are for both Graphical and Console modes. To open this window, from the main menu select **Tools → Options**.

General Global Settings

Setting	Options
Tooltips enabled	Select whether to turn on tooltips information. By default the **Tooltips enabled** check box is selected. This displays a small text box when you hover the cursor above some buttons or other items.
Localization	To change the GUI language from the **Localization** list choose **English** (the default), **Swedish**, **Spanish**, **German**, **Russian**, **Chinese**, **Japanese** or **French**.
Look and feel	To change the **Look and feel** of the GUI layout, choose **Windows** (the default), **Metal**, **Nimbus**, **CDE/Motif** or **WindowsClassic**.
Database directory	In the **Database directory** field, specify the path to the directory that contains the **data** directory. This is where the Thermo-Calc database directory called **data** is located. Do not specify the path to where the database files are found (that is, in the **data** directory). **Modifying the Database Directory Path (Graphical Mode Only)** Important Note: Database Directory Path (Graphical Mode only) When you open a database for the first time, it is stored in a cache (an intermediate storage area), that is maintained even if you close the program. This improves the speed of loading the next time you open that database. The cache is used for all databases that are listed in the database initialization file, typically all the standard databases that are included with Thermo-Calc. If you modify your own user-defined database, or if you choose to edit an unencrypted database, it is recommended that you create a separate directory to store these database files. This way the most recent version of the user-defined or edited databases are loaded into the program when the System Definer is performed. Also see *About the Database Files* on page 184.

Setting	Options
Log level	Select the level and type of information to display in the **Event Log** window using the **Log level** slide bar. Choose from a **Debug**, **Info**, **Warning** or **Error** level of detail.
Check update interval	From the **Check update interval** list, choose not to check (**Don't check**, the default) or **On startup**. You can also manually check for updates – choose **Help→Check for update**.
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Graphical Mode - Default Units

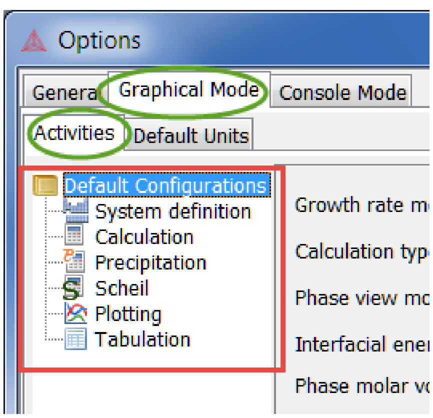


To open this window, from the main menu select **Tools → Options**and click the **Default Units** tab.

Default Units - Settings in Graphical Mode

Setting	Options
Temperature	Kelvin (the default), Celsius or Fahrenheit
Pressure	Pascal (the default), Atmospheres or Bar
Amount	Mole (the default), Gram, Kilogram or Pound
Composition	Mass percent (the default), Mole percent, Mass fraction or Mole fraction
Energy	Joule (the default), Calorie or Electron volt
Volume	Cubic meter (the default), Cubic decimeter or Cubic centimeter
Density	Kilogram per cubic meter (the default) or Gram per cubic meter
Entropy	Joules per Kelvin (the default), Calorie per Kelvin or Electron volt per Kelvin
Length	Meter (the default), Micrometer or Nanometer
Time	Seconds (the default), Hours or Days
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Graphical Mode: Activities



To open this window, from the main menu select **Tools → Options** and click the **Activities** tab.

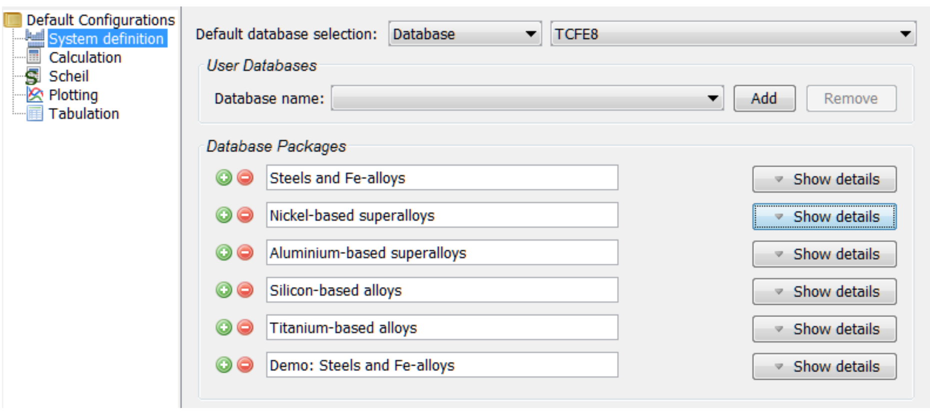
There are **System definition**, **Calculation**, **Precipitation**, **Scheil**, **Plotting** and **Tabulation** settings windows.

For Plotting, the default settings are shared with the Graphical and Console Modes. Any changes you make also apply to the default settings on the **Console Mode → Plotting** tab in the **Options** window (and vice versa). See *Global Settings: Graphical and Console Mode-Plotting* on page 200 for details.

Global Settings: System Definition (System Definer)

When you create a new System Definer, the default databases are taken from these defaults. The database selections for each System Definer can then be configured individually. Once created, the specific (local) settings for a System Definer are not affected if you later make changes to the global defaults.

To open this window, from the main menu select **Tools → Options** and click the **Activities** tab. Then click **System definition**.



Also see *System Definer* on page 54, *Creating a Database Package* on the next page and *Clearing a Database Package Message* on page 193.

System Definer Default Settings in Graphical Mode

Setting	Options
Default database selection	Choose to define a **Database** or a **Database package**. If you choose **Database**, then select the default installed database to use globally for all System Definer nodes. For example, choose **TCFE8** or **TCNI8**. If you choose **Database Package**, select an option from the list. These Database Packages are defined in *Database Packages*.
User Databases	

<table>
<tr><th>Setting</th><th>Options</th></tr>
<tr><td>Database name</td><td>Click the Add button to add a custom database. After you have added the user database it is available from the list to choose as the default.

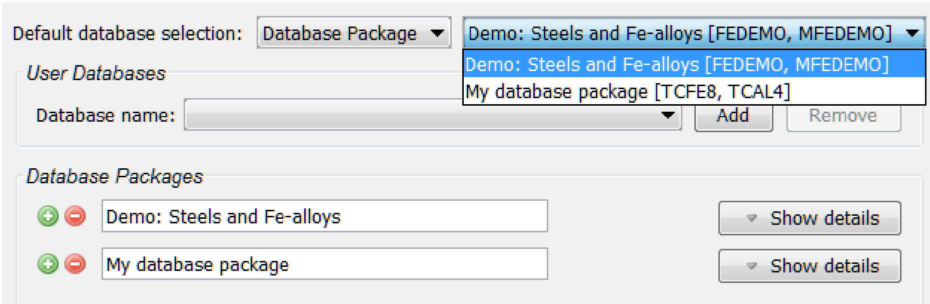
Once a user database is added to the Database name list, you can also click the Remove button to delete it from your project.</td></tr>
<tr><td colspan="2">Database Packages</td></tr>
<tr><td>Based on the installed databases, there are also default Database Packages available.</td><td>Create and remove Database Packages by clicking the Add a database package button or Remove this database package button.

The addition or removal of a database package does not take effect until a new System Definer node is defined.

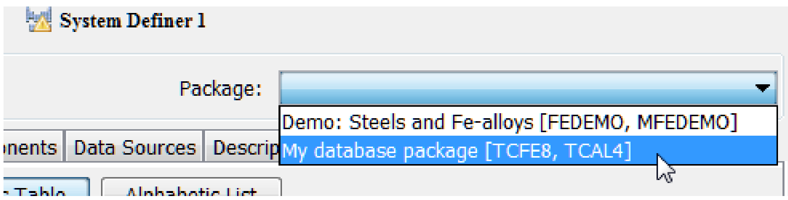
Click Show details to display the list of databases included with the package. Click Hide details to only display the name of the package. See Creating a Database Package below.</td></tr>
<tr><td>Reset to original settings</td><td>In the lower corner of the window, click the Reset to original settings button to restore the original settings. You may have to expand the window to view the button.</td></tr>
</table>

Creating a Database Package

1. From the main menu select **Tools → Options** .
2. Click the **Activities** tab and click **System definition**.
3. Under **Database Packages**, click the **Add a database package** button.
4. Replace the default text with a name for the package, for example **My database package**.
5. Select a database from the **Choose database** list, for example, **TCNI8**.
6. Click the **Add a database** button and from the **Choose database** list, select another database, for example, **TCAL4**
7. Click **OK.** The Database Package is now available to choose as the default for new System Definer activities.



The package is available for new System Definer nodes defined after adding the package.



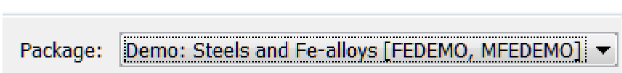
Clearing a Database Package Message below and *Global Settings: System Definition (System Definer)* on page 191.

Clearing a Database Package Message

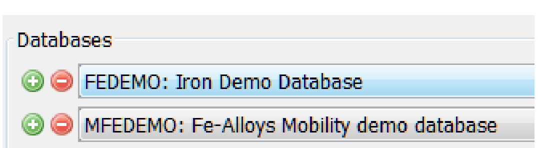
Sometimes when working on the System Definer Configuration window, you may add or delete Databases. If you originally added a Database Package, the addition or removal of a database means that the original package you selected does not now match the databases you are using. To clear the message *Selected databases and selected package may differ* and reset the databases you need to either:

- Change the newly added database back to the one that matches the database package, or
- Choose another package from the **Package** list, then select the package you want.

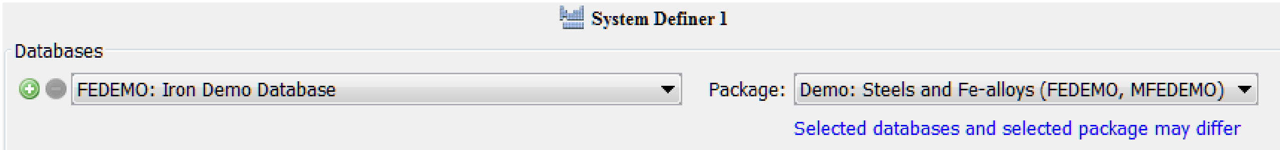
For example, from the **Package** list you select **Demo: Steels and Fe-alloys**.



This adds the **FEDEMO** and **MFEDEMO** databases.



If you add or remove **Databases**, in this example **MFEDEMO** is removed, then this does not match the **Package** and the message displays:



To clear the message, click the **Add a database** button and from the **Choose database** list, select the database that belongs with the package, in this example, **MFEDEMO**. The message is cleared.

You can also choose another package from the list, which changes the databases, and then re-select the original package to reset the message.

Creating a Database Package on page 192 and *Global Settings: System Definition (System Definer)* on page 191.

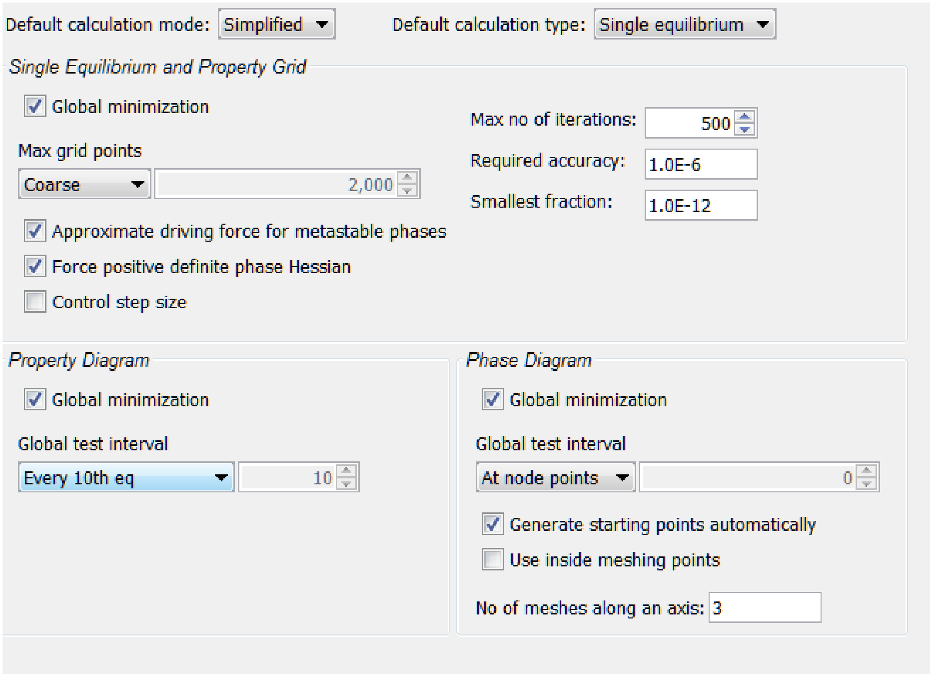
Global Settings: Calculation (Equilibrium Calculator)

When you create a new Equilibrium Calculator, its initial settings are taken from these defaults. The settings of each Equilibrium Calculator can then be configured individually. Once created, the specific (local) settings for an Equilibrium Calculator are not affected if you later make changes to the global defaults.

To open this window, from the main menu select **Tools → Options** and click the **Activities** tab. Then click **Calculation**.

Also see *Equilibrium Calculator* on page 72.

These settings can also be changed locally for a certain Equilibrium Calculator.



Equilibrium Calculator Default Settings in Graphical Mode

Setting	Options
Default calculation mode	Simplified (the default) or Advanced
Default calculation type	Single equilibrium (the default), Property diagram, Phase diagram or Property grid
	Single equilibrium and property grid
Global minimization	The check box is selected by default to ensure that the most stable minimum under the specified conditions is computed.
Max no of iterations	Default is 500
Max grid points	Coarse (2000 grid points, the default), Medium (20,000 grid points), Fine (200,000 grid points), or Custom (set your own number of grid points)
Required accuracy	The default is 1.0E-6
Smallest fraction	The default is 1.0E-12
Approximate driving force for metastable phases	The check box is selected by default

Setting	Options
Force positive definite phase Hessian	The check box is selected by default
Control stepsize	The check box is selected by default
Property diagram	
Global minimization	The check box is selected by default to ensure that the most stable minimum under the specified conditions is computed.
Global test interval	Every 10th eq (the default), Always or Custom
Phase diagram	
Global minimization	The check box is selected by default to ensure that the most stable minimum under the specified conditions is computed.
Global test interval	At node points (the default), Every 10th eq, Always or Custom
Generate starting points automatically	The check box is selected by default
Use inside meshing points	Click to select the check box as required
No of meshes along an axis	The default is 3
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Global Settings: Precipitation (Precipitation Calculator)

When you create a new Precipitation Calculator, its initial settings are taken from these defaults. The settings for each Precipitation Calculator can then be configured individually. Once created, the specific (local) settings for a Precipitation Calculator are not affected if you later make changes to the global defaults.

To open this window, from the main menu select **Tools → Options** and click the **Activities** tab. Then click **Precipitation**.

Also see *Precipitation Calculator* on page 78.

These settings can also be changed locally for a specific Precipitation Calculator.

Precipitation Calculator Default Settings in Graphical Mode

Setting	Defaults/Options
Growth rate model	Simplified (default) or Advanced.
Calculation Type	Isothermal (default), Non-isothermal or TTT diagram.
Phase mode view	Hide details (default) or Show details
Interfacial energy	Calculated (default) or User-defined
Phase molar volume	Database (default) or User-defined.
Matrix grain size	1.0E-4 m (default)
Matrix grain aspect ratio	1.0 (default)
Dislocation density	5.0E12 m^{-3} (default)
Nucleation sites	Bulk (default), Grain boundaries, Grain edges, Grain corners, or Dislocations.
Max time step fraction	0.1 (default)
No. of grid points over one order of magnitude in radius	150.0
Max no. of grid points over one order of magnitude in radius	200.0
Min no. of grid points over one order of magnitude in radius	100.0
Max relative volume fraction of subcritical particles allowed to dissolve in one time step	0.01
Max relative radius change	0.01
Relative radius change for avoiding class collision	0.5
Max overall volume change	0.001
Max relative change of nucleation rate in logarithmic scale	0.5
Max relative change of critical radius	0.1

Setting	Defaults/Options
Min radius for a nucleus to be considered as a particle	5.0E-10 m
Max time step during heating stages	1.0 s
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Global Settings: Scheil (Scheil Calculator)

When you create a new Scheil Calculator, its initial settings are taken from these defaults. The settings of each Scheil Calculator can then be configured individually. Once created, the specific (local) settings for a Scheil Calculator are not affected if you later make changes to the global defaults.

To open this window, from the main menu select **Tools → Options** and click the **Activities** tab. Then click **Scheil**.

Also see *Scheil Calculator* on page 108.

These settings can also be changed locally for a certain Scheil Calculator.

Scheil Calculator Default Settings in Graphical Mode

Setting	Options
Start temperature	Enter a default **Start temperature** that is higher than the liquidus temperature of the alloy, in other words, the temperature at which the alloy is completely melted.
Temperature step	Enter a default **Temperature step**. Decreasing the temperature step increases the accuracy, but the default value is usually adequate.
Global minimization	By default the **Global minimization** check box is not selected. Click to select it to perform a global minimization test when an equilibrium is reached. This costs more computer time but the calculations are more robust.
Allow BCC → FCC	By default the **Allow BCC → FCC** check box is not selected. Select the check box to allow transformations in the solidified part of the alloy caused by each of the components specified to be a *fast diffuser*. It is recommended that you only select this for steels.
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Global Settings: Tabulation (Table Renderer)

When you create a new Table Renderer, its initial settings are taken from these default settings. The settings of each Table Renderer can then be configured individually. Once created, the settings of a Table Renderer are not affected by changes in the default settings. To open this window, from the main menu select **Tools → Options** and click the **Activities** tab. Then click **Tabulation**.

Also see *Table Renderer* on page 101.

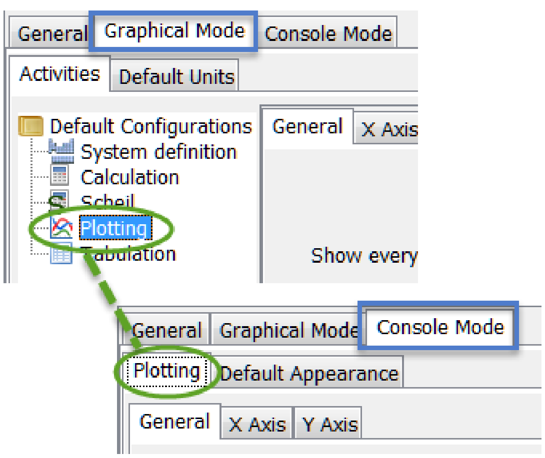
Table Renderer Default Settings in Graphical Mode

Setting	Options
Decimal digits	The default is 5
Number format	Auto (the default), Decimal or Scientific
Phase description detail	Composition (the default), Condensed, Constitution, or Composition and constitution

Setting	Options
Primary display color	Click **Modify** to change the default
Secondary display color	Click **Modify** to change the default
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Global Settings: Graphical and Console Mode-Plotting

The default settings are shared with both modes. Any changes you make also apply to the default settings on the **Console Mode → Plotting** tab in the **Options** window (and vice versa).



To make local changes to the appearance of a specific plot, in the **Results** window right-click a plot and select **Properties**. In the **Plot Properties** window a variety of settings can be made.

You can also edit some properties for individual plot lines (the color, the line width and type, and whether data points are included). In the **Results** window, hover the mouse over a plot line. The crosshair cursor turns into a cursor resembling a pointing hand when it is over a line that can clicked. Alternatively, hold down Ctrl while you move the cursor around the plot to only display it as a crosshair and prevent unintended edits.

To globally configure the default settings for plotting, from the main menu select **Tools → Options**. Click the **Graphical Mode → Activities** tabs and then click the **Plotting** node.

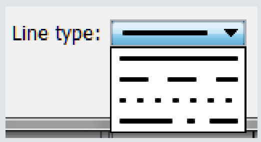
When you create a new Plot Renderer, its initial settings are taken from these default settings. The settings can then be configured individually. Once created, the settings of a Plot Renderer are not affected by changes to the default global settings.

To configure the settings of an individual Plot Renderer rather than the default settings, right-click a plot in the **Results** window and select **Properties** from the menu.

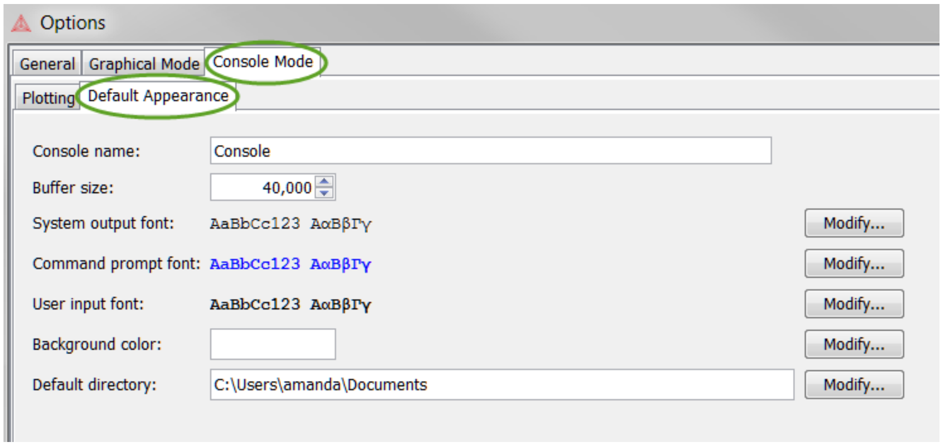
Plot Renderer Settings for Console and Graphical Modes

These global settings are available for both Console and Graphical Modes.

Setting	Options
Title	Enter a title. Select **Plain text** (the default) or **LaTeX formatted text**. Click α button to add symbols. Click a symbol category (for example, Latin characters or Greek and math symbols). Then click to choose the symbol located on the right-hand side. When you select **LaTeX formatted text** you can use the LaTeX math text to enter text. See *ADD_LABEL_TEXT* on page 190 and *Adding and Editing Plot Labels* on page 92 for more details.
Title font, Legend font, Label font, Header font	Click **Modify** to edit the Font Name, Style, Size and Color.
Legend phase caption style	Choose **All**, **Constitution description** or **Ordering description**. See *About Legend Styles* on page 91 for details about these options.
Legend background color, Background color, Canvas color, Border color,	Click **Modify** to choose a color or enter a specific color.
Label format	Select Plain text (the default) or LaTeX formatted text.
Show anchor	By default, an anchor between the label and the plot point is displayed.
Retain labels (Graphical Mode only)	By default the check box is selected. Plot labels are kept (retained) when plots are updated. It can be applied globally or locally to individual plots.
Show header (Console Mode only)	Click to select the **Show header** check box to display the basic details about the plot along the top. This includes the date and time the plot is generated, the database used and the properties.
Border width	Choose a plot border width between **1** (thin, the default) and **10** (thick).
Color option	For the plot line colors, choose from the options in the list: **Legacy**, **Printer friendly**, **JFree chart**, **Pastel**, **Medium dark**, **Bright dark**, **Vivid** or **Earth**.

Setting	Options
	This option is also available for specific plot lines. To change the color of an individual plot line double-click it in the Results window and use the color palette to define it or enter a specific color.
Line width	For the plot line width, choose from the options in the list. These options are also available for specific plot lines. To change an individual plot line width double-click it in the Results window.
Line type	This option is only available for individual plot lines. To change an individual plot line type double-click it in the Results window and choose an option from the list. 
Show data points	Click to select the check box to display the data points on the plot lines. This option is also available for specific plot lines. To show data points on an individual plot line double-click it in the Results window and select the check box.
Data point marker size	Choose the data point marker size between **1** (small) and **10** (large). The default is 3.
Plot area size	The **Fit the plot area size to the enclosing window** option is selected by default. Excluding triangular plots, click Define the plot area size to enter a Plot area height and Plot area width.
Reset to original settings	In the lower corner of the window, click the **Reset to original settings** button to restore the original settings. You may have to expand the window to view the button.

Global Settings: Console Mode Default Appearance



To open this window, from the main menu select **Tools → Options**. To change the name, buffer size, fonts and colours for a specific Console window, right-click the label for that particular Console and select **Properties**.

Default Appearance Settings in Console Mode

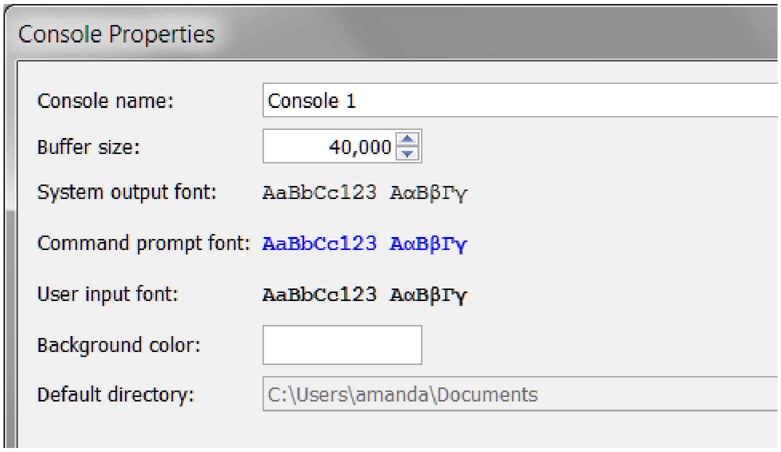
<table>
<tr><th>Setting</th><th>Options</th></tr>
<tr><td>Console name</td><td>The default is <code>Console</code> or enter another name to display</td></tr>
<tr><td>Buffer size</td><td>The default buffer size is 40,000.</td></tr>
<tr><td>System output font</td><td rowspan="3">Click Modify to open the Select Font window and makes changes to the font type, style, size or color as required.</td></tr>
<tr><td>Command prompt font</td></tr>
<tr><td>User input font</td></tr>
<tr><td>Background color</td><td>Click Modify to open the Select Color window to choose a background color for the Console Mode window where you enter commands.</td></tr>
<tr><td>Default directory</td><td>Set the default directory where log files and workspace files are saved.</td></tr>
<tr><td>Reset to original settings</td><td>In the lower corner of the window, click the Reset to original settings button to restore the original settings. You may have to expand the window to view the button.</td></tr>
</table>

Increasing the Buffer Size

Sometimes the output overflows the window text buffer (see *Controlling Console Output* on page 123). To increase the buffer size in a specific Console tab:

1. Right-click the Console tab header in the Console window (if this is the first Console tab, it is labelled **Console 1**).

2. Click **Properties**, and increase the **Buffer size** in the Console Properties window.



Console Mode Command Reference

Thermo-Calc Version 2016a

Thermo-Calc Software

Introduction to the Command Reference

In this section:

About the Console Command Reference

This command reference is for Console Mode users of Thermo-Calc software. This guide describes the function and syntax of the commands available in the Thermo-Calc modules.

If you have the Diffusion Module (DICTRA) also see the *Diffusion Module (DICTRA) Command Reference* found in the Diffusion Module (DICTRA) Documentation Set PDF.

The commands available in all or most modules are described in the first topic (*General Commands* on page 3). The next topics are listed in alphabetical order based on the module names, then in each subtopic, the module commands.

Abbreviations Used in this Guide

For the prompts, `Y` stands for Yes and `N` stands for No.

TC is for Thermo-Calc. For example, default file names often start with `tc.ini`.

File Types

Also see the *Thermo-Calc Installation Guide* and the *Data Optimization User Guide* included with this documentation set for information about EXP files.

In Console Mode, Thermo-Calc uses different kinds of files, including:

- Log files (*.LOG)
- Macro files (*.TCM or *.LOG)
- POLY workspace files (*.POLY3)
- Experimental data files (*.EXP)
- The GIBBS module uses a workspace file (*.GES5)
- The PARROT module uses files with the suffixes *.PAR and *.POP.

General Commands

In this section:

BACK

Also see *GOTO_MODULE* below.

Syntax	BACK
	Switch control back to the most recent module. Going from the POST module (post-processor), BACK goes only to the TAB or POLY module (from where the POST module is entered).

EXIT

Syntax	EXIT
	Exit the program and return to the operating system. Use a save command before exiting otherwise all data and results are lost (in either the GIBBS, POLY, PARROT or REACTOR module).

GOTO_MODULE

Switch between modules. The name of the module must be typed. To get a list of available modules, press the <Enter> key.

Syntax	GOTO_MODULE
Prompt	`MODULE NAME: <MODULE NAME>`

	If the <Enter> key is pressed without typing a unique module name (or typing a ? mark) it prompts for the `MODULE NAME` and lists the available modules
	`SYSTEM_UTILITIES`
	`GIBBS_ENERGY_SYSTEM`
	`TABULATION_REACTION`
	`POLY_3`
	`DICTRA_MONITOR`
	`BINARY_DIAGRAM_EASY`
	`DATABASE_RETRIEVAL`
	`DIC_PARROT`
	`REACTOR_SIMULATOR_3`
	`PARROT`
	`POTENTIAL_DIAGRAM`
	`SCHEIL_SIMULATION`
	`POURBAIX_DIAGRAM`
	`TERNARY_DIAGRAM`
	`MODULE NAME:`

HELP (? and ??)

Gives text help about the current prompt (whether this is the prompt of a module, or a prompt that requests you to specify the value of some parameter).

Syntax	? OR ??
	For some commands and prompts, more detailed help is given when two ?? are entered.

HELP

Lists the available commands or explains a specified command. Specifying a unique command displays an explanation on screen. Typing a command abbreviation that is not unique lists all matching commands. Get the command information by typing a unique abbreviation or the complete command name.

Syntax	HELP
Prompt	`COMMAND: <COMMAND NAME>` *Command name* is the name of the command (one of the-module commands) to get help. If the <Enter> key is pressed without typing a command name, then all the available commands are listed.

INFORMATION

Get basic information about topics related to the module you are in.

Syntax	INFORMATION
Prompt	`WHICH SUBJECT /PURPOSE/: <TOPIC NAME>` Specify a Subject (or its abbreviation as long as it is unique). Enter a question mark ? for a list of topics.

MACRO_FILE_OPEN

The macro file functionality is a way to predefine sequences of Thermo-Calc commands stored in a macro file (a basic text file usually with the default extension TCM for the Thermo-Calc Console Mode or DCM for DICTRA module) and then executing all of the sequences using this command (preceded by the macro file name). This command can be used within various modules (i.e. the SYS, POLY, PARROT and TAB modules in the Thermo-Calc (Console Mode) software; SYS, POLY, PARROT and DICTRA_Monitor).

Syntax	MACRO_FILE_OPEN
	<NAME OF A MACRO FILE *.TCM> An **Open** window displays to specify the filename of the macro file (*.TCM) with the macro command, so that the path (in the **Look in** field) and **File name** can be specified. The **Files of type** *.TCM for Thermo-Calc (Console Mode) calculations and *.DCM for DICTRA simulations, cannot be changed. Click **Open** or **Cancel** to continue. If the macro file contains commands to set *.LOG files, to save or read GIBBS, POLY or PARROT workspaces, switch user databases, compile experiments (from existing *.POP files), create new *.PAR files, append *.EXP files, plot or dump diagrams, etc., a window (e.g. Save, Open, Print, etc.) displays on screen. If required, these windows can be avoided by issuing the file names (preferably with file-type extensions; if the files are not in the current working area where the macro is located, the appropriate and complete paths of the files should also be specified) and sometimes with the required options after the corresponding commands or parameters/options. For details, see the related commands and modules. When using a macro file that intended to plot graphs on screen, but the command *SET_PLOT_FORMAT* on page 210 is used to alter the plotting environment from the default value, it is important to first use the command SET_PLOT_FORMAT again to change back to the default value.

See example 12 in the *Console Mode Examples Guide*.

This is useful when the same or similar calculations are made frequently with small changes [in terms of system definitions, data manipulations, conditions (for single points, stepping or mapping calculations), plotting settings, etc.). For example, use this feature when calculating phase/property diagrams during an assessment of thermodynamic data.

A macro file can be automatically generated by the Thermo-Calc (Console Mode) software, if in the SYS module the *SET_LOG_FILE* on page 242 command is used and a *.LOG file name is given before any other SYS, DATA, TAB, GIBBS, POLY, POST, PARROT, or ED-EXP command or any special-module command (e.g. BIN, TERN, POTENTIAL, SCHEIL, POURBAIX, or REACTION). Such a *.LOG file generated from the Thermo-Calc (Console Mode) software is a textual file, and using a textual editor (such as Notepad, WordPad, PFE, Emacs, vi, etc.) it can be edited, for example, by taking away unnecessary command lines, modifying some commands, settings and definitions, adding some pausing points, adding helpful commenting lines beginning with @@ signs, and so forth. Then save it as a macro file with the standard file extension TCM.

Experienced users can also write/edit an appropriate macro file for calculations/simulations, using any basic textual editor outside the Thermo-Calc program.

All commands can be used in a macro file. The file must be terminated with EXIT or be interrupted with SET_INTERACTIVE.

Within a macro file you can use comment-lines (for describing the problems and for explaining commands and inputs/outputs), which should always start with the @@ signs in the beginning of each comment-line. Such comment-lines help to document the macro file, while these are not considered as command lines and thus do not affect the proceeding of all the normal commands in modules of the Console Mode when the file is called by the Thermo-Calc Console Mode software.

You can put multiple-line comment-blocks in a macro file between the @ (" and "@)" signs. The former sign indicates where the comment begins and the latter sign indicates where the comment ends. A comment-block begins from a line started with the *begin comment* sign @ (and ends with the *finish comment* sign @); all the lines written in between are ignored, and the line started with @) is also ignored.

Another use for a macro file is to allow you to interact at some specific points, using the "@?" sign starting a text line (note that the text describing the expected user-specification/input should be written as a continuous string without any empty space), for user's on-time specifications of arguments/parameters or inputs of parameter-values that are requested by a certain command. The macro temporally stops at the "@?" sign, prompt on screen the text given after "@?", and waits for the specified argument/parameter/value. The Thermo-Calc (Console Mode) software then uses specified argument/parameter/value as the input(s) for the associated command. For example, you can input the values of lower and higher temperature limits for the second axis-variable as follows:

```
GO POLY-3

SET-AXIS-VAR 2 T  @?Low-temperature-limit:

@?High-temperature-limit:
```

You can have macro-variables denoted by the signs of @#n (for definition) and ##n (for usage); and up to nine variables inside a single macro file. Such a macro-variable can be assigned with its desired value, as for example:

```
@#3First-element?
```

This writes the text (note that the text describing the expected user specification/input is written as a continuous string without any empty spaces) after the "@#3" sign as prompted on screen and then wait for a specification. The input is assigned to the macro-variable ##3, which is directly called in different parts within the current macro file.

For example, the content of the macro-variable `##3` is inserted in the command:

```
DEFINE-SYSTEM ##3
```

It is also useful in more complicated commands, such as:

```
SET AXIS VAR 1 x(##3) 0 1,,,
```

This command sets the mole fraction of macro variable 3 as axis 1.

A macro file can have any number of pauses at the `@&` signs, for the purpose of checking the details/results of executing certain commands when running the macro file. However, you can also prevent the Thermo-Calc (Console Mode) software from temporarily stopping at any pause by typing any character (except for the `Y` character for confirming a Yes answer to a command prompt) after specifying the name of a macro file.

A macro file can have a maximum of five nested levels, i.e. a macro file can call another macro file, and if one sub-level macro is terminated by the command it resumes at the next command in the previous macro. If it is terminated by `end-of-file`, the Thermo-Calc (Console Mode) software stops.

This feature is especially useful for alloy design that requires many (hundreds) of calculations/simulations on similar material system/processes (specified in many different but appropriately-documented macro files that are organized in up to five levels) during a certain period of time (e.g. in an evening), you can run the main macro (on the top level) at a certain time (e.g. before leaving the office) and afterwards (e.g. next morning) you can systematically and efficiently check/compare/analyse the results (saved as graphical files, and/or EXP, TXT, XLS... files).

By adding the SYS command *SET_ECHO* on page 242 at the beginning of a macro file (or in the primary macro file on the top level if any sub-level(s) of macro files are used), it is useful to automatically display the complete/detailed meaning of various commands in all the sequential operations in the software, all enforced according to the macro file(s).

SET_INTERACTIVE

Syntax	SET_INTERACTIVE
	Resets the input and output units to the initial values, i.e. keyboard and screen. Add this as the last command to the macro files.

BINARY_DIAGRAM Commands

In this section:

BINARY_DIAGRAM Module

The BINARY_DIAGRAM module (short name, the BIN module) enables you to quickly calculate a simple binary phase diagram. Access to specific databases designed for BIN, such as TCBIN, is required.

To enter the module, at the SYS prompt type `GOTO_MODULE BINARY`. There are no other commands for this module. Follow the prompts to plot a diagram.

Also see the Graphical Mode equivalent to this command, *Binary Calculator* on page 104 in the *Thermo-Calc User Guide*.

Syntax	BINARY_DIAGRAM
Prompts	`DATABASE: /TCBIN/`
	`FIRST ELEMENT` `SECOND ELEMENT`
	`PHASE DIAGRAM, PHASE FRACTION (F), G- OR A-CURVES (G/A): /PHASE_ DIAGRAM/`

DATABASE_RETRIEVAL Commands

Search the online help or see in the *Thermo-Calc User Guide* for step-by-step instructions to define a system.

In this section:

AMEND_SELECTION

Use this command after defining the elements, species or the system (with the commands *DEFINE_ELEMENTS* on page 14, *DEFINE_SPECIES* on page 14, or *DEFINE_SYSTEM* on page 15). The prompts allow changes to the predefined system. By answering Y or N, each of the selected elements, species or phases can be accepted or rejected (although this is not the case for the constituents or the entire system).

Syntax	AMEND_SELECTION
Prompt	KEEP <NAME1> NO/QUIT /YES/ KEEP <NAME2> NO/QUIT /YES/
Options	KEYWORD-- ELEMENTS/SPECIES/PHASES NAME& Names of the pre-defined or pre-selected elements/species/phases

APPEND_DATABASE

Append data from a file or additional database to the current set of data already read from another database. Data already read from another database and stored in the Gibbs energy system is kept in the GES5 workspace.

This command also enters all additional parameters (phase constituents, G0 and interaction parameters, etc.) to already existing phases, and all existing parameters (phase constituents, excess model, G0 and interaction parameters, etc.) in the phase are replaced with the values retrieved from the appending database.

The command is equivalent to the SWITCH_DATABASE USER command sequence. It is also similar to *SWITCH_DATABASE* on page 20, but does not reinitialize the DATA module and GIBBS workspace.

All the directly connected databases as predefined by the original database initiation file (TC_INITD.TDB file in the /DATA/ area for Windows environments, or the `initd.tdb` file in the \data\ area for Linux), or by a user-specified database initiation file after using *NEW_DIRECTORY_FILE* on page 17, are listed after pressing <Enter> without giving any argument. You can supply your own database by giving the argument `USER`, the database name and the correct path if it is not located in the current working directory.

After the command is executed for the first time, the TDB_XYZ: prompt (XYZ stands for the name of the primary switched database) is changed to the APP: prompt to go to more commands related to appending data.

<table>
<tr><td>Syntax</td><td>APPEND_DATABASE</td></tr>
<tr><td>Prompt</td><td>Use one of these databases
<code>PURE = SGTE PURE ELEMENT DATABASE</code>
<code>SSUB = SGTE SUBSTANCE DATABASE 1997</code>
<code>USER = USER DEFINED DATABASE</code>
<code>DATABASE NAME /XYZ/: <ADDITIONAL DATABASE NAME></code></td></tr>
<tr><td>Options</td><td><code>ADDITIONAL DATABASE NAME</code>
The name of an existing database or a USER database definition file (***setup.TDB) that corresponds to the appending database.</td></tr>
<tr><td>Notes</td><td>If a USER database name or its path is not given on the same line of the APPEND_ DATABASE command, or if the name or path is incomplete or incorrect, an Open window displays to open an appropriate database-setup. You can then open the USER database, or can cancel the open session; in the latter case, the program lists all predefined databases in the installation area, and you can select one of these or proceed with the USER option again for appending a desired database.

When this command is used in a macro (*.TCM) file, if the USER option is selected, the database setup file name (*setup.TDB) containing the setup definitions of the USER database, and its correct path, are required.

In Linux, the filename of a USER database, or one of the predefined names, can be used under the prompt <code>FILENAME</code>: This is a valid filename for the USER database definition file (***setup.tdb) or a predefined database name, with the correct path. The default filename extension is .tdb.

This command can be called more than once if appending two or more databases to the primary switched database.

After this command, the commands to define a system (in terms of either elements or species), to reject/restore phases or species, and retrieve data must be repeated; however, the second keywords and parameter values might have to be different from the previous ones.</td></tr>
</table>

DATABASE_INFORMATION

<table>
<tr><td>Syntax</td><td>DATABASE_INFORMATION</td></tr>
<tr><td></td><td>A short description of the current database is normally given by typing this command.

This can include information on covered systems, used models, valid ranges for parameters in temperature and composition, major applications, and so on.</td></tr>
</table>

DEFINE_ELEMENTS

Define the system in terms of elements. All possible species that can be formed by the given elements are retrieved from the database. The names of elements must be separated with a space or comma. It is possible to use a wildcard * after a common part of element names so that all the elements, which start with that common part and are available in the currently switched or appending database, are defined in the system. Up to 40 elements can be defined into a single system.

Also see *Defining a System in Console Mode* on page 127 in the *Thermo-Calc User Guide* for step-by-step instructions to define a system with this command.

Syntax	DEFINE_ELEMENTS
Prompt	`ELEMENT& <ELEMENT1, ELEMENT2, …>` A list of elements to be defined into the system.
	When appending database(s), this command, or *DEFINE_SPECIES* below or *DEFINE_SYSTEM* on the next page must be repeated with the same or similar elements as defined in the first switched database.

DEFINE_SPECIES

Define the system in terms of species. Only those species given are retrieved. The different names of species must be separated with a space or comma. It is possible to use a wildcard * after a common part of species names so that all the species, which start with that common part and are available in the currently switched or appending database, are defined in the system. Up to 1000 species can be defined in a single system.

Also see *Defining a System in Console Mode* on page 127 in the *Thermo-Calc User Guide*.

Syntax	DEFINE_SPECIES
Prompt	`SPECIES& <SPECIES1, SPECIES2, …>` A list of species to be defined in the system.
	When appending database(s), this command, *DEFINE_ELEMENTS* above or *DEFINE_SYSTEM* on the next page must be repeated with the same or similar elements as defined in the first switched database.

DEFINE_SYSTEM

Define the system in terms of either elements (equivalent to *DEFINE_ELEMENTS* on the previous page) or species (equivalent to *DEFINE_SPECIES* on the previous page). Certain databases have a default value of the keyword (as either ELEMENTS or SPECIES) reflecting what is most appropriate when defining a system.

The different names of elements or species must be separated with a space or comma. It is possible to use a wildcard * after a common part of elements or species names so that all the elements or species, which start with that common part and are available in the currently switched or appending database, are defined in the system.

When appending database(s), this command (or DEFINE_ELEMENTS or DEFINE_SPECIES) must be repeated with the same or similar elements as defined in the first switched database. Different databases might contain different elements and have different species definitions: avoid defining elements/species that are missing in the appending database(s). Otherwise, the program indicates these missing elements/species, and ignores them in subsequent steps. But additional elements/species, and additional phases not available in the first switched database can be defined and retrieved from the appending database(s). Up to 40 elements and 1000 species can be defined in a single system.

Also see *Defining a System in Console Mode* on page 127 in the *Thermo-Calc User Guide* for step-by-step instructions to define a system with this command.

Syntax	DEFINE_SYSTEM
Prompt	`ELEMENTS: <ELEMENT1, ELEMENT2, …>, OR` `SPECIES: <SPECIES1, SPECIES2, …>`
Options	**Description**
`SPECIES OR ELEMENTS`	Default keyword.
`ELEMENT&`	Specify a list of elements for the defining system.
`SPECIES&`	Specify a list of species for the defining system.

GET_DATA

Only after executing this command can you go to any of the application programs such as GES, POLY or DICTRA and use the retrieved data. When appending database(s), this command must be repeated in order to obtain the additional system definitions, parameters and functions.

Syntax	GET_DATA
	Enter the defined system's elements, species, phases, and the connected parameters obtained from either the primary switched or additionally appending database(s) to the GIBBS and/or DICTRA workspace. This command is necessary for retrieval of all information concerning a defined system from the databank.

LIST_DATABASE

List all elements, species, phases or phase constituents in the database.

Syntax	LIST_DATABASE
Options	**Description**
`KEYWORD`	One of the keywords Elements, Species, Phases or Constituent must be used to indicate what to list
`ELEMENTS`	All available elements, the reference state, atomic mass, H298-H0 and S298. Some elements have spaces in the column for the reference state. This implies that there are no parameters stored for this element.
`SPECIES`	All available species together with the stoichiometric factors.
`PHASES`	All available phases together with the number of sublattices and the number of sites in each sublattice.
`CONSTITUENT`	All available phases, the number of sublattices, the number of sites in each sublattice and the species dissolved in each phase. Species in different sublattices are separated with a colon (:). It is important to realize, for example, a phase can consist of Fe, Mo, V, and Cr, and its thermodynamic parameters can come from the binary systems Fe-Mo, Fe-V, Fe-Cr, and Mo-Cr. These data can give a relatively good description of the corners of the Fe-Cr-Mo system, but would most certainly give a bad one for the system Mo-Cr-V, due to the fact that the interaction parameters are, by default, set to zero, which originates from binary systems not included in the database.

LIST_SYSTEM

List all elements, species, phases or phase constituents in the defined system. It works only after a system is defined.

Syntax	LIST_SYSTEM
Keyword	One of the keywords `Elements`, `Species`, `Phases` or `Constituent` (as described in *LIST_DATABASE* above) must be used to indicate what to list.

NEW_DIRECTORY_FILE

Open a new database initiation file (or called database directory file) generated by a local database manager or user, for accessing additional databases not predefined in the original database initiation file.

The original database initiation file is automatically copied by the installation script to the main database area, on a local computer for an independent installation locates or a connected server for a server installation.

- For Windows this is the TC_INITD (or TC_INITD.TDB) file in the \DATA\ area under the directory defined by the TCPATH parameter.
- For Linux, it is the initd.tdb file in /data/ area that is under the directory defined by the TC_DATA parameter. If the Thermo-Calc (Console Mode) is run on a Linux platform, the new database initiation file must be located in the current working directory.

If there are too many databases at one installation or if there are some user-specified databases for a particular purpose, a local database manager can generate additional database initiation files, or each user can have their own initiation file.

The command switches the working initialization of database groups in the DATA module among the original and additional database initiation files. The Thermo-Calc (Console Mode) can use additional database initiation files that define accessing paths to database groups (databases located in different subdirectories under the directory defined by the TCPATH or TC_DATA parameter). All directly accessible databases remain as in the same group, until this command is called or recalled.

Syntax	NEW_DIRECTORY_FILE

Prompt	`FILE WITH DATABASE DIRECTORY /TC_INITD/: <DATABASE-INITIATION-FILE NAME>`
	`ANOTHER DATABASE-INITIATION-FILE NAME`
	The name of the next database initiation file (either additional or original) that is to be switched on as follows sessions.

	If the file name or its path is not given on the same line of the NEW_ DIRECTORY_FILE command, or if it is incomplete or incorrect, an **Open** window displays to specify the new database initiation file.

REJECT

Reject elements, species, phases or phase constituents that can form from the defined elements and species. Phases/species/constituents that are possible to form in the defined system are removed from the list of system phases/species/constituents (shown by the command *LIST_SYSTEM* on the previous page). Phases/species/constituents that are not included in the list cannot be entered without first being restored. The different names must be separated with a space or comma. You can use a wildcard * after a common part of names so that all the elements/species/phases/constituents that start with that common part and are available in the currently switched or appending database, are rejected from the defined system.

This command can also reject a defined system, and thus reinitiate the entire DATA module memory and GES5 workspace.

Syntax	REJECT
Prompt 1	(keyword = ELEMENTS or SPECIES or PHASES) `IF KEYWORD: <NAME1, NAME2, …>`

Prompt 2	(if keyword = CONSTITUENT)
	`PHASE: <PHASE NAME>`
	`SUBLATTICE NUMBER: <SUBLATTICE NUMBER IN THE PHASE>`
	`CONSTITUENT: <CONSTITUENT(S) IN THE SUBLATTICE OF THE PHASE>`

Prompt 3	If keyword = SYSTEM the DATA module is reset to its initial state; GIBBS is reinitiated and data already entered to GES5 is lost.
Options	**Description and additional information**
`KEYWORD`	One of the keywords `Elements`, `Species`, `Phases`, `Constituent` or `System` must be used to indicate what is rejected.
`NAME&`	Names of the pre-defined elements/species/phases/constituents/system
`ELEMENTS`	The given elements are rejected.
`SPECIES`	The given species are rejected, making it impossible to form them from the defined elements.
`PHASES`	The given phases are rejected, making it impossible to form them from the defined elements or species.
`CONSTITUENT`	The given constituent in one phase are rejected. Add the following when prompted: `Phase`: The name of the phase containing the constituent to be rejected. `Sublattice number`: The sublattice where the constituent enters (the first sublattice is 1). The question is omitted if there exists only one possible sublattice. `Constituent`: The name(s) of the constituent(s) to be rejected.

RESTORE

Restore already explicitly rejected elements, species, phases or constituents; it is the opposite of the command *REJECT* on the previous page although it does not restore a completely rejected system.

Phases/species/constituents that are possible to form from the defined elements or species are entered to the list of system phases/species/constituents (shown by the command *LIST_SYSTEM* on page 16). Phases/species/constituents that are not included on the list can now be added to the list. The different names must be separated with a space or comma. It is possible to use a wildcard * after a common part of names so that all the elements/species/phases/constituents, which start with that common part and are available in the currently switched or appending database, are restored in the defined system.

Syntax	RESTORE
Prompt1	(if keyword = ELEMENTS or SPECIES or PHASES) `KEYWORD: <NAME1, NAME2, …>`

Prompt2	(if keyword = CONSTITUENT)
	`PHASE: <PHASE NAME>`
	`SUBLATTICE NUMBER: <SUBLATTICE NUMBER IN THE PHASE>`
	`CONSTITUENT: <CONSTITUENT(S) IN THE SUBLATTICE OF THE PHASE>`

Options	**Description and Information**
`KEYWORD`	One of the keywords `Elements`, `Species`, `Phases` or `Constituent` must be used to indicate what is rejected.
`NAME&`	Names of the pre-defined elements, species, phases, and constituents
`ELEMENTS`	The given elements are restored.
`SPECIES`	The given species are restored and thus possible to form from the defined elements.
`PHASES`	The given phases are restored and thus possible to form from the defined elements or species.
`CONSTITUENT`	The given constituent in one phase are restored. Add the following when prompted: `Phase`: The name of the phase containing the constituent to be restored. `Sublattice number`: The sublattice where the constituent enters (the first sublattice is 1). `Constituent`: The name(s) of the constituent(s) to be restored.

SET_AUTO_APPEND_DATABASE

Enforce an automatic action that appends thermodynamic data from a chosen database regarding the system to the data you already have about the system from the current database (that is, the default database or the database last chosen with *SWITCH_DATABASE* on the next page or *APPEND_DATABASE*

on page 12). This is useful to simultaneously retrieve both thermodynamic and mobility data for a defined system when performing a DICTRA module simulation.

This command should be used before defining a system (with the commands *DEFINE_ELEMENTS* on page 14, *DEFINE_SPECIES* on page 14, or *DEFINE_SYSTEM* on page 15) and retrieving the data with the *GET_DATA* on page 15 command from the primary switched database.

Syntax	SET_AUTO_APPEND_DATABASE
Prompt	`DATABASE NAME /XYZ/: <ADDITIONAL DATABASE NAME>`
Option	`ADDITIONAL DATABASE NAME` The name of an existing database or a USER database that corresponds to the automatically-appending database.

It works in a way that is similar to the APPEND_DATABASE<DATABASE-NAME> command sequence, but only the phases that are also available in the primary switched database are appended. There is no possibility to manually list, reject and/or restore any phase that is available in the appending database. When retrieving data with the GET_DATA command afterwards, all the phases that exist in the appending database but not in the primary switched database are automatically rejected. The prompt is kept as for the primary switched database, `TDB_XYZ`: (where `XYZ` stands for the name of the primary switched database) until the execution of the GET_DATA command.

Therefore, to selectively append more phases from a secondary database to a system that is defined and retrieved from a primary switched database, then the APPEND_DATABASE command and sequential commands (such as DATABASE_INFORMATION, DEFINE_SYSTEM, DEFINE_ELEMENT, DEFINE_SPECIES, LIST_SYSTEM, REJECT, RESTORE and GET_DATA) should be used instead, before this command.

Although it is impossible to manually list, reject and restore any phase from the appending database, the DATA module automatically ignores all the phases that do not exist in both the primary switched database and the appending database, and append the data for the phases (that are also available in the primary switched database) from the appending database, as informed on screen when executing *GET_DATA* on page 15 afterwards.

SWITCH_DATABASE

Switch (or change) from the current database to a new one, and reinitialize the entire DATA module for defining a system and the GES5 workspace for storing retrieved data. All the directly connected databases as predefined by the original database initiation file, or by a user-specified database initiation file after using *NEW_DIRECTORY_FILE* on page 17, are listed by pressing the <Enter> key without giving any argument. You can also supply your own database by giving the argument USER and then the database name and its correct path if it is not located in the current working directory. The second part of the prompt TDB_XYZ indicates the present database XYZ.

Syntax	SWITCH_DATABASE
Prompt	DATABASE NAME /XYZ/: <NEW DATABASE NAME> Use one of these databases: • PURE (SGTE pure element database) • SSUB (SGTE substance database 1997) • USER (user-defined database)
Options	**Description and Information**
NEW DATABASE NAME	The name of an existing database or a USER database definition file (***setup.TDB).
DATABASE NAME	The new database is specified by typing the abbreviation given in the front of one of the available predefined databases. When switching/appending databases or additionally purchased databases you can add these to the predefined database list in the database initiation file TC_INITD.TDB or initd.tdb of your installed software. When this command is used in a macro (*.TCM) file, if the USER option is selected, the database setup file name (*setup.TDB) that contains the setup definitions of the USER database, and its correct path, must be provided.
FILENAME	In Linux, the filename of a USER database, or one of the predefined names, can be used under this prompt. A valid filename for the USER database definition file (***setup.tdb) or a predefined database name, with the correct path. The default filename extension is .tdb.
Notes	If a USER database name or its path is not given on the same line of the SWITCH_DATABASE command, or if it is incomplete or incorrect, an Open window is displayed for the to-be-switched database. The USER database is used without reinitiating the Gibbs energy system, data from different databases can thus be combined. Therefore, the SWITCH_DATABASE USER command sequence is equivalent to the APPEND_DATABASE USER command sequence. The result of a combination must be examined carefully, because differences in standard states, phase models and names can be disastrous. If the same parameters occur in several such switched or appended databases, the last one retrieved is used in the calculations. It is not advisable to use this method for large databases since these load slowly. After this command, those commands for defining system (in terms of either elements or species), rejecting/restoring phases or species, retrieving data, as well as for appending data from additional database(s), can be proceeded.

EDIT_EXPERIMENTS Commands

These commands are accessed through the PARROT module.

Some special commands are designed only for the ED_EXP module and can also be used in *.POP or *.DOP files.

In this section:

Running the EDIT_EXPERIMENTS Command

To run the EDIT_EXPERIMENTS (ED_EXP) module:

- The first necessary command after entering the ED_EXP module for the first time (with *EDIT_EXPERIMENTS* on page 95 in the PARROT module) should always be READ_BLOCK in order to load the experimental data block from the current work *PAR-file for editing. Also, the read command must also be used prior to any other ED_EXP command if the module is reinitiated, or if no experimental data block has not previously been read from a work file compiled with a proper experimental data .POP/.DOP file, or if you want to change to another data block for editing.
- If any change is made in the ED_EXP module remember to use the command *SAVE_WORKSPACE* on page 232 before going BACK to the PARROT module.

Although the TABLE_HEAD, COMMENT and FLUSH_BUFFER commands are visible in the ED_EXP module, these cannot be used in the ED_EXP module; these are only functional in an *.POP or *.DOP file.

COMPUTE_ALL_EQUILIBRIA

In ED_EXP and PARROT modules, each experiment is treated as an individual equilibrium with some measured values. These are created with *CREATE_NEW_EQUILIBRIUM* on page 143, and stored in an experimental data *.POP file and then compiled and saved in the POLY3 workspace of a PARROT work *.PAR file.

With COMPUTE-ALL-EQUILIBRIA, all equilibria from the current to the last experimental points are calculated. If an equilibrium calculation fails, the calculation is stopped at that equilibrium. Equilibria with weight zero is skipped.

A current or present experimental point means the latest selected or calculated one. Once used the current point turns to the last point in the data block; in such a case you first use *SELECT_EQUILIBRIUM* on page 162 so that the current point is switched to a desired one.

Syntax	COMPUTE_ALL_EQUILIBRIA
	For this command there is always a list output on screen, which consists of six columns for all experimental points available in the current data block. The first column is the *equilibrium identifier* (a number) assigned by this command, the second the *data label* assigned with *LABEL_DATA* on page 28, the third the *number of iterations*, the fourth the *current weight*, and the fifth the *current temperature*. In the sixth column the *fixed stable phases* are listed together with any *comment text* given after a *COMMENT* on page 40 in the *.POP or *.DOP file. If the weight is zero for an equilibrium columns 3-5 are replaced by the text <*unused*>. If the alternate mode is used for some experimental points, the listing is slightly different for such points. The 3-4 columns are displayed with *alt*, instead. If an alternate calculation is failed at one experimental point, the point is automatically assigned with a zero weight, and a warning message is shown above the data line (with all six columns).

EXPERIMENT

This command is used in original experimental data files (*.POP or *.DOP), but can also be given interactively inside the ED-EXP module, to change the value or uncertainty of an experiment or to add more experimental information.

An experiment usually consists of two parts with a colon (:) separating them:

- as a quantity relation, and
- as the uncertainty of the value for the quantity,

The quantity relation can be a normal POLY-module condition or an inequality (which is similar to condition but the relation between the quantity and given value is not in equality, i.e. < or >). The uncertainty can be expressed as an absolute value or relative to the value (x%).

An experiment that uses an inequality gives zero contribution to the sum of errors in the optimization procedure of PARROT if the value is on the right side. If the value is on the wrong side, the value after the colon determines how steeply the error increases with the value.

Syntax	EXPERIMENT
	The experiment must be typed after the command. Several experiments can be given on the same line. The syntax of EXPERIMENT is similar to a CONDITION. Usually, a state variable set equal to a value with a given uncertainty is specified after the colon (:). For example, `T=1273.15:5`. An inequality, < or >, can also be used. For example: `ACR(C)<-0.01:0.001, W(BCC,AG)>0.05:10%`

Additional Information About the EXPERIMENT Command

A typical experiment added interactively specifies that a phase should not be stable in a certain experiment, because a phase may appear in a range of composition or of temperature where it has never been observed during the optimization. A phase is unstable if its driving force is negative, and you can add experimental information to enforce that. For example, you can suppress the HCP phase in an existing experimental point:

```
CHANGE-STATUS HCP=DORMANT
EXPERIMENT DGM(HCP)<-.001:.0001
```

`DGM` is the POLY variable for the driving force that is already divided by `RT`, and it is dimensionless. Only dormant phases can have driving forces larger than zero, as this indicates that these would be stable if allowed to form. The experiment in the above case tries to modify the model parameters to make the HCP phase unstable.

More general examples of experiment definitions:

```
EXPERIMENT X(LIQ,PB)=.10:.01 X(FCC,PB)=0.02:.01
EXPERIMENT ACR(PB)=0.8:5%
EXPERIMENT T=700:5
```

The first experiment above describes a tie-line where the experimentally measured mole fraction of Pb is 0.1 in the liquid phase and 0.02 in the FCC phase, and its uncertainty is 0.01 for both measurements. The second experiment is that the activity of Pb should be 0.8 with an uncertainty of 5 percent.

The reference state of the component Pb must be set with the command *SET_REFERENCE_STATE* on page 169. The final one is that the temperature should be 700 K with an uncertainty of 5 degrees.

Experiments that are functions of states variable(s) must be described as the defined functions. For example:

```
ENTER_FUNCTION HTR=HM(LIQUID)-HM(FCC);
EXPERIMENT HTR=7000:1000
```

EXPORT

Transfer/export a calculated value from an equilibrium to an optimizing variable in the PARROT workspace. Thus the value can be used, for example, in the Gibbs energy description of a phase. Of course, this variable should not be optimized (i.e. it should be set as a fixed variable in PARROT).

Syntax	EXPORT
Prompt	`FUNCTION NAME: <FUNCTION NAME>#<N>` Specify the name of the function, the value of which should be transferred to a `V` variable. The number (n) of the `V` variable must be given after the function name, separated by hash character #. The function name is UPPER/lowercase sensitive, and it should normally be given in UPPER case.

	Example
	
	`ENTER_FUNCTION STRNGY=GM(FCC).X(CU);`
	
	`EXPORT STRNGY#6`
	
	This transfers the value of the partial derivative of the Gibbs energy of the FCC phase with respect to the mole fraction of Cu to the optimizing variable 6 (i.e. `V6`).

IMPORT

Transfer/import the value of one of the optimizing variables to a constant. Normally, it is directly used inside an experimental data (*.POP or *.DOP) file. It is the inverse of *EXPORT* on the previous page.

This is useful if several experiments are done at the same activity of a component, but this activity is not known. The activity should then be optimized and all equilibria with the same activity should use the same value. In this case the variable must be set as an optimizing variable, and an initial guess of the activity should be given. During the optimization, the PARROT program tries to find the activity that gives the best fit.

If an approximate value of the activity is known, it can be supplied as an experiment.

See example 36 in the *Console Mode Examples Guide*.

Syntax	IMPORT
Prompt	`CONSTANT NAME: <CONSTANT NAME>#<N>` The value of the `V` variable must be assigned a symbolic constant. The number (`n`) of the `V` variable must be given after the constant name, separated by hash character `#`. The constant name is UPPER/lowercase sensitive, and it should normally be given in UPPER case.

	Example
	`ENTER_CONSTANT ACU=0.1`
	`IMPORT ACU#2`
	This transfers the value of the optimizing variable 2 (i.e. `V2`) to the constant `ACU`.

LABEL_DATA

Add a label to the experimental equilibrium point, either as a single point or several points given individually or in a table. The label is maximum four characters and must start with the letter A. Normally, it is directly used inside an experimental data (*.POP or *.DOP) file.

Several experimental equilibria can be given the same label, and the label can be used in the command *SET_WEIGHT* on page 33 to assign the same weight to all equilibria with the same label.

Syntax	LABEL_DATA
Prompt	`LABEL? /A1/: A<BCD>` Specify a textual label (as maximum characters and must start with the letter `A`) for the current experimental equilibrium point(s).

LIST_ALL_EQUILIBRIA

Syntax	LIST_ALL_EQUILIBRIA
	List all the details (including equilibrium indicators, labels, iteration numbers, weight, and fixed phases or comments) of all equilibrium points from the current one to the last one on screen. You can always use *SELECT_EQUILIBRIUM* on page 162 to select a specific point as the current equilibrium point for such a list.

MAKE_POP_FILE

It is possible in the ED_EXP module to make interactive changes and additions to the original experimental data file (the *.POP or *.DOP file).

To document or transfer the current or final experimental data to a *.POP/*.DOP file, you can use this command to write it to a new *.POP/*.DOP file or overwrite it on an old one.

Check this output carefully for errors. Tables in original *.POP/*.DOP files are written as individual experimental points. The output can be shown on screen or be saved as a new *.POP file where to write or overwrite the edited experimental information in a simple text format.

If an existing *.POP file name is specified, it is overwritten. In the DICTRA module, an *.DOP file is written or overwritten.

Syntax	MAKE_POP_FILE
Prompt	OUTPUT TO SCREEN OR FILE /SCREEN/: <POP/DOP-FILE NAME> Specify a name of the new *.POP/*.DOP file where to write or overwrite the edited experimental information in a simple text format.

READ

This command must be given each time the ED_EXP module is entered, unless it is already given once in ED_EXP and does not use *LIST_RESULT* on page 104 or *OPTIMIZE_VARIABLES* on page 109 in PARROT in between.

Syntax	READ
Prompt	BLOCK NUMBER /#/: <N> The number of data block that should be edited must be given. If there is no command *FLUSH_BUFFER* on page 41 in the original POP or DOP file, then there is only one data block with the number 1.
	READ_WORKSPACE = READ This command (previously named READ_BLOCK) is equivalent to *READ_WORKSPACE* on page 232 in the POLY module, but it only reads the POLY3 workspace from the work file set by *SET_STORE_FILE* on page 116 in the PARROT module.

RESTORE_ALL_WEIGHTS

Restore a specific weight-set (previously saved by *STORE_ALL_WEIGHTS* on page 34) and assign all the different experimental points in the current data block with the previously-set weights in the sequential assessments.

Syntax	RESTORE_ALL_WEIGHTS
Prompt	`SET NUMBER (0 FOR LIST) /0/: <WEIGHT-SET NUMBER>` The default value 0 is for a list on screen.

SAVE_WORKSPACES

In the PARROT workspace this command saves the current status of the EDIT_EXPERIMENT submodule before you switch back to the PARROT module. This is required when you want to save equilibria changes, including changes to conditions, experimental data, and/or computed equilibria results, as well as any newly added equilibria.

With this command all the changes made in the EDIT_EXPERIMENT submodule are saved in the PARROT workspace. This means you do not need to execute *SAVE_PARROT_WORKSPACES* on page 111 in the PARROT module.

However, when in the PARROT module, additional changes can only be saved with the SAVE_PARROT_ WORKSPACES command even if they have been made prior to executing SAVE_WORKSPACES in the EDIT_EXPERIMENT module.

When you exit the EDIT_EXPERIMENT module (or Thermo-Calc), the next time the file is opened it is in the same state as when you executed the SAVE_WORKSPACES command. However, if you use *OPTIMIZE_VARIABLES* on page 109 the computed results are automatically updated.

Syntax	SAVE_WORKSPACES
Prompt	`<FILE NAME> <Y OR N>` Sometimes good starting values (*SET_START_VALUE* on page 171) are required to successfully get or compute an equilibrium correctly. In this case, it is important to save the computed equilibrium in the PARROT workspace. It is important to remember that the computed results can be ruined by improper optimizations. For this reason, it is suggested you save the PAR file with a different name to back up the PARROT workspace.

SET_ALTERNATE_CONDITION

A special command used in the experimental data *.POP/*.DOP file but also possible in the ED_EXP module. It is used only when you have specified that the alternate mode is set in the PARROT module.

The command syntax is almost the same as for the POLY command *SET_CONDITION* on page 165 while the uncertainty should also be specified.

The alternate condition (including the normally POLY-module condition plus uncertainly; see below) must be given explicitly, but can be given on the same line or on separate lines with each one started with the command.

Syntax	SET_ALTERNATE_CONDITION
Prompt	`STATE VARIABLE EXPRESSION: <STATE VARIABLE NAME OR LINEAR EXPRESSION>` Give either a state variable or a linear expression of state variables.
Options	**Description and Information**
State variables	Some of the state variables that can be used in conditions are: • T (temperature in the system) • P (pressure in the system) • N (system size in moles) • B (system site in grams) • W(<component>) (mole fraction of a component in the system) • X(<component>) (fraction of a component in the system) • ACR(<component>) (activity of a component in the system) • MUR(<component>) (chemical potential of a component in the system) • W(<phase>,<component>) (mole fraction of a component in a phase) • X(<phase>,<component>) (mass fraction of a component in a phase) • ACR(<phase>,<component>) (activity of a component in a phase) • MUR(<phase>,<component>) (chemical potential of a component in a phase) • H (enthalpy in the system) • HM(<phase>) (enthalpy of a phase per mole)

Additional prompts	`FACTOR: <A FACTOR FOR THE STATE VARIABLE, OR A CONTINUATION>`
	This question means that the previous question was not answered. The program is then expecting a single state variable or a complete state variable expression, or the numeric factor in an expression with only one state variable. In a state variable expression a state variable may be preceded by a constant factor. An example of this is:
	`2*MUR(FE)+3*MUR(O)=-35000`
	This means that it should be a condition that two times the chemical potential of FE plus

Syntax	SET_ALTERNATE_CONDITION
	three times the chemical potential of O should be -35000 J/mol.
	`STATE VARIABLE: <A SPECIFIED STATE VARIABLE, OR A CONTINUATION>` This question is prompted if a single state variable name has not given in either the prompt `State variable expression` or `Factor`, or a state variable expression is given but the expression is incomplete, for example, `T-` or `2*MUR(FE)+`, for which the program is then expecting a continuation of the unfinished expression. You need to specify a state variable or a complete state variable expression, or complete the unfinished state variable expression. If a numeric factor is given before this prompt, only one state variable can be specified; otherwise, the program only takes the first state variable to complete the expression (i.e. the factor times the state variable).
	`VALUE /X/: <A NUMERIC VALUE, A CONSTANT OR A VARIABLE>` The value of the condition. This can be a numeric value, a constant or a variable. A suggestion is given as the default value. The special value `NONE` means that the condition is removed.
	`UNCERTAINTY /NONE/: <UNCERTAINTY OF THE CONDITION>` The uncertainty of the condition. This can be a numeric value, a constant or a variable. The default value `NONE` means that the uncertainty for the value specified above is zero. The uncertainty can either be expressed as an absolute value or relative to the value of the condition in percent (`x%`).

Notes	There are more state variables that can be used in conditions. For more information, type an INFO STATE_VARIABLES command in the POLY module. A condition is normally a value of a single state variable with its value.
	
	For example:
	
	`T=1273.15 P=1E5`
	`X(C)=.002`
	`W(CR)=1.5`
	`ACR(CR)=0.85`
	`X(FCC,C)=.001`
	`H=-250000`
	`HM(BCC)=-225000`
	
	A condition can also be a value of a linear expression involving more than one state variable.
	
	For example:
	
	`X(LIQ,S)-X(PYRR,S)=0`

Syntax	SET_ALTERNATE_CONDITION
	This means that it is a condition that the mole fraction of S should be the same in the LIQUID and PYRRHOTITE phases. In practice that should be the congruent melting point. After the equal sign only a numeric value is allowed.

SET_WEIGHT

Each experimental value has an uncertainty associated with it, specified by the value after the colon (:) with the command *EXPERIMENT* on page 24. During an optimization the absolute difference between the experimental and calculated values gives a contribution to the sum of error.

With the SET_WEIGHT command, you can change the scale of such a contribution (the uncertainty) for a single experimental point, or the contributions (all uncertainties) for a set of equilibria. The default weight is always unity. A value smaller than one means that the experiments should have less weight. A value larger than one that these should have higher weight. The special value zero means that the set of equilibria should be ignored in editing in ED_EXP (as *COMPUTE_ALL_EQUILIBRIA* on page 23 is given) and in *OPTIMIZE_VARIABLES* on page 109 in PARROT.

The weight may be needed to obtain a balance between different kinds of experiments. For example, if there are only five experimental values of the composition of a phase diagram but 500 experimental values of activities or enthalpies, then the five composition points may have to be given higher weight than unity, otherwise these are not well described by the optimizing procedure.

Syntax	SET_WEIGHT
Prompts	`VALUE /1/: <WEIGHT VALUE>` Specify a weight of the experiments in the specified equilibria (asked in the next prompt). The contribution to the sum of errors of these experiments are multiplied by this weight. The weight is squared, thus use 0.7 to make the error half (0.49) as big, and 1.4 to make it twice (1.96) as large.
	`EQUILIBRIA (RANGE) OR LABEL(S) /PRESENT/:<SELECTION>` Define the equilibria. These are available in the read data block from the current work file and are given the above-defined weight. The selection may be given as a range but the equilibrium numbers must then be separated only by a minus sign.

Syntax	SET_WEIGHT
	The start and end equilibrium numbers must be available in the read data block read from the current work file. For example, `5-101`, means that all equilibria starting from the identifier 5 until the identifier `101`. Some of numbers between these two identifiers may not exist in the data block. The selection may also be given as one or more labels (which should be separated only by a space). All equilibria with the specified label(s) in the read data block are given the above-defined weight. By pressing <Enter> to accept the default value `present`, the above defined weight is only assigned to the current or present experimental point (which is selected by the POLY command *SELECT_EQUILIBRIUM* on page 162). If *COMPUTE_ALL_EQUILIBRIA* on page 23 is used but no specific equilibrium point is selected afterwards, the last point in the read data block remains the `present` point.

STORE_ALL_WEIGHTS

Store a new weight-set that is the current situation of specified weights for all the different experimental points read from the currently-selected experimental data block in an assessment. This is useful when you want to try various weight-settings for some experimental points or for all the points, and then compare the assessment results from different weight-sets. Such a saved weight-set can be recalled and restored for all experimental data point with*RESTORE_ALL_WEIGHTS* on page 29 sequentially.

Syntax	STORE_ALL_WEIGHTS
Prompt	`COMMAND LINE: <WEIGHT-SET NAME>` Give a weight-set name (comment line texts) for the current weight set for all experimental points.

TRANSFER_START_VALUE

In the ED_EXP module it is time consuming to calculate all equilibria when the optimization is sensitive to start values of the composition of the phases. Usually in one data block there are several experimental points with the same kind of equilibrium and each must have its start value set. When you manage to calculate equilibriums of such experimental points, this command is useful to transfer the site fractions from a successfully-calculated equilibria to the present experimental point.

Syntax	TRANSFER_START_VALUE
Prompt	`FROM EQUILIBRIUM /PREVIOUS/: <EQUILIBRIUM NUMBER>` The equilibrium number is the numeric identifier for the equilibrium from which the start values should be copied to the present experimental point. `Previous` is the default if you press <Enter>.

Experimental Data Files

In this section:

Commands in the Experimental Data Files (*.POP, *.DOP)

In order to conduct an assessment for a system, the experimental data is described with a syntax, which is similar to the way that one calculates an equilibrium in the POLY module or you edit an experimental point in the ED_EXP module. Similar to a Thermo-Calc (TCC) macro file (*.TCM), an experimental data file (i.e. the *.POP or *.DOP file) is a basic text file and can be opened and edited by any text editor. For this reason, an *.POP or *.DOP file is not hardware dependent.

A POP or DOP file consists of various commands from the POLY and ED_EXP modules, as well some special commands which can only be used in such experimental data files. Various experimental information can be inputted in an *.POP or *.DOP file as different tables; the same type of experimental data are usually documented in the same table (see below).

An *.POP or *.DOP file is used in the PARROT module to provide experimental information for the optimization process, and is checked by a *syntax checker* when the PARROT command *COMPILE_ EXPERIMENTS* on page 92 is proceeded.

Many POLY and ED_EXP commands can be directly used in a POP or DOP file. But there are some differences with the set of POLY or ED_EXP commands, and some specially designed commands as described in this topic can only be used in the *.POP or *.DOP files.

Frequently Used Commands

Most of the commands in the experimental data file are the same as in the POLY module. The most frequently used are listed below:

- *CREATE_NEW_EQUILIBRIUM* on page 143
- *CHANGE_STATUS* on page 135
- *SET_CONDITION* on page 165
- *SET_REFERENCE_STATE* on page 169
- *ENTER_SYMBOL* on page 195
- *SAVE_WORKSPACES* on page 30

As the last command in an *.POP or *.DOP file, you must always have the SAVE command.

Other Commands

Other legal commands from the POLY module that are used less often are:

- *DEFINE_COMPONENTS* on page 143
- *EVALUATE_FUNCTIONS* on page 155

- *SET_ALL_START_VALUES* on page 162
- *SET_NUMERICAL_LIMITS* on page 168
- *SET_START_VALUE* on page 171
- *ADVANCED_OPTIONS* on page 125

The DEFINE-COMPONENTS command must be always used as the first command in an *.POP or *.DOP file, as it automatically reinitiates the whole workspace.

Most of the special commands for the ED_EXP module are also often used in an *.POP or *.DOP file. For example:

- *EXPERIMENT* on page 24
- *EXPORT* on page 26
- *IMPORT* on page 27
- *LABEL_DATA* on page 28
- *SET_ALTERNATE_CONDITION* on page 30
- *SET_WEIGHT* on page 33

The following special ED_EXP commands are illegal and should not be used in any *.POP or *.DOP file:

- COMPUTE-ALL-EQUILIBRIA
- MAKE-POP-FILE
- READ (READ-BLOCK OR READ-WORKSPACE)
- TRANSFER-START-VALUES

The special commenting identifier, i.e. a single dollar sign $ which starts a line, is used to document comment lines at any position in an *.POP or *.DOP file. You may remember that this feature is universally the same as in almost all kinds of text files in Thermo-Calc software, e.g. in *.TDB, *.DAT, *.TCM, *.POP/*.DOP and *.EXP files. Such a sign and all information afterwards in the same comment line is skipped and ignored by the syntax checker. Therefore, you shall feel free to write such comment lines in any of these types of text files.

TABLE_HEAD, TABLE_VALUES and TABLE_END

These commands are used only in the *.POP/*.DOP files. It represents a convenient way to enter many experimental measurements of the same type in a table format.

> These commands should always be used together and in a sequence, meaning that a table should always start with the TABLE-HEAD command, then follows the TABLE-VALUES command, and finish by the TABLE-END command. Between the two TABLEHEAD and TABLE-VALUES commands, there should exist some definition lines (e.g. on phase status, reference states, conditions, experiments, labels, etc.) for the experimental measurements of the current information type.

<table>
<tr><td>Syntax</td><td>TABLE_HEAD
TABLE_VALUES
TABLE_END</td></tr>
<tr><td>Prompt</td><td></td></tr>
<tr><td></td><td>After the TABLE-HEAD command, there must be an equilibrium description similar to a single experimental equilibrium but with some special notation. Then, there is always a TABLE-VALUES command, after which the actual data is given in columns. At the end of each table, there must be a TABLE-END command.

The TABLE-HEAD command must be followed by a numeric value. This is used to generate unique numeric identifiers for all the equilibria in the table. The numeric identifier is incremented by one for each experimental (equilibrium) point in the table.</td></tr>
<tr><td></td><td>Example

An example of the use of a table for enthalpy measurements in the liquid is given here. Much more elaborate tables can be used

<code>$ Enthalpies of mixing, Topor and Kleppa, Met Trans 1984 TABLE-HEAD 1</code>
<code>CREATE—NEW @@ 1</code>
<code>CHANGE-STATUS PHASE LIQ=FIX 1</code>
<code>SET-REFERENCE-STATE AU LIQ * 1E5</code>
<code>SET-REFERENCE-STATE CU LIQ * 1E5</code>
<code>SET-CONDITION P=1E5 T=1379 X(LIQ,AU)=@1</code>
<code>LABEL ALH</code>
<code>EXPER HMR=@2:5%</code>
<code>TABLE—VALUES</code></td></tr>
</table>

Syntax	TABLE_HEAD TABLE_VALUES TABLE_END

	`0.0115 -322`
	`0.0563 -1520`
	`0.8499 -2976`
	`0.9002 -2114`
	`TABLE—END`

	The equilibrium description between TABLE-HEAD and TABLE-VALUES is similar as for a single experiment, except for these details: • @@ Used to automatically generate a series of identifying numbers instead of creating only one with the command *CREATE_NEW_EQUILIBRIUM* on page 143. The program automatically generates the identifying numbers starting after the number given after the TABLE-HEAD command (in this case it is 1). • @1 values in the table are specified by an @ sign followed by a column number. In the above case, the mole fraction of Au in the liquid is in column 1 (see the line for SET-CONDITION). The columns may have other information than values, and you can use phase names or any text. If the text contains spaces or special characters, it must be surrounded by double quotes in the table, e.g. "`ABC DEF&ghi`". • For the *syntax checker* all commands describing the equilibrium must be in UPPER case after the TABLE-HEAD command. • After the TABLE-VALUES command, there should be one line for each experimental measurement of the same type giving the values or texts that should be copied into the places of the condition or experiment defined by `@1`, `t`, etc. You can have columns that are not used but there must be exactly the same number of columns of each line, otherwise the syntax checker gives an error message in the PARROT module.

COMMENT

This command is only used in the *.POP/*.DOP files to write brief comments or descriptive information on a newly created equilibrium (experimental) data point or a set of points given in a table.

Syntax	COMMENT
	A comment can have a maximum of about 60 characters written on the same line.

FLUSH_BUFFER

This command is only used in the *.POP/*.DOP files. It is needed if the number of experiments require more space than can be fitted into one buffer. When the PARROT program compiles experiments, it informs if any FLUSH commands are needed. The FLUSH_BUFFER command terminates the current block, save it to the work file and starts a new block. With the READ command in the EDIT-EXP module, you can select the block to edit.

Syntax	FLUSH_BUFFER
	After a FLUSH_BUFFER command, the workspace is reinitiated and all functions or constants must be entered again in the *.POP file. You can take advantage of this; for example, you can use the FLUSH command to have blocks with different components in one *.POP file.

GIBBS_ENERGY_SYSTEM Commands

In this section:

ADD_COMMENT

Syntax	ADD_COMMENT
	Add a comment or make notes about parameters.

AMEND_ELEMENT_DATA

The data for an element (in the SER, the Stable Element Reference, state) can be changed by this command. It should only be used for the elements that do not have any data in the database because the element data available in the database is set by the database-developer for the purpose of internal consistency. Except for the mass, the other values have no influence on the calculations.

Syntax	AMEND_ELEMENT_DATA
Prompts	`ELEMENT NAME: <ELEMENT NAME>` Specify the name of the element for which you want to change the data.
	`NEW STABLE ELEMENT REFERENCE /ABCD/: <NAME OF SER>` Press <Enter> to accept the default `SER` or specify a new SER for the element. Important: The default name should not be changed if this data is retrieved from a database. This name is used when parameters for a phase are listed and the database assumes that the stable element reference is the same as in the database. Only if the element's data have not been fetched from a database, e.g. entered manually, can you enter a new SER.
	`NEW ATOMIC MASS /XX.XXXX/: <YYYYYY>` Press <Enter> to accept the default atomic mass or specify a new value for the element. The atomic mass of the element is given in g/mol.
	`NEW H(298.15)-H(0) /XXX.XXX/: <YYYYY>` Press <Enter> to accept the default H(298.15)-H(0) or specify a new value for the element. H(298.15)-H(0) is the enthalpy difference between 298.15 K and 0 K for the element in its SER state.
	`NEW S(298.15) /XX.XXXX/: <YYYYY>` Press <Enter> to accept the default S(298.15) or specify a new value for the element. S (298.15) is the absolute value of the entropy at 298.15 K for the element in its SER state.
	`DEFAULT ELEMENT REFERENCE STATE SYMBOL INDEX /#/: <INDEX>` The index only changes the symbol, not any value. Normally the index is set correctly by the database. Only when manually entering data, you must set the index to get the correct symbol.

Syntax	AMEND_ELEMENT_DATA
	Specify an index for the default listing parameters (symbol), or press <Enter> to accept the pre-set index. 0 is for `G`, 1 is for `H298` and 2 is for `H0`. The index is to define the symbol printed in parameter listings. The symbol can be: • `G` the data are referred to Gibbs energy at a variable temperature (also called *lattice stability*). • `H298` the data are referred to the enthalpy of the element at 298.15 K and 1 bar. • `H0` is the same as `H298` but at the temperature 0 K.

AMEND_PARAMETER

Use this command to interactively change/amend the temperature-pressure function of a parameter. This is useful to correct typing errors because the old function is made available on the terminal for interactive editing. For most of these prompts, refer to *ENTER_PARAMETER* on page 99 for details.

This AMEND_PARAMETER command is for the GES module. There is also a PARROT module command with the same name (*AMEND_PARAMETER* on page 88).

Syntax	AMEND_PARAMETER

Prompts	`PARAMETER: <PARAMETER NAME>`
	Specify a correct parameter name. If a parameter name is not acceptable or you only press <Enter>, the error message displays:
	`ERROR, PLEASE RE-ENTER EACH PART SEPARATELY`
	The program prompts for separate input for each part for a parameter name.

	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>` Specify one of these legal identifiers as described for ENTER_PARAMETER.
	`PHASE NAME (/ABCD/): <PHASE NAME>` Specify the phase name.
	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>` Specify the constituent name.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` Specify the interacting constituent name: If there is no interacting constituent, press <Enter>.

Syntax	AMEND_PARAMETER
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` If there are more than one interacting constituents, specify them; otherwise press <Enter>.
	`DEGREE /#/: <DEGREE>` Specify a numerical number as the degree of the phase parameter.

	After the parameter name is specified correctly, the program lists its current definition (either present in the database or defined by the ENTER_PARAMETER command), such as:
	
	`L(PHASE2,AL,MG;1)=`
	`298.15<T<2000.00: +5000`
	`2000.00<T<4500.00: +4500`
	`4500.00<T<6000.00: +4000`
	
	Then you are prompted to change the parameter definition, as shown below:
	
	`DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: <Y OR N>`
	
	If you want to change the number of ranges for the chosen function, or change some of the temperature limits in the definition, for Y, retype both the low/high temperature limits and functions.
	
	If you do not want to change the number of ranges but want to change the function(s) in one or more ranges, press <Enter> to accept the default N, then the whole definition of the chosen parameter in all ranges (if any) is listed on screen, such as:
	
	`DIFFERENT FUNCTIONS IN THESE RANGES`
	`298.15<T<2000.00`
	`2000.00<T<4500.00`
	`4500.00<T<6000.00`
	
	The prompt is given:
	
	`DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y OR N>`
	
	If there is more than one range, this question is prompted press <Enter>.

	`RANGE NUMBER (0 TO EXIT) /0/: <RANGE NUMBER>` If the function of a parameter is different in two or more temperature ranges, you must specify the range of the function of which you want to amend. Or press <Enter> or type `0` to exit this command without making any changes.
	`FUNCTION:`

Syntax	AMEND_PARAMETER

The previous function is available for editing. The editing is performed within the general subroutine FOOLED, as described in *AMEND_SYMBOL* on page 90. This routine prompts as follows:

1:+:>

The prompt consists of the current position in the string and the character at that position between colons, (::).

Commands

These commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurrences> F<string>@
- Insert: I<string>@
- Move: <+-#positions> M
- Restore string: R
- Substitute: S<OLD>@<NEW>@
- Type string: T

Where CP denotes the current position in string, # means number of, @ is a terminator of an input or search string.

When the string is typed the character at the current position is replaced by an underscore _.

To finish the editing of the current function, type E at the prompt.

Solution Models in the GIBBS Module

The command *AMEND_PHASE_DESCRIPTION* on page 52 is used to specify/amend phase descriptions if a phase has a miscibility gap, uses a special excess energy model, or has a specific additional contribution to its Gibbs energy, for example.

This topic outlines the application of each option and includes important information about each command.

There are many solution models implemented in the GIBBS module (type `INFO MODELS` to see details about various thermodynamic models). However, these are switched on and handled differently via various GIBBS routines/commands:

- The first models the non-ideality as excess parts, i.e. by amending the phase’s EXCESS_MODEL and/or TERNARY_EXTRAPOLAT models.
- The second models the non-ideality as ADDITIONAL parts, i.e. by amending the phase’s other subjects in this command, e.g. MAGNETIC_ORDERING for the Magnetic Ordering Model, DISORDERED_PART for the CVM Approach in modelling chemical ordering/disordering phenomenon, QUASICHEM_IONIC for using a quasi-chemical entropy term for an ionic two-sublattice liquid solution phase, QUASICHEM_FACT00 or QUASICHEM_IRSID for describing a substitutional liquid solution phase, DEBYE_HUCKEL for the DHLL/SIT Model in describing a dilute aqueous solution phase, etc.
- The third implements the model entirely or partially into the GIBBS module and related database(s), such as for the electrostatic contribution in an AQUEOUS solution phase by the Complete Revised_HKF Model, the Murngham Model, Birch-Murngham Model or Generalized PVT Model for high-pressure/temperature volume contribution in a solid or liquid phase, the SUPERFLUID Model for the non-ideal EOS and non-ideal mixing in a gaseous mixture.

NEW_CONSTITUENT

Use this if you want to add a new constituent to a phase. It is illegal to add new constituents to an ionic two-sublattice liquid phase.

RENAME_PHASE

Use this to change the names of some specific phases. For example, a phase called *FE3O4_S* may be better named *Magnetite* to help identify it during the calculations and postprocessing. This is also a way to delete a phase by hiding it under a new name.

SITE_RATIOS

Use this to change the number of sites (i.e. the stoichiometric coefficients of various sublattices) in a sublattice phase.

COMPOSITION_SETS

Use this for solution phases that may have miscibility gap(s). However, this is less important today and often unnecessary to define additional composition set(s), since the implemented Global Minimization Technique can usually detect and add such composition set(s) in an automatic manner where it is really necessary during equilibrium calculations (of single-points, stepping or mapping).

MAJOR_CONSTITUENT

Use this to set major constituent(s) on each sublattice in each composition set for a solution phase. This is useful to make calculations converge faster because it may simplify giving start values when calculating the equilibrium as those phases with miscibility gaps should have different major constituents for each composition set. The databases often set major constituents for several phases automatically when data are retrieved.

FRACTION_LIMITS

Use this to limit the application range (in terms of mole-fractions of all the involved elements) of a particular solution phase. This is useful to avoid automatic creations (enforced by the Global Minimization Technique) of additional composition sets for some solution phases (such as AQUEOUS solutions or dilute Fe-based liquid mixtures) of which the used models [e.g. the `SIT Model` or the `Complete Revised_HKF Model` for AQUEOUS solution, or the modified dilute solution parameters (plus a quadratic term, according to Hillert (1986) based on the SigworthElliot Model (Sigworth and Elliot, 1974) for Fe-rich liquid mixture] cannot be appropriately applied on a full scale.

Globally set the composition limits (in terms of mole fractions of various elements) in a specific solution phase, so that whenever the program finds a potential phase composition or a new composition set of possible miscibility gap(s) but that is out of this globally-set composition range, the program automatically ignores such a phase composition in an equilibrium calculation. This can be done either permanently inside a database (by enforcing this option for amending the phase-description of the solution phase) or temporarily within the GIBBS module (by using this phase-description amendment option).

This is important for solution phases for which the thermodynamic models and assessed data for the phases are for specific composition ranges, for example, the FE_LIQUID phase in the SLAG database is only applicable for Fe-rich liquid mixture, and the AQUEOUS phase in the TCAQ/PAQ and AQS/PAQS databases is only applicable for H2O-dominant aqueous solution phase.

Specify the low and high mole-fraction limits for each of the elements possibly available within the considered phase (in the currently defined system, or in a certain database). Do this for all the possible elements (defined in the phase) in a single AMEND_PHASE_DESCRIPTION FRACTION_LIMITS command-sequence, for example:

```
TYPE_DEFINITION R GES AM_PH_DES FE_LIQUID FRACTION_LIMITS Fe 0 0.6
      Ag 0 0.01  Al 0 0.05  Ca 0 0.05  Co 0 0.01  Cr 0 0.01
      Cu 0 0.02  Mg 0 0.05  Mn 0 0.05  Mo 0 0.05  Nb 0 0.05
      Ni 0 0.05  Pb 0 0.05  Si 0 0.10  Sn 0 0.02  Ti 0 0.05
U  0 0.01  V  0 0.02  W  0 0.02  Zr 0 0.03
      B  0 0.01  C  0 0.01  H  0 0.01  N  0 0.01  O  0 0.01
P  0 0.01  S  0 0.01  !
```

DISORDERED_PART

This command is needed for the special treatment of chemically-ordered phases where the contributions from the disordered state are described by a phase without ordering sublattices (the disordered phase name must be specified).

Several checks are made that the ordered and disordered phases are compatible (sublattices, sites and constituents). A link is then created between the phases, and the disordered phase is hidden from application programs. The Gibbs energy for the ordered phase also includes the Gibbs energy for the disordered phase.

Phases which can have an order/disorder transformation have parameters split on two phases and are referred to as the *two phase* description in the GIBBS module. One of them has sublattices for chemical ordering, the other one represents the disordered state. Normally, the ordered BCC and FCC or HCP phases may have either 2 or 4 substitutional sublattice (plus one additional interstitial site), that are handled by the Two Substitutional-Sublattice Ordering Model or Four Substitutional-Sublattice Ordering Model, respectively.

Two phase means that the *ordered* phase has parameters that describe the ordering. The *disordered* phase has all parameters for the reference state and those for describing the disordered phase. The ordered phase may occur only is some systems whereas the disordered phase may occur frequently, typical examples are the disordered FCC and BCC which may become ordered as L12 or B2 in certain systems. In order to treat multicomponent systems where some subsystems have ordering it would be necessary to transform all parameters of BCC into a B2 model.

The parameters describing the disordered phase are not changed but the Gibbs Energy system is informed that the Gibbs Energy for the two phases should be added.

The method used calculates the Gibbs Energy of a two phase model with two or four substitutional sublattices for ordering.

$$G_m = G_m^{ord}(y_i^{'}, y_i^{''}) + G_m^{dis}(x_i) - G_m^{ord}(y_i^{'} = x_i, y_i^{''} = x_i)$$

The mole fractions *x*i are calculated from the site fractions *y*i. The last term means that the contribution from Gmord in a disordered state (same site fraction, equal to the mole fraction, on both sublattices) is subtracted. The effect of this is that the parameters in the ordered phase have no contribution to the disordered state.

MAGNETIC_ORDERING

Change the magnetic ordering parameter for a certain phase with magnetic ordering contribution, in terms of its anti-ferromagnetic factor. By default this is –1 for BCC phase and –3 for all other phases (FCC, HCP, etc.). The fraction value of the total enthalpy (due to short-range ordering above the magnetic transition temperature) is by default 0.40 for BCC phase and 0.28 for all other phases (FCC, HCP, etc.).

EXCESS_MODEL

Use this to change the default Excess Model (for interaction energies) from the default (REDLICH-KISTER_ MUGGIANU). The model handles the excess interaction energies in a solution phase is chosen from:

- REDLICH-KISTER_MUGGIANU: for binary (R-K) & ternary (R-K-M) interactions
- REDLICH-KISTER_KOHLER: for binary (R-K) & ternary (R-K-K) interactions

- FLORY-HUGGINS POLYMER MODEL: for interactions in a polymer mixture phase (F-H)
- MIXED-EXCESS-MODELS: (R-K default) for mixed binary excess model (R-K, Legendre & Polynom) of a substitutional solution phase
- HKF: for interaction in an AQUEOUS solution phase (HKF)
- PITZER: for interaction in an AQUEOUS solution phase (PIT)
- CENTRAL_ATOM_MODEL: for interaction in a liquid slag solution phase (C-A-M)

For extrapolations of excess energies from related binary systems to ternary or higher-order systems, the Redlich-Kister binary excess interaction parameters may be extrapolated with either a Muggianu extension (i.e. the default REDLICH-KISTER_MUGGIANU model) and a Kohler extension (i.e. the alternative REDLICH-KISTER_KOHLER model), where there is no ternary, quaternary or higher-order interaction parameter.

The REDLICHKISTER_KOHLER model is implemented only for ternary systems.

The MIXED-EXCESS-MODELS option works only for a substitutional phase with no sublattice (such as the metallic LIQUID phase) and it can be used to invoke asymmetrical simple or Legendre polynomial as binary excess energy models in addition to the default symmetrical Redlich-Kister model for the chosen pair of constituents in a substitutional phase with no sublattice. Asymmetrical here means that the power series depend only on one of the constituents, for example with expansions based on the [1-2*X(B)] term rather than [X(A)X(B)].

Binary interaction parameters for such a pair of constituents must be entered prior to turning on the non-default Legendre or Polynom models.

See example 52 in the *Console Mode Examples Guide*for an example.

TERNARY_EXTRAPOLAT

Use this to change the extrapolation method from the default REDLICH-KISTER_MUGGIANU to another extrapolation model. This method extrapolates from binary to ternary (and higher-order) excess interaction parameters in a solution phase and is chosen from:

- MUGGIANU for Muggianu Extrapolation from R-K parameters
- TOOP-KOHLER for Toop-Kohler Extrapolation Model
- KOHLER-ALL for Kohler Extrapolation Model
- MUGGIANU_RESTOR for Muggianu-Restor Extrapolation Model

Only when all the relevant binary excess energies in the current solution phase are treated by the default Redlich-Kister Model (i.e. the Mixed-Excess-Model should have not been used), the MUGGIANU_RESTOR method for ternary extrapolations is equivalent to the Redlich-Kister_Muggianu Model, or the KOHLER-ALL method to the RedlichKister_Kohler Model.

The default ternary extrapolation method MUGGIANU_RESTOR applies to a solution phase as a whole, whatever the binary excess model(s) enforced to each of individual binary pairs in the phase. In case that all the binary pairs in the phase use the default binary excess model REDLICH-KISTER, then the ternary extrapolation method becomes the so-called Redlich-Kister_Muggianu Model (in short as R-K-M), for extrapolations from binary parameters to ternary (and higher-order) excess energy terms; when no ternary L parameter is entered for that, such a default R-K-M Model is always used.

The KOHLER-ALL ternary extrapolation method can be turned on also for a solution phase as a whole, whatever the binary excess model(s) enforced to each of individual binary pairs in the phase. In case that all the binary pairs in the phase use the default binary excess model.

REDLICH-KISTER, then the ternary extrapolation method becomes the so-called Redlich-Kister_Kohler Model (in short as R-K-K), for extrapolations from binary parameters to ternary (and higher-order) excess energy terms; when no ternary L parameter is entered for that, such an R-K-K Model is used.

The TOOP-KOHLER ternary extrapolation method is applied in a special way: for a specific ternary system (e.g. A-B-C) in a solution phase, specify which two constituents as the so-called Kohler constituents and the remaining constituent as the so-called Toop constituent on a given sublattice. If for the A-B-C ternary system you have specified A and B, or B and A, as the Kohler constituents (entered as the basis constituent and first interacting constituent) and C as the Toop constituent, only the A-B binary interaction parameters are used in accordance with the Kohler ternary extrapolation formula for A-B-C ternary interaction, while any other binary interaction parameters involving the Toop species C (i.e. of A-C and B-C binaries) are used in line with the Toop-Kohler ternary extrapolation formula (for the A-C-B and B-C-A ternary interactions).

This concept is shown in the *Console Mode Examples Guide*.

DEBYE_HUCKEL

To use the DHLL (Debye-Hückel Limiting Law) model and SIT (Specific Ionic Interaction Theory) model for a dilute AQUEOUS solution phase, switch the DEBYE_HUCKEL part on, and it removes previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS phase.

HKF_ELECTROSTATIC

To use the hypothetical electrostatic contribution for the chosen phase (it must be an aqueous solution phase using the Complete Revised_HKF Model). It removes previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS solution phase.

QUASICHEM_IONIC

To use the Quasichemical-Ionic model for the chosen liquid phase, you must have already entered/retrieved a liquid phase as an ionic two-sublattice liquid solution phase (normally, that is the IONIC_LIQ phase), and then use this amending option to switch QUASICHEM_IONIC on. It creates a completely new liquid solution phase [namely the QUAS_IONIC phase which uses the Quasichemical Model for the entropy, according to Mats Hillert (2001: *J of Alloys and Compounds*, 320, 161-176)], while the original IONIC_LIQ phase remains in the system and is not changed by this option.

QUASICHEM_FACT00

To use the Quasichemical Model developed by Kongoli et al. in the F*A*C*T group in Montreal, you must have entered/retrieved a liquid phase using a normal substitutional liquid model with specified associates or species (on a single lattice site), and then turn on the option QUASICHEM_FACT00. This removes the previously-set (if any) ADDITIONAL part on the non-ideality for the chosen liquid phase.

See examples 49 and 50 in the *Console Mode Examples Guide*.

QUASICHEM_IRSID

To use the Kapoor-Frohberg-Gaye Quasichemical Cell Model (i.e. the Quasichemical Model developed by ISRID, France) for a liquid SLAG solution phase, you can use a normal liquid model with specified associates or species.

GLASS_TRANSITION

To use the special model for glass transition of a liquid phase. It removes previously set (if any) ADDITIONAL part on the non-ideality for the chosen liquid phase.

REMOVE_ADDITION

To remove all the selected ADDITIONAL part from the Gibbs energy description for the chosen phase. If preferred, set a specific phase-status bit for a phase, use the PHASE_BITS option as long you know the restrict meaning of each part of a phase- status bits.

DEFAULT_STABLE

Set phases as default-stable, which helps you to have a better guess of which phases that should be stable at the first calculation.

AMEND_PHASE_DESCRIPTION

Specify/amend phase descriptions if a phase has a miscibility gap, uses a special excess energy model, or has a specific additional contribution to its Gibbs energy, for example.

Also see *Solution Models in the GIBBS Module* on page 46 for detailed information about this command and its options.

Syntax	AMEND_PHASE_DESCRIPTION

Prompts	`PHASE NAME: <PHASE NAME>`
	Specify the name of the phase.
	`AMEND WHAT /COMPOSITION_SET/: <SUBJECT>`
	Several subjects for the phase can be amended but most often this command is used to enter two or more composition sets for a phase. If a phase has a miscibility gap it is necessary to have two composition sets, one for each possible composition that can be stable simultaneously.
	Enter a question mark at the prompt to get a list of all possibly amended subjects for a phase.

There are no additional prompts for DEBYE_HUCKEL, HKF_ELECTROSTATIC, GLASS_TRANSITION, QUASICHEM_FACT00, QUASICHEM_ISRID, REMOVE_ADDITION, and DEFAULT_STABLE.	
`RENAME_PHASE`	`NEW PHASE NAME /ABCD/: <PHASE NAME>` Give a new phase name for the chosen phase, or press <Enter> to keep the default shown.
`SITE_RATIOS`	`SITES IN FIRST SUBLATTICE /XX/ : <YY> SITES IN SECOND SUBLATTICE /XX/ : <YY>` Specify the site numbers for each of the prompted sublattices for the current phase. Press <Enter> to accept the previous definitions.

`NEW_CONSTITUENT`	`SUBLATTICE /#/: <SUBLATTICE NUMBER>`
	Specify the sublattice where the new constituents are located.
	`SPECIES: <SPECIES NAME>`
	Give a valid species name.

`COMPOSITION_SETS`	`NEW HIGHEST SET NUMBER /#/: <SET NUMBER N>`
	The default value (#) is usually one higher than the current value. All phases have initially one composition set. If a lower value (i.e. lower than the default one) is given, that specific composition sets are deleted.
	You cannot take away the first composition set.
	`MAJOR CONSTITUENT(S) FOR SUBLATTICE #: /AB/: <MAJOR CONSTITUENT(S)>`
	Specify the new major constituent(s) for the sublattice #, or press <Enter> to accept

Syntax	AMEND_PHASE_DESCRIPTION
	the default which was automatically set according to the specified composition set of the phase. This prompt is repeated for all available sublattices in the chosen phase. The major constituents in each sublattice can be given. This may simplify giving start values when calculating the equilibrium as phases with miscibility gaps should have different major constituents for each composition set.

MAJOR_ CONSTITUENT	`COMPOSITION SET /1/: <COMPOSITION SET NUMBER>`
	
	Give the composition set (digit number) for the chosen phase, or press <Enter> if you want to set major constituents for the specified composition set.
	
	`MAJOR CONSTITUENT(S) FOR SUBLATTICE #: /AB/: <MAJOR CONSTITUENT(S)>`
	
	Specify the new major constituent(s) for the sublattice #, or press <Enter> to accept the default which is automatically set according to the specified composition set of the phase.
	
	This prompt is repeated for all available sublattices in the chosen phase for the specified composition set.
	
	The major constituents in each sublattice can be specified. This is useful in order to make calculations converge faster and more easily (because it may simplify giving start values when calculating the equilibrium as those phases with miscibility gaps should have different major constituents for each composition set). The databases often set major constituents for several phases automatically when data are retrieved.

FRACTION_ LIMITS	`ELEMENT : <EL1>`
	`LOW FRACTION LIMIT /0/ :  <APPROPRIATE LOW LIMIT>`
	`HIGH FRACTION LIMIT /1/ : <APPROPRIATE HIGH LIMIT >`
	`ELEMENT : <EL2>`
	`LOW FRACTION LIMIT /0/ :  <APPROPRIATE LOW LIMIT>`
	`HIGH FRACTION LIMIT /1/ : <APPROPRIATE HIGH LIMIT>`
	`ELEMENT : <ELN>`
	`LOW FRACTION LIMIT /0/ :  <APPROPRIATE LOW LIMIT>`
	`HIGH FRACTION LIMIT /1/ : <APPROPRIATE HIGH LIMIT >`

MAGNETIC_ ORDERING	`THE ANTIFERROMAGNETIC FACTOR /XX/: <YY>`
	
	Specify the anti-ferromagnetic factor for the chosen phase. This should be –1 for BCC phase and –3 for all other phases (FCC, HCP, etc.).
	
	`SHORT RANGE ORDER FRACTION OF THE ENTHALPY /XX/: <YY>`

<table>
<tr><td>Syntax</td><td>AMEND_PHASE_DESCRIPTION</td></tr>
<tr><td></td><td>The magnetic ordering is a second-order transformation and part of the enthalpy due to this transformation is due to short-range order. This value is the fraction of the total enthalpy that is due to short-range ordering above the magnetic transition temperature.

The default value (xx) is 0.40 for BCC phase, and 0.28 for all other phases (FCC, HCP, etc.).</td></tr>
<tr><td>DISORDERED_PART</td><td>DISORDERED PHASE NAME: <PHASE NAME>

Give the disordered phase name for which there is no ordering sublattice.</td></tr>
<tr><td>EXCESS_MODEL</td><td>MODEL NAME /ABCDEFG/: <MODEL NAME>

The default model is the pre-set model for the solution phase, normally the REDLICH-KISTER_MUGGIANU model, or choose another model for the phase to be amended:

• REDLICH-KISTER_MUGGIANU
• REDLICH-KISTER_KOHLER
• FLORY-HUGGINS POLYMER MODEL
• MIXED-EXCESS-MODELS (R-K default)
• HKF
• PITZER
• CENTRAL_ATOM_MODEL

The MIXED-EXCESS-MODELS option has sub-options:

FIRST (THE INDEPENDENT) CONSTITUENT: <CONSTITUENT NAME>
SECOND (THE DEPENDENT) CONSTITUENT: <CONSTITUENT NAME>

Specify the binary pair of constituents in the current substitutional solution phase (the first one as the so-called independent constituent, and the second one as the dependent constituent), for which you wish to change the binary excess model from the default REDLICH-KISTER model to another model (LEGENDRE or POLYNOM).

You are repeatedly prompted with the first sub-option (i.e. <code>First (the independent) constituent</code> after you have specified the desired binary Excess model type. To make further changes of binary excess model for other specific binary pairs in the current substitutional solution phase, press <Enter> (implying that there are no more changes of binary excess model for all other possibly-remaining binary pairs that shall still use the default REDLICH-KISTER model).

EXCESS MODEL TYPE: /LEGENDRE/: <DESIRED BINARY EXCESS MODEL TYPE></td></tr>
</table>

Syntax	AMEND_PHASE_DESCRIPTION
	For the currently-specified binary pair, choose one the legal binary excess models: LEGENDRE, POLYNOM or REDLICH-KISTER. If the default binary excess model REDLICH-KISTER is used for a certain binary pair, you can either specify the model name (or in short as R-K) or press <Enter>. After this sub-option, it returns to the first sub-option *First (the independent) constituent:* either for further change(s) of binary excess model of any other binary pair(s) or for termination (by pressing <Enter>) in the MIXED-EXCESS-MODELS option.

TERNARY_ EXTRAPOLAT	`EXTRAPOLATION METHOD: /TOOP-KOHLER/: <TERNARY MODEL>`
	Choose a ternary extrapolation model for the current solution phase:
	• TOOP-KOHLER for Toop-Kohler Extrapolation Model
	• KOHLER-ALL for Kohler Extrapolation Model
	• MUGGIANU_RESTOR for Muggianu-Restor Extrapolation Model
	The TOOP-KOHLER option further prompts:
	`CONSTITUENT IN SUBLATTICE #: <BASIS CONSTITUENT NAME>`
	`FIRST INTERACTION CONSTITUENT: <INTERACTING CONSTITUENT NAME>`
	`TOOP CONSTITUENT: <TOOP CONSTITUENT NAME>`
	Specify the so-called Kohler constituents (entered as the basis constituent and first interacting constituent) and Toop constituent on a given sublattice in the current solution phase.

QUASICHEM_ IONIC	To use the Quasichemical Model for the chosen liquid phase, you must first enter it as an ionic liquid phase, and then use this amending option. It creates a completely new phase, the original IONIC_LIQUID phase remains in the system and is not changed. `NUMBER OF BONDS/ATOM /2/: <NUMBER OF BONDS PER ATOM>` Specify the Z value (number of bonds per atom).
STATUS_BITS	Correct the phase status bits. It is important to fully understand this command before using it: `NEW STATUS WORD (IN HEXADECIMAL) /04204000/: <NEW STATUS BITS>` Also see *LIST_STATUS* on page 80 for information on the status bits. Normally, these should never be changed in this way, because such phase status bits are automatically set on when all the phase descriptions (predefined in databases

Syntax	AMEND_PHASE_DESCRIPTION
	and amended in the GIBBS module) are defined properly. The only exceptions are for the Kapoor-Frohberg-Gaye cell model for liquid SLAG solution phase, which can be set only as 04208000; and for the complete Revised HKF AQUEOUS solution model, which can be set as `02084000`.

AMEND_SYMBOL

In the GIBBS and PARROT modules, interactively calculate the current values of a function or table (predefined by the switched database or previously entered with *ENTER_SYMBOL* on page 72) at the current temperature and pressure condition, and also possible to modify the definitions of variables or functions.

The current values of the temperature and pressure used to calculate the functions or tables are the ones listed with *LIST_STATUS* on page 80. There is no way to change the current temperature and pressure values interactively.

To modify the definitions of entered symbols (variable, functions or parameters), the performance of this command is slightly different one from another, as described below:

- For variables, the values can be changed.
- For functions the low and high temperature limits in various ranges (if any), as well the expressions, can be changed.
- For parameters the low and high temperature limits in various ranges (if any), as well the expressions, can be changed.

Syntax	AMEND_SYMBOL

Prompts	`NAME: <SYMBOL NAME>`
	Specify the name of an entered symbol.
	For tables and functions (or parameters which are treated as functions after these are entered by the ENTER_SYMBOL command with the `Parameter` keyword, but not by the ENTER_PARAMETER command), after the symbol name is specified here, the program automatically calculates the values under the current temperature and pressure conditions, and lists the current values, such as:
	`FUNCTION VALUE 2.52500000E+01`
	`TABLE VALUE 1.56000000E+02`
	For variables and functions (or parameters), there are additional prompts and depend on the symbol type.

	For a variable, its current value is displayed, and you can change it to a new value:
	`VALUE /CURRENT VALUE/: <NEW VALUE>`
	For a function (or a parameter entered as a symbol):
	`DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: <Y OR N>`
	To change the number of ranges for the function, or change some of the temperature limits in the definition, For `Y`, you must retype both the low/high temperature limits and functions (see all the remaining details in the ENTER_SYMBOL command.
	If you do not want to change it, press <Enter> to accept the default. The definition of the chosen function in all ranges (if any) is listed, for example:
	`DIFFERENT FUNCTIONS IN THESE RANGES`
	`298.15<T<2000.00`
	`2000.00<T<4500.00`
	`4500.00<T<6000.00`
	and this message displays:
	`Do you want to change range limits /no/: <y or n>`
	If a function is different in two or more temperature ranges, you must specify the range of the function of which you want to amend. Or press <Enter> or type `0` to exit this command without making any change.

	`RANGE NUMBER (0 TO EXIT) /0/: <RANGE NUMBER>`
	`FUNCTION:`
	The previous function is available for editing. The editing is performed within the general

Syntax	AMEND_SYMBOL

subroutine FOOLED. This routine prompts as follows:

1:+:>

The prompt consists of the current position in the string and the character at that position between colons (::).

Commands

These commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurrences> F<string>@
- Insert: I<string>@
- Move: <+-#positions> M
- Restore string: R
- Substitute: S<OLD>@<NEW>@
- Type string: T

where CP denotes the current position in the string, # means number of, @ is a terminator of an input or search string.

When the string is typed the character at the current position is replaced by an underscore _. To finish the editing of the current function, type E at the prompt.

CHANGE_STATUS

For the GIBBS_ENERGY_SYSTEM Module, the status of an element or species or phase can be either ENTERED or SUSPENDED. The suspended status can be either implicit or explicit. The implicitly suspended status can be set e.g. for a species if any of the elements in its chemical formula is explicitly suspended. A species that is implicitly suspended becomes entered automatically if all its elements are set entered. After this command, a message shows which elements/species/phases are suspended or restored (entered). Consequently, the status bits for the specified elements or species or phases are changed, as listed with *LIST_STATUS* on page 80.

Syntax	CHANGE_STATUS
Prompts	`FOR ELEMENT, SPECIES OR PHASE /SPECIES/: <KEYWORD>` Specify the keyword (`element`, `species` or `phase`).
	`SUSPEND /Y/: <Y OR N>` The status is changed from ENTERED to SUSPENDED (`Y`), or vice versa (`N`). If an element is suspended all species with this element become implicitly SUSPENDED too. A phase may become implicitly suspended if all its constituents or all constituents in a sublattice are suspended.
	`LIST OF ELEMENTS/SPECIES/PHASES: <NAME OF ELEMENTS OR SPECIES OR PHASES>` Specify the names or indices of those elements or species or phases that shall become suspended or active. For names, these should be separated by a space and terminated with a semicolon (;) or an empty line. For indices it is possible to give a range by separating two numbers by a hyphen (-). The list should be terminated by a semicolon (;). Example: `5 1 7-12 FE;`

ENTER_ELEMENT

Specify a system interactively. The program searches the currently switched or preset database for data for the given elements. The data for the elements in the database are the:

- mass in g/mol
- name of the Selected Element Reference State (SER) which normally is the stable phase for the element at 298.15 K
- enthalpy difference for the element in the SER state at 298.15 K and zero K
- absolute entropy for the element in the SER state at 298.15 K.

The two predefined elements, i.e. electrons and vacancies, have the chemical symbols `/-` and `VA`, respectively. Initially, these are suspended but can be entered either by this command or *CHANGE_STATUS* on the previous page.

If an aqueous solution phase is involved, in the GIBBS module you enter a special aqueous electron called, ZE. This is specially designed for appropriately calculating the standard electric potential in the aqueous solution system.

Syntax	ENTER_ELEMENT
Prompt	`ELEMENT NAME: <ELEMENT NAME>` Specify several elements on one line. The name of an element is its chemical symbol. The chemical symbols must be separated by spaces. Fictitious element names are legal but naturally no data are found in the database for them.
	An element name (maximum 2 characters) can either have its first letter in upper and the second (if any) in lower case (i.e. Lower Case Mode) or both letters can be in upper or lower case (i.e. Upper Case Mode). The Upper or Lower Case Mode is selected by the command *REINITIATE* on page 83, which removes all data, and should be executed before any other command.

About the ENTER_PARAMETER Command

For the PARROT Module, this command is the same and is described here. Also see *ENTER_PARAMETER* on page 99 for the command details.

In the descriptions of the standard thermochemical properties and special physical properties for a phase, there are a number of parameters which may depend on the temperature and pressure. The expressions for these parameters can be given in a free form as a sum of terms with powers of T and P and may also include the natural logarithm and exponential function. This type of expression is called *TP-functions*. Identical parameters (in terms of parameter-names) are stored only once in the GIBBS workspaces.

The composition-dependence of the Gibbs energy is described in the GIBBS module by the internal data structure, which is created when the phase is entered (see *GIBBS_ENERGY_SYSTEM Commands* on page 42). The Gibbs energy of a phase is always referred to one formula unit of the phase, i.e. the amount derived from the number of sites (i.e. the stoichiometric coefficient) for each sublattice. If vacancy is a constituent of a sublattice, the amount of matter per formula unit of the phase may vary with composition.

Defining the Parameter

A valid parameter should have the general form of:

```
<identifier>(<phase>, <constituent array>; <digit>) <xxx> <expression> <yyy> <keyword
Y or N> <zzz> !
```

The identifier must be followed by an opening parenthesis, a phase name, a comma and a constituent array. Optionally, the constituent array can be followed by a semicolon and a digit. The parameter name is terminated by a closing parenthesis. The parameter form is defined as:

- `<identifier>` is the parameter type;
- `<phase>` is the phase name (maximum 24 characters);
- `<constituent array>` is the specific constituent array in the phase;
- `<digit>` is the degree of composition-dependent interaction contribution (an integer number from 0 through 9), that is only for excess energy (L), Curie temperature (TC) and Bohr magneton number (BMAGN), as well as for volume-related parameters (V0 or VA or VB or VC or VK); if it is valued as zero, or if it is for the standard Gibbs energy (G) for which the degree is always zero, it can be omitted;
- `<expression>` is the mathematical relation to describe the parameter;
- `<xxx>` and `<yyy>` are the low and high temperature limits respectively for the applicable temperature range of the parameter expression;
- `<keyword Y or N>` is the indicator on if there is continuation for the parameter expression or not;
- `<zzz>` is the reference index/number for the assessment of this parameter;
- The exclamation point ! is used to indicate that the current parameter definition is ended.

PARAMETER NAME

The GES parameter name has a general form of:

```
<identifier>(<phase>,<constituent array>;<digit>)
```

Examples of parameter names:

- `G(GAS,C1O2)`: The Gibbs energy of formation of a CO2 molecule in gas.
- `G(FCC,FE:VA)`: The Gibbs energy of formation of fcc Fe with interstitials.
- `L(LIQ,Fe,Cr;0`: The regular solution parameter for Fe and Cr in liquid.
- `L(LIQ,Fe,Cr;1)`: The sub-regular solution parameter.
- `TC(BCC,Fe:Va)`: The Curie temperature of bcc Fe.
- `BMAGN(BCC,Fe:Va)`: The Bohr magneton number parameter of bcc Fe.

The parameter name consists of several parts. The first is a type-identifier and these can be used:

- `G`: Standard energy parameter (Gibbs energy of formation) or for interaction parameters;
- `L`: Excess energy parameter (Gibbs energy of interaction) always used for interaction parameters;

- TC: Curie temperature for magnetic ordering;
- BMAGN or BM: Bohr magneton number for magnetic ordering (or Born function ωPr,Tr for aqueous solute species).
- VO: Molar volume at 298.15 K and 1 bar (a numeric value only);
- VA: Integrated thermal expansivity; $\int_{298.15}^{T} \alpha\,(T)dT \int_{298.15}^{T} \alpha\,(T)dT$
- VB: Bulk modulus at 1 bar;
- VC: Isothermal compressibility;
- VK: High-pressure fitting parameter.

When necessary quantities as H (enthalpy), S (entropy), V (Volume), F (Helmholtz energy), etc., can be calculated from the Gibbs energy.

PHASE NAME

Specifying the PHASE NAME in uppercase is recommended; however, if you prefer to write it as a mixture of uppercase and lowercase, it automatically converts all lowercase to uppercase, as the GIBBS module only recognises uppercase phase names. It is important that if a phase bears a legal phase-type (among G, A, Y, L, I, F and B) in its phase definition (already by the PHASE keyword; such as GAS:G, LIQUID:L,SLAG:L, IONIC-LIQ:Y, SPINEL:I, FCC_L12:F, HCP_D021:F, BCC_B2:B, AQUEOUS:A), such a valid phase-type code should not be attached to the phase name when ENTER_PARAMETER is executed.

CONSTITUENT ARRAY

The constituent array consists of a list of constituent names. Interaction parameters have two or more constituents from the same sublattice separated by a comma. If the phase has sublattices, at least one constituent in each sublattice must be specified. The constituents in different sublattices must be given in sublattice order and are separated by a colon.

After the component array, a sub-index digit can be specified after a semicolon. This digit must be in the range 0 to 9. The interpretation of the sub-index depends on the excess energy model used for the phase. If no semicolon and digit are given, the sub-index value is assumed to be as zero.

The excess energy parameters, e.g. the regular/subregular (binary) parameter or ternary parameters, are multiplied with two or more fractions of the constituents from the same sublattice of the solution phase. These additional constituents must be given as interacting constituents (as the following prompt).

Solution phases with sublattices can have interacting constituents in each sublattice.

INTERACTION PARAMETER

An interaction parameter, which is used to describe the excess term of a quantity, must have two or more constituents that interact with each other on a specified sublattice site of the given phase. It is arbitrary which of these constituents is given as the first constituent and what is given as the interacting constituents. The software always sorts the constituents (in each sublattice) in alphabetical order when the parameter name is written as a prompt (for entering its parameter value) and when the parameter is listed (using the GIBBS commands *LIST_PARAMETER* on page 102 or *LIST_PHASE_DATA (from GIBBS)* on page 103). This is important for all asymmetric interaction parameters where the sign of the interaction parameter must depend on the appearance order.

Use an asterisk* to denote that the excess interaction parameter is independent of the constituents of a specific sublattice. For example, `L(FCC_L12,AL,NI:*)` means that the interaction parameter is for the binary interaction between constituents AL and NI on the first sublattice in the `FCC_L12` solution phase, while it is independent of all constituents on the second sublattice. An interaction parameter in the list of constituents is always added to the Gibbs energy and the asterisk * is calculated with the term of [1∑y (specified constituents)], which implies that in an A-B binary system these `L` parameters are identical (but in higher-order systems, these are different):

- `L(phase,A,B)` is multiplied with X(A)*X(B)
- `L(phase,A,*)` is multiplied with X(A)*(1-X(A))
- `L(phase,B,*)` is multiplied with X(B)*(1-X(B))

If you press <Enter> when you are asked for a parameter name or if you have improperly input the entire parameter name, you are asked for each of these items in the name.

ENTER_PARAMETER

For the PARROT Module, this command is the same and is described here.

Also see *About the ENTER_PARAMETER Command* on page 95 for details about each of the options.

Use this command to enter TP-function(s) for a specific parameter for a phase interactively with this command. If there is already a parameter expression defined (for this parameter), that is deleted and replaced with newly entered one.

The entered TP-Function(s) for the parameter can be changed later with the command *AMEND_ PARAMETER* on page 44.

<table>
<tr><td>Syntax</td><td>ENTER_PARAMETER</td></tr>
<tr><td>Prompts</td><td><code>PARAMETER: <PARAMETER NAME></code>

As explained in About the ENTER_PARAMETER Command on page 95, specify a correct and complete <code>Parameter Name</code>, which should contain all the necessary parts of the general form:

<code><IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)</code>

If a parameter name is not acceptable or <Enter>, is pressed, a message displays:

<code>*** Error, please re-enter each part separately</code>

and you are prompted for input for each required part for a parameter name.</td></tr>
<tr><td>Options</td><td>Description or Information</td></tr>
<tr><td><code>IDENTIFIER</code></td><td><code>IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK></code>

If this command is used one or more times, the previous value on this prompt is set as default. Press <Enter> for the same type identifier or specify a new type.</td></tr>
<tr><td><code>PHASE NAME</code></td><td><code>PHASE NAME (/ABCD/): <PHASE NAME></code>

Each parameter is valid for a specific phase only. The name of that phase must be supplied (maximum 24 characters). The name can be abbreviated.</td></tr>
<tr><td><code>CONSTITUENT</code></td><td><code>CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME></code>

A parameter is identified by the constituents on a specified sublattice site of the given phase, the stoichiometric coefficients of which are multiplied with the parameter. The name of the constituent can be abbreviated.

It is the species name, not the stoichiometric formula required here.

If this command is used one or more times, the previous value on this prompt is set as default. Accept it by pressing <Enter> if the constituent is the same, or specify a new species name.</td></tr>
<tr><td><code>INTERACTING CONSTITUENT</code></td><td>For phases with several sublattices, the program asks for one constituent in each sublattice.

<code>INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME></code>

If this command is used one or more times, the previous value on this prompt is set as default. Press <Enter> to accept it if the constituent is the same, or specify a new species name.</td></tr>
</table>

Syntax	ENTER_PARAMETER
	To cancel the default value of the interacting constituent type `NONE` or the name of another constituent. This question is repeated until all the interested interacting constituent(s) on a specific sublattice in the phase are specified, and finally an <Enter> is enforced.
DEGREE	`DEGREE /#/: <DEGREE>` Degree is model-dependent. Specify an integer number (a value from 0 through 9) as the degree of composition-dependent interaction contribution for the phase parameter. This is valid for excess energy (`L`), Curie temperature (`TC`) and Bohr magneton number (`BMAGN`), as well as for volume-related parameters (`V0`, `VA`, `VB`, `VC` or `VK`). For binary interaction parameters, the degree is usually the power in the Redlich-Kister expression. For ternary interaction parameters, it is usually the Hillert ternary index. For a standard `G` parameter for a pure component (end-member) its degree should be always 0 and this prompt should not display.

PHASE PARAMETER	After this prompt, the program echoes on the screen the full TP-Function of the phase parameter.
	
	`LOW TEMPERATURE LIMIT /298.15/: <LOWEST TEMPERATURE LIMIT IN K>`
	
	Specify the lowest temperature limit (in Kelvin), or the lowest-pressure limit (in Pascal but entered as a negative number), for the current TP-Function.
	
	`FUNCTION: <DEFINITION FOR A FUNCTION>`
	
	A TP-Function consists of terms in T and P.
	
	`& <CONTINUATION OF THE DEFINITION FOR THE CURRENT FUNCTION>`
	
	Continuation of a TP-Function definition.
	
	`HIGH TEMPERATURE LIMIT /6000/: <HIGH TEMPERATURE LIMIT IN K>`
	
	Specify the high temperature limit (in Kelvin), or the high-pressure limit (in Pascal; only if the *lowest limit* is entered as a negative number) for the current TP-Function.
	
	`ANY MORE RANGES /N/: <Y OR N>`
	
	Enter `Y` for more function(s) or `N` to end this command.

About the ENTER_PHASE Command

With the command *ENTER_PHASE* on page 69 , the phase name, phase-type, sublattice number, and constituents or constituent array(s) for the phase are entered into the GIBBS workspaces.

About Phase Names

A thermochemical system must have at least one phase (which is a homogeneous part of the system with a uniform composition and structure). In the GIBBS module, any number of phases can be entered for a system and for each phase there must be a description of how its Gibbs energy varies with temperature, pressure and composition. A phase has a large amount of data connected to it, e.g. it starts with a phase name:

- It may be treated as a special phase-type;
- It may have structural information about sublattice(s), etc.,
- There must be a list of constituents (for a substitutional phase with no sublattice) or of constituent arrays (for a sublattice phase);
- There may be basic information on what kind of EXCESS_MODEL (polynomial expression) is used to describe the binary, ternary and/or higher-order interactions between constituents;
- There may be so-called Additional contributions to the Gibbs energy of the phase from special physical phenomena, e.g. magnetic ordering, hypothetical electrostatic interaction, and so forth;
- There must exist all the parameters required for the descriptions of thermochemical properties (i.e. `G` terms for standard Gibbs energies, and `L` terms for binary, ternary or higher-order interaction excess energies) and of some special physical properties (e.g. the Curie temperature `TC` and Bohr magneton number `BMAGN` (or `BM`) for magnetic ordering, `V0`-`VA`-`VB`-`VC`-`VK` parameters for volume contributions, Born functions for hypothetical electrostatic interaction in an aqueous solution phase) stored in connection with the phase.

About Phase Type Code

A G phase (gaseous mixture) or an A phase (aqueous solution) is usually treated as a substitutional phase without sublattice, and that an L phase (ordinary liquid solution) is normally (but not always) modelled as a substitutional phase without sublattice, too.

The F and B phase-types are useful for ordered FCC (or HCP) and BCC solution phases handled by the so-called Four Substitutional-Sublattice Ordering Model, which always requires that the solution phase must have four sublattices for substitutional ordering and can additionally have an interstitial sublattice.

For ordered FCC or HCP phases, these four substitutional sublattices represent four corners of the regular tetrahedron on these lattices all of which are the nearest neighbours. A *Normal 4-Sublattice*

Model requires that all the G parameters for each of end-members with the same elements but distributed on different sites be given separately. However, as these corners are identical lattice points, the phase-type option F means that the G parameters need be given only once. The possible permutations are handled automatically. To be more clarified: An A-B binary solution phase (with the element A locates on one sublattice site and B on three sublattice sites) treated by the Normal 4-Sublattice Model has to have 4 G parameters for 4 end-members, i.e. G(phase,A:B:B:B), G (phase,B:A:B:B), G(phase,B:B:A:B) and G(phase,B:B:B:A), because of that in the general case these G parameters can be different from each other. But for the FCC and HCP orderings, these are identical and thus all G parameters of such end-members need be given only once, and the possible permutations are then automatically handled by the GIBBS module. This significantly simplifies the usage of this model (*Four Substitutional-Sublattice Ordering Model*) in multicomponent alloys.

There are restrictions on the excess parameters allowed with the phase-type option `F`. You can only have excess parameters of these types:

```
L(phase,A,B:*:*:*;0…9)
L(phase,A,B:C,D:*:*;0…9)
```

The asterisk * means that the interaction parameter is independent on the constituents on its corresponding sublattice. No ternary interaction parameters (i.e. with three elements on one sublattice site) are allowed. The reason for this restriction is that it would be too complicated to handle all possible permutations. In the current *Four Substitutional Sublattice Ordering Model*, the binary interaction between A and B atoms is thus independent of the constituents on the other sublattices, where there are many other parameters to model the composition-dependence of the Gibbs energy (both in the ordered and disordered parts of the phase). The model for these ordered phases are always partitioned in a disordered part (with a single substitutional sublattice) and an ordered part (with four substitutional sublattices for ordering).

For ordered BCC phases, the phase-type option B means the same thing but the situation is more complicated, as the 4-sublattice ordering phase represents an irregular tetrahedron with two pairs of sites that are next nearest neighbours. Thus, for an A-B binary solution phase (with the element A locates on two sublattice site and B on other two sublattice sites) treated by the Normal 4-Sublattice Model, the end-member described by the G(phase,A:A:B:B) term has four nearest neighbour bonds between A and B atoms, whereas the end-member described by the G(phase,A:B:A:B) term has two nearest neighbour bonds between A and B atoms and two next nearest neighbour bonds

For detailed reference visit the Crystal Lattice Structure web page http://www.nrl.navy.mil/mstd/branches/6390.

The first end-member (described by the G(phase,A:A:B:B) term) represents B2- ordering and the second (described by the G(phase,A:B:A:B) term) stands for B32-ordering. There are two permutations of the G (phase,A:A:B:B) term and four permutations of the G(phase,A:B:A:B) term, automatically conducted in the Four Substitutional-Sublattice Ordering Model. And there are also two kinds of reciprocal interaction parameters, i.e.

```
L(phase,A,B:C,D:*:*;0…9) L(phase,A,B:*:C,D:*;0…9)
```

An advanced feature with the phase-type options F and B is that a composition set that represents the solution phase has a suffix (indicating that it is really as an ordered or disordered phase) that is automatically added to its phase name in some listings of equilibrium calculations (when performing either single-point or stepping or mapping calculations, and when plotting the calculated property diagrams or phase diagrams). Such suffix indications can be:

- `_L12` or `_L10` for ordered FCC
- `_A1` for disordered FCC
- `_B2`, `_B32`, `_D03` or `_L21` for ordered BCC
- `_A2` for disordered BCC
- `_D019` and `_B19` for ordered HCP
- `_A3` for disordered HCP

Solution Phase	Suffix for Disordered Phase	Suffix for Ordered Phase
FCC PHASE	PHASE_A1	PHASE_L12
		PHASE_L10
BCC PHASE	PHASE_A2	PHASE_B2
		PHASE_B32
		PHASE_D03
		PHASE_L21
HCP PHASE	PHASE_A3	PHASE_D019
		PHASE_B19

ENTER_PHASE

See *About the ENTER_PHASE Command* on page 67 for details about the phase-type and other prompts.

With this command, the phase name, phase-type, sublattice number, and constituents or constituent array(s) for the phase are entered into the GIBBS workspaces. However, for the remaining parts of thermodynamic descriptions (i.e. thermochemical and physical parameters, excess model, and possibly additional contribution term) for the phase, the GIBBS commands *ENTER_PARAMETER* on page 99 and *AMEND_PHASE_DESCRIPTION* on page 52, should be used sequentially.

Normally, data for a system are retrieved from the preset database after the elements and species are specified. This is automatically done through *GET_DATA* on page 15. In some cases, specify a phase interactively.

Syntax	ENTER_PHASE
Options	**Description and Information**

PHASE NAME	`NAME OF PHASE: <PHASE NAME>`
	The phase name is any string starting with a letter and containing only letters, digits or the underscore character. The phase name must be unique.
	`TYPE CODE: <PHASE-TYPE CODE>`
	The `Phase-type code` must be specified for the phase if it is not an ordinary solution phase. Press <Enter> if the phase is ordinary.

PHASE-TYPE	
G	Gaseous mixture phase. There may only exist one gas phase in a system.
A	Aqueous solution phase.
L	Liquid solution phase but not `A` (aqueous) or `Y` (ionic liquid)
Y	Ionic liquid solution phase (that is specially treated by the Ionic Two Sublattice Liquid Model).
I	A phase with charged species but not `G` (gaseous), `A` (aqueous) or `Y` (ionic liquid).
F	An ordered FCC or HCP solution phase with four substitutional sublattices (additionally, such a phase can also have an interstitial sublattice).
B	An ordered BCC solution phase with four substitutional sublattices (additionally, such a phase can also have an interstitial sublattice).

SUBLATTICE NUMBER	`NUMBER OF SUBLATTICES /1/: <SUBLATTICE NUMBER>`
	Phases with stoichiometric constraints usually have two or more sublattices. On each sublattice one or more species can be entered as constituents. The number of sublattices must not exceed 10. Substitutional phases with no sublattices are treated as phases with one sublattice.
	`NUMBER OF SITES ON SUBLATTICE # /1/: <SITE NUMBER>`
	For phases with sublattices, the ratio of the sites on each sublattice (donated with #) must be given. The program asks for values for all sublattices (by repeating this question for each sublattice), but these may have a common factor extracted. It is

Syntax	ENTER_PHASE
	recommended to use integer numbers as sites if possible. For substitutional phases with no sublattices, this question does not show up.

CONSTITUENTS	`NAME OF CONSTITUENT: <CONSTITUENT NAME(S)>`
	For each sublattice specify at least one species as constituent. Several constituents can be given consequently (on the repeated prompt) or on the same line separated by a space. This question is repeated till pressing <Enter> or giving a semi-colon (;), meaning that the list of constituents is terminated by (;) or an empty line.
	`WILL YOU ADD CONSTITUENTS LATER /NO/: <Y OR N>`
	For `Y` add other constituents to the phase by specifying new components in e.g. an ENTER_PARAMETER command. The default is `N` or press <Enter>.
	If illegal constituents are used, you get a message e.g. when parameters are entered.
	`DO YOU WANT A LIST OF POSSIBLE PARAMETERS /NO/: <Y OR N>`
	Enter `Y` for a list of all possible parameters up to the fifth interaction order for the entered phase.

ENTER_SPECIES

Specify a species from the already entered elements. The stoichiometry of the species is the chemical formula of the species. For each species, its name and chemical formula must be given.

> All elements are automatically entered simultaneously as species. The constituents of a phase must be species, but there is no thermochemical data associated with the species except as constituents of a phase.

Syntax	ENTER_SPECIES

Prompt	SPECIES NAME: <SPECIES NAME>
	
	Enter a unique species name. This is usually its chemical formula but it can be any string which starts with a letter and contains only letters or digits. Parenthesis () and an underscore _ are permitted. The combination /- or /+ used to denote charge.
	
	Species Names Examples
	
	`NACL2, FE2O3, FEO3/2, FE1O1.5, CA.5MG.5SI1O3,`
	
	`AL1H1O1_Al(OH) , AL1H1O1_HAlO, C2H4O3_124TRIOXOLANE, NA+, SIO4-4, H1/+1, H1O1/-1, AGC2N2-1.`
	
	The species name can be used to separate isomers with the same stoichiometry, e.g. `C2H2Cl2, (CHCl)2_cis, (CHCl)2_trans, CH2_CCl2.`

	STOICHIOMETRY /SPECIES NAME/: <CHEMICAL FORMULA> The elements in the chemical formula are normally separated by stoichiometric numbers. Neither parenthesis () nor an underscore _ is allowed in the chemical formula, while the special combination /- or /+ can be used. In upper case mode (see *REINITIATE* on page 83), the stoichiometric number unity can be excluded only if the element symbol consists of two letters. In lower case mode, the element must be typed with the first letter in UPPER case and the second in lower, and it is thus possible to distinguish CO (carbon monoxide) from Co (cobalt) without any stoichiometry number. All elements in a species must be entered prior to entering the species.

ENTER_SYMBOL

In the GIBBS module, a symbol can be used to represent a numeric quantity (e.g. a *variable*), a *function*, or a *table*. The entered symbol (with a symbol name of maximum eight characters) can be used later when parameters are entered, and this is a flexible way to manipulate thermodynamic functions. Symbols are especially useful if the same function or table is used in several thermochemical parameters. A *parameter* symbol is used to assign the values of phase parameters (which are defined with the command *ENTER_PARAMETER* on page 99) to specific characterized symbols. If you want to define a constant quantity, enter it as a simple function with a constant value in all (temperature) ranges.

The symbols entered in the GIBBS module can also be listed and used in the *PARROT Commands* on page 86 (for data optimization) where you can use the GIBBS-entered symbols (variables, functions or parameters) to define parameters that shall be optimized.

Symbols entered in the GIBBS module are not the same symbols as defined in the POLY and POST modules.

Also see *ENTER_SYMBOL* on page 195 for the POLY and POST modules.

See example 44 in the *Console Mode Examples Guide*for an example of using variables and functions.

Syntax	ENTER_SYMBOL

Prompt	`VARIABLE, FUNCTION, TABLE OR PARAMETER? /FUNCTION/: <KEYWORD>`
	
	The `keyword` can be a variable, function, table or parameter.
	
	• `Variables` are similar to functions because these can also be expressions of state variables. It is possible to enter a variable with a new expression anytime.
	• `Functions` are expressions of state variables or other functions.
	• `Tables` are used for listing results. A table consists of a list of any number of state variables, functions or variables.
	• `Parameter` symbols are used to assign the values of phase parameters (which are defined by the command ENTER_PARAMETER) to specific characterized symbols. If the phase parameter is not defined yet, the parameter symbol is assigned zero or a symbol which is valued as zero. In this way it is easy to refer to entered phase parameters in further defining other phase parameters.
	
	`NAME: <NAME OF THE SYMBOL>`
	
	Each symbol has a unique name that must start with a letter and can have maximum 8 characters.
	
	The following prompts are based on the type of symbol.

`Variable` symbol	`VALUE: <NUMERIC VALUE OF A VARIABLE>` Only a constant numeric value, not an expression, is accepted.
`Function` symbol	Enter the lower-temperature limit, the expression, an upper-temperature limit and if there is another expression above this upper limit: `LOW TEMPERATURE LIMIT /298.15/: <LOWEST TEMPERATURE LIMIT IN K>`

Syntax	ENTER_SYMBOL

	Specify the lowest-temperature limit, below which its (first) expression is not applicable. The default lowest limit of all kinds of data is 298.15 K.
	
	If a negative number is given as the lowest-temperature limit, it is assumed that there are breakpoints in pressure for this function; in such a case, it is interpreted as the lowest-pressure limit (in Pascal), and the other limits in the current function are also taken as pressure limit values (in Pascal).
	
	The temperature/pressure limits for the functions are checked during calculations. An indicator is set if the actual temperature/pressure condition is below the lowest temperature/pressure limit or above the highest temperature/pressure limit; and in such a case, an extrapolation is conducted using the TP-FUNCTIONS valid in the nearest temperature/pressure range.
	
	`FUNCTION: <DEFINITION FOR A TP-FUNCTION>`
	
	A TP-Function consists of terms in T and P. The expression is a FORTRAN-like expression and operators +, -, *, = and ** can be used (** only with integer powers). Unary-functions LN or LOG (both for natural logarithm) and EXP (for exponential) can also be used. An expression can be continued on more than one line. A PT-function must be terminated by a semicolon (;) or an empty line. No more than 78 characters can be written on a line. If this is not sufficient space on one line, press <Enter> and continue on the next line. The program prompts with an ampersand `&` when a continuation is expected.
	
	A function can have several temperature ranges, and a different expression (i.e. a TP-Function) for each region; if required, a single function can have up to ten different temperature ranges (and thus have up to ten different TP-Functions).
	
	This question is repeated if the current function has more than one expression (TPFunction) in various temperature ranges [or pressure ranges; only if the *lowest limit* is entered as a negative number)].
	
	`& <CONTINUATION OF THE DEFINITION FOR THE CURRENT TP-FUNCTION>`
	
	This is for a continuation of the definition for the current TP-Function. The ampersand & means that you can continue to write the function on the new line if one line is not enough for the current TP-Function. If you are finished the current TP-Function, press <Enter> again.
	
	`HIGH TEMPERATURE LIMIT /6000/: <HIGH TEMPERATURE LIMIT IN K>`
	
	For a TP-Function, specify the high-temperature limit, above which its current expression is not applicable. The default high-temperature limit for all kinds of data is

Syntax	ENTER_SYMBOL
	always 6000 K. This question is repeated if the function has more than one expression (TPFunction) in various temperature ranges. All the temperature limits for the presently-be-entered TP-Function MUST be in an increasing order; otherwise the whole function is given a single value of zero. If a negative number is given as the lowest-temperature limit, it is interpreted as the low-pressure limit (in Pascal), and the high-temperature limit is also taken as the high-pressure limit (in Pascal) for the current TP-Function. The high-temperature/pressure limit for the current applicable expression (TPFunction) is the low temperature/pressure limit for the next range if several ranges are needed to describe the temperature/pressure-dependence of this function. The temperature/pressure limits for the functions are checked during calculations. An indicator is set if the actual temperature/pressure condition is below the lowest temperature/pressure limit or above the highest temperature/pressure limit; and in such a case, an extrapolation is conducted using the TP-Function valid in the nearest temperature/pressure range. `ANY MORE RANGES /N/: <Y OR N>` Answer `N` to finish the definition of the current function. If `Y` the program asks for a new function valid above the last high temperature limit and below a new high temperature limit. The maximum number of temperature ranges is 10. Important note: Make sure that the function and its first derivatives are continuous across break points.

`Table` symbol	Specify the low and high temperature limits, temperature step and tabled value at each specified temperature:
	
	`LOW TEMPERATURE LIMIT /298.15/: <LOW TEMPERATURE LIMIT IN K>`
	
	The lowest temperature where the table has a value.
	
	`HIGH TEMPERATURE LIMIT /6000/: <HIGH TEMPERATURE LIMIT IN K>`
	
	The highest temperature where the table has a value.
	
	`STEP IN TEMPERATURE /100/: <TEMPERATURE STEP>`
	
	The step in temperature between each value in the table. This step must be constant for the whole table. If a table cannot be described with a single step it must be divided

Syntax	ENTER_SYMBOL
	into several tables for different temperature ranges. The upper and lower limits are truncated to give reasonable values where the table values must be specified. `TABLE VALUE AT XX /YY/: <TABLE VALUE>` The value of the table at the specified temperature (`xx`) must be given. The default value (`yy`) is the last value.

`Parameter` symbol	Input the correct name of a phase parameter (including its identifier, phase name, constituent and interacting constituent(s) in various sublattices (if exist), and degree of the parameter. Most of the settings are the same as in the *ENTER_PARAMETER* on page 99 command or the Function symbol prompts above.
	`PARAMETER: <PARAMETER NAME>`
	Specify a correct and complete parameter name, which should contain all the necessary parts of the general form:
	`<IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)`
	If a parameter name is not acceptable or <Enter> is pressed, this error message displays:
	`*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY`
	The program prompts for separate input for each required part for a parameter name.
	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>`
	Specify one of these types of legal identifiers.
	`PHASE NAME (/ABCD/): <PHASE NAME>`
	Specify the phase name.
	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>`
	Specify the constituent name on the specified sublattice site of the given phase.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>`
	Specify the interacting constituent name on the specified sublattice site of the given phase; if there is no interacting constituent, press <Enter>.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>`
	If there is more than one interacting constituent on the specified sublattice site of the given phase, specify them; otherwise press <Enter>.
	`DEGREE /#/: <DEGREE>`
	Specify an integer number (a value from 0 through 9) as the degree for the phase

Syntax	ENTER_SYMBOL
	parameter.
	`LOW TEMPERATURE LIMIT /298.15/: <LOWEST TEMPERATURE LIMIT IN K>`
	Specify the lowest-temperature limit (in Kelvin), or the lowest-pressure limit (in Pascal but entered as a negative number), for the current TP-Function.
	`FUNCTION: <DEFINITION FOR A FUNCTION>`
	Specify the desired TP-Function for the current temperature range (or current pressure range; only if the *lowest limit* is entered as a negative number).
	`& <CONTINUATION OF THE DEFINITION FOR THE CURRENT TP-FUNCTION>`
	Specify the desired TP-Function continuation for the current temperature range (or current pressure range; only if the lowest limit is entered as a negative number).
	`HIGH TEMPERATURE LIMIT /6000/: <HIGH TEMPERATURE LIMIT IN K>`
	Specify the high-temperature limit (in Kelvin), or the high-pressure limit (in Pascal; only if the lowest limit is entered as a negative number) for the current TP Function.
	`ANY MORE RANGES /N/: <Y OR N>`

LIST_CONSTITUENT

Syntax	LIST_CONSTITUENT
	List all the constitutions (as site-fractions) for all the phases in the defined system.
	This is mainly for software managers. There are no commands in the GIBBS monitor to set the constitution.

LIST_DATA

All data for the current system that is defined (retrieved from database in the DATA module, or directly entered in the GIBBS module) within the current GIBBS workspace is written in a readable manner on the output file. If no output file is specified, the data display on screen.

The output data consists of a list of all elements and their data followed by a list of all phases and the data associated with each phase. The thermochemical parameters listed for each phase are always in SI units.

For encrypted commercial databases you may not be able to use this command to list any retrieved data. However, it can be used to list some basic definitions (such as phase models and phase-constituents) retrieved from an encrypted database, as well as all the references related to original assessments (but not thermodynamic parameters) for the currently-defined system.

Syntax	LIST_DATA
Prompts	`OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME>` Specify the name of a file in which the data shall be written in a readable manner. The default value is the screen (by pressing <Enter>).
	`OPTIONS?: <OPTION(S)>` Choose one or several of these options for output: • `N` the output is written as a user database format. • `P` the output is written as a macro file for future input. This is useful for creating `*SETUP` files for data assessments. • `S` the symbols are suppressed. • `R` the references for the parameters are listed (only for some databases in which references are available) • `L` the output is written suitable for a LaTeX preprocessor.

LIST_PARAMETER

This command is for both the GIBBS and PARROT modules.

List the TP-function(s) of a specific parameter for a phase. You must supply the name of the phase parameter:

`THE PARAMETER NAME: <IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)`

For encrypted commercial databases, you may not be able to use this command to list any retrieved parameter.

Syntax	LIST_PARAMETER

Prompts	`PARAMETER: <PARAMETER NAME>`
	
	Specify a correct and complete parameter name, which should contain all the necessary parts of the general form: <identifier>(<phase>,<constituent array>;<digit>)
	
	See *About the ENTER_PARAMETER Command* on page 95 to learn how to define the Parameter Name.
	
	If a parameter name is not acceptable or <Enter> is pressed, an error message displays:
	
	`*** Error, please re-enter each part separately`
	
	The program prompts for separate input for each required part for a parameter name.

	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>` Specify one of these types of legal identifiers.
	`PHASE NAME (/ABCD/): <PHASE NAME>` Specify the phase name.
	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>` Specify the constituent name on the specified sublattice site of the given phase.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` Specify the interacting constituent name on the specified sublattice site of the given phase; if there is no interacting constituent, press <Enter>.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` If there is more than one interacting constituent on the specified sublattice site of the given phase, specify them; otherwise press <Enter>.
	`DEGREE /#/: <DEGREE>` Specify a numerical number as the degree for the phase parameter.

LIST_PHASE_DATA

All data for a specific phase are written in a readable manner on screen. The thermochemical parameters listed for the phase are always in SI units.

For encrypted commercial databases you may not be able to use this command to list any phase data. However, this command can be used for listing some basic definitions (such as phase model and phase-constituents) for the specified phase, as well as all the references related to original assessments (but not thermodynamic parameters) for the currently-defined system.

Syntax	LIST_PHASE_DATA
Prompt	`PHASE NAME: <PHASE NAME>` Specify a phase name (if abbreviated, it should be unique).

LIST_STATUS

Syntax	LIST_STATUS
	For the GIBBS module, list the entered elements, phases and species with the status word. The command is included for system managers. Each element, species, phase and symbol has a set of status bits. The values of these bits are listed with this command. The bits are listed as hexadecimal, i.e. four bits are written as a hexadecimal number. Two hexadecimal numbers make a byte. In hexadecimal, 0 to 9 mean normal digits. A to F means values 10 to 15. The number `E4000000` has thus bit 1, 2, 3 and 6 equal to one. The bits are numbered starting with 1 for the most significant (leftmost) bit.

Element Status Word

Set the element status word (*Bit. Meaning*) if:

1. Element cannot be deleted (only vacancy and electron)
2. Suspended (inclusive OR of bit 3 and 4)
3. Explicitly suspended
4. Implicitly suspended (cannot occur for an element)

Species Status Word

Set species status word (*Bit. Meaning*) if:

1. Species record for an element (each element has a species record)
2. Suspended (inclusive OR of bit 3 and 4)

3. Explicitly suspended
4. Implicitly suspended (e.g. if a species element is suspended)
5. Charged (inclusive OR of bit 6 and 7)
6. Negative charge
7. Positive charge
8. Vacancy
9. Component (by default the elements are the components)

Phase Status Word

Set phase status word (*Bit. Meaning*) if:

1. Ideal (no sublattices and no excess parameters)
2. Suspended (inclusive OR of bit 3 and 4)
3. Explicitly suspended
4. Implicitly suspended (e.g. if all constituents suspended)
5. Gas phase
6. Liquid phase
7. Solution phase
8. Only one constituent (in each sublattice)
9. Ions (inclusive OR of bit 10 and 12)
10. New ionic liquid model (charge balance handled internally)
11. Constituents cannot be added
12. External charge balance needed
13. Aqueous model
14. Charged species (ions)
15. Dilute entropy
16. Last calculation exceeded temperature range for any parameter
17. Kapoor-Frohberg-Gaye cell model
18. Turbo calculation used
19. Turbo calculation impossible
20. Turbo calculation illegal

21. Phase is not ideal
22. Current site fractions saved
23. This phase is the ordered part
24. This phase is the disordered part (bit 2 and 3 also set)
25. Shadow phase with diffusion data
26. Error in ionic model
27. Sometimes
28. CVM-SRO (short-range ordering) entropy expression
29. CVM initialization is made
30. Used to test quasi-chemical ordering model
31. Major constituents check even if no miscibility gap (ordering)
32. Hoch-Arpshofen model

Symbol Status Word

Set symbol status word (*Bit meaning*) if:

1. Constant
2. Variable
3. Function
4. Table
5. Value must not be amended
6. Undefined

LIST_SYMBOL

List the symbols available in the current GIBBS workspace, which are defined (retrieved from database in the DATA module, or directly entered in the GIBBS module) and used in TP-functions for the entered model parameters for various phases in the defined system.

For encrypted commercial databases you may not be able to use this command to list any retrieved symbol.

Syntax	LIST_SYMBOL
Prompts	`NAME: <SYMBOL NAME>` Specify a symbol name (either as the full name of a defined symbol, or as the first few common characters of several symbols which have names starting with such common characters). Only those symbols that match this name are listed. Or press <Enter> for a list of all the symbols defined (retrieved from database in the DATA module, or directly entered in the GIBBS module) for the current system, which are available in the current GIBBS workspace.
	`OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME>` Specify the name of a simple-textual file in which the entered symbols shall be written in a readable manner (such a basic textual file can later on be opened and edited by any text editor). The default value is the screen (by pressing <Enter>).

READ_GES_WORKSPACE

The data area saved onto a *.GES5 file with *SAVE_GES_WORKSPACE* on the next page is read back into the GIBBS workspace.

Syntax	READ_GES_WORKSPACE
	Enter a **File name** and specify the working directory where to save the file in the **Look in** field. Normally the default file-type in the **Files of type** field is the correct one for the GIBBS workspace format (i.e. *.GES5 file). The saved *.GES5 files are unique for each CPU type, and therefore a *.GES5 file saved on one CPU type cannot be read and used on another CPU type.

REINITIATE

All data in the stored GIBBS workspace are erased, and all variables are initiated to their default values.

Syntax	REINITIATE
Prompts	`UPPER CASE ONLY /Y/: <Y OR N>` The name of elements and species can be either in only UPPER case (by typing `Y` or pressing <Enter>), or the elements with two letter names have the first letter in Upper and the second in lower case (by typing `N`). In upper case mode all input in lower case is automatically converted to upper case.
	`LOWER TEMPERATURE LIMIT /298.15/: <LOWEST T IN K>` This value is used as the lower temperature limit when data are entered interactively.
	`UPPER TEMPERATURE LIMIT /6000/: <HIGHEST T IN K>` This value is used as the upper temperature limit when data are entered interactively.
	`DEFAULT ELEMENT REFERENCE STATE SYMBOL INDEX /1/: <1 OR 2 OR 3>` Specify a proper index for the default listing parameters (symbol), or press <Enter> to accept the preset index 1. The *index* is for defining the symbol printed in listings of parameters. The symbol can be: • `G`: the data are referred to Gibbs energy at a variable temperature (also called *Lattice Stability*). • `H298`: the data are referred to the enthalpy of the element at 298.15 K and 1 bar. • `H0` is the same as H298 but at the temperature 0 K. The index only changes the symbol, not any value. Normally the index is set correctly by the database. Only when manually entering data must you set the index to get the correct symbol. Set the index value to 0 to get the symbol `G`, to 1 to get the symbol `H298` and to 2 to get `H0`.

SAVE_GES_WORKSPACE

The data area used by the GIBBS workspace can be saved on a file (with a default extension GES5 under MS-Windows, or ges5 under Linux). The default file name is RESULT.GES5 or equal to the one used in a previous *READ_GES_WORKSPACE* on the previous page command.

Syntax	SAVE_GES_WORKSPACE
	Enter a **File name** and specify the working directory where to save the file in the **Save in** field. Normally the default file-type in the **Files of type** field is the correct one for the GIBBS workspace format (i.e. *.GES5 file). The output is unformatted and cannot be printed.

SET_R_AND_P_NORM

The value of the gas constant (R) is used to define the energy units that are used for input of data. On output, all data are in SI units. The value of the pressure 1 atm is used to interpret values of the pressure given by application programs.

Syntax	SET_R_AND_P_NORM
Prompts	`VALUE OF GAS CONSTANT IN YOUR ENERGY UNITS /8.31451/: <NEW VALUE>` The default value of the gas constant is in SI units (i.e. 8.31451). If you change this value, it only affects the interpretation of energy data given interactively from the terminal, e.g. if the value is set to 1.98717 you can give values in Calories.
	`VALUE OF ONE BAR IN YOUR PRESURE UNITS /100000/: <NEW VALUE>` The value of the pressure given in application programs is divided by this value and then multiplied with the value of one bar expressed in Pascal before it is used inside the GIBBS module.

PARROT Commands

In this section:

AMEND_PARAMETER

Interactively modify the TP-function(s) for a specific parameter for a phase with this command. This is useful in order to correct typing errors because the old function is made available for interactive editing on the terminal.

This AMEND_PARAMETER command is for the PARROT module. There is also a GES module command with the same name (*AMEND_PARAMETER* on page 44).

Syntax	AMEND_PARAMETER

Prompts	`PARAMETER: <PARAMETER NAME>`
	Specify a correct parameter name. If a parameter name is not acceptable or <Enter> is pressed, the error message displays:
	`*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY`
	The program prompts for separate input for each part for a parameter name.

	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>` Specify one of these types of legal identifiers (see *ENTER_PARAMETER* on page 99).
	`PHASE NAME (/ABCD/): <PHASE NAME>` Specify the phase name.
	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>` Specify the constituent name.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` Specify the interacting constituent name; if there is no interacting constituent, press <Enter>.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` If there is more than one interacting constituent, specify these; otherwise press <Enter>.
	`DEGREE /#/: <DEGREE>` Specify a numerical number as the degree of the phase parameter.

	After the parameter name is specified correctly, the program lists its current definition (either preset in database or defined by the ENTER_PARAMETER command), such as:
	`L(PHASE2,AL,MG;1)=`
	`298.15<T<2000.00: +5000`
	`2000.00<T<4500.00: +4500`

Syntax	AMEND_PARAMETER

```
4500.00<T<6000.00: +4000
```

You are prompted to change the parameter definition:

```
DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: <Y OR N>
```

To change the number of ranges for the chosen function, or change some of the temperature limits in the definition, by typing `Y` you have to retype both the low/high temperature limits and functions

If you do not want to change the number of ranges but want to change the function(s) in one or more ranges, press <Enter> to accept the default `N` the whole definition of the chosen parameter in all ranges (if any) is listed such as:

```
DIFFERENT FUNCTIONS IN THESE RANGES

298.15<T<2000.00

2000.00<T<4500.00

4500.00<T<6000.00
```

This message prompts:

```
DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y OR N>
```

If there is more than one range, this question is prompted. Press <Enter>.

```
RANGE NUMBER (0 TO EXIT) /0/: <RANGE NUMBER>
```

If the function of a parameter is different in two or more temperature ranges, you must specify the range of the function of which you want to amend. Or press <Enter> or type `0` to exit this command without making any change.

```
FUNCTION
```

The previous function is available for editing. The editing is performed within the general subroutine `FOOLED`, as described in *AMEND_SYMBOL* on the next page. This routine prompts as follows:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons, (::).

Commands

These commands can be given:

- Help: `?`
- Move CP to last or first character: `<+/-> A`

Syntax	AMEND_PARAMETER
	• Delete characters from CP: `<+-#characters> D` • Exit: `E` • Find: `<#occurrences> F<string>@` • Insert: `I<string>@` • Move: `<+-#positions> M` • Restore string: `R` • Substitute: `S<OLD>@<NEW>@` • Type string: `T` where CP denotes the current position in string, # means number of, @ is a terminator of an input or search string. When the string is typed the character at the current position is replaced by an underscore `_`. To finish the editing of the current function, type `E`.
	`RANGE NUMBER (0 TO EXIT) /0/: <RANGE NUMBER>` Give a range number to edit that function, or press <Enter> or type `0` to exit this command.

AMEND_SYMBOL

In the GIBBS and PARROT modules, interactively calculate the current values of a function or table (predefined by the switched database or previously entered with *ENTER_SYMBOL* on page 72) at the current temperature and pressure condition, and also possible to modify the definitions of variables or functions.

The current values of the temperature and pressure used to calculate the functions or tables are the ones listed with *LIST_STATUS* on page 80. There is no way to change the current temperature and pressure values interactively.

To modify the definitions of entered symbols (variable, functions or parameters), the performance of this command is slightly different one from another, as described below:

- For variables, the values can be changed.
- For functions the low and high temperature limits in various ranges (if any), as well the expressions, can be changed.
- For parameters the low and high temperature limits in various ranges (if any), as well the expressions, can be changed.

<table>
<tr><td>Syntax</td><td>AMEND_SYMBOL</td></tr>
<tr><td>Prompts</td><td> <code>NAME: <SYMBOL NAME></code> Specify the name of an entered symbol. For tables and functions (or parameters which are treated as functions after these are entered by the ENTER_SYMBOL command with the <code>Parameter</code> keyword, but not by the ENTER_PARAMETER command), after the symbol name is specified here, the program automatically calculates the values under the current temperature and pressure conditions, and lists the current values, such as: <code>FUNCTION VALUE 2.52500000E+01</code> <code>TABLE VALUE 1.56000000E+02</code> For variables and functions (or parameters), there are additional prompts and depend on the symbol type. </td></tr>
<tr><td></td><td> For a variable, its current value is displayed, and you can change it to a new value: <code>VALUE /CURRENT VALUE/: <NEW VALUE></code> For a function (or a parameter entered as a symbol): <code>DO YOU WANT TO CHANGE THE NUMBER OF RANGES /NO/: <Y OR N></code> To change the number of ranges for the function, or change some of the temperature limits in the definition, For <code>Y</code>, you must retype both the low/high temperature limits and functions (see all the remaining details in the ENTER_SYMBOL command. If you do not want to change it, press <Enter> to accept the default. The definition of the chosen function in all ranges (if any) is listed, for example: <code>DIFFERENT FUNCTIONS IN THESE RANGES</code> <code>298.15<T<2000.00</code> <code>2000.00<T<4500.00</code> <code>4500.00<T<6000.00</code> and this message displays: <code>Do you want to change range limits /no/: <y or n></code> If a function is different in two or more temperature ranges, you must specify the range of </td></tr>
</table>

Syntax	AMEND_SYMBOL
	the function of which you want to amend. Or press <Enter> or type 0 to exit this command without making any change.

```
RANGE NUMBER (0 TO EXIT) /0/: <RANGE NUMBER>
FUNCTION:
```

The previous function is available for editing. The editing is performed within the general subroutine FOOLED. This routine prompts as follows:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons (::).

Commands

These commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurrences> F<string>@
- Insert: I<string>@
- Move: <+-#positions> M
- Restore string: R
- Substitute: S<OLD>@<NEW>@
- Type string: T

where CP denotes the current position in the string, # means number of, @ is a terminator of an input or search string.

When the string is typed the character at the current position is replaced by an underscore _. To finish the editing of the current function, type E at the prompt.

COMPILE_EXPERIMENTS

The descriptions of the experimental equilibria saved on a textual *.POP file, given as commands in the POLY and ED_EXP syntax, is compiled by this command into the PARROT structured data, i.e. stored into the current work *.PAR file which is created with *CREATE_NEW_STORE_FILE* on the next page, set by *SET_STORE_FILE* on page 116 , or opened with *READ_PARROT_WORKSPACES* on page 109, all done before COMPILE_EXPERIMENTS.

This command also lists the compilation details on screen or onto a listing file (*.TXT).

If a syntax error in the commands is detected, the compilation is terminated. You need to correct the error in the *.POP file (using a textual editor) and recompile it with the *.PAR work (store) file.

After a successful compilation, the program translates all the experimental data points in the *.POP file into a graphical experimental data file (*.EXP) in the DATAPLOT syntax, which can be further edited and used as a graphical experimental data file (*.EXP) for the purposes of imposing onto calculated/plotted phase diagrams and/or property diagrams for comparisons during the assessment process.

Also see the *DATPLOT User Guide* included with this documentation set.

Syntax	COMPILE_EXPERIMENTS
	A window displays to open the original experimental data file (*.POP). Enter a *.POP **File name** box, and specify the working directory in the **Look in** box. Normally the default file-type in the **Files of type** field is the correct one for the experimental data format (i.e. *.POP file). Click **Open** or **Cancel** button as applicable.
Prompts	`OUTPUT TO SCREEN OR FILE /SCREEN/: <*.POP FILE NAME>` A list of the source code in the *.POP file and error messages during compilation is written on screen (by pressing <Enter>) or on a list *.TXT file under a specific file name which can later on be opened and edited by any basic textual editor.
	`INITIATE STORE FILE /Y/: <Y OR N>` After a successful compilation, the data in the POLY3 workspace is stored as a new block of equilibria on the current work (store) file. If you specify that the store file should be initiated before compilation (`Y`), only the compiled block is stored on the current work file (consequently, the *.PAR file is updated). If a negative answer (`N`) is chosen here, the current work file is not changed. A window displays to generate/store the graphical experimental data file (*.EXP) in the DATAPLOT syntax. Enter a *.EXP file name in the **File name** field and specify the working directory when opening the *.EXP file in the **Look in** field. Normally the default file type in the **Files of type** field is the correct one for the graphical experimental data format (i.e. *.EXP file). Click **Open** or **Cancel** as applicable.

CONTINUE_OPTIMIZATION

This command continues the optimization using the same Hessian matrix.

In some cases it is not possible to attempt continuation. For example, if the optimization has already converged or if the optimizing parameters or the set of experimental equilibria used in the current optimization runs are changed. The program gives a warning message if you try to continue when you make these changes.

Syntax	CONTINUE_OPTIMIZATION

Prompts	`ARE YOU SURE? /N/: <N OR Y>`
	Prior to this question, a message displays to confirm if it is OK to continue:
	`It is safe to CONTINUE only after TOO MANY ITERATIONS and no change in variables and experiments ...`
	You can accept the default (`N`) in order to cancel this special attempt.

	`NUMBER OF ITERATIONS /N/: <INTEGRAL NUMBER OF ITERATION>` Specify an integral number for the optimization iteration. The specified iteration number is shown as the default number. The PARROT program tries this exact number of different sets of values of the optimizing variables without stopping the optimization even it would have converged earlier or should give up earlier.

CREATE_NEW_STORE_FILE

This file is hardware dependent and cannot be read by any text editor.

Create a binary file to be used as a *work file* (also called *store file*) before any optimization can be done. The workspace used by the GIBBS, POLY and PARROT modules is stored automatically on the work file but not any experimental information. The binary work file has a default extension of `PAR` under MS-Windows or `par` under Linux/UNIX.

A work file that is created at a previous run can be used in the PARROT module with the other command SET-STORE-FILE.

Syntax	CREATE_NEW_STORE_FILE
Prompt	<*.par file name> A **Save** window displays. Enter a **File name** and specify the working directory in the **Save in** field. Normally the default file-type in the **Files of type** field is the correct one for the PARROT workspace format (i.e. PAR file). Click **Save** or **Cancel** as applicable.

EDIT_EXPERIMENTS

Initialise and make available the sub-module for editing experimental equilibria, i.e. the ED_EXP module. All experimental equilibria compiled from the *.POP file can be accessed in the ED_EXP module.

The ED_EXP module is similar to the normal POLY module, but some commands are special and some POLY commands are not available. In the ED_EXP module, you can calculate each equilibrium separately or together, provide start values for equilibria which failed to converge, set weights, and modify the values of experiments or conditions.

Also see *Running the EDIT_EXPERIMENTS Command* on page 23.

Syntax	EDIT_EXPERIMENTS
	You are now in this submodule and can start using the commands.

About the ENTER_PARAMETER Command

For the PARROT Module, this command is the same and is described here. Also see *ENTER_PARAMETER* on page 99 for the command details.

In the descriptions of the standard thermochemical properties and special physical properties for a phase, there are a number of parameters which may depend on the temperature and pressure. The expressions for these parameters can be given in a free form as a sum of terms with powers of T and P and may also include the natural logarithm and exponential function. This type of expression is called *TP-functions*. Identical parameters (in terms of parameter-names) are stored only once in the GIBBS workspaces.

The composition-dependence of the Gibbs energy is described in the GIBBS module by the internal data structure, which is created when the phase is entered (see *GIBBS_ENERGY_SYSTEM Commands* on page 42). The Gibbs energy of a phase is always referred to one formula unit of the phase, i.e. the amount derived from the number of sites (i.e. the stoichiometric coefficient) for each sublattice. If vacancy is a constituent of a sublattice, the amount of matter per formula unit of the phase may vary with composition.

Defining the Parameter

A valid parameter should have the general form of:

```
<identifier>(<phase>, <constituent array>; <digit>) <xxx> <expression> <yyy> <keyword
Y or N> <zzz> !
```

The identifier must be followed by an opening parenthesis, a phase name, a comma and a constituent array. Optionally, the constituent array can be followed by a semicolon and a digit. The parameter name is terminated by a closing parenthesis. The parameter form is defined as:

- `<identifier>` is the parameter type;
- `<phase>` is the phase name (maximum 24 characters);
- `<constituent array>` is the specific constituent array in the phase;
- `<digit>` is the degree of composition-dependent interaction contribution (an integer number from 0 through 9), that is only for excess energy (L), Curie temperature (TC) and Bohr magneton number (BMAGN), as well as for volume-related parameters (V0 or VA or VB or VC or VK); if it is valued as zero, or if it is for the standard Gibbs energy (G) for which the degree is always zero, it can be omitted;
- `<expression>` is the mathematical relation to describe the parameter;
- `<xxx>` and `<yyy>` are the low and high temperature limits respectively for the applicable temperature range of the parameter expression;
- `<keyword Y or N>` is the indicator on if there is continuation for the parameter expression or not;
- `<zzz>` is the reference index/number for the assessment of this parameter;
- The exclamation point ! is used to indicate that the current parameter definition is ended.

PARAMETER NAME

The GES parameter name has a general form of:

```
<identifier>(<phase>,<constituent array>;<digit>)
```

Examples of parameter names:

- `G(GAS,C1O2)`: The Gibbs energy of formation of a CO2 molecule in gas.
- `G(FCC,FE:VA)`: The Gibbs energy of formation of fcc Fe with interstitials.
- `L(LIQ,Fe,Cr;0`: The regular solution parameter for Fe and Cr in liquid.
- `L(LIQ,Fe,Cr;1)`: The sub-regular solution parameter.
- `TC(BCC,Fe:Va)`: The Curie temperature of bcc Fe.
- `BMAGN(BCC,Fe:Va)`: The Bohr magneton number parameter of bcc Fe.

The parameter name consists of several parts. The first is a type-identifier and these can be used:

- `G`: Standard energy parameter (Gibbs energy of formation) or for interaction parameters;

- L: Excess energy parameter (Gibbs energy of interaction) always used for interaction parameters;
- TC: Curie temperature for magnetic ordering;
- BMAGN or BM: Bohr magneton number for magnetic ordering (or Born function ωPr,Tr for aqueous solute species).
- VO: Molar volume at 298.15 K and 1 bar (a numeric value only);
- VA: Integrated thermal expansivity; $\int_{298.15}^{T} \alpha\ (T) dT \int_{298.15}^{T} \alpha\ (T) dT$
- VB: Bulk modulus at 1 bar;
- VC: Isothermal compressibility;
- VK: High-pressure fitting parameter.

When necessary quantities as H (enthalpy), S (entropy), V (Volume), F (Helmholtz energy), etc., can be calculated from the Gibbs energy.

PHASE NAME

Specifying the `PHASE NAME` in uppercase is recommended; however, if you prefer to write it as a mixture of uppercase and lowercase, it automatically converts all lowercase to uppercase, as the GIBBS module only recognises uppercase phase names. It is important that if a phase bears a legal phase-type (among `G`, `A`, `Y`, `L`, `I`, `F` and `B`) in its phase definition (already by the `PHASE` keyword; such as GAS:G, LIQUID:L,SLAG:L, IONIC-LIQ:Y, SPINEL:I, FCC_L12:F, HCP_D021:F, BCC_B2:B, AQUEOUS:A), such a valid phase-type code should not be attached to the phase name when ENTER_PARAMETER is executed.

CONSTITUENT ARRAY

The constituent array consists of a list of constituent names. Interaction parameters have two or more constituents from the same sublattice separated by a comma. If the phase has sublattices, at least one constituent in each sublattice must be specified. The constituents in different sublattices must be given in sublattice order and are separated by a colon.

After the component array, a sub-index digit can be specified after a semicolon. This digit must be in the range 0 to 9. The interpretation of the sub-index depends on the excess energy model used for the phase. If no semicolon and digit are given, the sub-index value is assumed to be as zero.

The excess energy parameters, e.g. the regular/subregular (binary) parameter or ternary parameters, are multiplied with two or more fractions of the constituents from the same sublattice of the solution phase. These additional constituents must be given as interacting constituents (as the following prompt).

Solution phases with sublattices can have interacting constituents in each sublattice.

INTERACTION PARAMETER

An interaction parameter, which is used to describe the excess term of a quantity, must have two or more constituents that interact with each other on a specified sublattice site of the given phase. It is arbitrary which of these constituents is given as the first constituent and what is given as the interacting constituents. The software always sorts the constituents (in each sublattice) in alphabetical order when the parameter name is written as a prompt (for entering its parameter value) and when the parameter is listed (using the GIBBS commands *LIST_PARAMETER* on page 102 or *LIST_PHASE_DATA (from GIBBS)* on page 103). This is important for all asymmetric interaction parameters where the sign of the interaction parameter must depend on the appearance order.

Use an asterisk* to denote that the excess interaction parameter is independent of the constituents of a specific sublattice. For example, `L(FCC_L12,AL,NI:*)` means that the interaction parameter is for the binary interaction between constituents AL and NI on the first sublattice in the `FCC_L12` solution phase, while it is independent of all constituents on the second sublattice. An interaction parameter in the list of constituents is always added to the Gibbs energy and the asterisk * is calculated with the term of [1∑y (specified constituents)], which implies that in an A-B binary system these `L` parameters are identical (but in higher-order systems, these are different):

- `L(phase,A,B)` is multiplied with X(A)*X(B)

- `L(phase,A,*)` is multiplied with X(A)*(1-X(A))
- `L(phase,B,*)` is multiplied with X(B)*(1-X(B))

If you press <Enter> when you are asked for a parameter name or if you have improperly input the entire parameter name, you are asked for each of these items in the name.

ENTER_PARAMETER

For the PARROT Module, this command is the same and is described here.

Also see *About the ENTER_PARAMETER Command* on page 95 for details about each of the options.

Use this command to enter TP-function(s) for a specific parameter for a phase interactively with this command. If there is already a parameter expression defined (for this parameter), that is deleted and replaced with newly entered one.

The entered TP-Function(s) for the parameter can be changed later with the command *AMEND_ PARAMETER* on page 44.

Syntax	ENTER_PARAMETER

Prompts	`PARAMETER: <PARAMETER NAME>`
	As explained in *About the ENTER_PARAMETER Command* on page 95, specify a correct and complete `Parameter Name`, which should contain all the necessary parts of the general form:
	`<IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)`
	If a parameter name is not acceptable or <Enter>, is pressed, a message displays:
	`*** Error, please re-enter each part separately`
	and you are prompted for input for each required part for a parameter name.

Options	**Description or Information**
`IDENTIFIER`	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>` If this command is used one or more times, the previous value on this prompt is set as default. Press <Enter> for the same type identifier or specify a new type.
`PHASE NAME`	`PHASE NAME (/ABCD/): <PHASE NAME>` Each parameter is valid for a specific phase only. The name of that phase must be supplied (maximum 24 characters). The name can be abbreviated.

Syntax	ENTER_PARAMETER
CONSTITUENT	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>` A parameter is identified by the constituents on a specified sublattice site of the given phase, the stoichiometric coefficients of which are multiplied with the parameter. The name of the constituent can be abbreviated. It is the species name, not the stoichiometric formula required here. If this command is used one or more times, the previous value on this prompt is set as default. Accept it by pressing <Enter> if the constituent is the same, or specify a new species name.
INTERACTING CONSTITUENT	For phases with several sublattices, the program asks for one constituent in each sublattice. `INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` If this command is used one or more times, the previous value on this prompt is set as default. Press <Enter> to accept it if the constituent is the same, or specify a new species name. To cancel the default value of the interacting constituent type `NONE` or the name of another constituent. This question is repeated until all the interested interacting constituent(s) on a specific sublattice in the phase are specified, and finally an <Enter> is enforced.
DEGREE	`DEGREE /#/: <DEGREE>` Degree is model-dependent. Specify an integer number (a value from 0 through 9) as the degree of composition-dependent interaction contribution for the phase parameter. This is valid for excess energy (`L`), Curie temperature (`TC`) and Bohr magneton number (`BMAGN`), as well as for volume-related parameters (`V0`, `VA`, `VB`, `VC` or `VK`). For binary interaction parameters, the degree is usually the power in the Redlich-Kister expression. For ternary interaction parameters, it is usually the Hillert ternary index. For a standard `G` parameter for a pure component (end-member) its degree should be always 0 and this prompt should not display.
PHASE PARAMETER	After this prompt, the program echoes on the screen the full TP-Function of the phase parameter.

Syntax	ENTER_PARAMETER
	`LOW TEMPERATURE LIMIT /298.15/: <LOWEST TEMPERATURE LIMIT IN K>`
	Specify the lowest temperature limit (in Kelvin), or the lowest-pressure limit (in Pascal but entered as a negative number), for the current TP-Function.
	`FUNCTION: <DEFINITION FOR A FUNCTION>`
	A TP-Function consists of terms in T and P.
	`& <CONTINUATION OF THE DEFINITION FOR THE CURRENT FUNCTION>`
	Continuation of a TP-Function definition.
	`HIGH TEMPERATURE LIMIT /6000/: <HIGH TEMPERATURE LIMIT IN K>`
	Specify the high temperature limit (in Kelvin), or the high-pressure limit (in Pascal; only if the *lowest limit* is entered as a negative number) for the current TP-Function.
	`ANY MORE RANGES /N/: <Y OR N>`
	Enter `Y` for more function(s) or `N` to end this command.

LIST_ALL_VARIABLES

Create a list of the values and status of all variables in the PARROT workspace on screen or in a text file. After an optimization some statistical information is also written. The correlation matrix is written if that option is chosen.

Also see *SET_OUTPUT_LEVELS* on page 115.

Syntax	LIST_ALL_VARIABLES
Prompt	`OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME>` A list of the current values and status, as well statistical information (if after optimization) and the correlation matrix (optionally), for all variables is written on screen (by pressing <Enter>) or on the list file under a specific file name which can later on be opened and edited by basic text editor.

	Example Output
	The following is the listed variables for the example TCEX36, after the first optimization when having set all variables available for optimization.
	`== OPTIMIZING VARIABLES ==`
	`AVAILABLE VARIABLES ARE V1 TO V00`
	`VAR. VALUE START VALUE SCALING FACTOR REL.STAND.DEV`

Syntax	LIST_ALL_VARIABLES
	V1 2.03729090E+04 2.03688352E+04 2.03688352E+04 3.41524152E+00 V2 -2.94286372E+01 -2.94286372E+01 -2.94286372E+01 2.37944771E+00 V11 -2.17373936E+04 -2.18095983E+04 -2.18095983E+04 3.97086303E-02 V12 1.52107184E+01 1.51660547E+01 1.51660547E+01 5.84552327E-02 V15 2.42082351E+04 2.45139169E+04 2.45139169E+04 5.00914471E+00 V16 -8.38723972E+00 -8.83460472E+00 -8.83460472E+00 1.60961318E+01 V17 3.08916533E+03 3.15802558E+03 3.15802558E+03 9.63738108E+00 V19 2.20292586E+04 2.21385673E+04 2.21385673E+04 4.14274011E+01 V20 -7.04217974E+00 -7.22424882E+00 -7.22424882E+00 1.00561540E+02 NUMBER OF OPTIMIZING VARIABLES : 9 ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO THE SUM OF SQUARES HAS CHANGED FROM 8.00002729E+04 TO 8.00002719E+04 DEGREES OF FREEDOM 45. REDUCED SUM OF SQUARES 1.77778382E+03

LIST_CONDITIONS

In the PARROT module, generate a list of the current values of optimization conditions, as well the current status of listing, either on screen or in a basic text file which can be opened and edited by a text editor.

Syntax	LIST_CONDITIONS
Prompt	OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME> Press <Enter> to output to screen or on the list file under a specific file name.

LIST_PARAMETER

This command is for both the GIBBS and PARROT modules.

List the TP-function(s) of a specific parameter for a phase. You must supply the name of the phase parameter:

THE PARAMETER NAME: <IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)

For encrypted commercial databases, you may not be able to use this command to list any retrieved parameter.

Syntax	LIST_PARAMETER

Prompts	`PARAMETER: <PARAMETER NAME>`
	
	Specify a correct and complete parameter name, which should contain all the necessary parts of the general form: <identifier>(<phase>,<constituent array>;<digit>)
	
	See *About the ENTER_PARAMETER Command* on page 95 to learn how to define the Parameter Name.
	
	If a parameter name is not acceptable or <Enter> is pressed, an error message displays:
	
	`*** Error, please re-enter each part separately`
	
	The program prompts for separate input for each required part for a parameter name.

	`IDENTIFIER (/X/): <G OR L, OR TC, OR BM, OR V0 OR VA OR VB OR VC OR VK>` Specify one of these types of legal identifiers.
	`PHASE NAME (/ABCD/): <PHASE NAME>` Specify the phase name.
	`CONSTITUENT (IN SUBLATTICE # /ABC/): <SPECIES NAME>` Specify the constituent name on the specified sublattice site of the given phase.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` Specify the interacting constituent name on the specified sublattice site of the given phase; if there is no interacting constituent, press <Enter>.
	`INTERACTING CONSTITUENT (IN SUBLATTICE # /XYZ/): <SPECIES NAME>` If there is more than one interacting constituent on the specified sublattice site of the given phase, specify them; otherwise press <Enter>.
	`DEGREE /#/: <DEGREE>` Specify a numerical number as the degree for the phase parameter.

LIST_PHASE_DATA (from GIBBS)

All data for a specific phase are written on screen. The thermochemical parameters listed for each phase are always in SI units.

Syntax	LIST_PHASE_DATA
Prompt	`PHASE NAME: <PHASE NAME>` Specify a phase name.

LIST_RESULT

List the result of the current optimization run on screen or in a specified file. The level of detail listed can be prescribed by the command *SET_OUTPUT_LEVELS* on page 115.

The LIST_RESULT command has an option `G` for creation of an experimental data file with two columns, one for the experimental value and the other for the calculated value. It allows plotting a diagram and visualizing the fitting results. The plot is automatically plotted and you can scale it giving commands in the POST module. The option `D` lists all experiments, including those with fulfilled inequalities which are suppressed with the default option `C`.

Syntax	LIST_RESULT
Prompt	`FULL, CONDENSED, DETAILED OR GRAPHIC FORMAT: /C/: <C OR D OR F OR G>`
	Choose a format. • `C` (condensed) is the default. • `D` (detailed) lists all experiments, including those with fulfilled inequalities (not included with the default condensed file). • The `G` (graphical) format creates an experimental data file containing two columns, one with the experimental value and the other with the calculated value. This allows plotting of a diagram where all symbols should be on the diagonal if the fit is perfect. The plot is automatically plotted and you can scale it giving commands in the POST module. • The `F` (full) format is obsolete but retained for backward compatibility. `OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME>` A list of the current optimization results are written on screen (by pressing <Enter>) or on the list file under a specific file name which can later on be opened and edited by basic text editor.
	The output normally consists of these parts: • A title showing the data of the action; • A message on successful optimization with the iteration number in the last optimization;

Syntax	LIST_RESULT
	• A paragraph describing the optimization condition; • A list of the latest set of optimized and fixed variables; • A paragraph describing the optimization quality (some statistical information); • A correlation matrix for all optimizing variables (optionally shown, pre-set by the SET_OUTPUT_LEVELS command); • A list of all parameters (including their symbol names, status and current values or function expressions), and all phase descriptions (including phase name, model names, constituents, and phase's `G/L/TC/BM` expressions) pre-defined for each phase in the system (optionally shown, pre-set by the SET_OUTPUT_LEVELS command); • A paragraph describing the alternate equilibria (and possible error during optimization); • A detailed list on all the equilibrium points used in the current optimization. If the alternate-mode is used in the optimization, only an error value is shown. For normal-mode calculations, a list is shown for the experimental equilibrium numbers (first column) and corresponding original experimental data (second column, in the form `quantity = value`) which are used in optimization (i.e. non-zero weighted points). Such experimental data are by each point compared by the calculated value (third column) after the last optimization. Also listed are the experimental error (column 4), the difference between the calculated value and original experimental data (column 5), and the contribution to the sum of least square (column 6).

Example Output

The following is the listed result in the example TCEX36 (in the *Console Mode Examples Guide*.), after the first optimization based on the prescribed definitions in the TCEX36a.TCM and TCEX36b.TCM, but having set all variables available for optimization.

```
=====================================================

           OUTPUT FROM P A R R O T.
=====================================================

*** SUCCESSFUL OPTIMIZATION. ***      NUMBER OF ITERATIONS:  10

== OPTIMIZING CONDITIONS ==

   RELATIVE STANDARD DEVIATIONS FOR EXPERIMENTS: N
```

```
   MINIMUM SAVE ON FILE: Y

   ERROR FOR INEQUALITIES =  1.00000000E+00

   RELATIVE STEP FOR CALCULATION OF DERIVATIVES =  1.00000000E-04

ARGUMENTS FOR SUBROUTINE VA05AD (HSL)

   MAXFUN =   1   DMAX =  1.00000000E+02   H =  1.00000000E-04    ACC = (INITIAL SUM
OF SQUARES) *  1.00000000E-03

== OPTIMIZING VARIABLES ==

AVAILABLE VARIABLES ARE V1  TO V00

VAR.   VALUE             START VALUE       SCALING FACTOR    REL.STAND.DEV

V1      2.03749463E+04    2.03729090E+04    2.03729090E+04    3.41455863E+00

V2     -2.94286372E+01   -2.94286372E+01   -2.94286372E+01    2.37944774E+00

V11    -2.17395673E+04   -2.17373936E+04   -2.17373936E+04    3.98405298E-02

V12     1.52107184E+01    1.52107184E+01    1.52107184E+01    5.82861832E-02

V15     2.42106560E+04    2.42082351E+04    2.42082351E+04    5.07239609E+00

V16    -8.38723972E+00   -8.38723972E+00   -8.38723972E+00    1.69546796E+01

V17     3.08947424E+03    3.08916533E+03    3.08916533E+03    9.85220694E+00

V19     2.20314615E+04    2.20292586E+04    2.20292586E+04    4.16329629E+01  V20    -
7.04217974E+00   -7.04217974E+00   -7.04217974E+00    1.03161466E+02

NUMBER OF OPTIMIZING VARIABLES :   9

ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO

THE SUM OF SQUARES HAS CHANGED FROM   8.00002719E+04 TO   8.00002709E+04

DEGREES OF FREEDOM   45.  REDUCED SUM OF SQUARES   1.77778380E+03

Number of alternate equilibria           14

SYMBOL         STATUS   VALUE/FUNCTION

R          80000000  8.3145100E+00

RTLNP      20000000 +R*T*LN(1E-05*P)

V1         48000000  2.0374946E+04

V2         48000000 -2.9428637E+01

V11        48000000 -2.1739567E+04

V12        48000000  1.5210718E+01

V15        48000000  2.4210656E+04

V16        48000000 -8.3872397E+00

V17        48000000  3.0894742E+03

  21 V19       48000000  2.2031462E+04   22 V20       48000000 -7.0421797E+00

LIQUID

EXCESS MODEL IS REDLICH-KISTER_MUGGIANU                              CONSTITUENTS: A,B

      G(LIQUID,A;0)-G(BCC,A;0) =    500.00<T< 2000.00: +14000-10*T
```

```
      G(LIQUID,B;0)-G(BCC,B;0) =    500.00<T< 2000.00: +18000-12*T

      L(LIQUID,A,B;0) =    500.00<T< 2000.00: +V11+V12*T       L(LIQUID,A,B;1) =
500.00<T< 2000.00: +V13+V14*T

A2B

  2 SUBLATTICES, SITES  2:  1     CONSTITUENTS: A : B

      G(A2B,A:B;0)- 2 G(BCC,A;0)-G(BCC,B;0) =

             500.00<T< 2000.00: +V1+V2*T+V3*T*LN(T)

BCC

EXCESS MODEL IS REDLICH-KISTER_MUGGIANU                              CONSTITUENTS: A,B

      G(BCC,A;0)-G(BCC,A;0) =    500.00<T< 2000.00: 0.0

      G(BCC,B;0)-G(BCC,B;0) =    500.00<T< 2000.00: 0.0

      L(BCC,A,B;0) =    500.00<T< 2000.00: +V15+V16*T

      L(BCC,A,B;1) =    500.00<T< 2000.00: +V17+V18*T

FCC

EXCESS MODEL IS REDLICH-KISTER_MUGGIANU                              CONSTITUENTS: A,B

      G(FCC,A;0)-G(BCC,A;0) =    500.00<T< 2000.00: 408

      G(FCC,B;0)-G(BCC,B;0) =    500.00<T< 2000.00: +3300-3*T

      L(FCC,A,B;0) =    500.00<T< 2000.00: +V19+V20*T

      L(FCC,A,B;1) =    500.00<T< 2000.00: +V21+V22*T

======  BLOCK NUMBER  1

DEFINED CONSTANTS

    DX=2E-2, P0=101325, DH=500, DT=10

DEFINED FUNCTIONS AND VARIABLES%

    HTR=HM(LIQUID)-HM(A2B)

Alternate equilibrium calculation                  0.4183

Alternate equilibrium calculation                  0.1932

Alternate equilibrium calculation                  0.1016

Alternate equilibrium calculation                  1.4354E-03

Alternate equilibrium calculation                  2.5063E-02

Alternate equilibrium calculation                  8.3929E-03

Alternate equilibrium calculation                   141.4

Alternate equilibrium calculation                   141.4

Alternate equilibrium calculation                   141.4

Alternate equilibrium calculation                   141.4

Alternate equilibrium calculation                  1.2532E-03

Alternate equilibrium calculation                  5.4781E-04
```

```
Alternate equilibrium calculation              1.5404E-03
Alternate equilibrium calculation              1.2702E-03
ACR(B)=9.4E-1              0.9397     2.85E-02 -2.7745E-04 -9.7472E-03
ACR(B)=8.4E-1              0.8395     2.82E-02 -4.9038E-04 -1.7396E-02
ACR(B)=7.4E-1              0.7407     2.81E-02  7.3804E-04  2.6305E-02
ACR(B)=6.4E-1              0.6424     2.81E-02  2.3935E-03  8.5272E-02
ACR(B)=5.4E-1              0.5434     2.82E-02  3.4449E-03  0.1220
ACR(B)=4.4E-1              0.4428     2.85E-02  2.8265E-03  9.9024E-02
ACR(B)=3.4E-1              0.3394     2.90E-02 -5.8174E-04 -2.0040E-02
ACR(B)=2.3E-1              0.2320     2.97E-02  2.0260E-03  6.8208E-02
ACR(B)=1.2E-1              0.1194     3.06E-02 -6.4192E-04 -2.0981E-02
HMR(LIQUID)=-1964          -1957.     5.00E+02   7.439      1.4878E-02
HMR(LIQUID)=-3500          -3478.     5.00E+02   21.67      4.3338E-02
HMR(LIQUID)=-4588          -4565.     5.00E+02   22.69      4.5382E-02
HMR(LIQUID)=-5239          -5217.     5.00E+02   21.50      4.3008E-02
HMR(LIQUID)=-5454          -5435.     5.00E+02   19.11      3.8216E-02
HMR(LIQUID)=-5233          -5217.     5.00E+02   15.50      3.1008E-02
HMR(LIQUID)=-4575          -4565.     5.00E+02   9.691      1.9382E-02
HMR(LIQUID)=-3481          -3478.     5.00E+02   2.669      5.3385E-03   118 HMR
(LIQUID)=-1950          -1957.     5.00E+02  -6.561     -1.3122E-02
```

LIST_STORE_FILE

Syntax	LIST_STORE_FILE
	Displays the name of the store file and its full path.

LIST_SYMBOL_IN_GES

Lists TP-function(s) for the entered model parameters for phases in the system on screen. In many cases, the optimizing variables are parts of the TP-functions which in turn are entered in model parameters for various phases in the GIBBS workspace.

This command is a way to find out how the functions depend on the optimizing variables. In the PARROT module the list is only shown on screen, not written to any file (which the LIST_SYMBOL command in the GIBBS module does).

Syntax	LIST_SYMBOL_IN_GES
Prompt	NAME: <SYMBOL NAME> Specify a symbol name. Only those symbols that match this name are listed. Or press <Enter> for a list of all available symbols entered in the system.

OPTIMIZE_VARIABLES

Perform variable optimization. All system-definition data needed for the optimization is read from the current work file (*.PAR). The result of the optimization is automatically stored in the current work file.

Syntax	OPTIMIZE_VARIABLES
Prompt	NUMBER OF ITERATION /N/: <INTEGRAL NUMBER OF ITERATION> Specify an integral number for the optimization iteration. The specified iteration number is shown as the default number. The PARROT module tries this number of different sets of values of the optimizing variables unless it has converged earlier, or has given up earlier.
	It initially takes a small step in each variable to find the steepest slope. You can give zero iteration to calculate the error in all selected experiments, and then use *LIST_RESULT* on page 104to check how good (bad) the current fit is.

READ_PARROT_WORKSPACES

Read the previous PARROT/GIBBS/POLY3 workspaces back to replace the current PARROT/GIBBS/POLY3 workspaces. It is useful if late changes made through various PARROT/GIBBS/POLY commands are not satisfactory.

Syntax	READ_PARROT_WORKSPACES
	The previous PARROT/GIBBS/POLY3 workspaces are always associated with the latest action to either open a work file by *SET_STORE_FILE* on page 116, or to create a work file by *CREATE_NEW_STORE_FILE* on page 94 (either interactively in the PARROT module or through a *MACRO_FILE_OPEN* on page 5 of an *SETUP.TCM file), or to update the work file by *SAVE_PARROT_WORKSPACES* on page 111.
	Unlike the READ commands in other modules (e.g. GIBBS or PLOY), this command in the PARROT module does not ask for the file name where to read a previously opened/created/updated PARROT/GIBBS/POLY3 workspaces, for the reason described above.

Syntax	READ_PARROT_WORKSPACES
	You cannot use this command if there is no work file opened or created already.

RECOVER_VARIABLES

Syntax	RECOVER_VARIABLES
	The values of all variables are set back to the start values.

REINITIATE

This command should not be used unless you want to destroy the current PARROT workspace. However, this PARROT command does not reinitiate the GIBBS/POLY3 workspaces.

Syntax	REINITIATE
	The workspace used by the PARROT program is reinitiated. All output and optimizing conditions are given their default values. All variables are set fixed with their value equal to zero.

RESCALE_VARIABLES

Syntax	RESCALE_VARIABLES
	The current values of all the parameters are copied to their start values and the scaling factors for further optimization. Thus it should be done now and again, in particular if you think the optimization results are improved after the previous run(s), or if any variable changes more than a factor of 10.

SAVE_PARROT_WORKSPACES

Syntax	SAVE_PARROT_WORKSPACE
	If the latest changes made through various PARROT/GIBBS/POLY commands are as required, use this command to save the current workspace (i.e. data area) used by the PARROT program (also including the current GIBBS and POLY3 workspaces), onto the present work *.PAR file which has already been opened by *SET_STORE_FILE* on page 116 or been created by *CREATE_NEW_STORE_FILE* on page 94 (either interactively in the PARROT module or through a MACRO_FILE_OPEN of an *SETUP.TCM file). The PARROT/GIBBS/POLY3 workspaces are updated after each PARROT/GIBBS/POLY command. This means the current PARROT/GIBBS/POLY3 workspaces are always updated onto the latest work *.PAR file that is associated with the latest SET_STORE_FILE or CREATE_NEW_STORE_FILE command.
	Unlike the SAVE commands in other modules (e.g. GIBBS or PLOY), this command in the PARROT module does not ask for the file name where to save the current PARROT/GIBBS/POLY3 workspaces, for the reason described above. You cannot use this command if there is no work *.PAR file opened or created already.

SET_ALTERNATE_MODE

Turn the alternate mode on or off. The alternate mode is described in *PARROT Commands* on page 86. It should be used only to optimize start values of the model parameters in the beginning of the assessment.

With the ALTERNATE mode is possible to include functions to be evaluated together with an alternate calculation and to select ALTERNATE mode for each experimental equilibrium.

Syntax	SET_ALTERNATE_MODE
Prompt	`ON /Y/: <Y OR N>` The default `Y` turns on the alternate mode.

SET_EXTERNAL_PARAMETER

Optimize model parameters of the so-called external models, which may not be as fully-implemented/integrated parts inside the GIBBS system and thus are independently defined within user-specified/written source codes. This command makes links between the PARROT optimizing variables and external model parameters.

> In order to be able to build the specified external model and to use this SET_EXTERNAL_PARAMETER command, it requires you to first write codes for a preferred external model (as it is not within the standard GIBBS system of the Thermo-Calc software) and then provide it to the consultancy team (consult@thermocalc.se) of Thermo-Calc Software Company in Stockholm for the purpose of including the external model in a separate DLL that is interactively connected to the Thermo-Calc software. Such a special service can/should only be provided under the restrictive condition that a specific Consultancy Project Contract is bilaterally agreed and signed in advance between users and Thermo-Calc Software.

Syntax	SET_EXTERNAL_PARAMETER
Prompt	`EXTERNAL NAME: <PARAMETER NAME IN THE EXTERNAL MODEL>` Correctly specify the parameter name defined within a specified/written external model.

	`AS OPTIMIZING VARIABLE NUMBER: <N>`
	
	Give the number of the PARROT optimizing variable, which should have a unique link to the specified parameter name of the external model.
	
	For example, @@ Link PARROT variables to CAM parameters:
	
	`SET-EXTER-PAR GT(SLAG_A,O-2,SI+4,CA+2) 1`
	
	`SET-EXTER-PAR ET(SLAG_A,O-2,SI+4,CA+2) 2`
	
	`SET-EXTER-PAR EXT(SLAG_A,O-2,SI+4,CA+2) 3`

SET_FIX_VARIABLE

Prescribe a fixed value to a variable. The variable is considered as a constant at the optimization, usually after successful optimization runs for some specific variables (for some of the phases in the system). Such fixed variables can also be set back for further optimization run by using *SET_OPTIMIZING_VARIABLE* on page 114.

Syntax	SET_FIX_VARIABLE
Prompts	`VARIABLE NUMBER: <VARIABLE NUMBER(S)>` Specify the number of the variable. It is possible to give a range by giving two numbers with a hyphen in between (no spaces allowed), e.g. `1`. The parameters within the limits are set fixed to their current values (no question of values).
	`START VALUE /XXXX.XXXXX/: <A VALUE TO BE FIXED>` Specify a numerical value to be fixed for the selected variable (only when one variable number is specified at the previous prompt). The current value for the selected variable is

Syntax	SET_FIX_VARIABLE
	shown as default. If a range of variable numbers is specified at the previous prompt, this question is not prompted, and the current values are used as fixed values.

SET_OPTIMIZING_CONDITION

Specify the conditions for the optimization. The default values are chosen automatically by the PARROT program, and in most cases these should not be changed.

The optimization stops when the sum of errors decreases by this value, even though it might be possible to increase it further with a new OPTIMIZE command.

Syntax	SET_OPTIMIZING_CONDITION
	`RELATIVE STANDARD DEVIATION FOR EXPERIMENTS? /N/: <Y OR N>` Choose whether the standard deviation of the experimental determinations are absolute values (`N`) or if these are used as relative weighting factors (`Y`). The estimated standard deviations of the optimized variables might be different for the two cases.
Prompt	`MIN SAVE ON FILE? /Y/: <Y OR N>` Specify whether the program should minimize the transfer of data to and from the current work file during the optimization: If Y, it speeds up the optimization. After optimization use the *SAVE_PARROT_WORKSPACES* on page 111 command to update the progress onto the current work file. If minimum saving on file is not specified (`N`), the workspaces are stored on the current work file after every iteration in the optimization procedure.
	`ERROR FOR VARIABLE BOUNDS: /1/: <RETURN >` This feature has not been implemented yet. Press <Enter>.
	`RELATIVE STEP FOR CALCULATION OF DERIVATIVES: /1E-04/: <XXX>` In the calculation of the correlation matrix for equilibria with inaccuracy in the independent state variables, some numerical derivatives might have to be calculated. Specify the relative step (`xxx`) for the calculation of these derivatives.
	`MAXFUN (VA05AD): /100/: <N>` The maximum number of iterations in the optimization. The same value set by the OPTIMIZE command.

Syntax	SET_OPTIMIZING_CONDITION
	`DMAX (VA05AD): /100/: <N>` An estimate of the maximum distance between the start and the final values of the variables. A smaller value makes the program vary the parameters with smaller factors.
	`H (VA05AD): /1E-04/: <XXX>` The step used in the scaled variables for calculating numerical derivatives during the optimization.
	`ACC/(INITIAL SUM OF SQUARES) (VA05AD): /.001/: <XXX>` The break condition for the optimization. The accepted value is the difference between the true minimum and the calculated one.

SET_OPTIMIZING_VARIABLE

Specify which variable value should be estimated at the optimization. Such a variable can have a value of zero or another start value, or is fixed by *SET_FIX_VARIABLE* on page 112 prior to the previous optimization run.

For a good reference prior to this command, a list of all variables (which are used to define various parameters for phases in the optimizing system, as pre-entered in the *SETUP.TCM file or interactively entered in the GIBBS module) are obtained by using *LIST_ALL_VARIABLES* on page 101.

Syntax	SET_OPTIMIZING_VARIABLE
Prompts	`VARIABLE NUMBER: <VARIABLE NUMBER(S)>` Specify the number(s) for the variable(s) to be set. Any non-zero parameter within the range is allowed to be optimized. A parameter with a current value equal to zero must be specified explicitly here in order to be optimized. Specify a range by giving two numbers connected by a hyphen (no space allowed!), e.g. `2-8`. Under this circumstance, no question of start values are asked.
	`START VALUE /XXXX.XXXXX/: <A GUESS VALUE>` Specify a start guess of the optimum value (only when one variable number is specified at the previous prompt). The current numerical value for the selected variable is shown as default. This guess is critical as the initial guess of all parameters must make it possible to calculate the selected equilibria. If a range of variable numbers is specified at the previous prompt, this question is not prompted, and the current values (as start values) are not changed.

SET_OUTPUT_LEVELS

Choose the type of information the PARROT module gives during the optimization procedure and when listing results.

Syntax	SET_OUTPUT_LEVELS
Prompts	LIST INCREMENT /1/: <INCREMENT IN ITERATION NUMBER> Specify the increment in iteration number for which information is listed on screen during the optimization. By giving a larger number it shortens the list.
	LIST SUM OF SQUARES: /Y/: <Y OR N> Choose whether the sum of squares is listed on screen during the optimization procedure.
	LIST SCALED VARIABLES: /Y/: <Y OR N> Choose whether the scaled variable values are listed on the terminal during the optimization procedure.
	LIST WEIGHTENED RESIDUALS: /N/: <N OR Y> Choose whether the weighted residuals are listed on the terminal during the optimization procedure.
	LIST ALL PARAMETERS: /N/: <N OR Y> Choose whether all parameters in the models are listed in *LIST_RESULT* on page 104.
	LIST CORRELATION MATRIX: /N/: <N OR Y> Choose whether the correlation matrix of the variables are listed with the commands LIST_RESULT and *LIST_ALL_VARIABLES* on page 101.

SET_SCALED_VARIABLE

This command is similar to *SET_OPTIMIZING_VARIABLE* on the previous page in that it specifies start values for optimizing variables. The difference being it is functional for only one optimizing variable at one time. It also prescribes a minimum and maximum value for the variable. During the optimization runs, the variable value is limited within this min-max range.

Such a variable may have a value of zero or another start value, or may be fixed by *SET_FIX_VARIABLE* on page 112 prior to the previous optimization run. For a good reference prior to using this command, a list of all variables (which are used to define various parameters for phases in the optimizing system, as entered in the *SETUP.TCM file or interactively entered in the GIBBS module) is obtained by using *LIST_ALL_VARIABLES* on page 101.

Syntax	SET_SCALED_VARIABLE
Prompts	`VARIABLE NUMBER: <VARIABLE NUMBER(S)>` Specify the number for an optimizing variable to be set. Any non-zero parameter is allowed to be optimized, and a parameter with a current value equal to zero must be specified explicitly here in order to be optimized.
	`START VALUE /XXXX.XXXXX/: <A GUESS VALUE>` Specify a start guess of the optimum value. The current numerical value for the selected variable is shown as default. This guess is critical as the initial guess of all parameters must make it possible to calculate the selected equilibria.
	`MIN VALUE /XXXX.XXXXX/: <A GUESS VALUE>` Specify a minimum guess of the optimum value. It should be smaller than, at least equal to, the current start value for the selected variable.
	`MAX VALUE /XXXX.XXXXX/: <A GUESS VALUE>` Specify a maximum guess of the optimum value. It should be larger than, at least equal to, the current start value for the selected variable.

SET_STORE_FILE

Specify a store file (work file) to be used for compilation and optimization. The *.PAR work file must be created (*CREATE_NEW_STORE_FILE* on page 94) as a store file. The GIBBS, POLY and PARROT workspaces are read from the specified store file (with a default extension of PAR).

Syntax	SET_STORE_FILE
Prompt	Enter an `*.PAR file name` in the **File name** field and specify the working directory in the **Look in** field. Normally the default file-type in the **Files of type** field is the correct one for the PARROT workspace format (i.e. *.PAR file). Click **Open** or **Cancel** as applicable.

POLY_3 Commands

In this section:

ADD_INITIAL_EQUILIBRIUM

Add initial equilibrium points from which a phase diagram is calculated (through the *MAP* on page 158 command).

Syntax	ADD_INITIAL_EQUILBRIUM
Prompt	`DIRECTION /DEFAULT/: <DIRECTION CODE>` `direction code (s)` `1` or `2` for positive direction of axis 1 or 2, respectively; `-1` or `-2` for negative direction of axis 1 or 2, respectively; `Default` for all directions. The direction is important when the initial equilibrium point is in a single-phase region or when the phase diagram is an isopleth (tie-lines not in the plane of calculation). In such cases, the program searches for a line in the diagram (i.e. a line where the amount of a phase is zero, imply that if it starts to become stable on one side of the line and disappear on the other side) in the given direction.

Normally, this command is not needed to calculate a property diagram (with the *STEP_WITH_OPTIONS* on page 172 → `NORMAL` command-sequence). In many cases, the ADD_INITIAL_EQUILBRIUM command is not required for MAP commands either, as the mapping procedure starts from the current equilibrium state already calculated. In order to calculate a simple phase diagram, set the equilibrium conditions and the mapping axis variables, and then give a MAP command. But if a phase diagram has disconnected lines, the ADD_INITIAL_EQUILBRIUM command may still be needed to add two or more initial equilibria so that the MAP calculation starts from such initial equilibria at the specified directions to find all phase boundary lines.

This command becomes unnecessary in most cases, as the MAP routines that by default use the Global Minimization Technique handles all the initial equilibrium points in a robust and automatic way. Therefore, you do not need to have a good guess of the starting point and to add any initial equilibrium point prior to the calculations of various types of phase diagrams. However, if preferred, use this command to add any initial equilibrium points in certain directions, the POLY module uses the specified starting point(s) and corresponding user-added initial equilibrium point(s) for a mapping calculation; in such a case, the automatic procedure in the rewritten MAP routines are not enforced while the Global Minimization Technique can still be applied. Similar situations occur when a specific database, in which there are some definitions of initial-equilibrium adding direction(s) for accessed binary and/or ternary subsystems in its ACCESSED_SYSTEM section, is used in the BINARY or TERNARY module for calculating a binary or ternary phase diagram.

The ADD_INITIAL_EQUILBRIUM command with the default direction scans along the axis variables and generates start points each time the scanning procedure crosses a phase boundary. In addition, it generates four start points, scanning cross the middle of each axis, if there is any solubility line that does not reach the axes. At the MAP command, a search for lines in the diagram is made along each direction of the axis variables in the diagram. In this way, it should guarantee that all possible phase boundary lines in a phase diagram are found. Of course, it may take a little longer time to execute than using the minimum number of start points, as some lines may be calculated more than once. But the POLY module remembers all node points and subsequently stops calculations along a line when it finds a known node point.

It is also possible to create a sequence of start points from one initial equilibria by appending a > after the direction at the ADD command. For example:

```
Direction /default/: 2>
Direction /default/: -2>
```

This generates one start point for each set of phase change in the positive direction of the axis 2 (or negative direction of the axis 2); this ensures finding all possible phase boundary lines (not just the first one) along such an axis direction.

This is particularly useful when you have a phase diagram with several lines with no intersection. It is thus possible to calculate e.g. an isopleth for a much more limited composition range. It is also useful for calculating CVD diagrams and Pourbaix diagrams.

All the initial equilibrium points generated by the ADD_INITIAL_EQUILBRIUM command (previously and presently; saved in the current POLY workspace) can be easily listed out on screen with *LIST_INITIAL_EQUILIBRIA* on page 157. A certain initial equilibrium point (including its conditions and equilibrium results) can be loaded into the current equilibrium, if needed, with *LOAD_INITIAL_EQUILIBRIUM* on page 158. Any specific or all of the initial equilibrium points can be deleted from current POLY workspace, if desired, with *DELETE_INITIAL_EQUILIBRIUM* on page 151.

ADVANCED_OPTIONS Descriptions

Also see *ADVANCED_OPTIONS* on page 125 for information about the command.

These ADVANCED_OPTIONS can be set:

EQUILIBRIUM_CALCUL

Designed to decide how to perform the Ordinary POLY Minimization (i.e. the traditional Gibbs Energy Minimization) in the POLY module. By default, the Global Minimization Technique for various single point equilibrium calculations and for stepping or mapping calculations is used. You can use the advanced option *GLOBAL_MINIMIZATION* on page 236 to permanently turn off (for all sequential calculations throughout the current TCC run) the Global Minimization Technique and consequently use only the Ordinary POLY Minimization, and additionally adjust the ways to proceed the normal minimization (mainly in terms of how to control the steps in reaching Gibbs energy minima in an equilibrium state). When enforcing the Global Minimization Technique in the equilibrium calculations, you can further adjust the manners for assigning grid-points and for handling new possible compositional set(s) for solution phases during stepping/mapping; since the Global Minimization Technique also uses the normal POLY optimization routine for some calculations, it is also possible to change the ways on how to control the steps in reach Gibbs energy minima in an equilibrium state, in order to improve convergence.

GLOBAL_MINIMIZATION

Designed to decide how to perform the so-called Global Minimization in the POLY module. By default, the POLY module uses the Global Minimization Technique for various single-point equilibrium calculations and for stepping or mapping calculations. You can use this to permanently (for all sequential calculations throughout the current TCC run) turn off the Global Minimization Technique and consequently use only the Ordinary POLY Minimization, and additionally adjust the ways to proceed the normal minimization (mainly in terms of how to control the steps in reaching Gibbs energy minima in an equilibrium state). When enforcing the Global Minimization Technique in the equilibrium calculations, you can adjust the manners for assigning grid-points and for handling new possible compositional set(s) for solution phases during stepping/mapping; since the Global Minimization Technique also uses the normal POLY optimization routine for some calculations, it is also possible to change the ways on how to control the steps in reach Gibbs energy minima in an equilibrium state, in order to improve convergence.

LIST_PHASE_ADDITION

The values set as additional contributions (given by PHASE_ADDITION) to Gibbs energy *G*m (J/mol formula unit) to all the phases (stoichiometric or solution) are listed at the current calculated equilibrium.

MAJOR_CONSTITUENTS

Use this to set the major constituent(s) of a composition sets in a miscibility gap of a solution phase. Normally, the major constituents are specified when a new composition set is created by the NEW_ COMPOSITION_SET option; but for the first composition set, this option may be needed before using NEW_COMPOSITION_SET.

NEW_COMPOSITION_SET

A solution phase that can exist with two (or more) different compositions simultaneously must have two (or more) composition sets in its phase descriptions. Normally the database creates as many composition sets as is necessary but use this command to add or delete more composition sets. A complex solution phase in a defined multicomponent system can have up to 9 different composition sets for the purpose of appropriately handling its possible miscibility gap(s) under various temperature-pressure-composition conditions.

This option is unnecessary if the Global Minimization Technique is in use, as it can automatically detect all possible miscibility gap(s) for complex solution phases (normal or disordered/ordered phases) and then automatically add required composition sets for such phases in the defined system.

This option can automatically create composition sets also for disordered phase when it is executed for the ordered phase.

If you have your own data file, this option must be used to indicate that a solution phase can have a miscibility gap (or may exhibit some complex phase separations as to more than two composition sets), unless that the possibly-additional composition set(s) for the solution phase must have already been added inside the TDB file (through a TYPE_DEFINITION command to amend composition-set) or been manipulated through the GIBBS module (using the AMEND_PHASE_DESCRIPTION COMPOSITION_SET command-sequence). This option asks for default major constituents for the new sets and it is important that this is set correctly, otherwise the test for miscibility gaps may fail.

OUTPUT_FILE_FOR_SHOW

The name of a text file is asked for and all the results output from the command SHOW_VALUE is written to this file.

PARAEQUILIBRIUM

This calculates a paraequilibrium between two specific phases in an alloy system with one or more interstitial component(s) as fast diffusion species. Under the paraequilibrium state, two partially-equilibrated phases have the same chemical potential (but different contents) for one or more interstitial components (such as C, N, O, S, etc., as individual or combined)], along varied temperature or along a composition variable (of the matrix or one substitutional component) which has already set as the stepping variable with *SET_AXIS_VARIABLE* on page 164.

To ensure a successful point calculation of paraequilibrium state between two specific phases in a defined alloy system, it is important that you first have made a single-point equilibrium calculation with an initial overall composition in the current system before performing this advanced-option calculation; however, it is unnecessary to obtain an equilibrium in which either one or both of the target phases is stable. The initial overall composition must have a reasonable setting for the desired paraequilibrium calculation for the two target phases. This is especially true for cases where there are more than one interstitial components to be considered in the paraequilibrium state, because different interstitial components (for example C and N combined) may have significant different behaviours as partitioning into different structured phases; otherwise, for one chosen interstitial component the initial overall composition is OK for the paraequilibrium calculation between the specified two phases, but for other chosen interstitial component(s) it might be impossible to calculate the paraequilibrium state.

Note the following:

Always check if the chosen phases A and B have the exactly same definition of elements and if the chosen interstitial components are all in the vacancy sublattice sites of the two phases; otherwise the program cannot find the paraequilibrium state (as it is impossible to correctly calculate u-fractions).

Always have a comprehensive understanding of the normal phase diagram for the currently investigated system, so that you make the appropriate choice of the phase pair and staring bulk composition for the system.

Always set the status of the chosen interstitial components as SPECIAL using the POLY command: *CHANGE_STATUS* on page 135 `Component <interstitial component> = SPECIAL`. By doing this, you get a clear picture of u-fractions of various substitutional and interstitial components, which are different from the overall composition in the system. The SPECIAL status means that specified component(s) are not included in summations for mole or mass fractions. Therefore, all the composition variables plotted from paraequilibrium calculations are u-fraction related quantities.

PHASE_ADDITION

Sometimes it is interesting to add a constant contribution to the Gibbs energy of a phase (stoichiometric or solution). This can be done in the Database Module (DATA) or Gibbs Energy Module (GIBBS), for a stoichiometric phase, or for a pure end-member in a solution phase. However, if the addition is related to the equilibrium state, for example, strain energies or surface energies, interfacial energies or deformation energies, it may be more convenient to have this quantity related to the equilibrium state rather than the thermodynamic data.

Give a value of an addition to the Gibbs energy of a phase. The value should always be constant (implying that the addition is not a function of phase composition or temperature-pressure conditions in the equilibrium state) and always be given in the unit of J/mol formula unit of the phase.

PRESENT_PHASE

The phase specified must be stable at all equilibria calculated during a *MAP* on page 158 command. It is a way to limit the calculations of the monovariant lines in a ternary system to those in the liquidus surface. Normally such a calculation would have two compositional axes and a temperature axis and all monovariant lines, also those between 3 solids, would be mapped. If the liquid is set as PRESENT, only those with the liquid is mapped.

SHOW_FOR_T=

Display various thermodynamic properties (state variables, derived/partial variables or entered symbols) of the currently calculated (stable/meta-stable) equilibrium state but under a different temperature condition. This is useful for knowing, for example, a volume-related property of a frozen (stable/meta-stable) equilibrium state at a certain temperature, where the equilibrated phase assemblage and all the phase compositions are not adjusted while only the temperature condition is changed.

Use this option carefully. You must have successfully calculated a real equilibrium state under one temperature condition (normally the temperature for the last heat treatment). This can then be used to obtain the value(s) of any specified state variable(s) or derived/partial variable(s) or defined symbol(s) for thermodynamic properties of the entire system, of components, or of phases for the currently-defined system (being in a frozen state) under another temperature (normally at room temperature). No real equilibrium is re-calculated through this option, and thus the phase amounts and compositions in the system are the same as at the last real equilibrium calculation.

STABILITY_CHECK

In some composition ranges of a multicomponent system, it often happens that an unstable solution phase region locates inside a miscibility gap, and the stability limit (the *spinodal curve* or *spinodal*) may be not easy to find. A system inside a spinodal is thermodynamically unstable with respect to compositional fluctuations, and the system may experience the *spinodal decomposition* (i.e. decomposing to a mixture of regions with the two stable compositions, one on each side of the miscibility gap).

This option makes it possible to automatically check internal stability of both stable and unstable phases in all subsequent single-point equilibrium and MAP/STEP calculations. It can find out if any phase is subject to spinodal decomposition during the subsequent calculations. If there is an unstable phase located inside a miscibility gap in a calculation, it gives a warning so that you suspend the unstable phase, or use FORCED automatic start values for phase constituents (i.e. *SET_ALL_START_VALUES* on page 162 → `FORCE` command-sequence), or create other composition sets; you can also ignore the warning message if you know that the unstable phase is not formed in the current calculations.

STEP_AND_MAP

This is the default and is used to determine how to perform Global Minimization test and how to handle initial equilibrium points.

T-ZERO_TEMPERATURE

This calculates the temperature when two specific phases have the same Gibbs energy, i.e. the so-called T0 temperature. You must calculate an equilibrium state at an estimated temperature before performing this advanced-option calculation; however, it is unnecessary to obtain an equilibrium state in which either one or both of the target phases is stable.

TOGGLE_ALTERNATE_MODE

Toggle the ALTERNATE mode for experimental equilibrium calculation between DEFAULT, ALWAYS and

NEVER, during data-assessments using the PARROT optimization.

ADVANCED_OPTIONS

Also see *ADVANCED_OPTIONS Descriptions* on page 120.

Syntax	ADVANCED_OPTIONS
Prompt	`WHICH OPTION? /STEP_AND_MAP/: <OPTION>`
Options	**Description and Information**
NONE	Take away any set advanced option in all subsequent single-point equilibrium and MAP/STEP calculations. If there is any convergence problem in finding a stable solution, at any stage of an advanced option calculation conducted by an ADVANCED_OPTIONS command-sequence, these messages display: `Convergence problems, increasing smallest site-fraction from 1.00E-30 to hardware precision 2.00E-14. You can restore using SET-NUMERICAL-LIMITS` The smallest site fraction in the current POLY3 workspace is automatically increased from the default value 1.00E-30 to the hardware-dependent precision (under Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current TC run, you can use the POLY command *SET_NUMERICAL_LIMITS* on page 168 to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

STEP_AND_MAP (the default)	Settings for MAP and STEP:
	
	`GLOBAL TEST INTERVAL /0/: <INTEGER NUMBER>`
	
	Settings for MAP:
	
	`AUTOMATICALLY ADD INITIAL EQUILIBRIA /Y/: <Y OR N>`
	`NUMBER OF MESH ALONG AN AXIS /3/: <INTEGER NUMBER>`
	`USE INSIDE MESHING POINTS /N/: <Y OR N>`
	`GLOBAL TEST INTERVAL /0/: <INTEGER NUMBER>`
	
	The integer number determines how often the Global Minimization should be used during STEP_WITH_OPTIONS and MAP calculations. If it is set to 0 (zero), the recommended global test interval is used: i.e. every tenth step and at each phase change during STEP calculations, and only at node points during MAP calculations. Any other positive integer number, n, suggests it performs a Global Minimization test at every *n*-th step during STEP and MAP calculations. Of course, the Global Minimization test is always carried out at a phase change or a node

Syntax	ADVANCED_OPTIONS

	point. The Thermo-Calc software is installed with 0 (zero) as the default value, but such a configuration for the default value can be changed by using SET_TC_ OPTIONS in the SYS module.
	
	`AUTOMATICALLY ADD INITIAL EQUILIBRIA /Y/: <Y OR N>`
	
	When *ADD_INITIAL_EQUILIBRIUM* on page 119 is not used before MAP, a mesh of initial equilibrium points are added before the mapping itself takes place. This mesh is only available when two (2) axes are already defined. The default on start-up may be changed by SET_TC_OPTIONS in the SYS monitor.
	
	`NUMBER OF MESH ALONG AN AXIS /3/: <INTEGER NUMBER>`
	
	An integer for how many intervals of initial equilibrium points to be added along an edge in the diagram. For example, if set to 2 (two), initial equilibrium points are added at the beginning, centre, and end of each axis-variable forming 2 intervals.
	
	`USE INSIDE MESHING POINTS /N/: <Y OR N>`
	
	If `N` the INITIAL_EQUILIBRIUM mesh consists of initial equilibrium points added only along the edge/border of the diagram defined by the axis-variables. If `Y`, the initial equilibrium points added are also added inside edge.

EQUILIBRIUM_ CALCUL	The following prompts affect the ways how the POLY optimization does the Ordinary POLY Minimization to reach the minimum of an equilibrium state.
	
	Settings for the minimization of an equilibria:
	
	`FORCE POSITIVE DEFINITE PHASE HESSIAN /Y/: <Y OR N>`
	`CONTROL STEPSIZE DURING MINIMIZATION /Y/: <Y OR N>`
	`FORCE POSITIVE DEFINITE PHASE HESSIAN /Y/: <Y OR N>`
	
	Choose `Y` or `N` to determine how to reach the minimum of an equilibrium state in a normal POLY optimization procedure. This is related to the special quantity, the phase stability function QF (phase), for all kinds of phases in an equilibrium state [A phase stability function for a phase is negative when the phase composition is inside a spinodal, and positive everywhere else].
	
	If the composition of a solution phase is inside its spinodal, the enforcement of positive definite eigenvalues of a Hessian matrix (i.e. by answering Y) makes the step be taken towards the minima rather than the local maxima. Furthermore, if an eigenvalue (for a phase) of the Hessian matrix is near 0.0, the step-size is large and the answering `Y` sets a lower limit to this eigenvalue, and reduces the step-size and improves the convergence. The POLY command-sequence SHOW_VALUE QF(phase) shows the size of the lowest eigenvalue of a phase in an equilibrium state.

Syntax	ADVANCED_OPTIONS
	By answering `Y` it might help the Ordinary POLY Minimization to converge, if there is such a problem, especially if QF(phase)=0 for phases. By answering `N` the Ordinary POLY Minimization routine is applied. For each phase in a defined system, the molar Gibbs energy of the phase is a function of the temperature-pressure condition and its composition: $G_m(X) = G_m(T, P; y_1, y_2, ..., y_n)$ Compute a Hessian matrix as which describes the curvature of the Gibbs energy curve of this phase at the defined composition X: $\partial^2 G_m / \partial y_i \partial y_j$ Diagonalise this matrix and call the Eigenvalues as e1,e2,...,en . The QF(phase) quantity for this phase is the lowest of these eigenvalues divided by the same values for a corresponding ideal (stoichiometric) phase: $QF(ph) = \min(\{e_1, e_2, ..., e_n\}) / \min(\{se_1, se_2, ..., se_n\})$ Here, se1,se2,...,sen are the eigenvalues for the ideal phase. Therefore, for an ideal (stoichiometric) phase, the QF(phase) should always be 1.0. `CONTROL STEPSIZE DURING MINIMIZATION /Y/: <Y OR N>` Choose `Y` or `N` to determine whether to introduce a control of step-size while reaching the minimum of an equilibrium state in a normal POLY optimization procedure. By answering `Y` it might help the POLY optimization to converge, especially if when some site fractions are less than 1E-4.

`GLOBAL_ MINIMIZATION`	The following prompts affect how the Global Minimization is done. Settings for global minimization:
	
	`USE GLOBAL MINIMIZATION AS MUCH AS POSSIBLE /Y/: <Y OR N>`
	
	`USE GLOBAL MINIMIZATION FOR TEST ONLY? /N/: <Y OR N>`
	
	`MAXIMUM NUMBER OF GRIDPOINTS PER PHASE /2000/: <INTEGER NUMBER>`
	
	`USE GLOBAL MINIMIZATION AS MUCH AS POSSIBLE /Y/: <Y OR N>`
	
	Choose `Y` or `N` to decide if using the Global Minimization Technique from start of calculation or not. The default value on start-up may be changed by SET_TC_ OPTIONS in the SYS module.
	
	If `Y` the calculation is done when possible (depending on the condition settings), and a Global Minimization test is always performed when an equilibrium is

Syntax	ADVANCED_OPTIONS

	reached. This costs more computer time but the calculations are more accurate.
	
	If `N` the calculation is not used to reach the equilibrium state. `Use global minimization for test only?` determines if the Global Minimization test is made against the calculated equilibrium states obtained by the Ordinary POLY Minimization calculation.
	
	`USE GLOBAL MINIMIZATION FOR TEST ONLY? /N/: <Y OR N>`
	
	If `Y` a calculated equilibrium state obtained by the Ordinary POLY Minimization calculation is tested against the Global Minimization Technique after it is reached; and if found it as unstable, there is an error message.
	
	If `N` the Global Minimization Technique is never tested for or done for all sequential calculations throughout the current TCC run. This implies that Global Minimization Technique is permanently turned off, and that consequently only the Ordinary POLY Minimization routine is used.
	
	`MAXIMUM NUMBER OF GRIDPOINTS PER PHASE /2000/: <INTEGER NUMBER>`
	
	Specify the maximum number of grid points that are computed for each of the phases in the currently defined system, during the calculations enforced by the global minimization. More grid points give a higher robustness and take more computation time. The total number of computed grid points in all phases is limited to `2E6`.

`IGNORE_COMPOSI_SET_ORDER`	`IGNORE COMPOSITION SET ORDER /N/`

`NEW_COMPOSITION_SET`	The following prompts are to specify the additional composition set(s) to handle possible miscibility gap(s) of a certain solution phase:
	
	`PHASE WITH MISCIBILITY GAP: <NAME OF A PHASE>`
	
	Specify the name of the phase with miscibility gap.
	
	`NEW HIGHEST COMPOSITION SET NUMBER /2/: <#>`
	
	The default value is usually one higher than the current value. Each phase has initially one composition set. If a lower value is given, composition sets are deleted. You cannot take away the first composition set.
	
	A message displays to show that you need to specify the composition for the composition set # (2,3,…).
	
	`MAJOR CONSTITUENT(S) FOR SUBLATTICE 1: /XX/: <YY>`
	
	The major constituents (`YY`) in each sublattice can be given. This may simplify giving start values when calculating the equilibrium as phases with miscibility gaps

Syntax	ADVANCED_OPTIONS
	should have different major constituents for each composition set. This prompt is repeated for each sublattice in the phase, sometimes even for all sublattices in the first composition set if such major constituents have not been specified.

MAJOR_ CONSTITUENT	The following prompts are to specify the major constituent(s) on each of the sublattice sites for a composition set of a certain solution phase:
	
	`PHASE NAME: <NAME OF A PHASE>`
	
	Specify the name of the solution phase with a new set of major constituents.
	
	`COMPOSITION SET NUMBER /1/: <#>`
	
	The default value for the composition set number (#) is usually /1/ as the other composition sets are given major constituents when creating them. Each phase has initially one composition set.
	
	`MAJOR CONSTITUENT(S) FOR SUBLATTICE 1: /XX/: <YY>`
	
	The major constituents (`YY`) on each sublattice of a solution phase can be given. This may simplify giving start values when calculating the equilibrium as phases with miscibility gaps should have different major constituents for each composition set.
	
	This question is repeated for each sublattice in the phase.

PHASE_ADDITION	The following prompts are to specify the additional energy term (always being a constant) of a given phase.
	
	`PHASE NAME: <NAME OF A PHASE>`
	
	Specify the name of the (stoichiometric or solution) phase with the addition.
	
	`ADDITION TO G PER MOLE FORMULA UNIT: <XXXXX>`
	
	The value (`xxxxx`) given is added to the Gibbs energy of the (stoichiometric or solution) phase. It can represent a nucleation barrier, surface tension, elastic energy or whatsoever.
	
	It is not composition-, temperature- or pressure-dependent.

LIST_PHASE_ ADDITION	No additional prompt.
PRESENT_PHASE	Specify the name of the present phase a the name of the phase that should be present at all calculated equilibria:

Syntax	ADVANCED_OPTIONS
	PRESENT PHASE: PHASE NAME: <NAME OF A PHASE>
T-ZERO TEMPERATURE	NAME OF FIRST PHASE: <PHASE A> NAME OF SECOND PHASE: <PHASE B> The names of phases A and B must be given, for which the T0 temperature (where the Gibbs energies are equal) is to be calculated. If the T0 temperature between the two specified phases is successfully calculated, a message displays, e.g. `THE T0 TEMPERATURE IS 840.82 K. NOTE: LIST-EQUILIBRIUM IS NOT RELEVANT` The first message shows the calculated T0 temperature between the two specified phases. The second message indicates that after this option calculation with LIST_ EQUILIBRIUM is irrelevant and does not list the equilibrium for the system at the T0-temperature.
PARAEQUILIBRIUM	NAME OF FIRST PHASE: <PHASE A> NAME OF SECOND PHASE: <PHASE B> The names of the two target phases A and B, between which the paraequilibrium state is to be calculated, must be entered subsequently or on the same (first) line at once then separated by an empty space, e.g. `FCC#1 BCC` or `FCC#2 M23C6`. You need to understand what you are dealing with in terms of calculating a paraequilibrium state between the two specified phases. Specifically, there are four distinguished cases to understand: (1) both chosen phases must have similar interstitial/vacancy sublattices where the fast-diffusion interstitial component(s) occupy; (2) the choice on the target phase pair must be reasonable for the defined system and specified initial overall composition; (3) both target phases should have phase constitution definitions that cover all the defined substitutional and interstitial components of the current alloy system; or (4) it is impossible to calculate the paraequilibrium state between the target phase pairs with given interstitial component(s) in the currently defined system. FAST DIFFUSING COMPONENT: /C/: <INTERSTITIAL COMPONENT(S)> FAST DIFFUSING COMPONENT: /NONE/: <INTERSTITIAL COMPONENT(S)> The name(s) of the fast-diffusing component(s) (C as the default single

Syntax	ADVANCED_OPTIONS

	component) must be given at the above prompts subsequently or at the same (first) prompt. It is possible to specify more than one interstitial component as fast diffusion species.
	
	Note the following:
	
	Such specified interstitial component(s) must be appropriately defined according to the phase constitution definitions of the two selected phases: these must be located on the interstitial/vacancy sublattices in both chosen phases;
	
	If there is only one fast-diffusing component which is carbon, press the <Enter> key to accept the default input at the first prompt; if the single fast-diffusing component is another element (e.g. N), type its name at the first prompt;
	
	If there are two or more fast-diffusing components (e.g. `C` and `N`), type their names at the above prompts subsequently or at the same (first) prompt (separated by an empty space, such as `C N`);
	
	To finish the input of fast-diffusing elements, accept NONE at a repeated prompt, i.e. by pressing <Enter> key to start the paraequilibrium point calculation;
	
	If NONE or a non-existing component name is typed at the first prompt, it means no back diffusion is to be considered, and the para-equilibrium calculation is thus cancelled entirely.
	
	If the paraequilibrium state between the two specified phases is successfully calculated, the messages displays e.g.
	
	`NP(FCC) = 0.3586 with U-fractions C = 2.71821E-02  N = 4.1548129E-03`
	
	`NP(BCC) = 0.6414 with U-fractions C = 7.10061E-04  N = 2.3781027E-04`
	
	`All other compositions the same in both phases`
	
	`Note: LIST-EQUILIBRIUM is not relevant`
	
	The first and second lines list the phase amounts expressed in mole-percent [NP (phase)] and the contents of the interstitial components C and N in a specific phase expressed in the so-called u-fractions [u-f(phase,C) and u-f(phase,N)], for the phase A (in this case as FCC) and phase B (in this case as BCC), respectively. The third line states that the compositions of the matrix component and all the remaining compositions (regarding substitutional components) in both the target phase A and target phase B are the same at the current paraequilibrium state, while these are not shown on screen. The last line indicates that after this advanced-option calculation the LIST_EQUILIBRIUM command is irrelevant and does not list the paraequilibrium state for the system at the current condition.

Syntax	ADVANCED_OPTIONS

	However, if the single-point calculation of the paraequilibrium state between the two specified phases has failed, these messages display:
	`*** ERROR 4 IN NS01AD`
	`*** NUMERICAL ERROR`
	This implies that the chosen target phase pair may be unreasonable for the defined alloy system or for the defined initial overall composition, or one or both phases may have inappropriate phase constitution definitions regarding the specified interstitial component(s). Then, you must either modify the settings of initial overall composition or specify the reasonable target phase pair with an appropriate choice of the fast diffusion interstitials in the defined alloy system.

STABILITY_CHECK	`STABILITY CHECK ON? /Y/: <Y OR N>`
	The default is `Y` to switch on the automatic stability check during all subsequent single-point equilibrium and MAP/STEP calculations. By answering `N`, there is no stability check in various calculations.
	`CHECK ALSO FOR UNSTABLE PHASES? /Y/: <Y OR N>`
	If the automatic stability-check option is switched on choose to also check the stability for unstable phases. The default is `Y` and if an unstable phase is found to be located in a miscibility gas during a subsequent single-point equilibrium or MAP/STEP calculation, a warning message informs you to selectively make adjustments in the calculation settings [e.g. suspending the unstable phase, or using FORCED automatic start values for phase constituents by the S_A_S_V F command-sequence, or creating other composition sets, etc.]. By answering `N` the stability check is enforced only to stable phases in the system.

TOGGLE_ ALTERNATE_MODE	`SET ALTERNATE TOGGLE TO DEFAULT, ALWAYS, OR NEVER?`
	`SET GLOBAL TOGGLE TO DEFAULT, ALWAYS, OR NEVER?`
	`DEFAULT` means that the experiment points are calculated according to the Alternate Technique depending on the SET_ALTERNATE command.
	`ALWAYS` means the experiment points are always calculated according to the Alternate Technique even if the Alternate Mode is switched off in the PARROT module.
	`NEVER` means the experiment points are calculated as normal equilibria even if you SET_ALTERNATE in the PARROT module.

SHOW_FOR_T=	`TEMPERATURE (K) /298.15/: <TEMPERATURE_IN_K>` Specify the new temperature condition (in K) under which the values of some specific state variable(s) or derived/partial variable(s) or defined symbol(s) for

Syntax	ADVANCED_OPTIONS
	various thermodynamic properties of the entire system, of components, or of phases) in the currently-defined system (being in a *frozen* state) is shown on screen. `STATE VARIABLE OR SYMBOL /VM/: <STATE VARIABLE OR SYMBOL NAME(S)>` Specify the name(s) of the desired state variable(s) or derived/partial variable(s) or defined symbol(s) for various thermodynamic properties of the entire system, of components, or of phases) in the currently-defined system. More than one state variable or symbol of interest can be simultaneously specified on the same line. For example, you can choose to show the values of VM or GM (i.e. molar volume or molar Gibbs energy of the entire system), or of VM(*) or HM(*).T (i.e. molar volumes or isobaric heat capacity of all phases), or of ACR(*), DGM(*) and LNACR (*,FCC) [i.e. activities of all system components, driving forces for all phases, and activities (in logarithm) of all system components in the FCC solution phase], under the new temperature condition (being in a frozen state).
`OUTPUT FILE FOR SHOW`	A window opens with a default name for the file `tc_show.dat`. Choose a location to save the file and click Open.

AMEND_STORED_EQUILIBRIA

This command gives information about the calculated blocks (and phase regions included in blocks) after the STEP or MAP calculation(s). It allows you to list all or part of the calculation results, to suspend all or parts of the calculation results that are redundant or where metastable equilibria are calculated, and to restore all or parts of the calculation results (if having been suspended by another AMEND_STORED_EQUILIBRIA command).

The workspace for storing equilibria may overflow during stepping or mapping, and is then written to a file as blocks. Each block usually contains one or more ranges of equilibrium regions.

Use one these options:

- `L` to list the calculated equilibria (all or a specified block)
- `S` to suspend everything (all blocks and their regions)
- `Q` to suspend each set of equilibria individually (specified blocks and/or regions)
- R to restore everything (all blocks and their regions)

Syntax	AMEND_STORED_EQUILIBRIA
Prompts	`NAME: <NAME OF A DEFINED TABLE>` List (`L`) the calculated equilibria, suspend everything (`S`) or suspend each set of equilibria individually (`Q`), or restore everything (`R`).
	`BLOCK /*/: <BLOCK NUMBER>` Specify a block number in the option `L`, `S`, `Q` or `R`. Or include all the blocks in the amending option, by accepting the wildcard * (press <Enter>). For `L`, a block (if a block number is specified) or all the blocks (if the wildcard * is accepted/used), with the regions and the equilibrium details, that are calculated during stepping or mapping is listed on screen or in a textual file. For `Q`, each ranger in a block (if a block number is specified) or in all blocks (if the wildcard * is accepted/used) then choose:

	`S(USPEND) K(EEP) /K/: <S OR K>`
	
	For `S` or `R`, if a block is specified, you are asked what region in the block should be suspended or restored. However, if the wildcard * is accepted/used, you are asked the following:
	
	`REALLY SUSPEND ALL /N/: <Y OR N>`
	`REALLY RESTORE ALL /N/: <Y OR N>`
	
	If the answer is `N`, the program asks for which block(s) and which region(s) in a certain block to execute the `S` or `R` action.
	
	`REALLY SUSPEND ALL /N/: <Y OR N>`
	
	`Y` suspends everything.
	
	`REALLY RESTORE ALL /N/: <Y OR N>`
	
	`Y` restore everything.

	`RANGE: <RANGE(S) OF REGION>` Specify one or more ranges to be suspended (the `S` option) or restored (the `R` option). The wildcard * suspends or restores all ranges in the specified block. In order to know the ranges the `LIST` option must first be used.
	`S(USPEND) K(EEP) /K/: <S OR K>` The Query suspend option (`Q`) asks for each region in a certain block if it should be suspended or kept. Suspended regions are not included on sequentially generated plots.
	`OUTPUT FILE: /SCREEN/: <FILE NAME>`

Syntax	AMEND_STORED_EQUILIBRIA
	This prompt is only for listing (the `L` option), and after it the command is terminated. The file name must be given here; or accept the default `SCREEN` (terminal) by pressing <Enter>. A list of stored equilibria as various blocks (with all their ranges) are shown out on screen or the file.

CHANGE_STATUS

In the POLY module, set the status for components, species and phases in the defined system for all the sequential calculations (single-point, stepping and mapping) in equilibrium or local/partial equilibrium state. Each component, species and phase has a status. The default status is `ENTERED`.

The most important use is to calculate metastable equilibria and metastable phase diagrams by setting some phases (that would otherwise be stable) to the SUSPENDED or DORMANT phase-status. Another important applications is to calculate paraequilibria by setting some components to the SPECIAL component-status.

For a component and for a species, the status can be one of the following:

- `ENTERED` - the component(s) or species are included in the calculation. This is the default status.
- `SUSPENDED` - the component(s) or species are not considered in the calculation.
- `SPECIAL` means - specified component(s) are not included in summations for mole or mass fractions. It only works for component(s).

Only component(s) can have the status SPECIAL, which implies that these are not included in summations for mole or mass fractions.

For example, for the *u-fractions* or other normalized fractions, when one or more of the components are excluded from the summation, you must specify which component(s) should be excluded from the calculation of mole or mass fraction. This component status is particularly useful when calculating paraequilibrium states. Such component(s) are normally interstitial component, and must have the status SPECIAL. This is assigned by the CHANGE_STATUS command.

Syntax	CHANGE_STATUS
Prompts	`FOR PHASES, SPECIES OR COMPONENTS? /PHASES/: <KEYWORD>` `Keyword` = phase or species or components
	`PHASE NAME(S): <NAME(S) OF THE PHASE(S)>` For `phase` as the keyword, the names of the phases that have their status changed must be given (all on one line). A comma or space must be used as separator. The status to be

Syntax	CHANGE_STATUS
	assigned to the phases can also be given on the same line if preceded with an equal sign =. An asterisk, *, can be used to denote all phases. The special notations `*S`, i.e. a wildcard * directly followed by an `S`, sign, means all suspended phases. In the same way, `*D` means all dormant phases, and `*E` means all entered phases.
	`NAME(S): <NAME(S) OF THE SPECIE(S) OR COMPONENT(S)>` For `species` or `component` as the keyword, the names of the species or components that have their status changed must be given (all on one line). A comma or space must be used as separator. Similarly to the case of `phase` as the keyword, the status to be assigned to the species or components can also be given on the same line if preceded with an equal sign =. An asterisk, *, can be used to denote all species or components. The special notations `*S`, i.e. a wildcard * directly followed by an `S`, sign, means all suspended species or components. In the same way, `*E` means all entered species or components.
	`STATUS /ENTERED/: <NEW STATUS>` The `new` status to be assigned must be given. For species, the values `ENTERED` or `SUSPENDED` can be used. For components, the status `ENTERED`, `SUSPENDED` or `SPECIAL` can be given. SPECIAL means that this component is excluded from sums for mole fractions and mass fractions, which is useful when calculating the *ufractions* or other normalized fractions of system components. For phases, the status `ENTERED`, `SUSPENDED`, `DORMANT` or `FIXED` can be given. DORMANT means the same as suspended but the driving force is calculated. FIXED means that it is a condition that the phase is stable at a certain amount. For example, for the ufractions, when one or more of the components are excluded from the summation, you must specify which component should be excluded from the calculation of the mole fraction. This component must have the status SPECIAL. This is assigned by the CHANGE_STATUS command: `Change_Status comp C=special`.
	`START VALUE, NUMBER OF MOLES /0/: <INITIAL AMOUNT>` For ENTERED phases, an `initial amount` of the phase can be given. Normally, 0 is given if the phase is not likely to be stable, and 0.5 or 1 or any positive number if the

Syntax	CHANGE_STATUS
	phase could be stable, but such an initial amount is only used as the rough starting estimation in the equilibrium calculations.

	`NUMBER OF MOLES /0/: <EQUILIBRIUM AMOUNT>`
	
	For FIXED phases, the `equilibrium amount` of the phase [always using an initial estimation being the `NPF(phase)` value which it is the normalized mole number of components (per mole formula unit) of the specific status-fixed phase] must be given. If the equilibrium amount is zero, then the phase is at its stability limit.
	
	Special attentions should be paid when specifying a FIXED phase status in equilibrium calculations (for single points, stepping or mapping calculations), as described below:
	
	The phase amount variables, NP(phase), BP(phase) and VP(phase), as well as all their M/W/V-suffixed quantities, should not be used as conditions. Instead, use the CHANGE_ STATUS command to set a relevant condition, e.g. `CHANGE_STATUS phase <phase>=fix <amount>` where the fixed <amount> is roughly the same as the F-suffixed quantity `NPF(phase)`.
	
	The `NPF(phase)` quantity is the normalized mole number of components (per mole formula unit) of the specific phase in the defined system, which unlike other `F`-suffixed state variables [e.g. GF(phase), HF(phase) and DGF(phase)] cannot be directly applied in any POLY command, implying that it cannot be directly evaluated or listed/shown. If intended to shown such a normalized phase amount value in an equilibrium state, you should use a properly-entered symbol (function or variable), for example: `NPFabc = NP (abc)/NA` or `NPFabc = NPM(abc)/NA*N`. N is the total system size (in mole). The NA value is a quantity that is phase-dependent (and sometimes also equilibrium-dependent for ionic solution phases), and is the total atomic number in a mole-formula-unit of the specific phase abc (excluding interstitial component and, of course, vacancy). For example, the SIGMA, FCC, BCC and LIQUID phases (among others) in a defined Fe-Cr-Ni-C-N-O system (retrieved from a specific database) may be modelled by certain models, and their NA values must be evaluated in different ways, as described below:
	
	`LIQUID (C,Cr,CrO3/2,Fe,FeO,FEO3/2,N,Ni,NiO)1 ➔NA = 1`
	
	`FCC_A1 (Cr,Fe,Ni)1(Va,C,N,O)1 ➔NA = 1`
	
	`BCC_A2 (Cr,Fe,Ni)1(Va,C,N,O)3 ➔NA = 1`
	
	`SIGMA (Fe,Ni)8(Cr)4(Cr,Fe,Ni)18 ➔NA = 30`
	
	If in the same Fe-Cr-Ni-C-N-O system the liquid solution phase is modelled by the Two-Sublattice Ionic Liquid Model, i.e.:
	
	`IONIC_LIQ  (Cr+3,Fe+2,Ni+2)p(VA,C,N,O-2,FEO3/2)q,7`
	
	then the evaluation of its NA value becomes even more complicated:
	
	`NA = p + q*yC2 + q*yN2 + q*yO2□2 + q*yFeO2 3/ 2`

Syntax	CHANGE_STATUS
	where the stoichiometric coefficients *p* and *q* are also dependent upon the real equilibrium state (rather than having fixed values in the system). Similar situations occur for other (solid) phases which are described by a multiple sublattice model with ionic constituents, such as SPINEL and HALITE phases in some databases. There is no strange thing when using a zero value [i.e. `0`] in a FIXED phase-status, since it means the specified phase is stable in equilibrium state but has a zero-amount of mass in the equilibrium calculations; in other words, on a phase diagram, the specific phase is on a zero-fraction line (ZFL), i.e. it starts becoming stable on one side of a corresponding phase-boundary line or unstable on the other side of the same boundary. It is often and efficient to do so when calculating e.g. solidus equilibrium states. However, when a non-zero value [it must always be positive; e.g. 1 or 0.5 or 0.3 or 1.5] is to be specified in a FIXED phase-status, it is unnecessarily the exactly same stable amount of the specific FIXED-status phase in a calculated equilibrium state any longer; instead, the <equilibrium amount> value is the NPF(phase) value that is only roughly used as the estimated starting-value of the FIXED-status phase in the equilibrium calculations. Therefore, a FIXED-status for a liquid phase being unity does not necessarily imply that it is a liquidus equilibrium state (where the liquid phase is in equilibrium with some solid phases but the liquid phase takes all the mass in the defined system). A unity value for setting the liquid phase status in calculating liquidus equilibrium state can only be used when the liquid mixture phase is predefined as a single-sublattice solution phase (such as metallic liquid phase in multicomponent alloy systems) and the total system size as one mole (i.e. N=1). When a phase is described by a solution model in which two or more sublattices are considered and these sublattice sites may also have different stoichiometric coefficients [meaning that the mixture phase could have more than one atom in formula [NA>1; see some examples above], the unity value should not be used when setting the FIXED status for the phase; instead, you should use an appropriate value that ranges from 0 to a NPF (phase) value that equals to or is smaller than 1/NA (if the total system size N=1) or 1/NA*N (if N differs from unity). For this reason, if a multicomponent system bears an IONIC_LIQUID phase that is described by the Two-Sublattice Ionic Liquid Model (or any other multiple-sublattice ionic solution phases), it is difficult to use a proper `NPF(ION_LIQ)` value in setting its FIXED phase-status, because that should be less than (or equal to) the complex value of $N/[p + q*y_{C2} + q*y_{N}2 + q*{}^{*}y_{O}22 + q*{}^{*}y_{FeO}2\ 3/2]$.

Examples

For example, to obtain the metallic fraction in a system with carbon as an interstitial component, you can set the component status for carbon as `SPECIAL`:

```
Change_status comp C=special
```

The `SUSPENDED` status for components and species does not always work as expected.

For a phase, it may have one of these statues:

- `ENTERED` - the phase(s) are included in the equilibrium calculations and these are stable if that minimizes the total Gibbs energy in the defined system. This is the default status for all phases already retrieved from the chosen database(s). An ENTERED phase-status is always associated with an initially-estimated amount [in mole number; normally, as 0 if the phase is not likely to be stable, and as 0.5 or 1 or any positive number if the phase could be stable] but it is only used as the rough starting value in the equilibrium calculations.
- `SUSPENDED` - the phase(s) are not considered in the equilibrium calculations.
- `DORMANT` - the phase(s) are not considered in the equilibrium calculations but their driving forces for precipitation are calculated.
- `FIXED` - it is an equilibrium condition that the status-fixed phase must be stable, and be in equilibrium at a specified amount [always using an initial estimation being the `NPF (phase)` value which it is the normalized mole number of components (per mole formula unit) of the specific status-fixed phase]. See more descriptions at the end of this command.

COMPUTE_EQUILIBRIUM

The full equilibrium state is calculated for the given set of conditions. The Global Minimization Technique is by default enforced in this command (C_E), while it can be disabled temporarily (for the current single-point equilibrium calculation) if using C_E – or C_E * command-combination, or permanently (for all the sub sequential single-point calculations or stepping/mapping calculations within the current TCC run) if having decisively switched it off by a user (or possibly in some special modules) through changing the minimization option using the *ADVANCED_OPTIONS* on page 125 → `MINIMIZATION_OPTION` command-sequence.

Syntax	COMPUTE_EQUILIBRIUM
Prompt	The C_E command can be used, in order to enforce the ordinary POLY minimization routines in an equilibrium calculation; this is because of that the ordinary C_E command is now associated with the Global Minimization Technique, and only after the Global Minimization If technique is permanently switched off the C_E command makes no difference from the C_E – command-combination. Only certain types of equilibrium conditions [e.g. `T`, `P`, `N`, `N(<component>)`, `X(<component>)`, `B`, `B(<component>)`, and `W(<component>)`] are fully supported in the Global Minimization mode (called *Direct Global Minimization*); and when other types of equilibrium conditions are used, after the initial POLY, a Global Minimization test and corrections are performed until the lowest minimum is found (called *Indirect Global Minimization*). If there is any problem with convergence, you may try the C_E * command-combination. The character * enforces the command to use an advanced technique to obtain a complex equilibrium. However, after a successful C_E *calculation, you may repeat the C_E command and can check the status of phases, species or components (with *LIST_STATUS* on page 157 → CPS command-sequence) and equilibrium conditions (with *LIST_CONDITIONS* on page 155) and list out the calculation results (with *LIST_EQUILIBRIUM* on page 156), because such actions may tell you how to further modify various settings for your current calculation. This command-combination is not that useful anymore, because the Global Minimization Technique that is always associated with the C_E command is even more powerful and more precise in finding the most-stable equilibrium state in a complex heterogeneous interaction system; therefore, the C_E * command-combination is functional and can be used only after the Global Minimization mode has already been disabled temporarily or permanently. Some phases that are not stable in the current equilibrium state may not have their most favourable composition after this command, and thus their driving forces may not be correct. You can force the program to correctly calculate the driving forces of metastable phases, by giving repeated C_E commands until the number of iterations (that is shown on screen after this command) is reduced to 2. Also see the POLY command *SET_NUMERICAL_LIMITS* on page 168 which can set the approximate driving force for metastable phases option on or off in all the subsequent POLY calculations within the current TCC run. If an equilibrium state for the defined system is not found, an error message is given.

Syntax	COMPUTE_EQUILIBRIUM
	You can try repeating this command a few times, or change some of settings for the numerical limits, for starting variables and starting values, for starting constitutions of certain phases and for reference states of certain components, or to verify some of the defined conditions.

COMPUTE_TRANSITION

This command is a combination of the CHANGE_STATUS, SET_CONDITION and COMPUTE_EQUILIBRIUM commands. It allows a direct calculation when a new phase may form by varying one of the already-set conditions. It can be used only after at least one equilibrium is calculated successfully; otherwise, you are informed on the necessity of first making an equilibrium calculation to find out the stable phases under the current conditions.

When this command is used, the program calls the command *CHANGE_STATUS* on page 59 to temporarily change the phase status of a specified phase as FIXED at the *zero* amount, and at the same time to temporarily release one of the existing equilibrium conditions (which is chosen by you). The program calculates a new equilibrium in which that specific phase is stable but its equilibrium amount in the system is zero. The released condition is then assigned with a calculated value that ensures the calculated equilibrium. Afterwards, the program automatically changes the phase status of that specific phase back to ENTERED, and resets the temporarily released condition as one of the conditions and assigns it with the value that is calculated to ensure the zero-amount formation of that specific phase.

This command is useful to find melting temperature, boiling temperature, or solubility limits, and generally when you want to set the most optimal conditions for calculating an equilibrium where a specific phase becomes stable. It can also be used when you want to know exactly how far away the defined conditions are from the value that can ensure a zero-amount of a specific phase in the system when other conditions remain the same.

After a successful COMPUTE_TRANSITION calculation, you can issue a *COMPUTE_EQUILIBRIUM* on page 139 calculation to assure the calculated transitional equilibrium is a really stable one, and can also use *LIST_EQUILIBRIUM* on page 156 to see the details of transitional equilibrium state.

Syntax	COMPUTE_TRANSITION

Prompts	`PHASE TO FORM: <PHASE NAME>`
	
	A new `phase name`, e.g. `BCC` that is expected to form, is specified here. This changes the status of this new phase to be `FIXED` as `0` amount, and the program shows the information such as:
	
	`You must remove one of the these conditions`
	
	`P=100000, T=800, N=1, X(FE)=.5    DEGREE OF FREEDOM 0`
	
	If the key word `ANY` is used when prompted for `Phase to form` (instead of a specific phase name), it is possible to find out any new phase to be formed, in a given varying direction sign and at an estimated change of the released condition: a negative sign means at a lower value of the released condition any new phase is to be found, and a positive sign at a higher value; an estimated change of the released condition implies where any new phase is expected (but it is only estimated value, so any value within its reasonable scale would be enough). Such calculations can be repeated if required. This feature is useful to find out all possible phase transformations along a certain released condition.

	`GIVE THE STATE VARIABLE TO BE REMOVED /T/: <ONE CONDITION>` `One condition` must be removed, in order to calculate the transition equilibrium where the specified (or any) new phase to be formed at a calculated value of this released variable.

	Therefore, the message may display (after a successful calculation) if, for example, `X (Fe)` is entered:
	
	`To form BCC the condition is set to X(FE)=.48605791769`
	
	This calculated value is assigned as the parameter of that removed condition, in this case, the `X(FE)` variable. If the LIST_CONDITIONS command is typed this message displays:
	
	`P=100000, T=800, N=1, X(FE)=4.86057918E-1`
	
	`DEGREES OF FREEDOM 0`
	
	If the key word `ANY` (instead of a specific phase) is given as the phase name when it is prompted for Phase to form, the line is prompted for a given varying direction sign and an estimated change of the released condition before the calculation of transition equilibrium:

	`ESTIMATED CHANGE (WITH SIGN) /1/: <+/-#>` A given varying direction sign and an estimated change of the released condition, in this case as `X(FE)`, must be given here: a negative sign means at a lower value of the released condition any new phase is to be found, and a positive sign at a higher value; an estimated change of the released condition implies where any new phase is expected (but

Syntax	COMPUTE_TRANSITION

	it is only estimated value, so any value within its reasonable scale would be enough). For example, if a combination of -0.02 is input, this message may display (after a successful calculation):
	`To form BCC_A2#1 the condition is set to X(FE)=.493708756187`
	This calculated value is then assigned as the parameter of that removed condition, in this case, the `X(FE)` variable. The message is shown if the LIST_CONDITIONS command is typed:
	`P=100000, T=800, N=1, X(FE)=4.93708756E-1`
	`DEGREES OF FREEDOM 0`

CREATE_NEW_EQUILIBRIUM

During data-assessments using the PARROT/ED_EXP modules, you can, in the POLY module, create several equilibria with different sets of conditions and phases (but normally with the same set of components). By default, there is one equilibrium. To keep the set of conditions and phase for this equilibrium, create another one using this command, and use another set of conditions for that. Two equilibria may be useful to calculate easily the enthalpy difference between two states. In the PARROT module, the experimental information is stored as a sequence of equilibria.

Syntax	CREATE_NEW_EQUILIBRIUM
Prompts	`EQUILIBRIUM NUMBER /2/: <A NEW EQUILIBRIUM NUMBER>` Each equilibrium in the POLY3 workspace is identified by a unique integer number. Such an equilibrium number can be recalled with *SELECT_ EQUILIBRIUM* on page 162.
	`INITIATION CODE /2/:` When an equilibrium is created, you can choose to `ENTER` all components and phases (initiation code 2), `ENTER` the components only (initiation code 1) or `SUSPEND` everything (initiation code 0). No other values are legal. The entered components and phases can later be changed with *CHANGE_STATUS* on page 59.

DEFINE_COMPONENTS

Change the set of components. By default, the elements are used as components. The set of components can be important because some conditions are set using components, for example, the amounts, activities or chemical potentials.

For example, in the system Fe-Si-O, you can define FEO, FE2O3 and SIO2 as components, thus replacing the default FE, SI and O.

This implies a command *REINITIATE_MODULE* on page 161 and it should be given as the first command in the POLY module.

Syntax	DEFINE_COMPONENTS
Prompt	`GIVE ALL NEW COMPONENTS /EXISTING COMPONENTS/: <NEW COMPONENTS>` The `new components` must be specified all on one line. These replace the existing component definitions. The number of components cannot be changed with this command. Use *CHANGE_STATUS* on page 59 instead.
	To keep some existing components, it is recommended that you also enter these on the line. Otherwise, the added new components may not be defined correctly. This is especially important if some components are built out of several elements. The new components do not have to be present as species.

DEFINE_DIAGRAM

This allows automatic calculation and plotting of a diagram with a single command. It is the same as the DEFINE_MATERIAL command up to when the first equilibrium is calculated. The alloy OPTION feature is also available in this command to specify alloying compositions for a special alloy predefined by the OPTION keyword in a selected database (e.g. the TCNI Ni-based Superalloys Database).

Use this command to calculate all types of phase diagrams after specifying all composition value and an initial temperature (if temperature is used as an axis). However, for binary and ternary diagrams, the special BIN and TERN modules may be preferred.

It then lists all the independent variables for the defined system (i.e. temperature and the components) and asks for a variable as the X-axis. You must also specify a maximum and minimum for the X-axis. The second axis (Y-axis) can be another composition (or the temperature if that is not on the X-axis) from the independent variable list. The program then calculates and plots a *phase diagram*, as there are two independent quantities on the axes.

Alternatively, select a dependent quantity as the Y-axis variable from the second list on screen (e.g. the amount of all phases, composition of a specific phase, or fractions of a component in all phases), and the program calculates and plots how this quantity depends on the condition on the X-axis. This is a *property diagram*.

This command ends up within the POST module monitor. You can refine the calculated phase diagram or property diagram.

Moreover, many more property diagrams with axes other than compositions can also be plotted (after the calculation), using *SET_AXIS_VARIABLE* on page 164 in the sequent POST monitor.

Syntax	DEFINE_DIAGRAM
Prompts	These prompts are given: SAME ELEMENTS AS BEFORE /Y/? <Y OR N> MOLE PERCENT OF <ELEMENT> /##/: <VALUE> or MASS PERCENT OF <ELEMENT> /##/: <VALUE> DATABASE /ABCDE/: <DATABASE NAME> MAJOR ELEMENT OR ALLOY: <ELEMENT NAME> COMPOSITION IN MASS (WEIGHT) PERCENT? /Y/: <Y OR N> 1ST ALLOYING ELEMENT: <ELEMENT NAME> MASS (WEIGHT) PERCENT: <AMOUNT OF THE ABOVE SPECIFIED ELEMENT> 2ND ALLOYING ELEMENT: <ELEMENT NAME> NEXT ALLOYING ELEMENT: <ELEMENT NAME> MASS (WEIGHT) PERCENT: <AMOUNT OF THE ABOVE SPECIFIED ELEMENT> TEMPERATURE (C) /1000/: <TEMPERATURE OF INTEREST IN OC> REJECT PHASE(S) /NONE/: <LIST OF PHASE(S) TO BE REJECTED> RESTORE PHASE(S) /NONE/: <LIST OF PHASE(S) TO BE RESTORED> OK? /Y/: <Y OR N> SHOULD ANY PHASE HAVE A MISCIBILITY GAP CHECK? /N/: <Y OR N> PHASE WITH MISCIBILITY GAP: <PHASE NAME> MAJOR CONSTITUENT(S) FOR SUBLATTICE #: /AA/: <CONSTITUENT(S)> PHASE WITH MISCIBILITY GAP: <PHASE NAME> The first equilibrium is calculated, as with *DEFINE_MATERIAL* on page 147. Then a list of all independent conditions suitable to be chosen as X/Y-axis variables is given by the program.
	QUIT? /Y/: <Y OR N> This question is asked only when the axis variables are already defined, or if the command DEFINE_DIAGRAM is used. It then offers an opportunity to quit (Y) the calculation or to continue (N) the calculation but by defining other axes.
	GIVE THE NUMBER OF THE CONDITION TO VARY /1/: <A CONDITION INDEX>

Syntax	DEFINE_DIAGRAM
	Select one of the independent conditions by giving its index on the condition list as the X-axis variable.
	`MINIMUM VALUE /XXX/: <MINIMUM VALUE FOR X-AXIS>` Specify the minimum value of the chosen X-axis variable. A default value is shown automatically by the program; press <Enter> to accept it or input another value.
	`MAXIMUM VALUE /YYY/: <MAXIMUM VALUE FOR X-AXIS>` Specify the maximum value of the chosen X-axis variable. A default value is shown automatically by the program; press <Enter> to accept it or input another value. Then another list with some dependent quantities is given by the program, which can be selected as the Y-axis variable.
	`GIVE THE NUMBER OF THE QUANTITY ON THE SECOND AXIS /#/: <##>` Select one of the independent conditions or dependent quantities as the Y-axis variable, by giving its corresponding index given on the condition lists. It must be different from the X-axis variable already selected. If selecting one of dependent quantities (by giving the corresponding number from the second list) as the Y-axis, then a property diagram is to be automatically calculated (through a normal stepping procedure) and generated. For `composition of a phase` the phase name is asked for further specification subsequently. If selecting any of the other independent variables (conditions) on the first list as the Y-axis, then a phase diagram is automatically calculated (through a mapping procedure) and plotted.
	`NAME OF PHASE: /ABC/: <PHASE NAME>` This is prompted only in case of that the `composition of a phase` is selected as the Y-axis variable. The phase name for which the composition varied along with X-axis variable should be specified.
	`MULTIPLE START POINTS? /Y/: <Y OR N>` This question is to confirm whether or not to let the program to automatically generate and use multiple start points for mapping the defined phase diagram.
	`SAVE FILE /RESULT/: <FILE NAME>` The file name where the calculations are stored (saved as a *.POLY3 file) should be specified; the default file name is `RESULT.POLY3` (Windows) or `RESULT.poly3` (Linux).

DEFINE_MATERIAL

Read data for a system from a database in the POLY module. It is convenient to use for alloys when there is a major component and the amount of the other elements is known in mass (weight) fraction. The command reads the system from the specified database, sets the composition and temperature (and pressure equal to 1 bar) and calculates the equilibrium state before the prompting for a new command. You can list the results with *LIST_EQUILIBRIUM* on page 156 or set a new composition or set axis for a STEP or MAP command.

You cannot append data from different databases in this way. Use this command with data from a USER database.

<table>
<tr><td>Syntax</td><td>DEFINE_MATERIAL</td></tr>
<tr><td rowspan="4">Prompts</td><td><code>SAME ELEMENTS AS BEFORE /Y/? <Y OR N></code>

This question is asked only if some data is already read from the database, or if the command DEFINE_MATERIAL or DEFINE_DIAGRAM is used. It then offers a convenient way to change the composition and temperature with one command.

This command only works properly in cases where the composition of the material system is already defined as in the mole-percent or mass-percent unit.</td></tr>
<tr><td><code>MOLE PERCENT OF <ELEMENT> /##/: <VALUE></code>

or

<code>MASS PERCENT OF <ELEMENT> /##/: <VALUE></code>

If you have decided to use the same materials system (available in the current POLY3 workspace) by accepting the default answer (<code>Y</code>) to the previous prompt <code>Same elements as before /Y/?</code>, one of the alternative prompts display for each of the components in the defined system, depending on how the composition is defined (either in mole-percent, or in mass-percent).

Prompts are repeated until all the defined components are completed. Then, the program prompts to specify the temperature condition.</td></tr>
<tr><td><code>DATABASE /ABCDE/: <DATABASE NAME></code>

The database with the description for the material must be given, or press <Enter> if using the current database. It is possible to give a USER database.</td></tr>
<tr><td><code>MAJOR ELEMENT OR ALLOY: <ELEMENT NAME></code>

The material must have a major element, usually the element which is present in the largest amount. The fraction of this element is not set but is the rest.</td></tr>
</table>

Syntax	DEFINE_MATERIAL
	In some databases there are the alloys predefined. An alloy has a default major element and have limits of the amounts of the alloying elements. If you stay within these limits the calculation gives reasonable results.
	`COMPOSITION IN MASS (WEIGHT) PERCENT? /Y/: <Y OR N>` The default is that input is taken as mass percent, but it is possible to change to mole percent by answering `N`. Composition should be given in PERCENT not FRACTION, as it is required for the `W` and `X` state variables in the *SET_CONDITION* on page 165 command.
	`1ST ALLOYING ELEMENT: <ELEMENT NAME>` The first alloying element must be given. All alloying elements are asked for in a sequence. These can be given in any order. You must know if you are present as assessed systems in the database. There is no error or warning messages if data are missing. Check the documentation of the database selected. If an alloy is selected, a list of legal alloying elements and their maximum percent is listed on-line.
	`MASS (WEIGHT) PERCENT: <AMOUNT OF THE ABOVE SPECIFIED ELEMENT>` The amount of the alloying element in mass (weight) percent. Using the DEFINE_MATERIAL command you cannot use the normal flexibility of Thermo-Calc for conditions, but all must be given in mass percent. However, you can afterwards change the conditions using the SET_CONDITION command.
	`2ND ALLOYING ELEMENT: <ELEMENT NAME>` The second alloying element must be given. If only one, press <Enter>. If an element name is given then the program asks for its mass fraction.
	`MASS (WEIGHT) PERCENT: <AMOUNT OF THE ABOVE SPECIFIED ELEMENT>` The amount of the above specified alloying element in mass (weight) percent.
	`NEXT ALLOYING ELEMENT: <ELEMENT NAME>` Continue giving elements and mass (weight) fractions until all elements specified. When all alloying elements and their compositions (as in the above prompt) are specified, press <Enter> as answer to this question to finish the materials definition.
	`TEMPERATURE (C) /1000/: <TEMPERATURE OF INTEREST IN OC>`

Syntax	DEFINE_MATERIAL
	POLY makes the first calculation after retrieving the data for this temperature. By pressing <Enter> to accept the default temperature. The value should be given in Celsius (oC). The pressure is set to 1 bar.
	`REJECT PHASE(S) /NONE/: <LIST OF PHASE(S) TO BE REJECTED>` This is a question generated by the database allowing you to select the phases. Normally, all phases should be included when you press <Enter>. If a phase is to be rejected, the name of the phase must be supplied. Several phase names can be specified in one line. It is possible to reject all phase by giving an asterisk *. If the number of phases to be included is much smaller than the total number of phases, it may be convenient to first reject all phases and then restore those that should be included. The question is repeated until you press <Enter> after rejecting all unwanted phases or an asterisk *.
	`RESTORE PHASE(S) /NONE/: <LIST OF PHASE(S) TO BE RESTORED>` You can restore phases that are accidentally or deliberately rejected. It may also be possible to restore some hidden phases. If phases are to be restored the name of the phases must be supplied. Several phase names can be specified in one line. It is possible to restore all phase by giving an asterisk *. The question is repeated until you press <Enter> after restoring all desired phases.
	`OK? /Y/: <Y OR N>` All phases to be selected from the database are listed and the selection confirmed. If there are any errors or you want to amend the selection, answer `N` and you are returned to the question about rejecting phase(s). If the selection is confirmed by answering `Y` the software retrieves all thermodynamic data and available references from the chosen database.
	`SHOULD ANY PHASE HAVE A MISCIBILITY GAP CHECK? /N/: <Y OR N>` The database usually creates two or more composition sets for phases that can have miscibility gaps. However, for some phases this is not done automatically, for example the miscibility gap in the bcc phase in Fe-Cr is usually ignored. But if it is important to include a Cr-rich bcc phase, specify this here. It costs some computation time and may make the subsequent MAP or STEP more difficult to converge. If you do not want to have any phase with a miscibility gap in the calculation, press <Enter>. Then, the DEFINE_MATERIAL command starts calculating the equilibrium, and is terminated.
	To set such a phase with miscibility gaps in the calculation, answer `Y`. Then the software

Syntax	DEFINE_MATERIAL
	asks questions about the phase names and their constitutions, such as: `PHASE WITH MISCIBILITY GAP: <PHASE NAME>` You must supply the phase name, which has a miscibility gap under the specified system and conditions.
	`MAJOR CONSTITUENT(S) FOR SUBLATTICE #: /AA/: <CONSTITUENT(S)>` The software shows a default constituent in the sublattice `#(1,2,3,…)`, according to the existing phase definition in the chosen database. You can specify one or more major constituents for the sublattice # in the phase. This question is repeated until all sublattices are specified.
	`PHASE WITH MISCIBILITY GAP: <PHASE NAME>` You can supply another phase name with a miscibility gap under the specified system and conditions, and answer the questions concerning the major constituent(s) in associated sublattice(s). By pressing <Enter>, the command starts calculating the equilibrium, and then terminates.
	It is also possible to use this command to select an alloy from a specific database (e.g. the TCNI Ni-based Superalloys Database). Such alloys are predefined by the OPTION keyword inside the database, and have their default major elements and composition limits of their alloy elements. Such alloys available in the selected database (at the prompt `Database /ABCDE/` can be listed on the screen if typing a ? mark on the prompt `Major element or alloy`. When a specific predefined alloy (instead of a major element) is selected, the major element is staked from the alloy definition and shown on the screen (with a message like `Alloy found with major element NI`). You can only specify alloying elements and the compositions (weight percent or mole percent). Typing a ? at any of the prompts for the alloying element names, e.g. `1st alloying element`, `2nd alloying element`, lists all the alloying elements and the composition limits in the alloy. If the composition of an alloying element is outside of its limit, there is a message (such as `Amount above limit: 30.0000`) and a prompt `Override limit ? /N/`. If you decide to enforce the override by answering `Y` on this prompt (i.e. accepting the over-limit alloying composition), another warning message (such as `Amount of major element below limit: 70.0000`) and prompt `Override limit ? /N/`. Then further decide if enforcing the overriding: if `Y` then accepting the major element's composition below the limit; if `N` then using the predefined

Syntax	DEFINE_MATERIAL
	major element composition limit.

DELETE_INITIAL_EQUILIBRIUM

Delete ONE specific initial equilibrium point or ALL of the initial equilibria. The initial equilibria are used as starting points for all the sub-sequential MAP and STEP calculations.

Also see *ADD_INITIAL_EQUILIBRIUM* on page 119.

Syntax	DELETE_INITIAL_EQUILIBRIUM
Prompt	`NUMBER /ALL/: <NUMBER OF AN INITIAL EQUILIBRIUM>`
	Specify the number of an initial equilibrium (as a specific digit e.g. 3, or ALL) to be deleted from the POLY3 workspaces. It is recommended to use *LIST_INITIAL_EQUILIBRIA* on page 157 to figure out the numbers for all the existing initial equilibrium points that have already been added (*manually* through *ADD_INITIAL_EQUILIBRIUM* on page 119, or *automatically* by some other POLY commands) and stored in the current POLY3 workspace. By default, ALL the initial equilibrium points are deleted.

DELETE_SYMBOL

Delete symbols (i.e. constants, variables, functions or tables. These symbols are entered with *ENTER_SYMBOL* on page 72.

Syntax	DELETE_SYMBOL
Prompt	`NAME: <NAME OF A SYMBOL>` Specify the name of the symbol to be deleted. Only one symbol can be deleted each time.

ENTER_SYMBOL

Symbols are a useful feature of the POLY and POST modules to define quantities that are convenient. Symbols can be constants, variables, functions or tables.

Functions or tables (with defined functions as values) can be entered in the POST module after a stepping or mapping calculation, for purposes of plotting such entered functions or tables as axis variables.

Within the POLY module, symbols are normally defined prior to an equilibrium calculation (enforced by a C_E command), stepping calculation (enforced by the command *STEP_WITH_OPTIONS* on page 172) or mapping calculation (enforced by *MAP* on page 158). These can be entered after an equilibrium calculation; however, for defined functions, variable or tables, it requires using *EVALUATE_FUNCTIONS* on page 155 before showing the corresponding values in the calculated equilibrium state.

The symbols entered in the POST module are not saved in the currently-loaded POLY3 workspaces. Therefore, if you want to apply such symbols in other similar calculations for the same defined system, you must use the ENTER_SYMBOL command prior to the STEPPING or MAPPING calculation in the POLY module.

Also see *ENTER_SYMBOL* on page 72 for the GIBBS Module.

See example 44 in the *Console Mode Examples Guide*for an example of using variables and functions.

<table>
<tr><td>Syntax</td><td>ENTER_SYMBOL</td></tr>
<tr><td>Prompt</td><td>

`CONSTANT, VARIABLE, FUNCTION OR TABLE? /FUNCTION/: <KEYWORD>`

The `Keyword` can be specified as `CONSTANT`, `VARIABLE`, `FUNCTION` or `TABLE`.

- CONSTANTS can only be entered once and is a means of using a name for a numeric value. For example, the value of 1 atmosphere in Pascal can be denoted by `P0` after the command ENTER CONSTANT `P0=101325`. Defined constants can be used as values in condition assignments, for example, `SET-COND P=P0`.
- FUNCTIONS are expressions of state variables or other functions. These expressions are saved, and whenever a function value is requested all functions are evaluated. The reason for this is that they may depend on each other.
- VARIABLES are similar to functions because they can also be expressions of state variables. However, contrary to functions, they are only evaluated when they are entered or if they are explicitly named in an *EVALUATE_ FUNCTIONS* on page 155 command. It is possible to enter a variable with a new expression any time. This expression is evaluated directly and its value stored as the value of the variable. Defined variables can be used as values in the SET-CONDITION command.
- TABLES are used for listing results from the STEP or MAP commands. A table consists of a list of any number of state variables, functions or variables. Defined tables can also be used in the post-processor POST.

There is a special connection between tables and variables. If a variable is used in a table, it is evaluated for each line of the table in the TABULATE command or when the table is used in a plot.

</td></tr>
<tr><td></td><td>

`NAME: <NAME OF THE SYMBOL>`

Each symbol has a unique name that must start with a letter and can have maximum 8 characters.

Legal characters include letters (either UPPER or lower case), digits and underscore _. Any other special character, such as parentheses (and), plus +, minus -, slash / or \, full stop (.), are illegal for symbol names.

You can enter the symbol name and the value or function on the same line, these must be separated with an equal sign =, for example, `TC=T-273.15` or `T_C=T273.15` which stands for a definition of temperature in Celsius. Otherwise, these questions are asked.

</td></tr>
</table>

Syntax	ENTER_SYMBOL
	For different types of symbol (constant, function, variable or table), the questions have different prompts.

	`FUNCTION: <DEFINITION FOR A FUNCTION OR VARIABLE>`
	Functions and *variables* are evaluated from an expression of state variables or other functions, constants or variables. The expression is a FORTRAN-like expression and operators +, -, *, = and ** can be used (** only with integer powers).
	Unary functions like `LOG`, `LOG10`, `EXP`, `SIN`, `COS` and `ERF` can also be used. An expression can be continued on more than one line. An expression should be terminated by a semicolon (;) or an empty line (by pressing <Enter> at the next prompt).
	Examples of function expressions:
	• `GM(LIQUID)`: The Gibbs energy of liquid per mole component
	• `H.T/4.184`: The heat capacity of the system in calories
	• `ACR(CR)/X(FCC,CR)`: The activity coefficient for Cr in FCC
	• `T-273.15`: The temperature in Celsius
	`&: <CONTINUATION OF THE DEFINITION FOR THE SYMBOL>`
	The ampersand & means that you can continue to write the function on the new line if one line is not enough for the function. If you finish the function press <Enter> again.

	`VALUE: <VALUE FOR A CONSTANT>` A *constant* can only be assigned a numeric value once.
	`VALUE OR EXPRESSION: <VALUE OF EXPRESSION FOR A VARIABLE>` A *variable* can be assigned a numeric value or an expression. An expression is evaluated immediately and discarded. Only the numeric value is kept. This gives a possibility to save a value between calculations with different conditions because all state variables and functions are evaluated for the new conditions.

	`VARIABLE(S): <VARIABLE(S) IN A TABLE>`
	A *table* consists of a list of state variables or functions. One way to obtain results from a STEP command is through a table.
	Example:
	`ENTER TABLE K=T,X(LIQ,C),X(LIQ,CR),ACR(C)`
	Which means that the table called K contains four columns, i.e. the temperature, the mole

Syntax	ENTER_SYMBOL
	fractions of C and Cr in the LIQUID phase, and the activity of C. To show the temperature in Celsius in a table, give the command ENTER FUNCTION `TC=T-273`; and then use the symbol `TC` in the table. `& <CONTINUATION OF THE DEFINITION FOR THE TABLE>` The ampersand `&` means that you can continue to write the table on the new line if one line is not enough for the table. If you finish the table press <Enter> again.

EVALUATE_FUNCTIONS

The value of one or more or all entered functions or variables are evaluated and listed.

Syntax	EVALUATE_FUNCTIONS
Prompt	`NAME(S): <NAME(S) OF DEFINED FUNCTION(S)>` The names of one or more entered functions or variables must be specified. By typing a wildcard *, all functions and variables are evaluated. Variables are evaluated only if these are explicitly named.

LIST_AXIS_VARIABLE

Syntax	LIST_AXIS_VARIABLE
	Lists all the axis variables for a stepping or mapping calculation that have already been set by *SET_AXIS_VARIABLE* on page 164.

LIST_CONDITIONS

All the conditions that are set by the command SET_CONDITION and the command-sequence CHANGE_STATUS PHASE =FIXED <0 or 1 or alike>, are listed. The current conditions are also listed by the command LIST_EQUILIBRIUM. The degree of freedom in the defined system is also shown up. If this is zero, you can perform a COMPUTE_EQUILIBRIUM command. If it is larger than zero, some more conditions are required, and you must further set additional ones, using *SET_CONDITION* on page 165 or *CHANGE_STATUS* on page 135. If it is negative, a user has defined too many conditions and needs to take away the unnecessary ones, using the SET_CONDITION command (with a value of NONE for the to-be-deleted condition) or CHANGE_STATUS command (i.e. changing a FIXED status of a phase to another type of phase status, ENTERED or DORMANT or SUSPENDED).

Syntax	LIST_CONDITIONS

Example	`P=100000, T=800, N(NI)=1E-1, N=1`
	`FIXED PHASES`
	`FCC_A1=1 LIQUID=0   DEGREE OF FREEDOM 0`

LIST_EQUILIBRIUM

The result (always in SI units) from the last calculated equilibrium is listed on screen or in a textual file.

You can also execute this command if no calculation is made or if the calculation fails. It is your responsibility to interpret the results accordingly.

Syntax	LIST_EQUILIBRIUM
Prompt	`OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME>` The name of the text file where the list of the calculation results shall be written.
	`OPTIONS /VWCS/: <OPTION(S)>` Select the output units and formats by optionally specifying a combination of these letters. • *Fraction order:*`V` means VALUE ORDER; `A` means ALPHABETICAL ORDER. • *Fraction type*: `W` means MASS FRACTION; `X` means MOLE FRACTION. • *Composition*: `C` means only COMPOSITION; `N` means CONSTITUTION and COMPOSITION. • *Phase*: `S` means including only STABLE PHASES; `P` means including ALL NON-SUSPENDED PHASES.
	Default options are `VWCS`. If the output fraction type should be in mole fraction (rather than mass fraction), then give `VXCS` or type `X` (implying that in this case the options V, C and S are accepted as the listing manners for fraction order, composition and phase). If accepting all the default options, or if accepting all the altered options that had already changed when using this command previously, you can type `L_E,,,` or `LIST_ EQUILIBRIUM ,,,` LIST_EQUILIBRIUM displays ORD (ordered) or DISORD (disordered) (when available). See *CHANGE_LEGEND* on page 193 for details.

LIST_INITIAL_EQUILIBRIA

List all the equilibria added with *ADD_INITIAL_EQUILIBRIUM* on page 119.

Syntax	LIST_INITIAL EQUILIBRIA
	All the initial equilibria are used for the MAP (and STEP) calculations.

LIST_STATUS

In the POLY module, list the status of components, species or phases.

Syntax	LIST_STATUS
Prompt	Specify what to list: OPTION /CPS/: <KEYWORD(S)> KEYWORD = C OR P OR S,OR ANY COMBINATION • C means list component status • P means list phase status • S means list species status • CS for both components and species Default is CPS.
	The results depend on the key word specified in the options for *CHANGE_STATUS* on page 59, a table with the current status of phases or species or components, or the combinations, is shown: • For components, the statuses and reference states are listed. • For species, only the status is listed. • For ENTERED and FIXED phases, the status, driving forces and equilibrium amounts are listed.
	The metastable phases are listed in descending order of stability. To avoid long outputs only 10 metastable phases (in ENTERED status) are listed by lines, while all other less stable phases are merged onto one line. For DORMANT phases, their phase names and driving forces are listed. For SUSPENDED phases, only the phase names are listed.

LIST_SYMBOLS

For both the POLY and POST modules, list the definitions for all constants, functions, variables and tables.

In order to find the value of a function or variable, use *SHOW_VALUE* on page 172 or *EVALUATE_FUNCTIONS* on page 155. A table is tabulated with the TABULATE command.

Syntax	LIST_SYMBOLS
	The defined variables are listed up together with the defined functions, but variable names are followed by a percentage sign %.

LOAD_INITIAL_EQUILIBRIUM

Copies all conditions and calculated results from a specific added initial equilibrium to the current equilibrium. The current conditions and calculation results are lost, and the newly loaded initial equilibrium point gets into the POLY workspace.

Syntax	LOAD_INITIAL_EQUILIBRIUM
Prompt	`NUMBER: <NUMBER OF AN INITIAL EQUILIBRIUM>` Specify the number of an initial equilibrium to be loaded as current. The number can be found with *LIST_INITIAL_EQUILIBRIA* on the previous page.

MAP

This command starts the mapping procedure for making a calculation of phase diagrams in a defined multicomponent heterogeneous system, from one or more initial equilibria. A phase diagram is usually mapped within a specific space that is constructed by two (or more) defined independent mapping axis-variables.

Syntax	MAP
	A phase diagram consists of mapped phase boundary lines/curves; on one side of each such phase-boundary line/curve, the amount of one specific stable phase is zero (i.e. the zero-fraction lines). From a single MAP calculation, many different types of phase diagrams in the defined multicomponent heterogeneous system can be plotted, with some desired properties (that vary along the calculated phase-region boundaries) plotted as X/Y-axis variables. All different types of phase diagrams are generated by the mapping calculations through this command.

Normally, you need to have calculated at least one initial equilibrium point and have also defined at least two independent varying variables (i.e. the controlling conditions in the system) that are set with *SET_AXIS_VARIABLE* on page 164. You can also have three, or four or maximum five independent varying variables that are defined by the SET_AXIS_VARIABLE command).

This lists the current values of each of the independent axis variables for each of the calculated equilibrium points along each of the mapped phase boundaries, and also lists the corresponding information when the set of stable phases changes.

The Global Minimization Technique is by default enforced in this command while it can be disabled if having decisively switched it off (or possibly in some special modules) through changing the minimization option using the *ADVANCED_OPTIONS* on page 125 → MINIMIZATION_OPTION command-sequence. You can choose how often to do a Global Minimization test (via the Global Test Interval option) in the ADVANCED_OPTIONS STEP_AND_MAP command-sequence.

During a MAP calculation, the values of mapped axis-variables for presenting each phase boundary (lines/points) of the calculated phase-regions are listed, and the corresponding stable-phase sets are shown up. You can terminate the mapping of a line by pressing a single CTRL-A (Windows) or CTRL-C (Linux). This can be useful in order to stop a longish calculation without losing what is already calculated.

If there is any convergence problem in finding a stable solution at a certain stage during a calculation procedure enforced by this command, these messages display on screen:

```
Convergence problems, increasing smallest site-fraction from 1.00E-30   to hardware
precision 2.00E-14. You can restore using SET-NUMERICAL-LIMITS
```

This implies that smallest site fraction in the current POLY3 workspace is automatically increased from the default value 1.00E-30 to the hardware-dependent precision (under Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current TC run, you can use the POLY command *SET_NUMERICAL_LIMITS* on page 168 to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

In particular, for phase diagrams with tie-lines in the plane (i.e. most binary systems and ternary isotherms), there is a MAP procedure which checks for the best phase to use as axis variables in order to ensure reasonable increments between the tie-lines. This produces smoother curves and also gives a better stability in finding adjacent phase-regions.

Complex miscibility gaps of various solution phases are automatically detected during a mapping calculation, and two or more composition sets for each of such solution phases are automatically generated as well, through an automatic Global Minimization Test procedure (by specifying a Global Test Interval value and confirming the `Automatically add initial equilibria` in the ADVANCED_OPTION STEP_AND_MAP command-sequence). As a particular case, such an automatic Global Minimization Test ensures that you can start calculating from high temperatures in an austenitic steel (metallic FCC_A1 solution phase) where the MC carbides/nitrides/carbonitrides (i.e. the C-/N-rich sides of the FCC_A1 miscibility-gap, often (while not always) being referred to as `FCC_A1#2, FCC_A1#3`, etc.) are not stable, and during the MAP command the MC carbides/nitrides/carbonitrides may first become metastable with a composition different from the metallic FCC_A1 phase and later also stable. The advanced mapping procedure (enforced by the Global Minimization Technique) inside the Thermo-Calc software (both the Console Mode and Graphical Mode) can efficiently and effectively handle complex miscibility gaps in multicomponent heterogeneous systems.

To get a complete phase diagram, sometimes it may be necessary to have multiple starting equilibrium-points, and/or to have added multiple initial-equilibrium points (through *ADD_INITIAL_EQUILIBRIUM* on page 119) inside various phase-region boundaries under certain direction(s).

POST

Syntax	POST
	Switches to the post-processor (the POST module), which has its own command repertoire.
	See *POST Commands* on page 188 for all details.

READ_WORKSPACES

The POLY3 and GIBBS workspaces and the calculated results from the MAP and STEP commands can be READ from a file where they are saved with *SAVE_WORKSPACES* on page 30. Such an *.POLY3 file is not printable.

Syntax	READ_WORKSPACES
Options	`File name` is the name of a saved POLY3-file where the POLY3 and GIBBS workspaces shall be read from must be specified. You do not need to type the extension if it is the default *.POLY3 (Windows) or *.poly3 (Linux), otherwise type the whole POLY-file name. A window opens so that the path (in the **Look in** field) and **File name** can be specified. The **Files of type** (i.e. POLY3) cannot be changed. Click **Open** to open the POLY3 and GIBBS workspaces from the saved *.POLY3 file.
	When reading back an original POLY3 workspace that has already been saved as an *.POLY3 file in the current Thermo-Calc (Console Mode) run or had been read from an existing POLY3 file under the current work area, while some additional changes in the settings may be made but do not need to be kept in further steps in the current Thermo-Calc (Console Mode) run or any diagram is plotted in the POST module, you can type `READ,,` or `READ_WORKSPACE ,,`

RECOVER_START_VALUES

Syntax	RECOVER_START_VALUES
	Recovers the start values in equilibrium calculations.

REINITIATE_MODULE

Ensures that the whole POLY module (and thus the entire POLY workspace) is reinitiated to the state as it was when first entered.

Syntax	REINITIATE_MODULE
	All the defined components, defined conditions, changed status, entered symbols, defined independent axis-variables, calculated starting equilibrium-points, added initial equilibrium points, stepped/mapped equilibrium data, and so forth, are removed completely. The saved file name is restored to the default.

SAVE_WORKSPACES

Thermo-Calc allows saving the current status and workspaces of the program, including thermodynamic data, conditions, options and results from a single, stepping or mapping calculation on an *.POLY3 file. Do this for later use or when you need to terminate the current Thermo-Calc (Console Mode) run.

The POLY3 and GIBBS workspaces are saved on a file with this command. In the GIBBS workspace, all thermochemical data are stored. In the POLY3 workspace, all thermochemical data, all the last set of conditions and equilibrium state, changed status, entered symbols, advanced options, defined stepping/mapping variables, added initial equilibria, stepped/mapped results, etc., are stored, so it also contain the GIBBS workspace. After a SAVE_WORKSPACE command, you can always come back to exactly the state you had when you issued the SAVE_WORKSPACE command by giving a READ_ WORKSPACE command.

After saving the POLY3 and GIBBS workspaces on a file, you can leave the program and at a later time READ the file and continue from the saved state.

A STEP or MAP command automatically saves the work file with the most lately specified name. Do not SAVE after a MAP or STEP.

The results from the STEP or MAP commands are destroyed by the SAVE_WORKSPACE command. You may append several results obtained by sequential STEP or MAP calculations without destroying the previous results, whilst SAVE_WORKSPACE erases them all. Keeping this in mind is important and useful particularly for calculating various isothermal (or isoplethal) sections and plotting them on the same diagram in a single Thermo-Calc run.

You may append several results by STEP or MAP without destroying the previous results but the SAVE command erases them all. To suspend some of the STEP or MAP results, use the AMEND_STORED_ EQUILIBRIA command.

When you go to a response-driven module such as POTENTIAL or SCHEIL for example, a workspace file is automatically opened. In the workspace file, system definitions, conditions for the calculation, calculation results, and plot settings is saved. The file is saved in the current working directory, and is named after the name of the module that created it. For example, the POTENTIAL module saves a workspace file called POT.POLY3 (or POT.poly3 in Linux); the POURBAIX module saves a file called POURBAIX.POLY3; etc.

Syntax	SAVE_WORKSPACES
Options	A `file name` must be specified. The default extension of the POLY workspace file is *.POLY3 (Windows) or *.poly3 (Linux), while you can have any other extension. A **Save** window displays, so that the path (in the **Save in** field) and **File** name can be specified. The file type (i.e. POLY3) in the **Files of type** cannot be changed. When saving a POLY3 workspace under a name that already exists under the current work area, which is saved by default (after running a special module, e.g. BIN, TERN, POT, SCHEIL and POURBAIX) or in an earlier stage of the current Thermo-Calc (Console Mode) run or in a previous run (which is READ into the current POLY3 workspace), you can type SAVE,,y or `SAVE_ WORKSPACE ,,y`. However, this must also be avoided if some results from previous MAP or STEP calculations shall not be destroyed.

SELECT_EQUILIBRIUM

If you create more than one initial equilibrium (during data-assessments using the PARROT/ED_EXP modules), you can switch between them using this command.

Syntax	SELECT_EQUILIBRIUM
Prompt	`NUMBER /NEXT/: <CHOICE ON EQUILIBRIUM>` Answer `FIRST`, `LAST`, `NEXT`, `PREVIOUS` or `PRESENT`. Most commands affect only the `PRESENT` equilibrium. However, the commands *REINITIATE_MODULE* on the previous page and *DEFINE_COMPONENTS* on page 143 remove all the stored equilibria.

SET_ALL_START_VALUES

Set start values, e.g. if the calculation fails or if you have a miscibility gap or ordering. If temperature and pressure are not conditions, you are asked for values of them. Then for each phase prompt on if it should be stable and on its constitution.

Syntax	SET_ALL_START_VALUES
	`T /XXXX/: <TEMPERATURE IN K>` If the temperature is not a condition, supply a guess of its final value (in K).
	`P /100000/: <PRESSURE IN PA>`

<table>
<tr><td>Syntax</td><td>SET_ALL_START_VALUES</td></tr>
<tr><td></td><td>If the pressure is not a condition supply a guess of its final value (in pa).</td></tr>
<tr><td rowspan="4">Prompts</td><td><code>AUTOMATIC START VALUES FOR PHASE CONSTITUTIONS? /N/: <Y, N OR F></code>

Answer <code>N</code> (the default), <code>Y</code> or <code>F</code> (for force).

The reason for the <code>F</code> option is that in some cases the calculation has failed because impossible conditions are set, e.g. <code>W(C)=1.5 [W(C)</code> is mass fraction and must thus be less than unity]. The program tries to fulfil this condition by putting maximum amount of carbon in all phases, but can fail anyway. When you detect the error and set <code>W(C) to 0.015</code>, the calculation may still fail because it could start from the previous values. To get back to 'fresh' start values, give the answer <code>F</code> for <code>FORCE</code>.

If <code>Y</code> this command immediately terminates, and the program automatically sets the start values for phase constitutions in all possible phases.

For <code>N</code> supply an initial amount of each entered phase and the major constituents or site fraction of each constituent.</td></tr>
<tr><td><code>SHOULD <PHASE> BE STABLE /N/: <Y/1 OR N/2></code>

A guess on if this phase should be stable or not is requested. All entered phases are prompted for this question and next two.

You cannot have more phases stable than you have components, but at least one (which dissolves all constituents) must be set as stable. For backward compatibility, this question can be answered by 1 (for Yes) or 0 (for No).

The <code>phase</code> name may have a hash sign # followed by a digit, e.g. <code>BCC_A2#2</code>. For phases with miscibility gaps, there should be two phases with the same name but with different numbers after the hash sign.</td></tr>
<tr><td><code>MAJOR CONSTITUENT(S): <NAME OF MAJOR CONSTITUENT(S) IN THE PHASE></code>

The constituent with the largest fraction in the phase should be specified. If there are more than one constituent with a large fraction, give them all on the same line. If the default major constituents should be used answer with an asterisk (*). By giving $ the constitution is not changed. If there should be no major constituent give NONE and or if the major constituent(s) are improperly specified, you are asked for individual fractions in the phase.</td></tr>
<tr><td><code>Y(<PHASE>,<CONSTITUENT>) /.XXXXXXXXXX/: <.YYYYY></code>

The current value (<code>.XXXXXXXXXX</code>) is default. You can accept the default by pressing</td></tr>
</table>

Syntax	SET_ALL_START_VALUES
	<Enter> or give a new value (`.YYYYY`). The phase name or constituent name may have a hash sign # followed by a digit, e.g. `Y (BCC_A2#2,FE), Y(BCC_A2#2,C#2)`. For phases with miscibility gaps, there should be two phases with the same name but with different numbers after the hash sign. For phases with sublattices, the constituents in sublattice 2 or higher is also be suffixed with a hash sign # followed by a digit.

SET_AXIS_VARIABLE

In order to calculate a diagram, set at least one axis variable in a stepping calculation, or at least two axis variables in a mapping calculation. For property diagrams, one axis is enough; for phase diagrams two or more are necessary. Any condition that can be set to calculate an equilibrium can be used as an axis variable (with its lower and upper limits and step length) by using the SET_AXIS_VARIABLE command, and the POLY program does, after a STEP or MAP command, vary the value of the condition between the limits set on its related axis variable. As an extraordinarily unique and powerful feature of the Thermo-Calc software, up to five independent axis variables can possibly be used in a mapping calculation of a multicomponent heterogeneous system; however, the axis numbers 3, 4 and 5 must have chemical potentials of components (or temperature or pressure) as conditions.

Syntax	SET_AXIS_VARIABLE
	`AXIS NUMBER /#/: <AN AXIS NUMBER>` Specify a number between 1 and 5. The axis numbers 3, 4 and 5 must have chemical potentials of components (or temperature or pressure) as conditions.
Prompts	`CONDITION /NONE/: <ONE CONDITION>` Here the condition that should be varied along the axis must be given. The condition is specified as in *SET_CONDITION* on the next page, for example `W(C)` for mass fraction of carbon. By accepting `NONE`, the axis is removed.
	`MIN VALUE /0/: <MIN VALUE>` Specify the minimum value of the axis condition.
	`MAX VALUE /1/: <MAX VALUE>` Specify the maximum value of the axis condition.
	`INCREMENT /.025/: <STEP LENGTH>` Specify the maximum step length. By default, this is 1/40 of the total axis length.

You can give the SET_AXIS_VARIABLE command without having set a condition on the axis variable. Under such circumstances, the relevant condition is automatically created and the value set between the minimum and maximum axis limits; however, as a side effect, the POLY module creates two conditions, `P=1e5` and `N=1` (these have not been defined as a condition yet), in case you set an axis variable which is not already a condition.

You can use a logarithmic axis during calculations. This is useful for low fractions like in a gas phase where `1e-7` to `1e-2` might be an interesting range. The pressure is also suitable for logarithmic step. You specify the logarithmic axis by giving an asterisk * after the increment value.

The increment in this case is treated as a factor.

For example,

```
S-A-V 1 P 1E5 1E25 5*
```

This makes axis 1 a logarithmic axis where the difference between two calculated values makes a factor 5.

The factor must be larger than 1.0.

In some cases, such as when the DEFINE_DIAGRAM command is used or a special advanced module (e.g. BIN, TERN, POT, SCHEIL or POURBAIX) is called, some axis variables are automatically set by the program, not necessarily by this command.

SET_CONDITION

Specify the equilibrium conditions for the calculation. All kinds of state variables, as well as most of the M/W/V/F-suffixed state variables (for normalization) and R-suffixed state variables (with respect to chosen reference states), can be used as conditions. Beside temperature and pressure conditions, a multicomponent system can have mixed types of conditions; this brings the extreme flexibility and powerfulness to the complex calculations with the Thermo-Calc software system. You should repeat this command for a multicomponent system, till the degree of freedom in the defined system becomes zero. When a FIXED phase status is used on a specific phase (use *CHANGE_STATUS* on page 135), it is equivalent to one condition (implying that particular phase is stable in the defined system).

Syntax	SET_CONDITION

	Each condition must be given explicitly, but can be given on the same line (separated by a comma (,) or an empty space) or on separate lines with each one started with the command.
	
	Example
	
	`SET_COND T=1273, P=1E5, W(C)=.0015, X(LIQ,CR)=.22, ACR(N)=.2`
	
	or
	
	`SET_COND T=1273, P=1E5`
	
	`SET_COND W(C)=.0015, X(LIQ,CR)=.22 ACR(N)=.2`
	
	In this example, the temperature is set to 1273 K, the pressure to 1 bar (1E5 Pascal), the mass (weight) fraction of C to 0.0015 and the mole fraction of Cr to 0.22 and the activity of N to 0.2.

Prompt	`STATE VARIABLE EXPRESSION: <STATE VARIABLE NAME OR LINEAR EXPRESSION> OR A LINEAR EXPRESSION OF STATE VARIABLES.` Some of the state variables that can be used in conditions are: • `T`: temperature in the system (in K) • `P`: pressure in the system (in Pascal) • `N`: system size (mole number in moles) • `B`: system size (mass in grams) • `N(<component>)`: mole number of a component in the system • `X(<component>)`: mole fraction of a component in the system • `W(<component>)`: mass fraction of a component in the system • `ACR(<component>)`: activity of a component in the system • `MUR(<component>)`: chemical potential of a component in the system • `X(<phase>,<component>)`: mole fraction of a component in a phase • `W(<phase>,<component>)`: mass fraction of a component in a phase • `ACR(<species>,<phase>)`: activity of a species in a solution phase • `MUR(<species>,<phase>)`: chemical potential of a species in a solution phase • `H`: enthalpy in the system (in J) • `HM(<phase>)`: enthalpy of a phase (in J/mol)

Syntax	SET_CONDITION

	There are many state variables that can be used in conditions. For more information, enter an `INFO STATE_VARIABLES` command. A condition is normally a value of a single state variable with its value.
	Example
	`T=1273.15  P=1E5`
	`X(C)=.002`
	`W(CR)=0.5`
	`ACR(CR)=0.85`
	`X(FCC,C)=.001`
	`H=-250000`
	`HM(BCC)=-225000`
	A condition can also be a value of a linear expression involving more than one state variable. For example,
	`X(LIQ,S)-X(PYRR,S)=0`
	This means that it is a condition that the mole fraction of `S` component should be the same in the LIQUID and PYRRHOTITE phases. In practice, that should be the congruent melting point.
	After the equal sign only a numeric value is allowed in the expression.

	`FACTOR: <A FACTOR FOR THE STATE VARIABLE, OR A CONTINUATION>`
	This question means you did not answer the previous question. The program is then expecting a single state variable or a complete state variable expression, or the numeric factor in an expression with only one state variable. In a state variable expression a state variable may be preceded by a constant factor. An example of this is:
	`2*MUR(FE)+3*MUR(O)=-35000`
	This means that it should be a condition that two times the chemical potential of `FE` plus three times the chemical potential of `O` should be -35000 J/mol.

	`STATE VARIABLE: <A SPECIFIED STATE VARIABLE, OR A CONTINUATION>` This question is prompted if a single state variable name has not given in either the prompt `State variable expression` or `Factor`, or a state variable expression is given but the expression is incomplete, for example, `T-` or `2*MUR(FE)+`, for which the program is then expecting a continuation of the unfinished expression. You need to specify a state variable or a complete state variable expression, or complete the unfinished state variable

Syntax	SET_CONDITION
	expression. If a numeric factor is given before this prompt, only one state variable can be specified; otherwise, the program only takes the first state variable to complete the expression (i.e. the factor times the state variable).
	`VALUE /X/: <A NUMERIC VALUE, A CONSTANT OR A VARIABLE>` The value of the condition. This can be a numeric value, a constant or a variable. A suggestion is given as the default value. The special value `NONE` means that the condition is removed; for example `T=NONE` takes away the temperature condition.

SET_INPUT_AMOUNTS

Specify how a system is made up from mixing of various substances. It is useful with a substance database. In the C-H-O-N system, you can, for example, give:

```
S-I-A N(H2)=10, N(H2O)=25, N(C1O2)=5, N(N2)=100
```

The POLY module automatically converts this into conditions for the current set of components. In the case when the elements are defined as the components, the command above is equivalent to

```
SET-CONDITION N(H)=70, N(O)=35, N(C)=5, N(N)=200
```

Syntax	SET_INPUT_AMOUNTS
Prompts	`QUANTITY: <N(<SPECIE>) OR B(<SPECIE>)>` You can give the amount also preceded with an equal sign `=` [e.g. `N(H2)=10` or `B(H2O)=1000`], or press <Enter> for the next prompt on the amount of the quantity.
	`AMOUNT: <VALUE OF THE QUANTITY>` Specify the numerical value of the quantity. You can give negative amounts in the SET-INPUT-AMOUNTS command.

SET_NUMERICAL_LIMITS

Change the criteria for convergence. This is to speed up a calculation in a complex system.

If there is any convergence problem in finding a stable solution at any stage during a calculation procedure enforced by *COMPUTE_EQUILIBRIUM* on page 139, *STEP_WITH_OPTIONS* on page 172, *MAP* on page 158 or *ADVANCED_OPTIONS* on page 125), this message displays:

```
Convergence problems, increasing smallest site-fraction from 1.00E-30 to hardware
precision 2.00E-14. You can restore using SET-NUMERICAL-LIMITS
```

This implies that smallest site fraction in the current POLY3 workspace is automatically increased from the default value 1.00E-30 to the hardware-dependent precision (under Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current Thermo-Calc (Console Mode) run, you can use this command to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

Syntax	SET_NUMERICAL_LIMITS
Prompt	`MAXIMUM NUMBER OF ITERATIONS /200/: <XXX>` By default, the program tries 500 iterations before it gives up. As some models give computation times of more than 1 CPU second/iteration, this number is also used to check the CPU time and the calculation stops if 500 CPU seconds/iterations are used.
	`REQUIRED ACCURACY /1E-6/: <YYY>` This is a relative accuracy, and the program requires that the relative difference in each variable must be lower than this value before it has converged. A larger value normally means fewer iterations but less accurate solutions. The value should be at least one order of magnitude larger than the machine precision.
	`SMALLEST FRACTION /1E-12/: <ZZZ>` This is the value assigned to constituents that are unstable. It is normally only in the gas phase you can find such low fractions. The default value for the smallest site-fractions is `1E-12` for all phases, except for IDEAL phase with one sublattice site (such as the GAS mixture phase in many databases) for which the default value is always as `1E-30` (unless you have used the SET_NUMERICAL_ LIMITS command to reset an even-lower value (e.g. `1E-45`, that is naturally enforced to all the phases in the system).
	`APPROXIMATE DRIVING FORCE CALCULATION FOR METASTABLE PHASES /Y/: <Y OR N>` Normally the POLY module only requires that the set of stable phases is in equilibrium in order to terminate the iterations. The metastable phases are included in all iterations but these may not have reached their most favourable composition and thus their driving forces may be only approximate [corresponding to `Y`]. You can also force the calculation to converge for the metastable phases by answering `N` if it is important that these driving forces are correct, which may require more iterations, and the STEP and MAP command may also terminate due to bad convergence in a metastable phase.

SET_REFERENCE_STATE

The reference state for a component is important when calculating activities, chemical potentials and enthalpies. The reference state for a component is determined by the database. For each component the data must be referred to a selected phase, temperature and pressure, i.e. the reference state. All data in all phases where this component dissolves must use the same reference state. However, different databases may use different reference states for the same element. Thus you must be careful when, for example, mixing data from different databases.

By default, activities, chemical potentials, etc. are computed relative to the reference state used by the database and this may thus differ depending on the database. With this command select the reference state for a component if the reference state in the database does not suit.

You can set the reference state for a component as `SER`, i.e. the Stable Element Reference (which is usually set as default for a major component in alloys that is dominated by the component). Under such a case, the temperature and pressure for the reference state is not needed and thus is not prompted.

In order to specify conditions in the specified reference state, you can append an R to the state variables. Thus, AC is the activity (of a system component or of a species in a phase) with respect to the default reference state, and ACR is the activity with respect to the selected reference state; MU is the chemical potential of a system component with respect to the default reference state, and MUR stands for the chemical potential with respect to the selected reference state. The AC and ACR variables, for both components in a system and species in a phase, can be specified in natural logarithm, e.g. LNAC(Fe), LNACR(C), LNAC(O2,GAS), LNACR(O2,GAS).

Syntax	SET_REFERENCE_STATE
Prompts	`COMPONENT: <NAME OF THE COMPONENT>` The name of the component must be given.
	`REFERENCE PHASE: <NAME OF A PHASE USED AS THE NEW REFERENCE STATE>` The `Name of a phase`, that must be either ENTERED or DORMANT or SUSPENDED, must be given here. The component must, of course, be a constituent of this phase. For a phase to be usable as reference for a component the component need to have the same composition as an end member of the phase. The reference state is an end member of a phase. For example, in the Fe-C system, BCC can be a reference state for Fe but not for C since BCC cannot exist as pure C. If a component has the same composition as several end members of the chosen reference phase, then the end member is selected that at the specified temperature and pressure has the lowest Gibbs energy. For example, setting the reference state for component O as gas (one sublattice) with constituents O1, O2 and O3 results in O2 being the reference state if, at the present temperature, pure O2 has the lowest energy of O1, O2 and O3. If the reference state is set above a critical higher temperature, then O1 has the lowest energy and consequently become the reference state. The actual calculation of the reference state is only performed when SET_ REFERENCE_STATE is executed. For example, if GAS is chosen as the reference phase of component O at such a high temperature that O1 has the lowest energy, then O1 remains the reference state even at calculations performed at lower temperatures where O2 has a lower energy than O1.
	`TEMPERATURE /*/: <TEMPERATURE FOR THE REFERENCE STATE>`

Syntax	SET_REFERENCE_STATE
	Select the temperature (in K) for the reference state. The wildcard value * means the current temperature is used for the calculation.
	`PRESSURE /1E5/: <PRESSURE FOR THE REFERENCE STATE>` Select the pressure (in Pa) for the reference state. The wildcard value * means the current pressure is used for the calculation.

	Examples
	`S-R-S Fe SER`
	`S-R-S Cr FCC * 100000`
	`S-R-S H2O AQUEOUS * 100000`
	`S-R-S ZE REF_ELECTRODE * 100000`

SET_START_CONSTITUTION

This command is similar to *SET_ALL_START_VALUES* on page 162, but is used for an individual phase that may have a wrong constitution.

Syntax	SET_START_CONSTITUTION
Prompts	`PHASE NAME: <NAME OF A PHASE, AND POSSIBLE MAJOR CONSTITUENT(S)>` Specify the name of a phase for which the constitution shall be set. If there is a major constituent of the phase, you must specify this on the same line as the phase name. By giving an*, select the default major constituents. A $ means keeping the same constitution, and `NONE` means to give individual site-fractions.
	`Y(<PHASE>#<COMPOSITION_SET>,<CONSTITUENT>#<SUBLATTICE>) /XXX/: <SF>` The site fraction (`SF`) of the constituent shall be given. The default value `xxx` is the last calculated one.

SET_START_VALUE

Set a start value of an individual variable.

This command is not often required as the automatic start values can be handled for most problems.

Syntax	SET_START_VALUE
Prompts	`STATE VARIABLE: <NAME OF A STATE VARIABLE>` The state variable must be given.
	`VALUE` Specify the value to be assigned the state variable as start value.

SHOW_VALUE

Show the current value of any state variable, function or variable.

Syntax	SHOW_VALUE
Prompt	`STATE VARIABLE OR SYMBOL: <NAME(S) OF STATE VARIABLE(S) OR SYMBOL(S)>` A single or several state variables or symbols (entered functions or variables) can be specified.
	The wildcard * can be used to denote all phases or all components. You can use the dollar-sign $ to denote all stable phases. Thus, the command SHOW W (*,*)lists the mass fraction of all phases, whereas `SHOW W($,*)` lists the mass fraction of all stable phases. If you SHOW a function, all functions are evaluated using the current values of the state variables. But if you SHOW a variable, it retains its value from the time it was ENTERED or last EVALUATED. Therefore to save a value from one calculation to another, ENTER it to a variable. This is frequently done in the PARROT module to transfer values between equilibria.

STEP_WITH_OPTIONS

Initiates the stepping procedure to make a calculation of property diagrams in a multicomponent heterogeneous system. A property diagram is usually stepped under one defined independent stepping axis-variable, and is often constructed by the defined independent mapping axis-variable against a chosen property (for the system, or for a specific phase, or for a certain species).

A property diagram consists of property lines that change (normally) along with the stepping axis-variable. From a single STEP calculation, many different types of property diagrams in the defined multicomponent heterogeneous systems can be plotted. Various types of property diagrams are generated by the stepping calculations through this STEP_WITH_OPTION command

Normally, before it can be invoked, you must have already calculated an initial equilibrium point and have also set one independent varying variable (i.e. the controlling condition in the system) using *SET_AXIS_VARIABLE* on page 164.

The Thermo-Calc Console Mode lists the current values of the axis variable for each calculated equilibrium, as well as the corresponding information when the set of stable phases changes.

The Global Minimization Technique is by default enforced in this command (STEP), while it can be disabled if having decisively switched it off by a user (or possibly in some special modules) through changing the minimization option using the *ADVANCED_OPTIONS* on page 125 → MINIMIZATION_OPTION command-sequence.

You can choose how often to do a Global Minimization Test (via the `Global Test Interval` option) in the `ADVANCED_OPTIONS STEP_AND_MAP` command-sequence.

- NORMAL: Allows a stepping calculation with the chosen independently-varying equilibrium condition (axis-variable).
- INITIAL_EQUILIBRIA: An initial equilibrium is stored at each calculated equilibria. It is intended to generate a matrix of calculated initial equilibria, first stepped under a STEP INITIAL_EQUILIBRIA command-sequence and then repeated by another STEP NORMAL command-sequence (or even continued by a MAP command). For example, you can first use the temperature condition as the stepping axis-variable, and give a STEP INITIAL_EQUILIBRIA command sequence; this calculates a number of initial equilibrium points and automatically ADDs each one as an initial equilibria for another STEP NORMAL command-sequence (or even for a MAP command). Before the second STEP command (i.e. the STEP NORMAL command-sequence), you can alternatively chose a compositional condition as the new stepping axis-variable; then the STEP NORMAL command-sequence uses those created starting equilibrium points (at different temperatures) and steps in such a new compositional variable. As a result, these two sequential STEP commands together give a matrix of values. Actually, you can carry this option even deeper.
- SEPARATE_PHASES: Each of the entered phases are calculated separately and in parallel at each step, i.e. all the phases are calculated for the same composition (if the phases exist for that composition). In this way, you can calculate how the Gibbs energy for a number of phases varies for varying compositions; and this option is particularly useful for calculating Gibbs energies for complex phases with miscibility gaps and for ordered phase which is never disordered (e.g. SIMGMA-phase, G-phase, MU-phase or alike).

However, this option may sometimes be unable to find all the ordered phases and their disordered pairs during a STEP SEPARATE_PHASES calculation; and it is thus recommended to always use the newer ONE_PHASE_AT_TIME option instead, for STEPPING heterogeneous systems with both ordered phases and their disordered pairs.

- ONE_PHASE_AT_TIME: The stepping calculation is conducted for only one phase at a time individually (stepped over the whole stepping variable range), but the stepping is repeated for all the entered phase in the defined system. This new option is particularly useful for properly stepping in composition for an equilibrium heterogeneous system with both ordered phases and their disordered pairs, because of that, instead of calculating for the same composition for each of the phases at each composition step, it at least starts the stepping in each of the phases in the default most-stable compositions (the major constitutions). Therefore, this option is always recommended for STEPPING heterogeneous systems with both ordered phases and the disordered pairs.
- EVALUATE: The variables given are evaluated after each step, which may change other conditions. This option is useful for performing the Ordinary ScheilGulliver Approach (non-equilibrium transformations) for simulating alloy solidifications without back diffusion of fast-diffusing interstitial components in solid phases. However, users are always recommended to go directly to the SCHEIL module and run the desired Scheil-Gulliver type of simulations in a much easy, straightforward and automatic way.
- MIXED_SCHEIL: A stepping calculation designed for the Extended Scheil-Gulliver Approach (partial-equilibrium transformations) to simulate alloy solidifications with back-diffusion of one or more fast-diffusing interstitial components (such as C, N, O, S, etc.) in solid phases is performed. However, you are always recommended to go directly to the SCHEIL module and run the desired Scheil-Gulliver type of simulations in a much easy, straightforward and automatic way.
- T-ZERO: T0 (T-zero) lines in a diffusionless transformation [where two specific partially-equilibrated phases have the same Gibbs energy] are calculated along a composition variable (set as the stepping variable).
- PARAEQUILIBRIUM: The paraequilibrium state in a partly partitionless transformation [under which two partially-equilibrated phases have the same chemical potential (but different contents) for one or more interstitial components, such as C, N, O, S, etc.] is calculated along varied temperature or along a composition variable (of the matrix or one substitutional component).

Syntax	step_with_options
Prompt	`OPTION? /NORMAL/: <A CHOSEN OPTION, N OR I OR S OR O OR E OR M OR T OR P>`
Options	**Choose from these options**
`NORMAL`	Allows a stepping calculation with the chosen independently-varying equilibrium condition (axis-variable). By repeating the STEP NORMAL command-sequence, you can make several stepping calculations while still keep all the stepping-calculation results in the same current POLY3 workspaces (saved in the same `RESULT.POLY3` file default-named by the program, or in the same `<MyName>.POLY3` file that is specified before the first STEP NORMAL command-sequence). Only when there is no SAVE_WORKSPACES command issued before the repeated STEP NORMAL command-sequence, such repeated stepped results can be restored into the same current POLY3 workspaces/file. Choose to alter one or more controlling conditions for calculating starting equilibrium point(s) and can also optionally change the independently-varying condition (stepping-variable) and its settings for the stepping. All such alternations/changes are properly stored into the current POLY3 workspaces/file. Therefore, with one or more repeated STEP NORMAL command-sequence (without any SAVE_WORKSPACES command in between for overwriting stepped results), it enables you to easily/efficiently generate some special types of property diagrams and even to plot special phase boundary lines on phase diagrams, as described below: Calculate/plot missing part(s) of a specific property [that are calculated under another controlling condition inside the range of the stepping-variable] on the same property diagram. Calculate/plot two or more sets of a specific property [that are calculated under different controlling conditions and stepped over the same stepping-variable range] on the same property diagram for the same system. Calculate some special phase-boundary lines [especially for some defined secondary phase transformations or alike, e.g. the BCC_A1 <--> BCC_B2 phase-boundary, or equal-Gm for two specific phases, or equal-fraction or equal-activity for a certain species in two specific phases, that shall be calculated under different controlling-conditions and also be stepped over different stepping variable(s)], and then plot (or impose) them onto a corresponding phase diagram (which can be resulted from several such STEP NORMAL calculations or even from a separate

Syntax	step_with_options
	MAP calculation) for the same system.
INITIAL_ EQUILIBRIA	An initial equilibrium stored at every step. This option has not been completely implemented yet. It is intended to make a stepping calculation by generating a matrix of calculated initial equilibria, first stepped under a STEP INITIAL_ EQUILIBRIA command-sequence and then repeated by another STEP NORMAL command-sequence (or even continued by a MAP command). For example, you can first use the temperature condition as the stepping axis-variable, and give a STEP INITIAL_EQUILIBRIA command-sequence; this calculates a number of initial equilibrium points and automatically ADDs each one as an initial equilibria for another STEP NORMAL command-sequence (or even for a MAP command). Before the second STEP command (i.e. the STEP NORMAL command-sequence), you can chose a compositional condition as the new stepping axis-variable; then the STEP NORMAL command-sequence uses those created starting equilibrium points (at different temperatures) and steps in such a new compositional variable. As a result, these two sequential STEP commands together give a matrix of values.
SEPARATE_PHASES	Each phase is calculated separately. This option performs a stepping calculation in which all the entered phases in the system are calculated separately and in parallel at each step, i.e. all the phases are calculated for the same composition (if the phases exist for that composition). In this way, you can calculate how the Gibbs energy for a number of phases varies for varying compositions. This option is useful when you want to plot `Gm` curves versus composition for a given temperature, particularly for calculating the Gibbs energy differences for phases with miscibility gaps and for ordered phase which is never disordered (e.g. SIMGMA-phase, G-phase, MU-phase or alike). However, this option may sometimes be unable to find all the ordered phases and their disordered pairs during a STEP SEPARATE_PHASES calculation procedure; and it is thus recommended to always use the newer ONE_PHASE_AT_TIME option instead, for STEPPING heterogeneous systems with both ordered phases and their disordered pairs.
ONE_PHASE_AT_ TIME	One phase calculated at a time and repeated for all entered phase. It allows a stepping calculation for only one phase at a time individually (stepped over the whole stepping variable range), but the stepping is repeated for all the entered phase in the defined system. The STEP ONE_PHASE_AT_TIME command-sequence properly STEP in composition for an equilibrium heterogeneous system with both ordered phases and their disordered pairs, instead of calculating for the same composition for each of the phases at each composition step, and it at least starts the stepping in each of the phases in their default most-stable compositions (the major constitutions). Therefore, this option is always recommended for STEPPING

Syntax	step_with_options
	heterogeneous systems with both ordered phases and their disordered pairs.
EVALUATE	Specified variables evaluated after each step. This is an advanced option for experts of Thermo-Calc, when additional conditions (rather than the stepping variable) should be changed during a stepping calculation. It allows a stepping calculation in a single axis with simultaneous evaluation of one or more variables after each step. As variables can be used as conditions, it means that you can change the conditions during the stepping. After a successful STEP EVALUATE calculation, you can go to the POST module to manually define and plot various desired property diagrams. Specify the prompt: VARIABLE NAME(S): <VARIABLE NAME> The names of the variables that shall be evaluated after each step must be typed here. During a STEP EVALUATE calculation in some earlier TCC versions, it used to calculate all kinds of property diagrams, phase fraction plots, etc., some problems may occur, especially when new phases want to appear. These problems are simplified by the general improvement of convergence but a number of additional fixes are added to the STEP EVALUATE command-sequence to handle the problems. The miscibility gap test is automatically used during stepping (see the command *ADVANCED_OPTIONS* on page 125) if a phase has two or more composition sets. The miscibility gap test means that you can now start calculating from high temperatures in a steel, where the MC carbide is not stable; and during a STEP EVALUATE calculation procedure, the MC carbide first becomes metastable with a composition different from the metallic FCC phase and later also stable. One good case of using this advanced option is described in the command *INFORMATION* on page 5 with the subject as SOLIDIFICATION (Solidification Path Calculations). With this option, you can simulate the Ordinary Scheil-Gulliver Solidification Processes (i.e. non-equilibrium transformations) by changing the overall composition to the new liquid composition after each step, but without considering back-diffusion of fast-diffusing interstitial components in solid phases. See example 30A in the *Console Mode Examples Guide*. It is recommended to go directly to the SCHEIL module and run the desired Scheil-Gulliver type of simulations.

Syntax	step_with_options
MIXED_SCHEIL	Extended Scheil-Gulliver type solidification with back diffusion. This option leads to a stepping calculation (always with T as the stepping axis variable) for simulating Extended Scheil-Gulliver solidification processes (i.e. *partial-equilibrium transformations*) with considering back-diffusion of one or more fast-diffusing interstitial components (such as C, N, O, S, etc.) in formed solid phases; it also allows considering BCC-->FCC phase transformation (practically for steels) along the alloy solidification process. To ensure a successful stepping calculation of mixed Scheil-Gulliver simulation of solidification process of a defined alloy system with a certain initial overall composition, it is important to pay special attention to the following four aspects, before performing this advanced-option stepping calculation: For solution phases with possibility of miscibility gap existence, you have appropriately added the necessary composition set(s); You have already set the temperature condition as the stepping SET_AXIS_VARIABLE command, with a minimum and maximum temperature points as well as an appropriate s temperature step for the cooling process); It is equally important there must be a composition condition for each of the back-diffusion components (i.e. fast-diffusing interstitial components, such as C, N and/or other elements) that are already defined in terms of mole-fraction or mass-fraction, e.g. `X(C)` and `W(N)`. Otherwise, a stepping calculation with this option can fail. An initial equilibrium in which the LIQUID mixture phase is the only stable must be calculated [for this reason, it is always good to start with a relatively high temperature condition for the initial equilibrium calculation; and it is always necessary to reject or suspend the GAS mixture phase (if exists) before the C-E and STEP commands, to avoid its formation along with LIQUID]. In cases where fast-diffusing interstitial components are specified as back-diffusion components, there is a possibility to choose an option to `allow BCC>FCC` phase transformation (practically for steels) during the alloy solidification process. With this stepping calculation option, these prompts need to be specified: `FAST DIFFUSING COMPONENTS: <FAST DIFFUSION INTERSTITIAL(S)>` Specify one or more interstitial component(s) as the back-diffusion component(s). If there is only one fast-diffusing interstitial component (e.g. C or N), type its name at the prompt;

Syntax	step_with_options

	If there are two or more fast-diffusing interstitial components (e.g. C, N and S), always type their names on the same line at once (separated by an empty space in between, e.g. C N S);
	
	If there is no fast-diffusing interstitial component to be considered, type `NONE` to ignore back diffusion entirely, meaning a normal Scheil-Gulliver simulation is performed.
	
	Such specified back-diffusion interstitial component(s) must be appropriately defined according to the phase constitution definitions of some major alloy solution phases (e.g. FCC, BCC, HCP, etc.): These are located on the interstitial/vacancy sublattices in such alloy solution phases; otherwise, such a Scheil-Gulliver simulation may not make sense at all.
	
	`ALLOW BCC -> FCC ? /Y/: <Y OR N>`
	
	Type `Y` (or press <Enter> to accept the default) or `N` to allow or not allow the `BCC-->FCC` phase transformation (which is usually a typical phenomenon along the cooling processes of steels/Fe-alloys).
	
	During the STEP MIXED_SCHEIL calculation procedure, the simulated solidification path, including the locally-equilibrated phase assemblage (region), captured temperature condition (`T` in K), remaining liquid phase (`NL`, mole fraction), overall formed solid phases (`NS`, mole fraction) and latent heat formation (`NH`, J/mol) along the cooling process are shown after the information on starting temperature point of the alloy solidification process, e.g.
	
	`Solidification starts at  1743.15 K`
	`Phase Region from   1.744150E+03 for:      LIQUID`
	`Phase Region from   1.742525E+03 for:`
	`     LIQUID`
	`     FCC_A1#1`
	`       1742.5250          0.9960          0.0040          -1.1824`
	`       1742.4000          0.9795          0.0205        -202.1585`
	`. . . . . .`
	`Phase Region from   1.733150E+03 for:`
	`     LIQUID`
	`     BCC_A2`
	`     FCC_A1#1`

Syntax	step_with_options

	`1733.1500  0.3294  0.6707  -8032.6240`
	`1733.0250  0.3237  0.6763  -8095.1490`
	. . . . . .
	
	After a successful STEP MIXED_SCHEIL calculation, you can go to the POST module to manually define and plot various property diagrams for the alloy solidification process according to the chosen Scheil-Gulliver model, or to impose existing relevant experimental information or other types of calculated solidification results [e.g. Scheil-Gulliver simulation with or without considering fast-diffusing interstitial component(s), full-equilibrium approach, or Diffusion Module (DICTRA)-type simulation with moving-boundary conditions] onto the same plotted property diagrams. Normally, the solidus temperature (`T` in Celsius) is set as one of the axis variables (usually as Y-axis in most cases), while the other plotted quantity on the other axis can be the amount of overall formed solid alloy phases (`NS` in mole-fraction or `BS` in mass-fraction), the amount of remaining liquid phase (`NL` in mole-fraction or `BL` in mass fraction), the heat of latent along the solidification process (`NH` in J/mol or `BH` in J/gram), among many other properties in the solidified alloy phases or in the whole alloy system.
	
	However, you are always recommended to go directly to the SCHEIL module and run the desired Scheil-Gulliver type of simulations in a much easy, straightforward and automatic way.

`T-ZERO`	T0 lines between two specific phases calculated This option allows a stepping calculation of the so-called T0 (T-zero) line in a *diffusionless transformation* [where two specific partially-equilibrated phases have the same Gibbs energy, along a composition variable which has already set as the stepping variable with *SET_AXIS_VARIABLE* on page 164.
	
	The temperature condition cannot be set as the stepping variable if you want to make a STEP T-ZERO calculation.
	
	To ensure a successful calculation of T0 line between two specific phases after an initial equilibrium calculation in the current system, it is recommended to make a single T0 point calculation by the ADVANCED_OPTION T-ZERO command-sequence, prior to performing this STEP T-ZERO command-sequence, although this might not be always necessary for some systems.
	
	Specify these prompts:
	
	`NAME OF FIRST PHASE:  <PHASE A>`
	`NAME OF SECOND PHASE: <PHASE B>`

<table>
<tr><th>Syntax</th><th>step_with_options</th></tr>
<tr><td></td><td>The names of the two target phases, for which the Gibbs energies equal to each other at each point on the T0 line, must be typed at the above two prompts subsequently.

During the STEP T-ZERO calculation procedure, the calculated T0 values are shown after the corresponding scanned conditions (of the stepping composition variable), e.g.

<code>Phase Region from 1.000000E-01 for:</code>
<code>BCC_A2</code>
<code>FCC_A1</code>
<code>1.000000E-01 940.24</code>
<code>9.250000E-02 941.20</code>
<code>. . . . . .</code>
<code>2.500000E-03 977.61</code>
<code>7.500000E-09 979.34</code>
<code>Phase Region from 1.000000E-01 for:</code>
<code>BCC_A2</code>
<code>FCC_A1</code>
<code>1.000000E-01 940.24</code>
<code>1.075000E-01 939.62</code>
<code>. . . . . .</code>
<code>2.950000E-01 1084.87</code>
<code>3.000000E-01 1080.99</code>

After a successful STEP T-ZERO calculation, you can go to the POST module to plot the T0 line against the stepping composition variable or another varying axis, or to impose the calculated T0 line onto a normal phase diagram [normally plotted as a <code>T-X</code> isopleth].

See examples 23 and 41 in the Console Mode Examples Guide.</td></tr>
<tr><td><code>PARAEQUILIBRIUM</code></td><td>Paraequilibrium state between two specific phases is calculated with this. It allows a stepping calculation of so-called paraequilibrium lines in a partly partitionless transformation between two specific partially-equilibrated phases in an alloy system where the chemical potential for one or more fast-diffusing interstitial components but not for the substitutional components in two specified phases are equal.</td></tr>
</table>

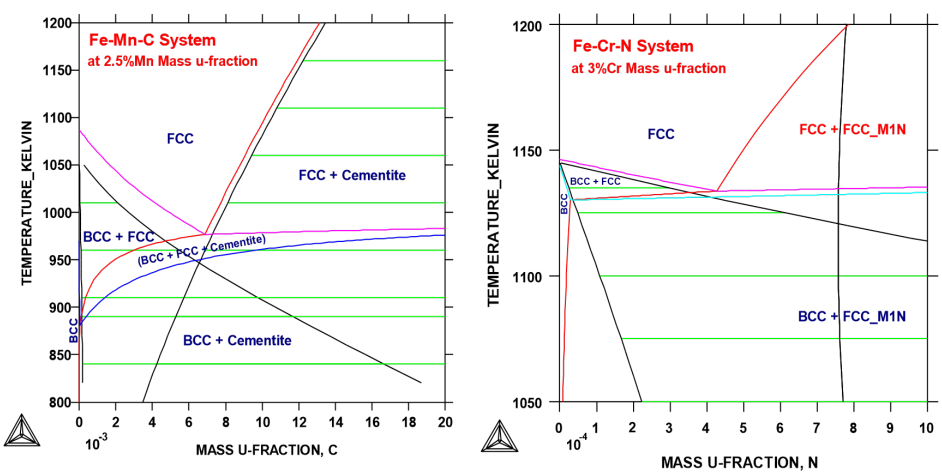
Syntax	step_with_options
	Therefore, the STEP PARAEQUILIBRIUM command-sequence ensures that the two partially equilibrated phases have the same chemical potential (but different contents) for one or more fast-diffusing interstitial components (such as C, N, O, S, etc., as individual or combined)], along varied temperature or along a composition variable (of the matrix or one substitutional component) which has already set as the stepping variable with *SET_AXIS_VARIABLE* on page 164. More than one interstitial components can be treated as fast-diffusion species (such as C, N, O, S, etc., as individual or combined) in a paraequilibrium stepping calculation. The composition condition for the interstitial components cannot be set as the stepping variable if you want to make a STEP PARAEQUILIBRIUM calculation. To ensure a successful stepping calculation of paraequilibrium states between two specific phases in a defined alloy system, it is important to first make a starting-point equilibrium calculation with an initial overall composition in the current system before performing this advanced-option stepping calculation; however, it is unnecessary to obtain an equilibrium in which either one or both of the target phases is stable. The initial overall composition must have a reasonable setting for the desired paraequilibrium calculation for the two target phases. This is especially true for cases where there are more than one interstitial components to be considered in the paraequilibrium state, because different interstitial components (for example C and N combined) may have significant different behaviours as partitioning into different structured phases; otherwise, for one chosen interstitial component the initial overall composition is OK for the paraequilibrium calculation between the specified two phases, but for other chosen interstitial component(s) it might be impossible to calculate the paraequilibrium state. Always check if the chosen `phaseA` and `phaseB` have the exact same definition of elements and if the chosen interstitial components are all in the vacancy sublattice sites of the two phases; otherwise the program cannot find the paraequilibrium state (as it is impossible to correctly calculate u-fractions). Always have a comprehensive understanding of the normal phase diagram for the currently investigated system, so that you make the appropriate choice of the phase pair and staring bulk composition for the system. Always set the status of the chosen interstitial components as SPECIAL using the command:

Syntax	step_with_options

	`CHANGE_STATUS COMPONENT <interstitial component> = SPECIAL`
	
	By doing this, you get a clear picture on u-fractions of various substitutional and interstitial components, which are different from the overall composition in the system. The SPECIAL status means that specified component(s) are not included in summations for mole or mass fractions. Therefore, all the composition variables plotted from the paraequilibrium calculations are u-fraction related quantities.
	
	Always make a single-point paraequilibrium calculation by the command-sequence of ADVANCED_OPTIONS PARAEQUILIBRIUM, prior to performing this STEP PARAEQUILIBRIUM command-sequence, although this might not be always necessary for some systems.
	
	With this stepping calculation option, these prompts need to be specified:
	
	`NAME OF FIRST PHASE:  <PHASE A>`
	
	`NAME OF SECOND PHASE: <PHASE B>`
	
	The names of the two target phases `A` and `B`, between which the paraequilibrium states establish, must be typed at the above two prompts subsequently or on the same (first) line at once (separated by an empty space, e.g. `FCC BCC`, `FCC#2 M23C6`).
	
	It is important to understand that this is the calculation of a paraequilibrium state between the two specified phases.
	
	Particularly, there are four issues which you need to consider:
	
	1. Both chosen phases must have similar interstitial/vacancy sub-lattices where the fast-diffusion interstitial component(s) occupy.
	2. The choice on the target phase pair must be reasonable for the defined system and specified initial overall composition.
	3. Both target phases should have phase constitution definitions that cover all the defined substitutional and interstitial components of the current alloy system; or
	4. It is impossible to calculate the paraequilibrium state between the target phase pairs with given interstitial component(s) in the currently defined system.

Syntax	step_with_options
	FAST DIFFUSING COMPONENT: /C/: <INTERSTITIAL COMPONENT(S)> FAST DIFFUSING COMPONENT: /NONE/: <INTERSTITIAL COMPONENT(S)> The name(s) of the fast-diffusing interstitial component(s) (`C` as the default single component) must be given either one by one at each of the above prompts subsequently or in a combination (separated by an empty space in between, e.g. `C N S`) at the same (first) prompt; otherwise, press <Enter> key to start the paraequilibrium line stepping-calculation. Note the following: Such specified fast-diffusing interstitial component(s) must be appropriately defined according to the phase constitution definitions of the two selected phases: These must be located on the interstitial/vacancy sublattices in both chosen phases. If there is only one fast-diffusing component which is carbon, press <Enter> to accept the default value (`C`) at the first prompt; if the single fast-diffusing interstitial component is another element (e.g. N), appropriately type interstitial component the name at the first prompt; If there are two or more fast-diffusing components (e.g. `C` and `N`), type their names at the above prompts subsequently or at the same (first) prompt (separated by an empty space, such as `C N`); To finish the input of fast-diffusing elements, accept `NONE` at a repeated prompt; If NONE or a non-existing component name is typed at the first prompt, it means no back diffusion is to be considered, and the para-equilibrium calculation is thus cancelled entirely. During the STEP PARAEQUILIBRIUM calculation procedure, the calculated paraequilibrium states include the: • corresponding stepping conditions (e.g. temperature in K, or the stepping composition variable of the matrix component or of one of substitutional components) which has already set as the stepping variable by the SET_AXIS_VAR command); • amounts of the phase `A` in mole number [i.e. `NP(phaseA)`]; • amounts of the phase `B` in mole number [i.e. `NP(phaseB)`]; • contents of the interstitial component(s) in the phase `A` expressed in u-fractions [i.e. `u-f(phaseA,component(s))`];

Syntax	step_with_options
	• contents of the interstitial component(s) in the phase `B` expressed in u-fractions [i.e. `u-f(phaseB,component(s))`]; • `LNACR(component)` value(s) for the interstitial(s). After a successful STEP PARAEQUILIBRIUM calculation, you can go to the POST module to plot a paraequilibrium phase diagram, or to impose the calculated paraequilibrium states onto a normal phase diagram [normally plotted as a `T-X` isopleth with one of the considered interstitial component(s) as the x-axis variable]. See examples 23 and 42 in the *Console Mode Examples Guide*. However, the stepping calculation procedure for the currently specified paraequilibrium states between the two specified phases may find difficulties at some steps due to some possible numerical problems (normally regarding to the u-fractions of the interstitial components under some composition ranges), or it may completely fails. In the latter case, it implies that the chosen target phase pair may be unreasonable for the defined alloy system or for the defined initial overall composition, or one or both phases may have inappropriate phase constitution definitions regarding the specified interstitial component(s) and substitutional components. Then, you must either modify the settings of initial overall composition or specify the reasonable target phase pair with an appropriate choice of the fast diffusion interstitials in the defined alloy system.



The figure above shows the plot of a paraequilibrium calculation of isopleths: Formation of para-pearlite in two Fe-based alloys, with equilibrium results appended.

The calculated paraequilibrium phase boundaries (black lines) are shown with tie-lines (in green-colour). In the Fe-2.5Mn-C alloy system, C is considered as fast diffusing interstitial component in the BCC+FCC and FCC+Cementite paraequilibrium stepping calculations; while in the Fe-3Cr-N alloy system, N is treated as fast diffusing interstitial components in the BCC+FCC and FCC+FCC_M1N (i.e. FCC#1+FCC#2) paraequilibrium stepping calculations.

During a STEP calculation, the value of the stepping axis variable for each calculated equilibrium is listed and also the set of stable phases.

You can stop the mapping of a line by pressing a single CTRL-A (under MS Windows) or CTRL-C (under Linux). This can be useful in order to stop a longish calculation without losing what is already calculated.

If there is any convergence problem in finding a stable solution at any stage during a calculation procedure enforced by a STEP_WITH_OPTIONS command-sequence, this message displays:

```
Convergence problems, increasing smallest site-fraction from 1.00E-30   to hardware
precision 2.00E-14. You can restore using SET-NUMERICAL-LIMITS
```

This implies that smallest site fraction in the current POLY3 workspace is automatically increased from the default value `1.00E-30` to the hardware-dependent precision (under Linux, as `2.00E-14`). For other subsequent POLY-module calculation in the current TC run, you can use the POLY command *SET_NUMERICAL_LIMITS* on page 168 to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

Complex miscibility gaps of various solution phases are automatically detected during a stepping calculation, and two or more composition sets for each of such solution phases are automatically generated as well, through an automatic Global Minimization Test procedure (by specifying a `Global Test Interval` value and confirming the `Automatically add initial equilibria` in the ADVANCED_OPTION STEP_AND_MAP command-sequence). As a particular case, such an automatic Global Minimization Test ensures that you can now start calculating from high temperatures in an austenitic steel (metallic FCC_A1 solution phase) where the MC carbides/nitrides/carbonitrides (i.e. the C-/N-rich sides of the FCC_A1 miscibility-gap, often (while not always) being referred to as `FCC_A1#2`, `FCC_A1#3`, etc.) are not stable, and during the MAP command the MC carbides/nitrides/carbonitrides may first become metastable with a composition different from the metallic FCC_A1 phase and later also stable.

Occasionally, it may be necessary to have a specified increment other than the default value for the defined stepping axis-variable, in order to calculate/plot a smoother property diagram.

To get a complete property diagram, sometimes (especially in some earlier Thermo-Calc versions) it may also be required to have more than one starting equilibrium-points (under another starting equilibrium condition), and repeat the stepping calculation over the same stepping axis-variable range (while without over-writing the same POLY3 workspaces via a SAVE_WORKSPACE command). However, using the current STEP procedure such a requirement become less necessary.

TABULATE

For any entered table, this command gives a table of values from equilibria calculated with a STEP command. It works only for a table that is entered in either the POLY or POST module.

Unlike the TAB module's command (also called *TABULATE* on page 253) it is not possible to plot columns from the tabulated tables. For this use normal POST commands.

Syntax	TABULATE
Prompts	`NAME: <NAME OF A TABLE ENTERED IN EITHER POLY OR POST>` Specify a table name that is entered in either the POLY or POST module.
	`OUTPOUT ON SCREEN OR FILE /SCREEN/: <FILE NAME OR RETURN>` Specify a file name if you want to save on a file the table values along the defined STEP calculation, or press <Enter> if want to see the table values displayed.

POST Commands

The POST module (post-processor) is available from the POLY module. See *POLY_3 Commands* on page 117 for other information about the POLY module. Read *Using the POST Module* on page 190 before beginning.

In this section:

Using the POST Module

Use the POST module after a *STEP_WITH_OPTIONS* on page 172 or *MAP* on page 158 calculation to generate *graphical* or *tabulated* presentations on screen or into files.

The graphical files are available in different formats (e.g. PS, EMF, PNG, BMP, PDF, JPG, TIF) or data forms (e.g. EXP, TAB and WRL). The graphical output can be phase diagrams, property diagrams, diffusion profiles and other types of plots. The tabulated presentation is into a basic text file (TXT) or a spreadsheet (e.g. an MS Excel file with an extension of .XLS).

The POST module is available through these modules:

- POLY
- TAB
- BIN
- TERN
- POT
- SCHEIL
- POURBAIX

Within the POST module, you can choose any state variable, any derived variable, or any entered symbol (functions or variable) as the X/Y-axis.

Also see *Thermodynamic Variables and Units* on page 269.

When a diagram is plotted, appearance parameters for defining a high-standard graph can be further specified, e.g. curve labelling options, diagram titles and subtitles, plot size, axis length, axis types, axis-tic type, tie-line status, automatic or manual scaling and zooming, semi-automatic or manual labelling on phase boundaries and phase regions, graphic formats, text fonts, colours, raster plot, etc.

You can append experimental data onto a plotted diagram. You can also save the coordinates onto a textual file, which can be edited and used as an experimental data file to be merged onto another diagram or as a part of a setup file for PARROT-module assessment.

Phases can be suspended or restored in a plotted diagram. The reference states for components can also be modified for the resulted diagrams. Moreover, you can translate a plotted property diagram (after a stepping calculation) into a tabulated form, such as a simple list on screen or a textual file, and a spreadsheet .

ADD_LABEL_TEXT

Add a text to an area in a phase diagram or a property diagram. The label is written starting from the specified coordinates. Optionally, you can also let the program automatically add the text by first calculating an equilibrium state at the specified coordinates and then making a label out of the stable phase names.

The label texts and coordinates, either added by this command or modified with *MODIFY_LABEL_TEXT* on page 200 (or manually edited in the **Console Results** window), are stored in the workspace, and these may later be saved onto an *.EXP file (with *MAKE_EXPERIMENTAL_DATAFILE* on page 199).

In some cases such an optional calculation for automatic phase region labelling may not work. It is not possible to automatically add labels for certain sets of axes, like entropy, enthalpy, pH, Eh, etc.

Adding and Editing Plot Labels on page 92 in the *Thermo-Calc User Guide*.

Syntax	ADD_LABEL_TEXT
Prompts	`GIVE X COORDINATE IN AXIS UNITS: <VALUE OF THE X COORDINATE>` Specify the value of the X coordinate where the label is to start.
	`GIVE Y COORDINATE IN AXIS UNITS: <VALUE OF THE X COORDINATE>` The labels disappear if any axis is changed afterwards. Specify the value of the Y coordinate where the label is to start.
	`AUTOMATIC PHASE LABELS? /Y/: <Y OR N>` If you answer Y, the program automatically calculates the equilibrium at the given coordinates and generates a label with the names of the stable phases. However, the automatic calculation works only for phase diagrams that are mapped with two axes in the POLY module, and sometimes it may not work correctly especially for complex heterogeneous interaction systems; in such cases, a message displays.

	`TEXT: <TEXT FOR THE LABEL>`
	
	If you answer N enter text at the prompt. You can enter simple text, use LaTeX, rotate the text, as well as adjust the font size.
	
	LaTeX: You can use LaTeX commands to enter text and symbols. First enter the command followed by the expression or string, for example to display Al2O3 as subscripts:
	
	`\latex Mole-fraction Al_2O_3`

Syntax	ADD_LABEL_TEXT

For a list of supported LaTeX commands, see http://www2.ph.ed.ac.uk/snuggletex/documentation/math-mode.html .

By default when using the LaTeX command, you are in math mode and Thermo-Calc automatically interprets the entered text as math. The above link uses the dollar sign ($) in all the examples but this is NOT required. For example, if you enter `$E=mc^2$` this is displayed including the $ instead of the equation.

To add a mixture of plain text and equation text, you can exit math mode using the string `\textrm` followed by the text to display enclosed in curly brackets. For example `\latex \textrm{Mole-fraction }Al_2O_3` displays this label Mole-fraction Al_2O_3 .

Rotate the text: If you want to rotate the label, start a text string with the rotation command (which is case sensitive and must be in lower case letters) then enter the number of degrees to rotate the label. For example, to rotate the label 45 degrees enter:

```
\rotation 45
```

To rotate the text and use LaTeX, the rotation command must be used first. For example:

```
\rotation 45 \latex Mole-fraction Al_2O_3
```

Rotates the text 45° and includes subscripts in the title Mole-fraction Al2O3.

```
TEXT SIZE: /.35/: <SIZE FOR THE LABEL>
```

A smaller size of the text may be necessary to fit the label text into the diagram. A default size is given using the latest size in adding label texts, or .35 if the command is called for the first time. A recommended size is e.g. 0.25.

These changes to plot labels can also be manually edited in the **Console Results** window.

APPEND_EXPERIMENTAL_DATA

Add experimental data and text onto a calculated diagram. This is done by placing the experimental data and text on a formatted textual file (always having an extension of EXP) that is prepared outside of

Thermo-Calc software (Console Mode and Graphical Mode), according to the syntax of the DATAPLOT Graphical Language.

The picture generated from the data is superimposed on the ordinary graphical output from the postprocessor. Such an experimental data file (*.EXP) may be created by an ordinary textual editor.

Another important usage of this command is to superimpose plots from several independent calculations. For this purpose, there is a command *MAKE_EXPERIMENTAL_DATAFILE* on page 199, which dumps a calculated diagram on an EXP file according to the DATAPLOT syntax. With the aid of a normal text editor, many such files may be merged together. Remember that you can only have one prologue section on each such EXP file.

Also see the *DATPLOT User Guide* included with this documentation set.

Syntax	APPEND_EXPERIMENTAL_DATA
Prompts	`USE EXPERIMENTAL DATA (Y OR N) /N/: <Y OR N>` Specify whether the data from an experimental data file should be included in the next plot. If the answer is `c`, no experimental data are plotted. An **Open** window displays to specify the name of the file with the experimental data.
	`PROLOGUE NUMBER: /0/: <PROLOGUE NUMBER>` Select which prologue to use. In a prologue, you can, for example, give the scaling of an axis, the axis texts, and so on. By answering -1, a list of all prologues on the file is given; and by 0, no experimental data are appended.
	`DATASET NUMBER(S) /-1/: <DATASET NUMBER(S)>` Select from which dataset(s) data should be read. Several datasets may be given separated with commas or spaces. By answering -1, a list of all datasets on the file is given; and by 0, no experimental data are appended.

CHANGE_LEGEND

When there is a legend in the diagram you can append constitution and/or ordering descriptions to the phase name.

LIST_EQUILIBRIUM on page 156 also displays `ORD` (ordered) or `DISORD` (disordered) (when available).

In some cases the legend only shows the phase name for a line when the *SET_LABEL_CURVE_OPTION* on page 209 option is set to F.

Syntax	CHANGE_LEGEND
Prompt	SET SUFFIX FOR PHASE NAMES IN LEGEND? (ALL,NONE,CONSTITUTION,ORDERING) /NONE/: When you select None the legend includes the standard information about the phases. When you select All the Constitution description and the Ordering description (when there is ordering) are also included. When you select Constitution, this is added to the standard description. When you select Ordering and when there is ordering, this is added to the standard description.
	Also see *About Legend Styles* on page 91 in the *Thermo-Calc User Guide* for examples of these labels.

DUMP_DIAGRAM

After a Thermo-Calc graph is created (and displays with the PLOT_DIAGRAM command) you can save (dump) a diagram to a file so you can quickly print it. It is similar to the PRINT_DIAGRAM command.

Similar to *PRINT_DIAGRAM* on page 201, it does not matter what graphic device is set up for the graph, this command is always able to convert the graph shown on screen to the selected graphic format.

Syntax	DUMP_DIAGRAM
Prompt	OUTPUT FORMAT /: <FORMAT OR RETURN> Specify an appropriate graphic format for dumping the graph or pressing <Enter> to accept the default format.
	A **Save** window opens if you have not specified a filename. Select a directory path and enter a **File name**. If a file with the specified name already exists, then you are prompted whether to overwrite the old file or not. The third argument of the command allows you to specify whether to overwrite (Y) the old file or not (N).

After the printing process is finished, the temporarily set graphic device for conversion and printing is switched back to the one set by the command *SET_PLOT_FORMAT* on page 210, so that you can save the graph in file(s) using *PLOT_DIAGRAM* on page 200.

ENTER_SYMBOL

Symbols are a useful feature of the POLY and POST modules to define quantities that are convenient. Symbols can be constants, variables, functions or tables.

Functions or tables (with defined functions as values) can be entered in the POST module after a stepping or mapping calculation, for purposes of plotting such entered functions or tables as axis variables.

Within the POLY module, symbols are normally defined prior to an equilibrium calculation (enforced by a C_E command), stepping calculation (enforced by the command *STEP_WITH_OPTIONS* on page 172) or mapping calculation (enforced by *MAP* on page 158). These can be entered after an equilibrium calculation; however, for defined functions, variable or tables, it requires using *EVALUATE_FUNCTIONS* on page 155 before showing the corresponding values in the calculated equilibrium state.

The symbols entered in the POST module are not saved in the currently-loaded POLY3 workspaces. Therefore, if you want to apply such symbols in other similar calculations for the same defined system, you must use the ENTER_SYMBOL command prior to the STEPPING or MAPPING calculation in the POLY module.

Also see *ENTER_SYMBOL* on page 72 for the GIBBS Module.

See example 44 in the *Console Mode Examples Guide*for an example of using variables and functions.

Syntax	ENTER_SYMBOL
Prompt	`CONSTANT, VARIABLE, FUNCTION OR TABLE? /FUNCTION/: <KEYWORD>` The `Keyword` can be specified as `CONSTANT`, `VARIABLE`, `FUNCTION` or `TABLE`. • CONSTANTS can only be entered once and is a means of using a name for a numeric value. For example, the value of 1 atmosphere in Pascal can be denoted by `P0` after the command ENTER CONSTANT `P0=101325`. Defined constants can be used as values in condition assignments, for example, `SET-COND P=P0`. • FUNCTIONS are expressions of state variables or other functions. These expressions are saved, and whenever a function value is requested all functions are evaluated. The reason for this is that they may depend on each other. • VARIABLES are similar to functions because they can also be expressions of state variables. However, contrary to functions, they are only evaluated when they are entered or if they are explicitly named in an *EVALUATE_ FUNCTIONS* on page 155 command. It is possible to enter a variable with a new expression any time. This expression is evaluated directly and its value stored as the value of the variable. Defined variables can be used as values in the SET-CONDITION command. • TABLES are used for listing results from the STEP or MAP commands. A table consists of a list of any number of state variables, functions or variables. Defined tables can also be used in the post-processor POST. There is a special connection between tables and variables. If a variable is used in a table, it is evaluated for each line of the table in the TABULATE command or when the table is used in a plot.
	`NAME: <NAME OF THE SYMBOL>` Each symbol has a unique name that must start with a letter and can have maximum 8 characters. Legal characters include letters (either UPPER or lower case), digits and underscore _. Any other special character, such as parentheses (and), plus +, minus -, slash / or \, full stop (.), are illegal for symbol names. You can enter the symbol name and the value or function on the same line, these must be separated with an equal sign =, for example, `TC=T-273.15` or `T_C=T273.15` which stands for a definition of temperature in Celsius. Otherwise, these questions are asked.

Syntax	ENTER_SYMBOL
	For different types of symbol (constant, function, variable or table), the questions have different prompts.

	`FUNCTION: <DEFINITION FOR A FUNCTION OR VARIABLE>`
	Functions and *variables* are evaluated from an expression of state variables or other functions, constants or variables. The expression is a FORTRAN-like expression and operators +, -, *, = and ** can be used (** only with integer powers).
	Unary functions like `LOG`, `LOG10`, `EXP`, `SIN`, `COS` and `ERF` can also be used. An expression can be continued on more than one line. An expression should be terminated by a semicolon (;) or an empty line (by pressing <Enter> at the next prompt).
	Examples of function expressions:
	• `GM(LIQUID)`: The Gibbs energy of liquid per mole component
	• `H.T/4.184`: The heat capacity of the system in calories
	• `ACR(CR)/X(FCC,CR)`: The activity coefficient for Cr in FCC
	• `T-273.15`: The temperature in Celsius
	`&: <CONTINUATION OF THE DEFINITION FOR THE SYMBOL>`
	The ampersand & means that you can continue to write the function on the new line if one line is not enough for the function. If you finish the function press <Enter> again.

	`VALUE: <VALUE FOR A CONSTANT>` A *constant* can only be assigned a numeric value once.
	`VALUE OR EXPRESSION: <VALUE OF EXPRESSION FOR A VARIABLE>` A *variable* can be assigned a numeric value or an expression. An expression is evaluated immediately and discarded. Only the numeric value is kept. This gives a possibility to save a value between calculations with different conditions because all state variables and functions are evaluated for the new conditions.

	`VARIABLE(S): <VARIABLE(S) IN A TABLE>`
	A *table* consists of a list of state variables or functions. One way to obtain results from a STEP command is through a table.
	Example:
	`ENTER TABLE K=T,X(LIQ,C),X(LIQ,CR),ACR(C)`
	Which means that the table called K contains four columns, i.e. the temperature, the mole

Syntax	ENTER_SYMBOL
	fractions of C and Cr in the LIQUID phase, and the activity of C. To show the temperature in Celsius in a table, give the command ENTER FUNCTION `TC=T-273`; and then use the symbol `TC` in the table. `& <CONTINUATION OF THE DEFINITION FOR THE TABLE>` The ampersand `&` means that you can continue to write the table on the new line if one line is not enough for the table. If you finish the table press <Enter> again.

FIND_LINE

Identifies various property curves on a plotted property diagram (after a stepping calculation); it also works well in find stable phase regions on a phase diagram (after a mapping calculation).

Syntax	FIND_LINE
Prompts	`X COORDINATE /.484623611/: <VALUE OF THE X COORDINATE>` Specify the `value of the X coordinate` where you want to find the details of property curve (or phase region).
	`Y COORDINATE /.5153737962/: <VALUE OF THE Y COORDINATE>` Specify the `value of the Y coordinate` where you want to find the details of property curve (or phase region). The POST module displays the identified property (for a property curve on a property diagram), or the identified stable phase-assemblage (for a phase region on a phase diagram).

LIST_DATA_TABLE

This command is used for listing various properties (i.e. the Y-axis variable on a property diagram) that change with an independent variable (i.e. the X-axis variable on a property diagram), which have already been defined in the POST module (and often are plotted on the **Console Results** window), but only after a STEP calculation.

The listing output can be either on screen or in a spreadsheet file in the MS Excel format (under specified file name with the default file extension of XLS), which can be further edited or used for graphical handling by opening the file (using the MS-Excel program) outside of the Thermo-Calc software system.

Syntax	LIST_DATA_TABLE
Prompt	OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME OR RETURN > Specify a file name if you want to save on a spreadsheet file (in the MS Excel format, with the default extension of XLS) on which the already-defined properties on a property diagram (after a STEP calculation) is saved, or press <Enter> if want to see a simple textual table on screen (for the already-defined properties that change alone with an independent variable). This command only works for a property diagram after a stepping calculation but not for any phase diagram after a mapping calculation.

LIST_PLOT_SETTINGS

Syntax	LIST_PLOT_SETTINGS
	Lists the present values of most parameters specifying the type of diagram to be plotted.

LIST_SYMBOLS

For both the POLY and POST modules, list the definitions for all constants, functions, variables and tables.

In order to find the value of a function or variable, use *SHOW_VALUE* on page 172 or *EVALUATE_FUNCTIONS* on page 155. A table is tabulated with the TABULATE command.

Syntax	LIST_SYMBOLS
	The defined variables are listed up together with the defined functions, but variable names are followed by a percentage sign %.

MAKE_EXPERIMENTAL_DATAFILE

Write the graphical information onto screen or a formatted text file (always having an extension of EXP) using the DATAPLOT graphical format. In order to merge two or more diagrams from separate calculations, you can write them out on files with this command, and then add/edit them together with a normal textual editor.

Also see the *DATPLOT User Guide* included with this documentation set.

Syntax	MAKE_EXPERIMENTAL_DATAFILE
Prompt	`OUTPUT TO SCREEN OR FILE /SCREEN/: <NAME OF A FILE OR RETURN>` Specify the name of the desired file. Accept `SCREEN` (press <Enter>) for a list Otherwise, a file name where the graphical information is written must be given here. The default file extension is EXP (under MS Windows) or exp (under Linux).

MODIFY_LABEL_TEXT

Move a label created with the command *ADD_LABEL_TEXT* on page 190 to another position, or replace its text with another one.

The label texts and the coordinates, either added by ADD_LABEL_TEXT command or modified by this command or manually edited on the Console Results window, are stored in the workspace, and can be saved onto an *.EXP file (through *MAKE_EXPERIMENTAL_DATAFILE* on the previous page).

Syntax	MODIFY_LABEL_TEXT
Prompts	`WHICH LABEL TO MODIFY? /#/: <NUMBER OF THE LABEL>` Before this prompt, all the labels created by the ADD_LABEL_TEXT command is listed with an identifying number. Specify the `number of the label` you want to modify. The default one (#) is the last added label.
	`NEW X COORDINATE: /XXX/: <NEW X POSITION>` Specify the `new X position`. The previous X coordinate is displayed. Press <Enter> to accept.
	`NEW Y COORDINATE: /YYY/: <NEW Y POSITION>` Specify the `new Y position`. The previous Y coordinate is displayed. Press <Enter> to accept.
	`NEW TEXT /ABCDEFGH.../: <NEW LABELING TEXT>` Specify the `new labeling text`. The previous text is displayed. Press <Enter> to accept. The new label text must not be longer than the previous text.

PLOT_DIAGRAM

With this command, a calculated/plotted diagram is displayed if the default graphic device has already been defined by the *SET_PLOT_FORMAT* on page 210 command. It can also be saved onto a file.

The diagram axes must be set first.

The functionality of this command is shared in two ways:

- in PLOT_DIAGRAM for normal screen display and file saving and
- in *PRINT_DIAGRAM* below (under MS-Windows environment) for direct hard-copy printing.

Syntax	PLOT_DIAGRAM
Prompt	`OUTPUT TO SCREEN OR FILE /SCREEN/: <NAME OF A FILE OR RETURN FOR SCREEN>`
	Specify the name of the desired file (Note that under Linux, the prompt is `PLOTFILE /SCREEN/`). Consequently, a file with a proper extension for the selected graphic format (set by the SET_PLOT_FORMAT command) is saved under the current working directory. If no extension is given in the file name, the default extension e.g. ps for Postscript portrait/landscape modes, P7 or p7 for HPGL landscape, P8 or p8 for HPGL portrait, etc.) is automatically given to the file name. If the default device is set (by the command SET_PLOT_FORMAT), you can either accept `SCREEN` to plot the graph in the Thermo-Calc Graph window on screen by pressing <Enter>, or save it on a file by giving a file name.

PRINT_DIAGRAM

Syntax	PRINT_DIAGRAM
	Print with high quality to a file after a Thermo-Calc graph is created (and is displayed with *PLOT_DIAGRAM* on the previous page). It has the same function as the Print Plot option on the Console Results windows.

QUICK_EXPERIMENTAL_PLOT

This is similar to *APPEND_EXPERIMENTAL_DATA* on page 192 but can be used when there is no graphical information to be plotted in the POLY3 workspace. It defines a pair of axis, sets the axis labels to X and Y, and scales both X- and Y-axes between 0.0 and 1.0 unless a prologue is read from the DATAPLOT (*.EXP) data file.

Also see the *DATPLOT User Guide* included with this documentation set.

Syntax	QUICK_EXPERIMENTAL_PLOT
Prompts	PROLOGUE NUMBER: /0/: <PROLOGUE NUMBER> An **Open** window displays. Select the prologue to use. In a prologue, you can, for example, give the scaling of an axis, the axis texts, and so on. By answering -1, a list of all prologues on the file is given; and with 0, no experimental data are appended.
	DATASET NUMBER(S) /-1/: <DATASET NUMBER(S)> Select from which dataset(s) data should be read. Several datasets may be given separated with commas or spaces. By answering `-1`, a list of all datasets on the file is given; and with `0`, no experimental data are appended.

REINITIATE_PLOT_SETTINGS

All parameters describing the diagram are given default values.

Syntax	REINITIATE_PLOT_SETTINGS
	All plot settings made in the POST module are deleted, and it returns to the initial settings when the POST module is entered.

RESTORE_PHASE_IN_PLOT

Restores a phase previously suspended from the plot with *SUSPEND_PHASE_IN_PLOT* on page 216. It works only after at least one phase is suspended.

Syntax	RESTORE_PHASE_IN_PLOT
Prompt	PHASE NAME: <NAME OF A PHASE> Give the name of the phase to be restored.

SELECT_PLOT

Select a plot within the current Results tab in the Console Results window.

Syntax	SELECT_PLOT
Prompt	PLOT <N/NEW> Where `N` is a positive integer. Enter the number of the plot that is selected in the **Console Results** window (in the currently selected **Results** tab). If a plot with that number does not exist, it is created. The default parameter value is `NEW`. If this is entered, then a new plot is created. Create a new plot can also be done by pressing Ctrl+Shift+T.

SET_AXIS_LENGTH

Change the real length of an axis in inches. The default number of tic-marks on an axis is 10 when the relative length is 1. The number and units per tic-mark must be a multiple of 1, 2 or 5 to obtain a reasonable scaling of an axis.

Syntax	SET_AXIS_LENGTH
Prompts	`AXIS (X, Y OR Z): <NAME OF AN AXIS>` Specify which axis to set the length.
	`AXIS LENGTH /11.5/: <NEW RELATIVE AXIS LENGTH>` Specify the new real axis length in inches. The relative length 1 corresponds to 10 tic-marks on the axis.

SET_AXIS_PLOT_STATUS

Specify whether the axis title texts and axis label texts are to be plotted or not on a diagram. It does not work for the axis lines and tic-marks. If you want to plot a diagram without tic-marks, use *SET_TIC_TYPE* on page 215 first.

This can be used to merge different diagrams on a pen-plotter or to obtain the diagram faster. The default status is that all the set axes are to be plotted.

Syntax	SET_AXIS_PLOT_STATUS
Prompt	`AXIS PLOT (Y OR N) /Y/: (Y OR N)` Specify to plot axis texts (`Y`) or no (`N`).

SET_AXIS_TEXT_STATUS

Change the axis text from the automatic text given by the axis specification to a text.

Syntax	SET_AXIS_TEXT_STATUS
Prompts	`AXIS (X, Y OR Z): <NAME OF AN AXIS>` Specify which axis text status to change.
	`AUTOMATIC AXIS TEXT (Y OR N) /N/: <Y OR N>` Specify if automatic axis text is to be used (`Y`) or (`N`). If you answer `N`, enter axis text.
	`AXIS TEXT: <ENTER AXIS TEXT>` You can enter simple text, use LaTeX, rotate the text, as well as adjust the font size.

Syntax	SET_AXIS_TEXT_STATUS
	Also see *ADD_LABEL_TEXT* on page 190.

SET_AXIS_TYPE

Change the axis type among linear, logarithmic and inverse axis.

Syntax	SET_AXIS_TYPE
Prompts	`AXIS (X, Y OR Z): <NAME OF AN AXIS>` Specify which axis to change the axis type.
	`AXIS TYPE /LINEAR/: <NEW AXIS TYPE>` Specify which new axis type to set. Select `LIN`ear (default), `LOG`arithmic or `INV`erse. Only the three first characters are relevant.

SET_COLOR

Some devices support colours or line thickness. With this command, select different colors or line types on some types of information on a diagram.

This command can fail if a chosen device does not support colors or line thickness.

This command gives four sequential prompts (i.e. Text and axis, Invariant equilibria, Tie-lines, and All other lines), with the Keyword and default option (both listed below). Depending on if the switched-on graphic device (by the SET_PLOT_FORMAT command) supports colors or line types, the POST automatically switches on the appropriate Keyword.

Syntax	SET_COLOR

Prompt	`TEXT AND AXIS KEYWORD /DEFAULT OPTION/`
	`<RETURN OR NEW OPTION>`
	`INVARIANT EQUILIBRIA KEYWORD /DEFAULT OPTION/: <RETURN OR NEW OPTION>`
	`TIE-LINES COLOR KEYWORD /DEFAULT OPTION/: <RETURN OR NEW OPTION>`
	`KEYWORD OF ALL OTHER LINES /DEFAULT OPTION/: <RETURN OR NEW OPTION>`
	This is usually for the solubility lines.

Syntax	SET_COLOR
	The `Keyword` is either `Color` or `LineType`, depending on if color or line type is supported by the switched-on graphic device (by the *SET_PLOT_FORMAT* on page 210 command); the POST module *automatically* switches on the appropriate `Keyword`.

Color Options

The options for `Color` are:

BACKGROUND	BLUE	PURPLE	GRAY
FOREGROUND (default)	YELLOW	GOLD4	ORANGERED3
RED (default for invariant equilibria)	MAGENTA	TURQUOISE4	MAROON
GREEN (default for tie lines)	CYAN	PINK	PLUM
	SIENNA	OLIVEDRAB	SEAGREEN
	ORANGE1	CORAL1	USERDEF

LineType Options

The options for `LineType` are:

INVISIBLE	VERY_THICK (default for invariant equilibria)
NORMAL (default)	THIN (default for tie lines)
THICK	VERY_THIN
DASHED	DOTTED

SET_CORNER_TEXT

Adds text to the corners on a plotted diagram. Normally, you can write such texts as a subtitle.

The primary database (not the appended ones) used in calculations and the calculated conditions are automatically plotted at the upper-left corner as a part of the header on all plotted diagram. Unless the plotting option WRITE CONDITIONS is switched off by the command *SET_PLOT_OPTIONS* on page 211, the calculated conditions are always written on plotted diagrams; and only when the plotting option PLOT HEADER is switched off by the command SET_PLOT_OPTION, the used database also always appear. Under such circumstance, avoid writing texts at the UPPER_LEFT corners.

Syntax	SET_CORNER_TEXT
Prompt	`CORNER /LOWER_LEFT/: <RETURN OR NEW OPTION>` Choose an option: • LOWER_LEFT • UPPER_LEFT • UPPER_RIGHT • LOWER_RIGHT • TOP_OF_TRIANGLE
	`TEXT: <TEXTS AS A SUBTITLE OR A NOTE>` Write the texts that are to be written on the specified diagram corner.

SET_DIAGRAM_AXIS

Specify an axis variable for a plot or a listing. To be able to plot a diagram, at least two axis variables (`X` and `Y`) must be specified. In the Thermo-Calc Console Mode, three axes (`X`, `Y` and `Z`) can be specified for a diagram.

The axis variables in the plot can be different from the number of axis used for the mapping.

If you want to plot a phase diagram with tie-lines in the plane after a MAP calculation, then the composition axis must be `mole-fraction` or `weight-fraction`, or `mole-percent` or `weight-percent` of one specific component. You must not use the stable variable `x(comp)` or `w(comp)` even if that is what you used when calculating the diagram, because that only gives one side of the two-phase region. It may be sufficient to indicate that mole-fraction is the same as the stable variable `x(*,comp)`, and weight-fraction is the same as the stable variable `w(*,comp)`.

If you plot a phase diagram where the tie-lines are not in the plane after a MAP calculation, then `mole-fraction` and `x(comp)` are identical.

When you plot a property diagram after a STEP calculation and want a composition axis variable, you should normally use x(comp) and not mole-fraction.

The axis must have exactly the same number of columns, or one axis must have one column. In the first case the columns are plotted matching them one by one; in the latter all columns on one axis is plotted against the single column. For example, you can have the temperature on one axis and the amount of each phase on the other. The amount of each phase is the state variable `NP(*)` in mole fraction or `BPW(*)` for mass fractions of phases.

Automatic diagram axis: The POST module can set automatic diagram axis identical to those used in the *MAP* on page 158 command.

If the state variable `x(comp)` is used in a MAP calculation, then mole-fraction of the component is used as diagram axis; if a potential or some other state variables are used, then that is used as a diagram axis.

After a STEP calculation, automatic diagram axis is not set as there is only one axis used in the calculation.

Syntax	SET_DIAGRAM_AXIS
	`AXIS (X, Y OR Z): <AXIS NAME>` Specify which axis (X or Y or Z) to set with a variable.
Prompts	`VARIABLE TYPE: <VARIABLE NAME>` Specify a variable for the chosen axis here. The valid variables are: • TEMPERATURE-CELSIUS as temperature in oC • TEMPERATURE as temperature in K • PRESSURE as pressure in Pa • ACTIVITY for a component • LNACTIVITY for a component [ln(ACR(component)] • MOLE-FRACTION for a component • MOLE-PERCENT for a component • WEIGHT-FRACTION for a component • WEIGHT-PERCENT for a component • NONE to clear the axis • Any valid state variable including those with wildcards, e.g. NP(*) or x (*,component). Due to the use of wildcard * in such a state variable, the program asks for a column number. • Any entered function or variable. • Any entered table.
	If you specify an entered table as the variable, then you are given these prompts: `FOR COMPONENT: <COMPONENT NAME>`

Syntax	SET_DIAGRAM_AXIS
	When an activity, mole or weight fraction or percent of a component shall be plotted, the component name must be supplied here.
	COLUMN NUMBER: <COLUMN NUMBER(S)> Specify the column number(s) in the chosen table that are to be plotted onto the earlier specified axis. For example: • 1 for column 1 • 2,3 for columns 2 and 3 • 2,3 >5 for column 2,3 and all columns above column 5 • * for all columns

SET_DIAGRAM_TYPE

Choose the diagram type as perpendicular plot or triangular plot (Gibbs triangle, especially for ternary systems). The default is with a perpendicular axes.

For phase diagrams of ternary or pseudo-ternary systems, it is usually need to plot isothermal sections as triangular grams. If desired, all lines outside the region limited by a line joining the end points of the X- and Y-axis is removed.

In order to create a 3D-plot file (in the VRML format, *.WRL) for a tetrahedron diagram, this command in the sequence SET_DIAGRAM_TYPE N Y should be used first.

Syntax	SET_DIAGRAM_TYPE
Prompts	TRIANGULAR DIAGRAM (Y OR N) /N/: <Y OR N> Specify a triangular plot Y. Otherwise (by pressing <Enter> to accept N), perpendicular axis is set and this command is terminated. By default, perpendicular diagrams have (almost) the same scaling on both axes.
	CREATE TETRAHEDRON WRML FILE (Y OR N) /N/: <Y OR N> If a triangular plot is not selected (at the above prompt), answer Y in case a tetrahedron diagram is used to generate VRML (*.WRL) file by sequentially using the CREATE_3D_ PLOTFILE command.
	PLOT 3:RD AXIS (Y OR N) /N/: <Y OR N> If a triangular plot is selected, specify if a 3rd axis (by answering Y here), connecting the end points of the X- and Y-axes, is plotted.

Syntax	SET_DIAGRAM_TYPE
	`CLIP ALONG 3:RD AXIS (Y OR N) /N/: <Y OR N>` If a triangular plot is selected, all lines outside the region limited by a line joining the end points of the X- and Y-axis are removed by answering Yes (`Y`).

SET_FONT

Select the font to use for labels and numbers when plotting the diagram under the currently selected graphic device (*SET_PLOT_FORMAT* on the next page). For some devices there may be other fonts available and these are selected by the SET_PLOT_FORMAT command.

Syntax	SET_FONT
Prompts	`SELECT FONTNUMBER /1/: <#>` Specify an available font number # as the default font for the current graphic device, or accept the font number 1 by pressing <Enter>. By typing a question mark ? here, the program lists the available fonts under the currently selected graphic device.
	`FONT SIZE /.34/: <##>` The size of the chosen font is specified. A value around 0.34 is recommended.

SET_INTERACTIVE_MODE

Syntax	SET_INTERACTIVE_MODE
	Resets the input and output units to their initial values, i.e. keyboard and screen. Remember to add this as the last command to your MACRO files, in order to stop execution of the command file in the POST module. It has no meaning in interactive mode.

SET_LABEL_CURVE_OPTION

Identify the curves drawn in the post-processor by marking each curve with a digit and then list the meaning of these digits beside the plot.

The number of digits can be quite large in some cases. If so, use the SET_FONT command and reduce the font size; usually, 0.2 is sufficient.

Syntax	SET_LABEL_CURVE_OPTION
Prompt	`CURVE LABEL OPTION (A, B, C, D, E, F OR N) /A/: <OPTION>`

	The options:
	• `A`: List stable phases along line
	• `B`: As A but curves with same fix phase have same number
	• `C`: List axis quantities
	• `D`: As `C` but curves with same quantities have same number
	• `E`: As `B` with changing colors
	• `F`: As `D` with changing colors
	• `N`: No labels
	
	This question is rather cryptic but usually the option B or E is good for phase diagrams (after a MAP command) and option D or F for property diagrams (after a STEP command). Those interested can try out the option A and C by themselves.
	
	Option B or E lists the fixed phases along each curve, whereas option D or F gives the axis variables used along each curve. E provides varied colors for different stable-phase curves on phase diagrams (after a MAP command); F provides varied colors for different property curves on property diagrams (after a STEP command). The option `N` (NONE) disables all curve labels and lists.
	
	If, for example, you use `T` (temperature in K) as the variable axis in a STEP command and then plot the amount of stable phases (with *SET_DIAGRAM_AXIS* on page 206 `Y NP(*)`) on the Y-axis versus `T-C` on the X-axis, then the list may have lines like:
	`1: T-273.15, NP(LIQUID)`
	`2: T-273.15, NP(FCC_A1)`
	
	This means that for curve `1` the X-axis is `T-273.15` (the same for all curves, of course) and `NP(LIQUID)` on the Y-axis. Curve `2` has the same X-axis but `NP(FCC_A1)` on the Y-axis.

SET_PLOT_FORMAT

Set the format of the graphical output to different graphical devices. The default graphic device is normally for screen output (number 22 - TCUNITE Driver). Legacy devices can still be used (1 for Windows and 9 for XWindow for Linux). This default graphic device value may also be altered with *SET_PLOT_ENVIRONMENT* on page 243 in the system monitor or by your TC.INI file.

Syntax	SET_PLOT_FORMAT
Prompt	GRAPHIC DEVICE NUMBER /#/: <NUMBER OF THE DEVICE> Specify a graphic device number. Depending on the hardware, different plot formats (graphic devices) may be available. These are listed online by giving a question mark ?. For some formats there can be additional sub-prompts, asking for if using (YES) or not using (NO) the fonts available for the chosen format, and if YES for further specifications, e.g. font type and size.

SET_PLOT_OPTIONS

You are prompted for and can toggle on (`Y`) or off (`N`) the plotting of some options on all subsequently generated diagrams.

For the `Always initiate POST on re-entering` prompt, the default is Y which implies that the POST module automatically reinitiates whenever the POLY3 workspace is changed upon re-entering. If answered N the same diagram axis variables are kept even if the workspaces in the POLY module are read from or written to an *.POLY3 file. The N answer is useful when there are several sets of *.POLY3 files with the same calculation but for different sets of model parameters.

For the `Always solid line` prompt, the default answer is `N` which means that the POST module plots solid lines for stable phases but dashed lines for metastable phases. This is normally for plotting property diagram after a STEPPING calculation; however, when there is a metastable extrapolation in a phase region on a phase diagram after a MAPPING calculation, this is also enforced. When the alternative answer Y is chosen, all the lines are always plotted in solid lines.

Syntax	SET_PLOT_OPTIONS
Prompts	PLOT HEADER /Y/: <Y OR N> Toggle the plotting of the Thermo-Calc header text above a diagram.
	PLOT LOGO /Y/: <Y OR N> Toggle the plotting of the Thermo-Calc logotype at the lower-left corner of a diagram.
	PLOT FOOTER /Y/: <Y OR N> Toggle the plotting of the footer identifier text at the right margin of a diagram.
	WHITE-CONTOURED-PS-CHARS /Y/: <Y OR N> This makes it possible to write PostScript characters in white contoured status.
	PLOT REMOTE EXPONENT(S) /Y/: <Y OR N>

Syntax	SET_PLOT_OPTIONS
	This sets (`Y`) or removes (`N`) the plotting of the remote exponent on the axis.
	`PLOT SYMBOLS AT NODE POINTS /0/: <#>` This makes it possible to plot symbols at the node points on plotted lines on a diagram.
	`SYMBOL SIZE /.1/: <.#>` Set the size of symbols which are plotted at the node points.
	`WRITE CONDITION? /Y/: <Y OR N>` The initial equilibrium conditions for the current calculations are automatically plotted at the upper-left corner as a part of the header. With this option switched off (by answering `N`), it makes possible to not write the calculated conditions on a plotted diagram.
	`WRITE DATABASE NAME? /Y/: <Y OR N>` Primary database (not the appended ones) used in the current calculations are automatically plotted at the upper-left corner as a part of the header. With this option switched off (by answering `N`), it makes possible to not write the database name on a plotted diagram.
	`ALWAYS INITIATE POST ON RE-ENTERING: /Y/: <Y OR N>` `Y` lets the POST module automatically reinitiate whenever the POLY3 workspaces are changed upon re-entering. If answered `N`, the same diagram axis variables are kept even if the workspaces in the POLY module is read from or written to an *.POLY3 file. Answering `N` is useful when there are several sets of *.POLY3 files with the same calculation but for different set of model parameters.
	`ALWAYS SOLID LINE: /N/: <Y OR N>` `N` allows the POST module to plot solid lines for stable phases but dashed lines for metastable phases. This is normally for plotting property diagram after a STEPPING calculation; however, when there is a metastable extrapolation in a phase region on a phase diagram after a MAPPING calculation, this is also enforced. When answered `Y` all the lines are always plotted in solid lines.

SET_PLOT_SIZE

Change the size of the diagram by specifying a global plot size (as a relative scaling factor). The default value of the relative scaling factor is 1, while the real size of the plotted diagram depends on what output device is chosen with *SET_PLOT_FORMAT* on page 210. The default plot size (corresponding to the default global plot size 1) is adjusted to the chosen device.

Syntax	SET_PLOT_SIZE
Prompt	GLOBAL PLOT SIZE /1/: <RELATIVE SCALING FACTOR> Enter the relative scaling factor as a numerical number (e.g. 0.5, 0.8, 1.0, 1.5, etc.). By pressing <Enter>, the default relative scaling factor (1) is accepted.

SET_PREFIX_SCALING

Set the prefix scaling of remote exponents with certain powers (default is 3), by taking as argument with an axis name and a YES, or NO, or an integer number # (as the powers for the remote exponents):

NO switches off its action.

YES arranges the prefix scaling so that the axis scaling is done with the remote exponents being powers of three, i.e. ...,-6,-3,0,3,6,...

sets the remote exponents as being powers of #.

Syntax	SET_PREFIX_SCALING
Prompt	AXIS (X, Y OR Z): <NAME OF AN AXIS> Specify which axis is to have a prefix scaling.
	USE PREFIX SCALING /Y/: (Y OR N OR #) Answer either Y or N or an integer number as described above.

SET_RASTER_STATUS

Plot a raster (i.e. with gridlines on both axis directions) in the diagram. The default status is no raster plotted.

Syntax	SET_RASTER_STATUS
Prompt	RASTER PLOT (Y OR N) /N/: <Y OR N> Enable (Y) or disable (N) the raster plot.

SET_REFERENCE_STATE

The reference state for a component is important when calculating activities, chemical potentials and enthalpies. After calculating with *STEP_WITH_OPTIONS* on page 172 or *MAP* on page 158, the reference state for a component can also be changed for plotting various properties of the component in the entire system or in a specific phase by using this command. Afterwards, you can set diagram axes as chemical potential or activity quantities with the R suffix, i.e. MUR(comp), MUR(comp,ph), ACR(comp), ACR (comp,ph) or their common logarithms [e.g. LNACR(comp,ph)].

Syntax	SET_REFERENCE_STATE
Prompts	`COMPONENT: <NAME OF THE COMPONENT>` The name of the component must be given.
	`REFERENCE STATE: <NAME OF A PHASE USED AS THE NEW REFERENCE STATE>` The `Name of a phase`, that must be either ENTERED, DORMANT or SUSPENDED, is given here. The component must, of course, be a constituent of this phase. A subtle problem is that if the component exists in several species in the phase, for example, oxygen as O, O2 and O3 in a gas. Normally, you would like to have the most stable species as reference state of oxygen, i.e. O2 in this case. When setting the reference state, a one-time calculation is made for the Gibbs energy. This is for all possible states with the pure component phase at the current temperature and the command selects the most stable one as the reference state.
	`TEMPERATURE /*/: <REFERENCE TEMPERATURE>` Select the temperature (in K) for the reference state. The value * means the current temperature is used for the calculation.
	`PRESSURE /1E5/: <REFERENCE PRESSURE>` Select the pressure (in Pa) for the reference state. The value * means the current pressure is used for the calculation.

SET_SCALING_STATUS

When an axis variable is selected by *SET_DIAGRAM_AXIS* on page 206, the scaling status for the axis is always set to automatic scaling. By this command, choose between manual or automatic scaling on a specified axis. If manual scaling is chosen, specify a minimum and a maximum value. Manual scaling can be used to magnify interesting parts of a diagram.

Syntax	SET_SCALING_STATUS
Prompts	`AXIS (X, Y OR Z): <NAME OF AN AXIS>` Specify which axis that you want to set the scaling status.
	`AUTOMATIC SCALING (Y OR N) /N/: <Y OR N>` Choose between automatic (Y) and manual scaling (N). For manual scaling specify these parameters:
	`MIN VALUE: <MINIMUM VALUE>` The minimum value at the start point of the specified axis.
	`MAX VALUE: <MAXIMUM VALUE>`

Syntax	SET_SCALING_STATUS
	The maximum value at the end point of the specified axis.

SET_TIC_TYPE

Change the drawing of the axis tics. You may alter the placement of tic marks on the axis, e.g. inside or outside or no tic marks. The size of tic marks cannot be altered by this command, but these are adjusted based on the selected graphic device and the defined relative scaling factor (global plot size).

Syntax	SET_TIC_TYPE
Prompt	`TIC TYPE /1/: <1 OR -1 OR 0>` The tic type 1 is default, i.e. the tics are drawn outside the diagram. -1 means that the tics are inside the diagram and 0 means no tics.

SET_TIELINE_STATUS

If the tie-lines are in the plane of the calculation, you can select to plot the tie-lines in two-phase fields with this command.

Syntax	SET_TIELINE_STATUS
Prompt	`PLOTTING EVERY TIE-LINE NO /0/: <NUMBER OF TIE-LINES>` The tie-lines plotted are not equally spaced by the graphics software. Instead, you can select to plot a subset of the calculated tie-lines, i.e. every one (1), every second (2), every three (3), etc. By accepting the default value 0, no tie-line is plotted.

SET_TITLE

Specify a title that displays on all listed tables and plotted diagrams from the POST module in the same run of Console Mode.

Syntax	SET_TITLE
Prompt	`TITLE: <TITLE>` Enter the title you want to appear on all tabulation or graphic outputs. There is a maximum length of about 60 characters for the title. You can enter simple text, use LaTeX, rotate the text, as well as adjust the font size.
	Also see *ADD_LABEL_TEXT* on page 190.

SET_TRUE_MANUAL_SCALING

The tic-marks on the axes are normally placed in even intervals using the whole axis length. The scaling routine adjusts the given minimum and maximum values slightly to accomplish this. If such an automatic adjustment behaviour is unwanted, it can be avoided by using this command (followed by a message showing the TRUE MANUAL SCALING set for a specific axis).

The command works like a toggle. To reset the automatic scaling behaviour, repeat the command a second time for that specific axis (followed by a message saying the SEMI_MANUAL SCALING set for the specific axis.

Syntax	SET_TRUE_MANUAL_SCALING
Prompt	AXIS (X, Y OR Z): <NAME OF AN AXIS> Specify which axis (X or Y or Z) you want to toggle between automatic adjustments or avoid the adjustment of the given maximum and minimum values.

SUSPEND_PHASE_IN_PLOT

Specify that lines originating from the presence of a certain phase shall not be plotted on a diagram. If you want to bring the suspended phase back to the diagram, use *RESTORE_PHASE_IN_PLOT* on page 202.

Syntax	SUSPEND_PHASE_IN_PLOT
Prompt	PHASE NAME: <NAME OF A PHASE> Specify the name of the phase to be suspended from the diagram.

TABULATE

For any entered table, this command gives a table of values from equilibria calculated with a STEP command. It works only for a table that is entered in either the POLY or POST module.

Unlike the TAB module's command (also called *TABULATE* on page 253) it is not possible to plot columns from the tabulated tables. For this use normal POST commands.

Syntax	TABULATE
Prompts	NAME: <NAME OF A TABLE ENTERED IN EITHER POLY OR POST> Specify a table name that is entered in either the POLY or POST module.
	OUTPOUT ON SCREEN OR FILE /SCREEN/: <FILE NAME OR RETURN> Specify a file name if you want to save on a file the table values along the defined STEP calculation, or press <Enter> if want to see the table values displayed.

POTENTIAL_DIAGRAM Module

In this section:

POTENTIAL_DIAGRAM

The POTENTIAL_DIAGRAM module (short name, the POTENTIAL module) is a potential phase diagram calculation module.

If you have a metal-oxide/sulphide-gas interaction system, you might want to calculate a so-called potential phase diagram given certain temperature and pressure. A potential diagram uses the activities of two major species in the gaseous mixture phase as mapping variables. The activities (that is, the fugacities at 1 bar) of these two species are typically set as the X- and Y-axis when the diagram is plotted. The phase relations between the gaseous mixture and various metal forms, metal-oxides, metal-sulphides, or other metal-bearing solids, are represented by different fields controlled by these two potential quantities.

To enter the module, at the SYS prompt type `GOTO_MODULE POTENTIAL`.

There are no commands for this module. Follow the prompts and see *About Potential Diagrams* on page 158 in the *Thermo-Calc User Guide* for details about how to calculate using potential diagrams.

See example 35 in the *Console Mode Examples Guide*.

Syntax	POTENTIAL_DIAGRAM
Prompts	`DATABASE: /POT/` Press <Enter> and the program lists the available databases. Type one of the database names and press <Enter> to select. The following default elements are then based on the chosen database.
	`MATRIX ELEMENT: /FE/`
	`FIRST POTENTIAL SPECIES: /S1O2/`
	`SECOND POTENTIAL SPECIES: /O2/`
	`TEMPERATURE: /1000/`

POURBAIX_DIAGRAM Commands

Also search the online help or see the Pourbaix diagrams educational material and the *Thermo-Calc User Guide*.

In this section:

POURBAIX_DIAGRAM Commands

Also search the online help for additional information in other guides.

See examples TCEX40 and TCEX53 in the *Console Mode Examples Guide*.

After you enter the POURBAIX_DIAGRAM module, you are prompted with the following:

```
NEED ANY INSTRUCTION ON THE POURBAIX MODULE? /N/:Y
```

If you answer `Y` the following notes are displayed.

- The Pourbaix diagram is actually a calculated equilibrium phase diagram mapped and plotted with the independently-varied electropotential (Eh, as defined with regard to the standard hydrogen electrode as its reference) and acidity (pH), that represents all the equilibrated phase boundaries among aqueous solution, gaseous mixture, and various primary and secondary solids (modelled as either complex solution or simple stoichiometric phases) in a certain multicomponent heterogeneous interaction system, under a defined T-P condition and a specific initial bulk composition (which is, by default, always set as 1 kg of water dissolving a specified amount of metals/alloys and other acids/alkalines/salts).
- At least one database is required that contains an AQUEOUS solution phase (with thermodynamic data for water and various aqueous solutes), that shall be selected from TCAQ (PAQ) or AQS (PAQS) [which use the SIT Model or the Complete Revised HKF Model, respectively] within the Thermo-Calc database spectrum, or be chosen from an appropriate USER-specified database (it must be in the Thermo-Calc TDB format).
- Due to the restrictions of aqueous solution model used within Thermo-Calc, such a database must be designed in the same format as in the default TCAQ (PAQ) or AQS (PAQS) for AQUEOUS solution phase. Among others, one should keep in mind the following regulations:
    - The ELECTRON is defined as an special element (ZE) and as the only constituent in its reference phase REFERENCE_ELECTRODE (for determining the electropotential that is defined as Eh with the standard hydrogen electrode as the reference), but it is not defined as an aqueous species;
    - The vacancy (VA) is unnecessary for AQUEOUS solution phase and it should be avoided in the definition of phase-constituents in the AQUEOUS phase;

- The AQUEOUS solution phase should always be defined as a constitutional solution phase, implying that all the aqueous solution species must be included in a single site, rather than in two or multiple sublattices.

- Beside the AQUEOUS solution phase, there shall exist a GAS mixture phase containing at least H2O1, O2 and H2; and for multicomponent systems, normally there shall also contain some solid (stoichiometric or solution) phases. Of course, if desired, you could also choose to calculate and generate a Pourbaix diagram without considering the GAS mixture phase entirely; however, such a plot is not really a complete Pourbaix diagram, due to the fact that thermodynamically-stable phase boundaries between the AQUEOUS solution phase and GAS mixture phase will then not be calculated at all!
- All the required thermodynamic data for calculations of Pourbaix diagrams or other diagrams must be retrieved either from one (Single) database which consists of an AQUEOUS solution phase, a GASeous mixture phase, a REF_ELECTRODE phase, and some SOLID phases (being solutions and/or stoichiometric compounds; for primary metals/alloys and for secondary products formed from heterogeneous chemical/electrochemical interactions, or from several (Multiple) databases that respectively contain various solutions/compounds (as listed above).

Such databases suitable for calculations of aqueous-bearing heterogeneous interaction systems can be those default-prompted ones [i.e., in the Single-Database case, the PAQ or PAQS; and in the Multiple-Database case, the TCAQ or AQS as primarily-switched database, plus the SSUBx as firstly-appended one and the SSOLx as secondly-appended one if it is necessary; even more databases can be appended). Of course, you could also choose to append required data from other appropriate databases (such as TCFE, TCSLD, TCNI/TTNi, TCAL/TTAl, TCMG/TTMg, TTZr, TCMP, SLAG, etc.) for GASeous mixture phase and for varous solid solution and stoichiometric compound phases. Furthermore, an experienced user can also utilize his/her own USER-specified databases in various cases.

- The current advanced POURBAIX-Module has been designed and developed in an efficient and effective way that it only requires the user to just answer some simple and necessary questions, rather than to go through basic modules (i.e., TDB, GES5, POLY3 and POST) and type the ordinary command-lines. Beside the default plotted Pourbaix diagram, it also allows the user to easily and quickly plot many different properties of the system, stable phases and aqueous species, varied along the calculated phase boundaries for the same defined heterogeneous interaction system. Moreover, it permits the user to directly change some plotting settings and manipulate all kinds of plotted diagrams.

- The current advanced POURBAIX-Module has been extended so that it is additionally able to directly perform a normal STEPPING calculation (varied with a specified independent variable) and to easily generate various types of property diagrams, for the same heterogeneous interaction system that has been defined in a previous POURBAIX or TDB-GES5-PLOY3-POST calculation.

Pause Options

After this information is displayed, you are prompted:

```
ENFORCE A PAUSE AFTER PLOTTING WHEN RUNNING A MACRO? /N/:
```

If you answer `Y` the following options are displayed.

```
1. START A COMPLETELY NEW POURBAIX DIAGRAM CALCULATION
2. OPEN AN OLD FILE & PLOT OTHER PROPERTY DIAGRAMS
3. OPEN AN OLD FILE & MAKE ANOTHER POURBAIX CALCULATION
4. OPEN AN OLD FILE & MAKE ANOTHER STEPPING CALCULATION
```

Option 1: Start a completely new POURBAIX diagram calculation

Make a completely new POURBAIX diagram calculation and automatically plot a pH-Eh diagram. i.e., define a new chemical system; specify the T-P-X conditions; calculate the initial equilibria; perform the pH-Eh mapping calculation; plot pH-Eh & various property diagrams.

```
CONSIDER THE GAS PHASE IN CALCULATING A POURBAIX DIAGRAM? /Y/
USE SINGLE DATABASE? /Y/:
COMBINED DATABASE: /PAQ2/:
```

DEFINE A CHEMICAL SYSTEM AND ITS INITIAL BULK COMPOSITION:

Normally a POURBAIX diagram and related equilibrium property in a heterogeneous interaction system are calculated under a certain bulk composition which is usually 1 kg of water with defined amounts of dissolving solute substances. The solutes may either be completely dissolved into the aqueous solution, or be partially dissolved and simultaneously form some solid phases.

CHEMICAL SYSTEM (ELEMENTS):

Default defined elements (solvent H2O): H & O and specially assigned ZE(electron) & VA(vacancy).

Prompt specified elements (solutes ELEM): Fe Ni Co Cr C N S Na Cl

INITIAL BULK COMPOSITION: Default defined composition (solvent): 1.0 kg of H2O

Prompt specified composition (solutes): x mole of ELEM

Notes: For accepting a default value, just RETURN at prompt;

For changing to a specific value, enter it at prompt.

IMPORTANT NOTE for Entering Solutes in Chemical Formulas:

First element letter in UPPER case, and second lower case! such as NaCl CO2 CoCl3 Fe0.93S NaSO4-1 H2SO4

Mass of Water (weight) = 1 kg

Option 2: Open an old file and plot other property diagrams

Open an existing POLY3 file created by POURBAIX Module (from a previous POURBAIX calculation Option 1 or 3 or a previous STEPPING calculation Option 4), and just selectively plot other property diagrams. i.e., open the old GES and POLY3 workspaces; plot pH-Eh or various property diagrams.

```
FILE NAME /POURBAIX/
X-AXIS VARIABLE: /PH/:
Y-AXIS VARIABLE: /EH/:
CHANGE AXIS TYPE? /N/:
ZOOM IN? /N/:
CHANGE CURVE-LABEL OPTION FOR THE DIAGRAM? /N/:
ADD LABEL-TEXTS ONTO THE SPECIFIED DIAGRAM? /N/:
```

```
CHANGE THE SUBTITLE OF THE DIAGRAM? /N/:
CHANGE AXIS-TEXTS? /N/:
FURTHER REFINE THE DIAGRAM IN POST MODULE? /N/:
HARD COPY OF THE DIAGRAM? /N/:
SAVE X-Y COORDINATES OF CURVE ON TEXT FILE? /N/:
MODIFY THE DIAGRAM? /N/:
ANY MORE DIAGRAM? /N/:
```

Option 3: Open an old file and make another POURBAIX calculation

Open an existing POLY3 file created by POURBAIX Module and make another POURBAIX diagram calculation i.e., open the old GES and POLY3 workspaces; adopt the defined chemical system; modify the T-P-X conditions; calculate the initial equilibria; perform the pH-Eh mapping calculation; plot pH-Eh & various property diagrams.

```
CONSIDER THE GAS PHASE IN CALCULATING A POURBAIX DIAGRAM? /Y/:
FILE NAME /C:\USERS\AMANDA\DOCUME~1\POURBA~1.POL/:
ENFORCE GLOBAL MINIMIZATION TECHNIQUE IN CALCULATIONS? /N/:
SAVE ALL FUNCTIONS, VARIABLES AND TABLES IN POLY3 FILE? /Y/:
SET NUMERICAL LIMITS? /N/:
SET CALCULATING PARAMETERS:
LIST OF DEFAULT AND PRE-DEFINED CALCULATION CONDITIONS:
  UNITS: T IN K, P IN PASCAL, B(H2O) IN GRAM, N(ELEM) IN MOLE
T=298.15, P=1E5, B(H2O)=1000, N(H+1)=0, N(ZE)=0
DEGREES OF FREEDOM 0
CONFIRM DEFINED CONDITIONS? /Y/:
```

Additional Information About Option 3

The default definitions of the Eh and pH quantities in the advanced POURBAIX-Module (and in the ordinary TDB-GES-POLY calculation routines) should ALWAYS be as below:

```
Eh = MUR(ZE)/RNF
pH = -log10[AI(H+1,AQUEOUS)]
= -log10[ACR(H+1,AQUEOUS)*AH2O]
```

where RNF is the Faraday constant (96485.309 C/mol), and AH2O is the molecular weight of H2O (55.508435 g). MUR(ZE) is the electrochemical potential (ECP; in the unit of J/mol; w.r.t. the standard hydrogen electrode). ACR(H+1,AQUEOUS) is the site-fraction-based activity of the H+1 aqueous species in AQUEOUS solution phase, but AI(H+1,AQUEOUS) [that equals ACR(H+1,AQUEOUS)*AH2O] is the molality-based activity of the H+1 aqueous species that should be used for defining the pH quantity.

Within an aqueous-bearing heterogeneous interaction system, the fundamental system-components must be H2O, H+1 and ZE, which are corresponding to the basic elements O & H and the hypothetical electron (ZE) in the aqueous solution phase. For the additional chemical elements in the system, their corresponding system-components shall be defined as in their element forms (such as Fe, Cr, Mn, Ni, Na, Cl, S) or (for some) in their molecular forms (e.g., NaCl, H2S). The reference state for the H2O component must always be defined as the solvent species H2O in the AQUEOUS solution phase under the current temperature (*) and 100000 Pascal (i.e., 1 bar). The reference states for the H+1 and ZE components are by default set as their SER.

Various conventional properties of aqueous solute species I are converted in the following manners:

```
ML = Y(AQUEOUS,I)*AH2O/YH2O
RC = ACR(I,AQUEOUS)*YH2O/Y(AQUEOUS,I)
AI = RC*ML
= ACR(I,AQUEOUS)*AH2O
```

where YH2O [i.e., Y(AQUEOUS,H2O)] and Y(AQUEOUS,I) are the site-fractions of solvent H2O and solute species I.

LIST THE DEFINED AXIS-VARIABLES:

```
Axis No 1: LNACR(H+1)Min: -34.532525Max: 2.3025851 Inc: 0.8
Axis No 2: MUR(ZE)   Min: -150000   Max: 200000    Inc: 7718.85
```

NOTE: The default settings (listed above) for two mapping variables [in terms of lnACR(H+1) and MUR(ZE), and their minimum/maximum values and increment steps] are covering the following pH-Eh ranges/steps:

```
pH:  0.00 -> 14.00, at a step of 0.35
Eh: -1.55 ->  2.07, at a step of 0.08 (V)
```

The maximum pH limit has been calculated precisely and determined automatically by the POURBAIX-Module, as a function of the temperature-pressure conditions and initial bulk compositions of the current defined interaction system.

```
ACCEPT THE DEFAULT SETTINGS FOR TWO MAPPING VARIABLES? /Y/:
ANY MISSING PHASE BOUNDARY YOU COULD POSSIBLY THINK OF? /N/:
CHANGE THE PH/EH STEPS FOR SMOOTHER CURVES? /N/:
ZOOM IN? /N/:
CHANGE CURVE-LABEL OPTION FOR THE DIAGRAM? /N/:
ADD LABEL-TEXTS ONTO THE POURBAIX DIAGRAM? /N/:
CHANGE THE SUBTITLE OF THE DIAGRAM? /N/:
CHANGE AXIS-TEXTS? /N/:
FURTHER REFINE THE DIAGRAM IN POST MODULE? /N/:
HARD COPY OF THE DIAGRAM? /N/:
SAVE X-Y COORDINATES OF CURVE ON TEXT FILE? /N/:
MODIFY THE DIAGRAM? /N/:
ANY MORE DIAGRAM? /N/:
```

Option 4: Open an old file and make another STEPPING calculation

Open an existing POLY3 file created by POURBAIX Module (from a previous POURBAIX calculation Option 1 or 3 or a previous STEPPING calculation Option 4), and make a normal STEPPING calculation. i.e., open the old GES and POLY3 workspaces; adopt the defined chemical system; specify one of the T-P-X conditions as the stepping variable; calculate the initial equilibria; perform the stepping calculation; plot various property diagrams.

Because of that in a STEPPING calculation will change some system definitions in a POURBAIX-type POLY3 file, it is highly advisable to make a backup of the existing POURBAIX-type POLY3 file (which you are going to open in this STEPPING calculation) before you proceed the further steps; otherwise, the structure of the loaded POLY3 file will be alternated by the option "4", and consequently the existing POLY3 file may be not suitable for other POURBAIX-type calculation any longer. Therefore, if you have not done so, you should make the copy right now!

However, a POLY3 file which is modified by this option can be further used for another STEPPING calculation inside the POURBAIX Module (by a repeated Option 4) or in normal POLY3 Module.

```
FILE NAME /C:\USERS\AMANDA\DOCUME~1\POURBA~1.POL/:
ENFORCE GLOBAL MINIMIZATION TECHNIQUE IN CALCULATIONS? /N/:
SAVE ALL FUNCTIONS, VARIABLES AND TABLES IN POLY3 FILE? /Y/:
SET NUMERICAL LIMITS? /N/:
   SET CALCULATING PARAMETERS:
   List of Default and Pre-defined Calculation Conditions:
   --------------------------------------------------------------
     Units: T in K, P in Pascal, B(H2O) in gram, N(ELEM) in mole
   --------------------------------------------------------------
   T=298.15, P=1E5, B(H2O)=1000, N(H+1)=0, N(ZE)=0
   DEGREES OF FREEDOM 0
FURTHER MODIFY SYSTEM DEFINITIONS & CONDITIONS IN POLY3 MODULE? /N/:
CONFIRM DEFINED CONDITIONS AS THE STARTING POINT? /Y/:
```

The following parameters are suitable for a STEPPING calculation for the defined aqueous-bearing system. Choose one of them as the stepping variable:

```
   T Temperature (C)
   P Pressure (bar)
   pH Acidity
   Eh Electronic Potential (V)
   N(Comp) Mole of One Component
```

where "Comp" is the name of a specific component as listed below. One should not use any key component (H2O or H+1 or ZE) as such a stepping variable, because of that their compositional definitions and variations are determined by the AQUEOUS solution phase in the POURBAIX Module.

An interacting component defined as an element (such as Fe or Na or Cl) can be selected as a stepping variable.

However, if you want to make a stepping calculation with a non-element chemical formula (such as NaCl or CaS2) as the stepping variable, one must have made some necessary modifications in the POLY3 Module, before reaching this point, by conducting some POLY3 commands as given below (as an example for the system Fe-Na-Cl-H-O):

```
POLY_3: def-component H2O H+1 ZE Fe NaCl Cl
POLY_3: set-ref-state H2O AQUEOUS * 1E5
POLY_3: set-ref-state ZE REF_ELEC * 1E5
POLY_3: change-status phase REF-ELEC=SUSPEND
POLY_3: set-cond b(H2O)=1000 n(H+1)=0 n(ZE)=0
POLY_3: set-cond P=1E5 T=298.15**
POLY_3: set-cond n(Fe)=1E-3 n(Cl)=0 n(NaCl)=3 **
```

** Here the P-T-n(Fe) conditions can be set as what you wanted, while the initial n(NaCl) value must be set as between its minimum and maximum values that you are interested in. Later on, it will be prompted for specifying the n(NaCl) Min-Max-Step values of the stepping calculation.

```
STEPPING VARIABLE: /T/:
MINIMUM TEMPERATURE (C) /15/:
MAXIMUM TEMPERATURE (C) /35/:
INCREMENT IN TEMPERATURE (C) /1/:
CHANGE AXIS TYPE? /N/:
ZOOM IN? /N/:
CHANGE CURVE-LABEL OPTION FOR THE DIAGRAM? /N/:
ADD LABEL-TEXTS ONTO THE SPECIFIED DIAGRAM? /N/:
CHANGE THE SUBTITLE OF THE DIAGRAM? /N/:
CHANGE AXIS-TEXTS? /N/:
FURTHER REFINE THE DIAGRAM IN POST MODULE? /N/:
HARD COPY OF THE DIAGRAM? /N/:
SAVE X-Y COORDINATES OF CURVE ON TEXT FILE? /N/:
MODIFY THE DIAGRAM? /N/:
ANY MORE DIAGRAM? /N/:
```

REACTOR_SIMULATOR_3 Commands

In this section:

CREATE_DIVIDERS

Define the segment boundaries and the ways on how to distribute phases between various segments and to split the feeds of heat and mass.

Syntax	CREATE_DIVIDERS
Prompts	`NUMBER OF DIVIDERS /4/: <THE NUMBER OF DIVIDERS>` Specify the number of dividers (boundaries between various segments) in the reactor. It is normally the segment number plus one: for example, if the stage box number is defined as 4, the divider number should be defined as 5.
	`NUMBER OF OUTPUTS FOR DIVIDER # /3/: <THE NUMBER OF OUTPUTS>` Specify the number of outputs for a specific divider #.
	`PERCENT OF INPUT TO OUTPUT 1 /100/: <PERCENT OF INPUT>` Define the percentage of input to the output 1 in the current divider #.
	`PERCENT OF INPUT TO OUTPUT ## /100/: <PERCENT(S) OF INPUT(S)>` Define the percentages of inputs to the outputs ## in the current divider #. It is repeated till the last output is specified.

CREATE_PIPES

Define the pipes between various segments and the ways on how the mass and heat transport between the surroundings (with the record number as 0) and the reactor, among various segments in the reactor, and along various dividers (segment boundaries in the reactor).

Syntax	CREATE_PIPES
Prompts	`FROM RECORD: <A RECORD NUMBER>` Specify a record number. If there are four segments in the reactor, these records are available: • `0` surroundings • `1-4` stage boxes (segments) • `5-9` dividers (segment boundaries)
	`PIPE TO RECORD: <ANOTHER RECORD NUMBER>` Specify another record number to which the mass or heat flows from the current record (specified above). Before this prompt, a message shown on the screen indicates what kind of mass or heat flow (according to the definitions of stage boxes and dividers, which have already been created prior to this command) should be specified in the pipe. This is

Syntax	CREATE_PIPES
	repeated for all inputs and/or outputs for mass and/or heat in current record.

CREATE_STAGE_BOXES

Create several reaction stages as boxes (segments), where the you define the feed in either mass input and initial temperature or heat input from surroundings, as well as the mass out and initial temperature in each of the segments.

Syntax	CREATE_STAGE_BOXES
Prompts	`NUMBER OF STAGE BOXES /4/: <THE NUMBER OF STAGE BOXES>` Specify the number of stage boxes in the reactor. Then define the feed of mass and/or energy (from surroundings to) the rector.
	`GIVE FEED TO SYSTEM: <FEED>` One feed (of mass or heat) to the reactor must be specified here. Two types of feeds can be used: • As input amount of heat (enthalpy): `H=Value`. • As input amount of specific species or element and its initial temperature: `N(SPECIES)=Value, Input temperature` (in one line, or the input temperature in the next prompt).
	`INPUT TEMPERATURE /298.15/: <INPUT TEMPERATURE>`

	`GIVE FEED TO SYSTEM: <FEED>`
	
	Specify other feeds (mass or energy) to the reactor. Press <Enter> to finish the feed inputs, and then:
	
	`GIVE FOR STAGE BOX #`
	
	Now start to define the type of stage box (segment is specifically numbered as #), and to specify a guess of initial temperature and an output for each phase in the segment.

	`TYPE OF BOX /EQUILIBRIUM/: <TYPE OF BOX>` Only the `EQUILIBRIUM` type is allowed. Press <Enter> to accept it.
	`IS THE STAGE HEAT CONTROLLED? /Y/: <YES OR NO>` `Y` means the current segment is heat controlled.
	`INITIAL GUESS OF TEMPERATURE? /1000/: <INITIAL TEMPERATURE>` Give a guess of initial temperature (in K) for the current segment; press <Enter> to accept

Syntax	CREATE_STAGE_BOXES
	the default.
	`GIVE INITIAL AMOUNT: <INITIAL AMOUNT OF MASS OR HEAT>` The initial amount of mass and/or heat in the current segment can be specified here. Two types of initial amount can be used: • As initial amount of heat (enthalpy): `H=Value`. • As initial amount of specific species or element and its initial temperature: `N(SPECIES)=Value, Input temperature` (in one line, or the input temperature in the next prompt).
	`INPUT TEMPERATURE /298.15/: <INPUT TEMPERATURE>`

	`GIVE INITIAL AMOUNT: <INITIAL AMOUNT OF MASS OR HEAT>`
	
	Specify initial amount (mass or energy) in the current segment. Press <Enter> to finish the initial amount inputs, and then this message displays:
	
	`Each phase may have a separate output, give these`
	
	Now start to define the output for each phase.

	`PHASE NAME /REST/: <NAME OF PHASE>` Specify a phase that has a separate output, e.g. `GAS`; by pressing <Enter> to accept that all `REST` phases do not have separate outputs. Then the program asks for further definitions of all the other segments (one by one), repeating some of the above questions, till the last segment in the reactor is defined.

EXECUTE_POLY3_COMMAND

Syntax	EXECUTE_POLY3_COMMAND
	To directly use all kinds of POLY commands inside the REACTOR module. Any legal *POLY_3 Commands* on page 117 can be entered.

LIST_RECORDS

Syntax	LIST_RECORDS
	Lists all the already-created records for the reactor (including stage boxes, dividers, pipes, etc.).

READ_WORKSPACE

The REACTOR workspace (along with the related POLY3 and GIBBS workspaces) and the simulated results from the SIMULATE command can be read by this command from a binary file where they must be saved with *SAVE_WORKSPACE* below. The *.RCT file is not printable.

Syntax	READ_WORKSPACE
Options	`File name`: The name of a saved *.RCT file where the REACTOR and POLY3/GIBBS workspaces shall be read from must be specified. You do not need to type the extension if it is the default .RCT, otherwise type the whole *.RCT file name. An **Open** window displays if a file name is not given after the command or its path is incorrect, so that the path (in the **Look in** field) and **File name** can be specified. The **Files of type** (i.e. RCT) cannot be changed.

SAVE_WORKSPACE

The REACTOR (and POLY3/GIBBS) workspaces are saved on a file with this command. In the GIBBS workspace, all thermochemical data are stored. In the POLY3 workspace, all thermochemical data, all conditions, changed status, entered symbols, etc., are stored. In the REACTOR workspace, all definitions of multi-stage steady-state rector and its distribution coefficients are saved, together with the GIBBS and POLY3 workspaces. After a SAVE command, you can always return to the state when you issued the SAVE command by giving a READ command.

After saving the REACTOR (and POLY3/GIBBS) workspaces on a file, you can leave the program and at a later time READ the file and continue from the saved state.

A START_SIMULATE command automatically saves on the work file with the most lately specified name. To avoid destroying the simulated results, do not SAVE after a START_SIMULATE command is enforced, similar to that you should not SAVE after a STEP or MAP command! You may append several results by START_SIMULATE without destroying the previous results but SAVE erases them all.

Syntax	SAVE_WORKSPACE
	`FILE NAME` A file name must be specified. The default extension of the RCT file is .RCT, while or any other extension as required.
	A **Save** window displays if a file name is not given after the command, so that the path (in the **Save in** field) and **File name** can be specified. The **Files of type** (i.e. RCT) cannot be changed. If there is already a file with the same file name under the directory a warming message displays. Click **OK** to overwrite the current REACTOR (and POLY3/GIBBS) workspaces onto the existing *.RCT file. Click **Cancel** to return to the REACTOR module. You

Syntax	SAVE_WORKSPACE
	can use the SAVE command with an unspecified name (i.e. through `SAVE ,,,` command-sequence) to save the REACTOR (and POLY3/GIBBS) workspaces.

START_SIMULATION

Start the reactor simulation, and list the results for all iterations. The output for each iteration consists of the conditions set in each segment, and you can also select some state variables to list. After each loop, the temperatures in all segments are listed.

Syntax	START_SIMULATION
Prompts	`MAX NUMBER OF LOOPS /10/: <THE MAX NUMBER OF LOOPS>` Give the max number of loops. The REACTOR module repeats the reactor simulation until this loop number is reached.
	`OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME OR RETURN>` Decide on either listing simulation results on the screen (by pressing <Enter>) or saving them on an experimental file (with the default extension .EXP). The conditions in each segment are not saved in the file, only on the screen.
	`OUTPUT VARIABLES /T BP($)/: <VARIABLE NAME(S)>` Specify the variable name(s) that is also listed on the screen together with the conditions in each segment, or in the output experimental file.

Example Output

For a four-stage reactor, the output at a certain iteration may look like this:

```
H=-991745.1, P=100000, N(C)=2.8468936, N(N)=6.78399966E-4, N(O)=3.8171615,
N(SI)=1.77031
DEGREE OF FREEDOM 0
BP(GAS)=43.464341, BP(C1SI1_BETA)=25.175981, BP(C_S)=8.1588099,
BP(QUARTZ_S3)=68.193896
1.78803E+03
BP(GAS)=61.971213, BP(C1SI1_BETA)=38.144346, BP(QUARTZ_L)=66.230717 2.06569E+03
BP(GAS)=67.66262, BP(C1SI1_BETA)=35.480983, BP(QUARTZ_L)=58.219292
2.08691E+03
BP(GAS)=71.23129, BP(SI_L)=22.471787
2.20831E+03
```

```
H=-924751,87, P=100000, N(C)=2.0594558, N(N)=5.17999978E-4, N(O)=4.0142358,
N(SI)=2.7551438
DEGREE OF FREEDOM 0
BP(GAS)=43.464341, BP(C1SI1_BETA)=25.175981, BP(C_S)=8.1588098,
BP(QUARTZ_S3)=68.193896
1.78803E+03
BP(GAS)=61.971213, BP(C1SI1_BETA)=38.144346, BP(QUARTZ_L)=66.230717 2.06569E+03
BP(GAS)=67.66262, BP(C1SI1_BETA)=35.480983, BP(QUARTZ_L)=58.219292
2.08691E+03
BP(GAS)=71.23129, BP(SI_L)=22.471787
2.20831E+03
H=-813239., P=100000, N(C)=1.6592668, N(N)=3.59999988E-4, N(O)=3.7549293,
N(SI)=2.896635
DEGREE OF FREEDOM 0
BP(GAS)=43.464341, BP(C1SI1_BETA)=25.175981, BP(C_S)=8.1588098,
BP(QUARTZ_S3)=68.193896
1.78803E+03
BP(GAS)=61.971213, BP(C1SI1_BETA)=38.144346, BP(QUARTZ_L)=66.230717 2.06569E+03
BP(GAS)=67.66262, BP(C1SI1_BETA)=35.480983, BP(QUARTZ_L)=58.219292
2.08691E+03
BP(GAS)=71.23129, BP(SI_L)=22.471787
2.20831E+03
H=7374.6403, P=100000, N(C)=8.84911857E-1, N(N)=1.99999996E-4, N(O)=1.9379203,N
(SI)=1.853872
DEGREE OF FREEDOM 0
BP(GAS)=43.464341, BP(C1SI1_BETA)=25.175982, BP(C_S)=8.1588097,
BP(QUARTZ_S3)=68.193896
1.78803E+03
BP(GAS)=61.971213, BP(C1SI1_BETA)=38.144346, BP(QUARTZ_L)=66.230717 2.06569E+03
BP(GAS)=67.66262, BP(C1SI1_BETA)=35.480983, BP(QUARTZ_L)=58.219292
2.08691E+03
BP(GAS)=71.23129, BP(SI_L)=22.471787
2.20831E+03
10  1788.03  2065.69  2086.91  2208.31     0.00     0.00     0.00
```

SCHEIL Commands

Search the online help or see the *Thermo-Calc User Guide* included with this PDF documentation set for details about how to calculate using Scheil either in Graphical Mode (the Scheil Calculator) and Console Mode (the wizard Scheil module).

If you also have a license for the Diffusion Module (DICTRA), you can import a previously calculated Scheil segregation profile into the software using the command INPUT_SCHEIL_PROFILE. Search the online help or see the *Diffusion Module (DICTRA) Command Reference Guide*.

In this section:

BCC_TO_FCC_TRANSFORMATION

Use this command for the Scheil module to allow transformations in the solidified part of the alloy caused by each of the components specified to be Fast Diffuser. It is recommended that you only select this for steels.

Syntax	BCC_TO_FCC_TRANSFORMATION
Prompt	`ALLOW BCC 🕙 FCC ? /N/` The default is `N`. Enter `Y` to allow transformations.

EVALUATE_SEGREGATION_PROFILE

Use this command for the Scheil module to evaluate the segregation profile.

Syntax	EVALUATE_SEGRATION_PROFILE
Prompt	`EVALUATE SEGREGATION PROFILE? /N/` If answer `Y`, then this feature is not supported with back diffusion.
	`NUMBER OF GRID POINTS /100/` Segregation profile is dumped to a file XF.TXT in the current directory.

GLOBAL_MINIMIZATION

Use this command for the Scheil module to use global minimization.

Also see the POLY commands topic *ADVANCED_OPTIONS Descriptions* on page 120 for information about that GLOBAL_MINIMIZATION command.

Syntax	GLOBAL_MINIMIZATION
Prompt	`USE GLOBAL MINIMIZATION? /N/` If `Y`, the Global Minimization is done when possible (depending on the conditions setting), and a Global Minimization test is performed when an equilibrium is reached. This costs more computer time but the calculations are more robust. If `N` the Global Minimization is not used to reach the equilibrium.

LIQUID_PHASE_NAME

Use this command for the Scheil module to use the liquid phase name from the database definitions or enter another phase name.

Syntax	LIQUID_PHASE_NAME
Prompt	LIQUID PHASE NAME /DEFAULT/ Default selects the liquid phase from the database definitions, otherwise enter a phase name.

SAVE_FILE_NAME

Use this command for the Scheil module to save the file. The files get the console number added to this name, e.g. 005.->Scheil_006.poly3 and Scheil_segregation_profile.txt).

Syntax	SAVE_FILE_NAME
Prompt	Name of files to be saved, poly3 and segregation profiles.

START_WIZARD

Use this command for the Scheil module to use the wizard with step-by-step prompts to create a Scheil simulation.

This wizard walks you through a basic Scheil simulation. You are prompted for settings including the major element or alloy, composition input in mass (weight) percent, the alloying elements, the temperature, fast diffusing components, and whether any phase should have a miscibility gap check.

Use the other commands to make changes to settings that are not included in this wizard.

- BCC_TO_FCC_TRANSFORMATION
- EVALUATE_SEGREGATION_PROFILE
- GLOBAL_MINIMIZATION
- LIQUID_PHASE_NAME
- SAVE_FILE_NAME
- STORED_COMPOSITION_TYPE
- TEMPERATURE_STEP
- TERMINATION_CRITERIA

Syntax	START_WIZARD

Prompt	`DATABASE /<DEFAULT>/:`
	`MAJOR ELEMENT OR ALLOY:`
	`COMPOSITION INPUT IN MASS (WEIGHT) PERCENT? /Y/:`
	`1ST ALLOYING ELEMENT:CR 10 C 1`
	`NEXT ALLOYING ELEMENT:`
	`TEMPERATURE (C) /2000/:`
	`FAST DIFFUSING COMPONENTS: /NONE/:C`
	`SHOULD ANY PHASE HAVE A MISCIBILITY GAP CHECK? /N/:`

TEMPERATURE_STEP

Use this command for the Scheil module to define a temperature step.

Syntax	TEMPERATURE_STEP
Prompt	`SET TEMPERATURE STEP (C) /1/`

TERMINATION_CRITERIA

Use this command for the Scheil module to define the termination criteria for the fraction of liquid or temperature.

Syntax	TERMINATION_CRITERIA
Prompts	`FRACTION OF LIQUID OR TEMPERATURE (F OR T)? /F/`
	`FRACTION OF LIQUID /.01/`

SYSTEM_UTILITIES Commands

In this section:

ABOUT

Syntax	ABOUT
	Lists basic information about the development history and ownership of the Thermo-Calc software and its components.

CLOSE_FILE

Close an opened text file.

Syntax	CLOSE_FILE
Prompt	`UNIT NUMBER: <UNIT NUMBER>` A unit number given in an OPEN_FILE command must be specified.

DISPLAY_LICENSE_INFO

Use this command to launch a license diagnostics program implemented in the SYS module. It detects the license information for the Thermo-Calc installations on the current client computer (and on the connected license server), and lists the details, including the current status of available license file and all its included license features for the current installation(s), as well as the Safenet Environment Variables and detailed Local (Client) Information, on screen or into a textual file (*.TXT).

This is useful for a list of license details, and to obtain basic technical support by sending the *.TXT file to support@thermocalc.se. If you are running a network client installation of Thermo-Calc, you can see how many of the client example licenses that are currently checked out and how many of them are left. You can also see which clients computers it is that have checked these client example licenses out.

Syntax	DISPLAY_LICENSE_INFO
Prompt	`OUTPUT TO SCREEN OR FILE /SCREEN/: <MYLICENSEINFO.TXT>` Press <Enter> to display a list of license information or into a *.TXT file under a specific file name which can later on be opened and edited by any basic text editor.

HP_CALCULATOR

This is the QBA simple interactive calculator using reverse Polish notations.

Syntax	HP_CALCULATOR
	Available OPCODEs (HPC codes) are listed by entering the command HELP. Use the BACK command to quit the HP_CALCULATOR and go back to the SYS module. Input are numbers, + - * / and ^ and OPCODEs. Several numbers an operations can be given on one line. The content of the X register is displayed after each operation. For example, to compute 2*EXP(1.5**3-30000/(8*1273)), you should input the following: `30000 8/1273/chs 1.5 3 ^ + exp 2 *`

OPEN_FILE

A text file is opened for use in other commands where a unit number is necessary. The program automatically assigns a unit number.

Syntax	OPEN_FILE
Prompt	`FILE NAME: <FILE NAME>` A legal file name must be specified.

SET_COMMAND_UNITS

This command is useful for reading inputs already prepared by a textual editor on a file. Such inputs can be a table of values or a large number of parameters. The file must be opened with *OPEN_FILE* above.

The first two lines of an input file are skipped before any input is read.

Syntax	SET_COMMAND_UNIT
Prompt	`INPUT UNIT NUMBER /5/: <INPUT UNIT NUMBER>` Specify the input unit number returned from an OPEN_FILE command. The next command is taken from the file connected to this unit number. The last command on such a file must be EXIT or SET_INTERACTIVE when you are in POLY, POST or SYS module, in order to go back to read input from the keyboard. The default value is the current input unit.
	`OUTPUT UNIT NUMBER /6/: <OUTPUT UNIT NAME>` Specify the output unit number returned from an OPEN_FILE command.

SET_ECHO

Syntax	SET_ECHO
	Add this command at the beginning of a Console Mode MACRO *.TCM file (or to the primary MACRO file on the top level if any sub-level(s) of MACRO files are used). Automatically displays on screen the complete details of various commands in all the sequential operations in the software, that are enforced according to the MACRO file(s). This command is not applicable if MACRO files are not used.

SET_INTERACTIVE_MODE

Syntax	SET_INTERACTIVE_MODE
	Resets the input and output units to their initial values, i.e. keyboard and screen. Remember to add this as the last command to your MACRO files, in order to stop execution of the command file in the POST module. It has no meaning in interactive mode.

SET_LOG_FILE

Use it to save, in a basic text file, everything typed in the Thermo-Calc software. When having problems executing a command sequence, this command can be used to save the command typed onto a log file. The manager can then check what the problem might be.

This command also makes the system echo the full command for all commands typed. This feature is useful when demonstrating the system since the abbreviated commands are often difficult.

The saved log file can then be edited as a MACRO file by using a basic text editor. This is useful to run the MACRO file(s) for similar calculations (the same system but slightly varied temperature-pressure-composition conditions), or run the example MACRO files.

Syntax	SET_LOG_FILE
Prompt	A **Save** window opens to specify a file name for the *.LOG file, so that the path (in the **Save in** field) and **File name** can be specified. If an *.LOG file with the same name exists in the current working directory, it is overwritten. The **Files of type** (i.e. LOG) cannot be changed. Click **Save** or **Cancel** as applicable.

SET_PLOT_ENVIRONMENT

Allows you, preferably in the initiation file tc.ini, to set the plot devices you normally have access to as this can vary for different Thermo-Calc installations.

This command must be terminated with an empty line or two commas.

Syntax	SET_PLOT_ENVIRONMENT
Prompts	`DEFAULT PLOTDEVICE NUMBER /DEFAULT NUMBER/: <DEVICE NUMBER>` The number given is unique for each type of graphical device. For plotting various types of Thermo-Calc graphs on screen, the default graphical device number should normally be selected. A question mark ? gives you a list of all available graphical devices used in the Thermo-Calc Console Mode.
	`PSEUDO FILE NAME: <PSEUDO-FILE NAME>` The name given here is a symbol which can be used to refer to a physical graphics device when asked for graphical output file.
	`PLOTDEVICE NUMBER /1/: <DEVICE NUMBER>` The number that specifies the type of graphical device.

	`PLOT FILE NAME: <FILE NAME OR PRINTER NAME>`
	
	The name of the graphical device on system level (name of file or printer).
	
	Example
	
	`SET_PLOT_ENVIRONMENT 1 lasp 5 a0tr,,`
	
	This example sets the default plot device number to `1`, and defines an alias named `lasp` with plot device `5` and connected to a printer named `a0tr`.

SET_TC_OPTIONS

Predefine the default values of three general Thermo-Calc options for performing all the sequential single-points, stepping and mapping calculations.

Syntax	SET_TC_OPTIONS
Prompts	`USE_GLOBAL_MINIMIZATION /YES/: <Y OR N>` If `Y`, it is done when possible (depending on the conditions setting), and a Global Minimization test is performed when an equilibrium is reached. This costs more computer time but the calculations are more robust. For `N` Global Minimization is not used to reach the equilibrium.
	`GLOBAL_TEST_INTERVAL /0/: <AN INTEGER VALUE>` The integer number determines how often the Global Minimization should be used during calculations with *STEP_WITH_OPTIONS* on page 172 and *MAP* on page 158. If it is set to 0 (zero), the recommended global test interval is used: i.e. every tenth step and at each phase change during STEP calculations, and only at node points during MAP calculations. Any other positive integer number, n, suggests it performs Global Minimization tests at every nth step during STEP and MAP calculations. Of course, the Global Minimization test is always carried out at a phase change or a node point. The Thermo-Calc Console Mode is initially configured with 0 (zero) as the default value, but such a configuration for the default value can be changed by this option.
	`AUTOMATICALLY_ADD_INITIAL_EQUILIBRIA /YES/: <Y OR N>` When *ADD_INITIAL_EQUILIBRIUM* on page 119 is not used before a MAP calculation, a mesh of INITIAL_EQUILIBRIUM are automatically added before the mapping procedure itself takes place. This mesh is only available when the two independent axis-variables have already been defined by the POLY command.
	`SET_AXIS_VARIABLE` For `Y` the POLY module always enforces the automatic procedures for adding initial equilibrium points during mapping. For `N` no initial equilibrium points are then added during mapping.

STOP_ON_ERROR

Useful for batch jobs in order to prevent that an erroneous sequence of commands to the program causes waste of computer activity.

Syntax	STOP_ON_ERROR
	Give the preferred argument (`ON` or `OFF`) after the STOP command. If the default argument ON is given, the program terminates after an illegal or ambiguous command. It is possible to reset the effect of this command by giving the argument OFF.

TABULATION_REACTION Commands

In this section:

ENTER_FUNCTION

Used to define a new thermodynamic function to be included in the sixth column of tabular output. The last entered function is listed in the 6th column of all tables, and can be plotted (if choosing the Plot Column value 6) in all graphs, for all subsequently tabulated substances or reactions.

Syntax	ENTER_FUNCTION
Prompt	`NAME: <NAME>` `FUNCTION: <DEFINITION>`
	A limited number of state variables, G, H, S, T, P, V and H298 can be used in the function definition. For example, -(G-H)/T can be defined as a function which is identical to S for a substance or ΔS for a reaction, G+T*S-P*V can be defined as a function which is U (internal energy) for a substance or ΔU (internal energy change) for a reaction.
Options	`Name`of the function (maximum 8 characters) and `Definition` of the function. A long function can be typed over several lines. A function should be terminated by a semicolon (;) or an empty line.

Example Output

For the pure substance Fe (using the SSUB database), by entering a function as `G+T*S-P*V`, the following table is obtained by typing <Enter> in the prompt `Output file /SCREEN/:`, and the figure by typing `6` in the prompt `Plot column /2/:`

```
TAB: ENTER-FUNCTION
Name: InEnergy
Function: G+T*S-P*V;
TAB: TABULATE_SUBSTANCE
Substance (phase): Fe
Pressure /100000/: <Enter>
Low temperature limit /298.15/: <Enter>
High temperature limit /2000/: <Enter>
Step in temperature /100/: <Enter>
Output file /try1/: try2
Graphical output? /Y/: <Enter>
Plot column? /2/: 6
```

This gives the following output:

```
O U T P U T  F R O M  T H E R M O - C A L C
Column 6: InEnergy G+T*S-P*V
Phase : FE_S                          Pressure :      100000.00
```

```
Specie: FE
***************************************************************
    T      Cp            H             S             G             InEnergy
   (K)    (Joule/K)      (Joule)      (Joule/K)      (Joule)
***************************************************************
  298.15   2.48446E+01   2.17972E-06   2.72800E+01  -8.13353E+03   2.17972E-06
  300.00   2.48905E+01   4.60049E+01   2.74338E+01  -8.18414E+03   4.60049E+01
  400.00   2.71299E+01   2.64957E+03   3.49085E+01  -1.13138E+04   2.64957E+03
  500.00   2.93561E+01   5.47211E+03   4.11976E+01  -1.51267E+04   5.47211E+03
  600.00   3.19293E+01   8.53245E+03   4.67701E+01  -1.95296E+04   8.53245E+03
  700.00   3.50985E+01   1.18777E+04   5.19207E+01  -2.44667E+04   1.18777E+04
  800.00   3.92042E+01   1.55830E+04   5.68623E+01  -2.99068E+04   1.55830E+04
  900.00   4.49645E+01   1.97726E+04   6.17903E+01  -3.58387E+04   1.97726E+04
1000.00   5.42147E+01   2.46891E+04   6.69619E+01  -4.22728E+04   2.46891E+04
1100.00   4.55851E+01   2.99025E+04   7.19412E+01  -4.92328E+04   2.99025E+04 $ Stable
phase is FE_S2
1200.00   3.40840E+01   3.51037E+04   7.64466E+01  -5.66322E+04   3.51037E+04
1300.00   3.49398E+01   3.85549E+04   7.92086E+01  -6.44162E+04   3.85549E+04  1400.00
3.57994E+01   4.20918E+04   8.18293E+01  -7.24692E+04   4.20918E+04
1500.00   3.66636E+01   4.57149E+04   8.43287E+01  -8.07780E+04   4.57149E+04
1600.00   3.75330E+01   4.94247E+04   8.67226E+01  -8.93314E+04   4.94247E+04 $ Stable
phase is FE_S3
1700.00   4.05217E+01   5.41173E+04   8.95609E+01  -9.81363E+04   5.41173E+04
1800.00   4.12595E+01   5.82055E+04   9.18975E+01  -1.07210E+05   5.82055E+04 $ Stable
phase is FE_L
1900.00   4.60000E+01   7.74165E+04   1.02377E+02  -1.17099E+05   7.74165E+04  2000.00
4.60000E+01   8.20165E+04   1.04736E+02  -1.27456E+05   8.20165E+04
```

ENTER_REACTION

This is the same as *TABULATE_REACTION* on page 258 in that both the chemical reaction can be specified and the thermodynamic property changes for the reaction can be generated as a table, and plotted as a graph or saved as an experimental file (with an extension of .EXP). The thermodynamic data are automatically retrieved from the SSUB5 database (by default), or the current database set by *SWITCH_DATABASE* on page 20 (thus there is no need to use the DATA module before this command). You can select any other database by using the SWITCH_DATABASE command in the TAB module. The rules to specify a reaction are the same as in the command TABULATE_REACTION.

Syntax	ENTER_REACTION
Prompts	SAME REACTION? /Y/: <Y OR N> This displays if there is at least one reaction already defined either by the command ENTER_REACTION or TABULATE_REACTION. For Y the next prompt for defining reaction is not asked.
	REACTION: <CHEMICAL REACTION EQUATION> & <RA+RB=PC+PD> A long reaction can be typed over several lines. A reaction should be terminated by a semicolon (;) or an empty line. A list showing the used database, defined elements, getting-data sequence, references, etc.: Pressure /100000/: <pressure of interest, in pa> Low temperature limit /298.15/: <T-low, in K> High temperature limit /2000/: <T-high, in K> Step in temperature /100/: <T-step>
	OUTPUT FILE /SCREEN/: <RETURN FOR SCREEN, OR TYPE A FILE NAME> If you type Return for screen a list of thermodynamic property changes for the defined/chosen reaction is shown up as a table (which is demonstrated in the example outputs given below), and the command is terminated. If typing a file name, the table is displayed and saved as an *.EXP or an *.TAB file. Then prompts:
	GRAPHICAL OUTPUT /Y/: <Y OR N> For N a table is output to the screen and this table is saved as a basic text file with the default extension .TAB under the current working directory. In this case, the graph cannot be plotted. For Y a table is created with all thermodynamic functions as normal (which is displayed on screen) and generates a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which is plotted on screen and saved as an *.EXP file), and it further asks which column is to be plotted on the resulting graph:
	PLOT COLUMN? /2/: <1 OR 2 OR 3 OR 4 OR 5 OR 6> Specify which property (as column number) to be plotted as the Y-axis (versus temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (i.e. the plotted column) for the graph is written into an *.EXP file using the DATAPLOT format. The default column 2 is heat capacity, 3 enthalpy, 4 entropy and 5 Gibbs energy, and the additional column 6 is a user-entered function. The table displays on the screen. The plot then displays on the screen, followed by a POST prompt. The POST module (postprocessor) automatically opens, and all types of the POST-module commands to refine the plotted diagram may be used. Possibilities include scaling the

Syntax	ENTER_REACTION
	X/Y-axis, changing the X/Y-axis texts, etc. The command BACK or EXIT at the `POST` prompt always takes you back to the TAB module.

LIST_SUBSTANCES

List all species in the current database with a certain set of elements. This is useful to specify species as pure substances in the command *TABULATE_SUBSTANCE* on page 260.

Syntax	LIST_SUBSTANCES
Prompts	`WITH ELEMENTS /*/ <* OR A SET OF ELEMENTS>` The elements building up various species must be specified. The wildcard * means all elements in the current database. If some elements are specified, it also prompts for how to list species with such elements.
	`EXCLUSIVELY WITH THOSE ELEMENTS /Y/: <Y OR N>` All species (not phases) in the current database are searched. If the answer is `Y` (the default), no other elements are allowed in searching, except for those specified. If `N` all species containing at least one of the specified elements are listed.

Example Output

For the SSOL5 database, there are different lists, as shown below:

```
TAB: l-sub
With elements /*/: <Enter>
VA                      AG                      AL
AM                      AS                      AU
B                       BA                     BE BI   BR C
CA                      CD                      CE
CL                      CO                      CR
CS                      CU                      DY
ER                      EU                      F
FE                      GA                      GD
GE                      H                       HF
HG                      HO                      I
IN                      IR                      K
LA                      LI                      LU
MG                      MN                      MO
```

```
NA                      NB
ND                      NI                      NP
OS                      P
PA                      PB                      PD
PR                      PT                      PU
RB                      RE                      RH
RU                      S                       SB
SC                      SE                      SI
SM                      SN                      SR
TA                      TB                      TC
TE                      TH                      TI
TL                      TM                      U
V                       W                       Y
YB                      ZN                      ZR
C1                      C2                      C3
C4                      C5                      C6
C7                      N2                      O2
P1                      P2                      P4
NA/+1                   K/+1                    CL/-1
H2O                     H1/+                    LI1/+
CS1/+                   RB1/+                   O1H1/-
F1/-                    BR1/-                   I1/-
H1/+1                   H1O1/-1                 SI1O2 CA1O1 TAB:
TAB: l-sub
With elements /*/: fe cr ni c n o
Exclusively with those elements? /Y/: <Enter>
C                CR                     FE
N                NI                     O
C                C2                     C3
C           C5                     C6
C                N2                     O2
TAB:l-sub fe cr ni c n o
Exclusively with those elements? /Y/: n
C                CR                     FE N   NI O
C1               C2                     C3
C4               C5                     C6
```

```
C7          N2          O2
H2O         O1H1/-      H1O1/-1 SI1O2 CA1O1 TAB:
```

SET_ENERGY_UNIT

Set the energy unit as calories or Joule, in all the subsequent outputs (tables, graphs, and files) in the current calculation operation.

Syntax	SET_ENERGY_UNIT
Prompt	`ENERGY UNIT (C=CAL, J=JOULE) /J/:` `<UNIT>`

SET_PLOT_FORMAT

Set the plotting format while the result is plotted on screen or the file is saved as EXP file [using DATAPLOT graphical format] and TAB file (as a basic text file and as displayed on screen).

This command is different from *SET_PLOT_FORMAT* on page 210 in the POST module, but is identical to *SET_PLOT_ENVIRONMENT* on page 243 in the SYS (system utility) monitor.

Syntax	SET_PLOT_FORMAT
Options	The default set `Unit`is `1` (Windows) or `9` (Linux) for graphical presentation on screen, or all other units for outputs as *.EXP and *.TAB files. Also set the format of the graphical output to different graphical devices. The default unit 22 (1 for legacy Windows, and 9 for legacy Linux) may be altered with the SET_PLOT_ ENVIRONMENT command in the SYS monitor or by the TC.INI file.

SWITCH_DATABASE

By default, the TAB module always automatically retrieves thermodynamic data for substances or reactant/product substances in a defined reaction, or for pure phases or ideal solution phases, from the SSUB Substances Database. However, you can select thermodynamic data for complex non-ideal solution phases from another database. This is possible with this command.

This command switches (or changes) from the current database to a new one, and reinitializes the entire TAB module for defining a substance or reaction and the GIBBS workspace for storing retrieved data. All the directly connected databases as predefined by the original database initiation file (TC_INITD.TDB file in the /DATA/ area for Windows environments, or the initd.tdb file in the \data\ area for Linux) are listed by pressing the <Enter> key without giving any argument. You can also supply your own database by giving the argument USER and then the database name and, if it is not located in the current working directory, its correct path.

This command (when required to use another database for subsequent tabulations) must be used before executing the TAB commands *TABULATE_SUBSTANCE* on page 260 for substances, and *ENTER_REACTION* on page 247 or *TABULATE_REACTION* on page 258 for reactions.

When a predefined database or an appropriate USER database is switched on, the DATA command *GET_DATA* on page 15 is automatically executed; therefore, the TAB module immediately displays a message

Example

Example of TAB module message with the PSUB database specified

```
TAB: SW PSUB
THERMODYNAMIC DATABASE Module running on PC/Windows NT
Current database: TC Public Substances Database
VA DEFINED
REINITIATING GES5
VA DEFINED
ELEMENTS
SPECIES
PHASES
PARAMETERS
FUNCTIONS
-OK- TAB:
```

Syntax	SWITCH_DATABASE

Prompt	`DATABASE /XYZ/: <NEW DATABASE NAME>`
	Specify a `New database name`. `XYZ` stands for the default SSUB database or the current database which is switched on. If an appropriate database name is not given and the `USER` option is not used, this list displays:
	`Use one of these databases`
	`PURE4 = SGTE Pure Elements Database v4`
	`SSUB3 = SGTE Substances Database v3`
	`USER = user defined database`
	`DATABASE NAME /XYN/: <new database name>`

Notes	DATABASE NAME: The new database is specified by typing the abbreviation given in the front of one of the available predefined databases. For convenience when switching/appending databases or recently purchased databases from Thermo-Calc Software or its agents, you can add them to the predefined database list in the database initiation file TC_INITD.TDB or initd.tdb of your installed software.

Syntax	SWITCH_DATABASE
	When this command is used in a MACRO (*.TCM) file, if the USER option is selected, the database setup file name (*setup.TDB) containing the setup definitions of the USER database, and its correct path, must be provided. If a USER database name or its path is not given on the same line of the SWITCH_DATABASE command, or if it is incomplete or incorrect, an Open window displays to specify the filename of the database setup file name (*setup.TDB) of the to-be-switched USER database. Unlike in the DATA module, if a USER database is used, the Gibbs energy system is also reinitialized and only data from the USER databases are retrieved for tabulations. After this command, those commands to define systems (in terms of either elements or species), to reject/restore phases or species, retrieve data, as well as to append data from additional database(s), can be done.

TABULATE

Gives the values of an already-defined table (with various properties varied with the stepping axis variable) from equilibria calculated with a STEP command. You can tabulate a table even if the table is entered after the STEP calculation.

Syntax	TABULATE
Prompts	NAME: <NAME OF A DEFINED TABLE> The name of the table must be given. The table must be ENTERED.
	OUTPUT TO SCREEN OR FILE /SCREEN/: <FILE NAME> The table can be listed on the screen (by pressing <Enter>), on a file (the file name must be specified here).

TABULATE_DERIVATIVES

This command is mainly for system debugging purposes and for experienced users. It automatically calculates all partial derivatives for a pure phase or solution phase at a given composition.

This is not the same as the chemical potential.

Before using this command, the system must be defined and thermodynamic data retrieved from an appropriate solution database in the DATA module. The phase name can be given in either upper, lower or mixed cases, and can be abbreviated until it is unique, e.g. fcc, BCC, cem, Liq, etc. as the SSOL database is used for the tabulations, or GAS, FE-S, wustite, Fe2O3-hem, etc. as the SSUB database is used. The module then optionally goes through a specific sublattice or all the possible sublattices for the specified phase, as well as through the whole framework of a currently defined system (including all the defined elements/species, as well as the default defined vacancy and electron if necessary for the current database), when it prompts for inputs of site fractions for various species possibly occupying such sublattices. Therefore, thermodynamic properties can be tabulated for a pure phase, or for a solution phase as a pure end-member (with a composition definition for the corresponding non-interacting species on all necessary sublattices in the end-member), or for a solution phase as a real solution (with a composition definition for related interacting species on all necessary sublattices in the solution).

<table>
<tr><td>Syntax</td><td>TABULATE_DERIVATIVES</td></tr>
<tr><td>Prompts</td><td>
<code>PHASE NAME /XXXX/: <NAME OF THE PHASE></code>

Specify the phase name of a pure phase or a solution phase.

<code>XXXX</code> is the last phase the TAB module accounted for. It automatically prompts for other options and necessary inputs for the definition of the composition of the specified phase.

For a pure phase (such as <code>Fe-S</code>, <code>Wustite</code>, and <code>Fe2O3-Hematite</code>), no further composition definition is needed. For a solution phase with one sublattice (such as <code>AQUEOUS</code> solution, <code>GAS</code> mixture and <code>SLAG</code> solution), it requests n-1 site-fraction inputs for the phase constituents (if there are n species in the defined phase within the whole framework of currently defined system, including all the defined elements/species, as well as the default defined vacancy and electron if necessary for the current database); the nth species is automatically assigned as the rest.

The sum of input site fractions must not exceed unity.
</td></tr>
<tr><td></td><td>
Example

For example, these prompts and inputs can be seen for a <code>LIQUID</code> solution phase (from the SSOL database) within the Fe-Cr-Ni-C-N-O framework (Note that by default, the unprompted <code>Ni</code> species are assigned as the rest):

<code>FRACTION OF CONSTITUENT (RETURN FOR PROMPT): <RETURN> C /1/: .05</code>
<code>CR /1/: .1</code>
<code>FE /1/: < RETURN></code>
<code>SUM OF FRACTIONS EXCEED UNITY, PLEASE REENTER</code>
<code>FE /1/: .8</code>
<code>N /1/: .005</code>

For a solution phase with two or more sublattices (such as FCC alloy solution, and ION_
</td></tr>
</table>

Syntax	TABULATE_DERIVATIVES

	LIQ ionic liquid solution), it first asks for which sublattice the site-fractions of constituents shall be specified: the default value 0 for all possible sublattices, and a given positive number for a specific sublattice (of course, this number must be reasonable for the currently specified phase, i.e. it must be smaller than the total sublattice number of the phase). Then, for the given sublattice or for all sublattices, it prompts for all the necessary inputs of site-fractions for the possible constituents on each sublattice n-1 times if there are n species on such a sublattice in the defined phase; the nth species on each of the sublattices is automatically assigned as the rest. For example, the following prompts and inputs can be seen for an FCC solution phase (from the SSOL5 database) within the Fe-Cr-Ni-C-N-O framework (Note that by default, the unprompted O species on its sublattice 1 and VA on its sublattice 2 are assigned as the rest):
	`SPECIFY SUBLATTICE (0 FOR ALL) /0/: <RETURN>`
	`FRACTIONS IN SUBLATTICE 1`
	`CR /1/: .1`
	`FE /1/: .8`
	`NI /1/: .0995`
	`FRACTIONS IN SUBLATTICE 2`
	`C /1/: .05`
	`N /1/: .05`

	`TEMPERATURE /2000/: <TEMPERATURE OF INTEREST, IN K>` Specify the temperature of interest in K.
	`PRESSURE /100000/: <PRESSURE OF INTEREST, IN PA>` Specify the pressure of interest in Pa.

Example Output

For the FCC phase in the Fe-Cr-Ni-C-N-O system (using the SSOL database), the following table is obtained at a specified composition:

```
TAB: TAB_DER
Phase name /BCC/: FCC
SPECIFY SUBLATTICE (0 FOR ALL) /0/: <Enter>
FRACTIONS IN SUBLATTICE 1
CR /1/: .1
FE /1/: .8
NI /1/: .3
SUM OF FRACTIONS EXCEED UNITY, PLEASE REENTER
```

```
NI /0/: .0995

FRACTIONS IN SUBLATTICE 2

C /1/: .05

N /1/: .05

Temperature /1800/: <Enter>

Pressure /100000/: <Enter>

Gibbs energy: .................... -1.27432533E+05

Helmholz energy: ................. -1.27433205E+05

Enthalpy: ........................  5.95773994E+04

Internal energy: .................  5.95767279E+04

Entropy: .........................  1.03894407E+02

Molar volume: ....................  6.71473258E-06

Thermal expansivity: .............  7.63886416E-05

Isothermal compressibility: .......  6.02925387E-12

Heat capacity at constant pressure:  4.33555074E+01

First partial derivative with respect to CR in sublattice  1     of Gibbs energy:
................ -1.26034739E+05     of enthalpy: ....................  4.63000206E+04
of entropy: ....................  9.57415334E+01     of volume:
.....................  6.87203263E-06

Second partial derivative of Gibbs energy with respect to also

  CR                       in  1:  1.54392858E+05

  FE                       in  1: -1.53353158E+04

  NI                       in  1: -1.71750366E+04

  O                        in  1:  0.00000000E+00

  C                        in  2: -1.82016870E+05

in  2: -3.73062665E+05

  VA                       in  2: -9.36260878E+04

irst partial derivative with respect to FE in sublattice  1     of Gibbs energy:
................ -1.02869265E+05

    of enthalpy: ....................  6.11738912E+04     of entropy:
.....................  9.11350866E+01     of volume: ......................
7.53441165E-06

Second partial derivative of Gibbs energy with respect to also

FE                       in  1:  1.82508696E+04

NI                       in  1: -3.07043434E+03

in  1:  0.00000000E+00

C                        in  2: -1.36027071E+05

in  2: -2.13007485E+05
```

```
VA                       in  2: -1.11741180E+05

First partial derivative with respect to NI in sublattice  1     of Gibbs energy:
............... -1.32427029E+05     of enthalpy: ...................  5.21563580E+04
of entropy: ...................  1.02546326E+02     of volume:
.....................  0.00000000E+00

    Second partial derivative of Gibbs energy with respect to also

  NI                       in  1:  1.48390257E+05

in  1:  0.00000000E+00

  C                        in  2: -7.70620431E+04

in  2: -1.61551726E+05

VA                       in  2: -1.12772206E+05

First partial derivative with respect to O in sublattice  1     of Gibbs energy:
............... -2.62929308E+05     of enthalpy: ...................  5.02555370E+04
of entropy: ...................  1.73991581E+02     of volume:
.....................  0.00000000E+00

Second partial derivative of Gibbs energy with respect to also

in  1:  2.99322360E+07

C                        in  2:  0.00000000E+00

N                        in  2:  0.00000000E+00

VA                       in  2: -1.82377137E+05

First partial derivative with respect to C in sublattice  2     of Gibbs energy:
............... -1.59508417E+05     of enthalpy: ...................  1.21269096E+05
of entropy: ...................  1.55987507E+02     of volume:
.....................  1.06885187E-05

Second partial derivative of Gibbs energy with respect to also

C                        in  2:  2.99322360E+05

N                        in  2: -1.90144000E+04

VA                       in  2: -2.78465070E+04

First partial derivative with respect to N in sublattice  2     of Gibbs energy:
............... -2.39671400E+05     of enthalpy: ...................  4.04456947E+04
of entropy: ...................  1.55620608E+02     of volume:
.....................  0.00000000E+00

Second partial derivative of Gibbs energy with respect to also

N                        in  2:  2.99322360E+05

VA                       in  2: -3.30383665E+04

First partial derivative with respect to VA in sublattice  2     of Gibbs energy:
............... -9.52042686E+04     of enthalpy: ...................  5.37142294E+04
of entropy: ...................  8.27324989E+01     of volume:
.....................  6.86700739E-06

Second partial derivative of Gibbs energy with respect to also

VA                       in  2:  1.66290200E+04
```

TABULATE_REACTION

Tabulate thermodynamic property changes for a chemical reaction as a table, or variation of one chosen property can be plotted against temperature as a graph, or saved as an experimental file (with an extension of .EXP). The thermodynamic data are automatically retrieved from the SSUB5 database (by default), or the current database set by the SWITCH_DATABASE command (thus there is no need to use the DATA module before this command). You can select any other database by using *SWITCH_DATABASE* on page 251 in the TAB module.

It is important to remember the rules to specify a reaction:

- The reaction is specified, by giving the reactants and products, such as `Ga+S=GaS`, `5Al+2Fe=Al5Fe2`.
- Under Linux environments, the notation for a chemical formula treats upper and lower cases as the same (such as `ga+s=GaS, 5Al+2Fe=al5fe2`); but under Windows, the TAB module only accepts upper cases, e.g. `GA+S=GAS, 5AL+2FE=AL5FE2`.
- Elements with a single letter symbol must be followed by a stoichiometry factor, even if it is 1; thus, the symbol `CO` is taken as cobalt, carbon monoxide must be given as `C1O1` (or `O1C1`).
- All the reactants and products must be defined as species-stoichiometries in the currently-used database; in other words, the reaction definition only accepts already-defined species-stoichiometries as reactants and products. For example, if a database does not have a species defined with a stoichiometry of Al5Fe2 {e.g. the Al and Fe (but not Al5Fe2) may be defined as species, and the Al5Fe2 phase may be defined as a two-sublattice stoichiometric phase [Al]5[Fe]2, rather than a single sublattice stoichiometric phase [Al5Fe2]}, then the reaction 5Al+2Fe=Al5Fe2 cannot be tabulated.
- Make sure that the Gibbs free energy expression for a stoichiometric phase is really corresponding to the species-stoichiometry. For example, if a database has a definition of Function GAL5FE2 for 1/7 of the AL5FE2 formula, then the relation G(Al5Fe2)-)=7*GAL5FE2 must be defined; otherwise, the tabulated thermodynamic properties for the reaction 5Al+2Fe=Al5Fe2 is wrong.

Syntax	TABULATE_REACTION
Prompts	`SAME REACTION? /Y/: <Y OR N>` This prompt displays if there is at least one reaction already defined either with the commands *ENTER_REACTION* on page 247 or TABULATE_REACTION. If you answer `Y`, then the next prompt is not asked.
	`REACTION: <CHEMICAL REACTION EQUATION> & <RA+RB=PC+PD>` A long reaction can be typed over several lines. A reaction should be terminated by a

Syntax	TABULATE_REACTION

	semicolon (;) or an empty line. A list showing the used database, defined elements, getting-data sequence, references, etc.
	`Pressure /100000/: <pressure of interest, in pa>`
	`Low temperature limit /298.15/: <T-low, in K> Specify the starting temperature K.`
	`High temperature limit /298.15/: <T-high, in K> Specify the ending temperature K. Step in temperature /100/: <T-step> Specify the temperature step for the tabulation.`

	`OUTPUT FILE /SCREEN/: <RETURN FOR SCREEN, OR TYPE A FILE NAME>` If you type `Return for screen`, a list of thermodynamic property changes for the defined/chosen reaction is shown as a table (which is demonstrated in the example outputs given below), and the command is terminated. If typing a file name, the table is both displayed on screen and saved as an *.EXP or a *.TAB file, and the program prompts:
	`GRAPHICAL OUTPUT /Y/: <Y OR N>` • For N a table is output to the screen, and the same table is saved as a basic text file with the default extension .TAB under the current working directory. In this case, the graph cannot be plotted. • For `Y` a table is created with all thermodynamic functions as normal (which is displayed on screen) and generates a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which is plotted on screen and saved as an *.EXP file), and it also asks which column is to be plotted on the resulting graph:
	`PLOT COLUMN? /2/: <1 OR 2 OR 3 OR 4 OR 5 OR 6>` Specify which property (as column number) to be plotted as the Y-axis (versus temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (i.e. the plotted column) for the graph is written into an *.EXP file using the DATAPLOT format. The default column 2 is heat capacity, 3 enthalpy, 4 entropy and 5 Gibbs energy, and the additional column 6 is a user-entered function. The table displays on the screen. The plot then displays on the screen, followed by a POST: prompt. The POST module (postprocessor) automatically opens, and all types of the POST-module commands to refine the plotted diagram may be used. Possibilities include scaling the X/Y-axis, changing the X/Y-axis texts, etc. The command BACK or EXIT at the POST: prompt always takes you back to the TAB module.

Syntax	TABULATE_REACTION
	Also see the *DATPLOT User Guide* included with this documentation set.

Example Output

For the reaction Ga+S=GaS (using the SSUB5 database), the following table is obtained by typing <Enter> in the prompt Output file /SCREEN/, and the figure by typing `2` in the prompt `Plot column /2/`:

```
    O U T P U T   F R O M   T H E R M O - C A L C

     Reaction:  S+GA=GA1S1

  stable as S_S

  GA stable as GA_S

  GA1S1 stable as GA1S1_S

  Delta-Cp      Delta-H       Delta-S       Delta-G

   (K)    (Joule/K)      (Joule)      (Joule/K)      (Joule)

  ***************************************************************

  298.15  -2.79489E+00  -2.09200E+05  -1.50580E+01  -2.04710E+05

  300.00  -2.87516E+00  -2.09205E+05  -1.50755E+01  -2.04683E+05

  302.   ---- GA becomes GA_L ,delta-H =   5589.80

  367.   ---- S becomes S_S2 ,delta-H =    401.00

  389.   ---- S becomes S_L ,delta-H =   1721.00

  400.00  -1.22278E+01  -2.17521E+05  -4.07488E+01  -2.01222E+05

  500.00  -1.57686E+01  -2.19323E+05  -4.47280E+01  -1.96959E+05

  600.00  -1.02518E+01  -2.20583E+05  -4.70410E+01  -1.92359E+05

  700.00  -6.58954E+00  -2.21430E+05  -4.83534E+01  -1.87583E+05

  800.00  -4.46246E+00  -2.21963E+05  -4.90686E+01  -1.82709E+05

  900.00  -2.82671E+00  -2.22328E+05  -4.94995E+01  -1.77778E+05

  1000.00  -1.21787E+00  -2.22530E+05  -4.97138E+01  -1.72816E+05

  1100.00   3.71702E-01  -2.22572E+05  -4.97552E+01  -1.67841E+05
```

TABULATE_SUBSTANCE

Tabulate thermodynamic properties of a substance (with a given chemical formula but it is not certain in which phase/state it may form) or a pure stoichiometric phase (getting data from a specific substance database such as SSUB5, or from a specific solution database such as SSOL5 and TCFE8), or of a solution phase with a fixed composition (getting data from a specific solution database such as SSOL5 and TCFE8), under a constant pressure and various temperatures.

In case of a pure stoichiometric phase or a solution phase with a fixed composition, you must already define the system and get thermodynamic data from an appropriate solution database in the DATA module before using this TAB command. The phase name can be given either upper or lower or mixed cases, and can be abbreviated until it is unique, e.g. `fcc`, `BCC`, `Liq`, `cem`, `Al5Fe2`, etc. as the SSOL database is used for the tabulations, or `GAS`, `FE-S`, `wustite`, `Fe2O3-hem`, etc. as the SSUB database is used. The module then optionally goes through a specific sublattice or all the possible sublattices for the specified phase, as well as through the whole framework of a currently defined system (including all the defined elements/species, as well as the default defined vacancy and electron if necessary for the current database), when it prompts for inputs of site fractions for various species possibly occupying such sublattices. Therefore, you can tabulate thermodynamic properties for a pure phase, or for a solution phase as a pure end-member (with a composition definition for the corresponding non-interacting species on all necessary sublattices in the end-member), or for a solution phase as a real solution (with a composition definition for related interacting species on all necessary sublattices in the solution).

Moreover, the variation for one of the tabulated properties for a substance, a pure stoichiometric phase, or a solution phase with a fixed composition can be plotted against temperature as a graph or saved as an experimental file (with an extension of .EXP).

Syntax	TABULATE_SUBSTANCE
Prompt	`SUBSTANCE (PHASE): <NAME OF THE SPECIES OR PHASE>` In case of a substance, give its chemical formula, e.g. Fe, H2, C1H6, FeC1, CaC1O3, MgSiO3, etc. When the TABULATE_SUBSTANCE <SUBSTANCE> command sequence is used for the first time in the TAB module, it always uses the SSUB Substances Database as the default database. If the SSUB database is not available for the current Thermo-Calc installation and a specific database has not been switched on yet, it may prompt to specify an appropriate substance or solution database (e.g. typing `SSUB2`, `SSUB3`, `SSUB4`, `SSUB5`; `SSOL2`, `SSOL4`, `SSOL5`; `TCFE3`, `TCFE4`, `TCFE5`, `TCFE6`, `TCFE7`, `TCFE8` etc.). Prior to this command, you can also use the *SWITCH_DATABASE* on page 251 command in the TAB module to set the current database. If a solution database is set as the current database, only the neutral species that are valid substance standing by themselves as phases can be tabulated. Before other prompts, a list of the used database, defined elements, getting-data sequence, references, etc. is displayed on screen. From such information, you know what the TAB module is performing. In case of a pure stoichiometric phase or a solution phase, give its phase name, e.g. FCC, CEMENTITE, LIQUID, SLAG, AQUEOUS, GAS, Al5Fe2, Fe-S, Wustite, Fe2O3-Hematite, etc. REMEMBER: It is important that you have already defined the system and gotten thermodynamic data from an appropriate solution database in the DATA module before using this TAB_SUB command. It automatically prompts for other options and necessary inputs for the definition of the composition of the specified phase. For a pure stoichiometric phase defined with one single sublattice (such as Fe-S, Wustite, and Fe2O3-Hematite), no further composition definition is needed. For a solution phase defined with one sublattice (such as AQUEOUS solution, GAS mixture and SLAG solution), it requests n-1 site-fraction inputs for the phase constituents (if there are *n* species in the defined phase within the whole framework of currently defined system, including all the defined elements/species, as well as the default defined vacancy and electron if necessary for the current database); the *n*th species is automatically assigned as the rest.

Syntax	TABULATE_SUBSTANCE
	The sum of input site fractions must not exceed unity. For example, the following prompts and inputs can be seen for a LIQUID solution phase (from the SSOL database) within the Fe-Cr-Ni-C-N-O framework (Note that by default, the unprompted Ni species is assigned as the rest):

	`FRACTION OF CONSTITUENT (RETURN FOR PROMPT): <RETURN>`
	`C /1/: .05`
	`CR /1/: .1`
	`FE /1/: <RETURN>`

	`SUM OF FRACTIONS EXCEED UNITY, PLEASE REENTER`
	`FE /1/: .8`
	`N /1/: .005`
	
	For a pure stoichiometric phase defined with two or more sublattices (such as Al5Fe2 intermetallic stoichiometric phase) and for a solution phase defined with two or more sublattices (such as FCC alloy solution, and ION_LIQ ionic liquid solution, Al5Fe4 intermetallic solution phase), it first asks for which sublattice the site-fractions of constituents shall be specified: the default value 0 for all possible sublattices, and a given positive number for a specific sublattice (this number must be reasonable for the currently specified phase, i.e. it must be smaller than the total sublattice number of the phase). Then, for the given sublattice or for all sublattices, it prompts for all the necessary inputs of site-fractions for the possible constituents on each sublattice *n*-1 times if there are n species on such a sublattice in the defined phase; the *n*th species on each of the sublattices are automatically assigned as the rest. For example, these prompts and inputs can be seen for an FCC solution phase (from the SSOL database) within the Fe-Cr-Ni-C-N-O framework (Note that by default, the unprompted `O` species on its sublattice `1` and `VA` on its sublattice `2` is assigned as the rest):

	`SPECIFY SUBLATTICE (0 FOR ALL) /0/: <RETURN>`
	`FRACTIONS IN SUBLATTICE 1`
	`CR /1/: .1`
	`FE /1/: .8`
	`NI /1/: .0995`
	`FRACTIONS IN SUBLATTICE 2`
	`C /1/: .05`
	`N /1/: .05`

Syntax	TABULATE_SUBSTANCE
	`PRESSURE /100000/: <PRESSURE OF INTEREST, IN PA>` Specify the constant pressure condition in Pa.
	`LOW TEMPERATURE LIMIT /298.15/: <T-LOW, IN K>` Specify the starting temperature K.
	`HIGH TEMPERATURE LIMIT /298.15/: <T-HIGH, IN K>` Specify the ending temperature K.
	`STEP IN TEMPERATURE /100/: <T-STEP> SPECIFY THE TEMPERATURE STEP FOR THE TABULATION.`
	`OUTPUT FILE /SCREEN/: <RETURN FOR SCREEN, OR TYPE A FILE NAME>` If you type `Return for screen` a list of basic thermodynamic functions for the chosen substance or for the specified pure or solution phase with the fixed composition is shown up as a table (which is demonstrated in the example outputs given below), and the command is terminated. If typing a file name, the table is both displayed on screen and saved as an *.EXP or a *.TAB file, and the program further prompts:
	`GRAPHICAL OUTPUT /Y/: <Y OR N>` • For `N` a table is output to the screen, and the same table is saved as a basic text file with the default extension .TAB under the current working directory. In this case, the graph cannot be plotted. • For `Y` a table is created with all the thermodynamic functions as normal (which is displayed on screen) as well as generates a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which is plotted on screen and saved as an *.EXP file), and it also ask which column is to be plotted on the resulting graph:
	`PLOT COLUMN ? /2/: <1 OR 2 OR 3 OR 4 OR 5 OR 6>` Specify which property (as column number) to be plotted as the Y-axis (versus temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (i.e. the plotted column) for the graph is written into an *.EXP file using the DATAPLOT format. The default column 2 is heat capacity, 3 enthalpy, 4 entropy and 5 Gibbs energy, and the additional column 6 is a user-entered function. The table displays on the screen. The plot then displays on the screen, followed by a POST: prompt. The POST module (postprocessor) automatically opens, and all types of the POST-module commands to refine the plotted diagram may be used. Possibilities include scaling the X/Y-

Syntax	TABULATE_SUBSTANCE
	axis, changing the X/Y-axis texts, etc. The command BACK or EXIT at the POST: prompt always takes you back to the TAB module.
	Also see the *DATPLOT User Guide* included with this documentation set.

Example Output 1

For the `FCC` alloy phase as a non-ideal solution with a fixed composition

```
[Fe0.80,Cr0.10,Ni0.0995,O0.005][C0.05,N0.05,VA0.90]
```

The following table is obtained by typing `Return for screen` at the prompt `Output file /SCREEN/`:

```
O U T P U T   F R O M   T H E R M O - C A L C

Phase : FCC                           Pressure :       100000.00

Specie: CR1/--2

***************************************************************

    T      Cp             H             S             G

   (K)    (Joule/K)      (Joule)      (Joule/K)      (Joule)

***************************************************************

298.15   2.70517E+01   6.23824E+03   4.40241E+01   -6.88755E+03

300.00   2.70889E+01   6.28832E+03   4.41916E+01   -6.96915E+03

400.00   2.87304E+01   9.08420E+03   5.22235E+01   -1.18052E+04

500.00   2.99904E+01   1.20222E+04   5.87742E+01   -1.73649E+04

600.00   3.10889E+01   1.50770E+04   6.43408E+01   -2.35275E+04

700.00   3.21116E+01   1.82375E+04   6.92106E+01   -3.02100E+04

800.00   3.30994E+01   2.14982E+04   7.35633E+01   -3.73524E+04

900.00   3.40742E+01   2.48569E+04   7.75182E+01   -4.49094E+04

1000.00   3.50483E+01   2.83130E+04   8.11586E+01   -5.28456E+04

1100.00   3.60268E+01   3.18667E+04   8.45449E+01   -6.11327E+04

1200.00   3.70143E+01   3.55187E+04   8.77219E+01   -6.97476E+04

1300.00   3.80149E+01   3.92700E+04   9.07241E+01   -7.86713E+04

1400.00   3.90311E+01   4.31222E+04   9.35784E+01   -8.78875E+04

1500.00   4.00649E+01   4.70768E+04   9.63064E+01   -9.73827E+04

1600.00   4.11174E+01   5.11358E+04   9.89256E+01   -1.07145E+05

1700.00   4.21896E+01   5.53010E+04   1.01450E+02   -1.17165E+05

1800.00   4.33555E+01   5.95774E+04   1.03894E+02   -1.27433E+05
```

```
1900.00    4.58528E+01    6.40379E+04    1.06306E+02   -1.37943E+05
2000.00    4.75402E+01    6.87138E+04    1.08704E+02   -1.48694E+05
```

Example Output 2

For a phase, as either an end-member or real solution, the listed species name is irrelevant.

For the H2 species as a pure substance (using the SSUB5 database), the following table is obtained by typing `Return for screen` at the prompt `Output file /SCREEN/`, whilst the figure is generated by typing 5 at the prompt Plot column /2/:

```
           O U T P U T F R O M T H E R M O - C A L C
  Phase : GAS                             Pressure :      100000.00
Specie: H2
****************************************************************    T       Cp
H              S              G
   (K)     (Joule/K)      (Joule)      (Joule/K)      (Joule)
****************************************************************  298.15    2.88369E+01
3.17684E-06    1.30680E+02   -3.89622E+04
300.00    2.88473E+01    5.33580E+01    1.30858E+02   -3.92042E+04
400.00    2.91591E+01    2.95686E+03    1.39209E+02   -5.27268E+04
500.00    2.92650E+01    5.87874E+03    1.45729E+02   -6.69856E+04
600.00    2.93441E+01    8.80908E+03    1.51071E+02   -8.18336E+04
700.00    2.94579E+01    1.17488E+04    1.55602E+02   -9.71730E+04
800.00    2.96320E+01    1.47027E+04    1.59547E+02   -1.12935E+05
900.00    2.98786E+01    1.76776E+04    1.63050E+02   -1.29068E+05
1000.00    3.02043E+01    2.06810E+04    1.66214E+02   -1.45533E+05
1100.00    3.05319E+01    2.37171E+04    1.69108E+02   -1.62302E+05
1200.00    3.09281E+01    2.67897E+04    1.71781E+02   -1.79348E+05
1300.00    3.13615E+01    2.99040E+04    1.74274E+02   -1.96652E+05
1400.00    3.18115E+01    3.30625E+04    1.76614E+02   -2.14197E+05
1500.00    3.22641E+01    3.62664E+04    1.78824E+02   -2.31970E+05
1600.00    3.27094E+01    3.95151E+04    1.80921E+02   -2.49958E+05
1700.00    3.31406E+01    4.28078E+04    1.82917E+02   -2.68151E+05
1800.00    3.35526E+01    4.61426E+04    1.84823E+02   -2.86539E+05
1900.00    3.39415E+01    4.95175E+04    1.86648E+02   -3.05113E+05
2000.00    3.43045E+01    5.29300E+04    1.88398E+02   -3.23866E+05
```

TERNARY_DIAGRAM Commands

In this section:

TERNARY_DIAGRAM Module

The TERNARY_DIAGRAM module (short name, the TERN module) enables you to quickly calculate a simple ternary phase diagram.

To enter the module, at the SYS prompt type GOTO_MODULE TERNARY. There are no other commands for this module. Follow the prompts to plot a diagram.

Also see the Graphical Mode equivalent to this command, *Ternary Calculator* on page 106 as described in the *Thermo-Calc User Guide*.

Syntax	TERNARY_DIAGRAM
Prompts	DATABASE: /FEDEMO/
	FIRST ELEMENT: SECOND ELEMENT THIRD ELEMENT
	PHASE DIAGRAM, MONOVARIANTS, OR LIQUIDUS SURFACE: /PHASE_ DIAGRAM/
	TEMPERATURE (C) /1000/
	GLOBAL MINIMIZATION ON: /Y/

Thermodynamic Variables and Units

The following topics are about the different thermodynamic variables recognised by Thermo-Calc. The topics have information about how to define new derived variables and functions and how to express thermodynamic variables in non-default units. The POLY module only recognises default units, which are the SI units.

See example 44 Thermo-Calc Console Mode Examples to learn about using variables and functions.

In this section:

Thermodynamic Variables

There are different thermodynamic variables in Thermo-Calc. Some are *state variables* such as temperature, pressure and mole fraction. These characterize the equilibrium state of a system. You set them when you define your system in the POLY module before performing a calculation. They are also used in other modules such as DATA, GIBBS and PARROT. State variables that involve components can be used for the defined components, but not for any species. (To define new components in a defined system, use *DEFINE_COMPONENTS* on page 143).

Other useful thermodynamic variables are derive from state variables with a mathematical function. Some derived variables are pre-defined by Thermo-Calc, such as normalized energetic and compositional extensive state variables for example. You use these by appending various normalizing suffixes to abbreviations that are associated with the state variables.

Thermodynamic variables can also be divided into extensive variables and intensive variables. An extensive variable is a variables whose value depends on the size of the system, whereas an intensive variable is a variable whose value is independent of system size.

More information about thermodynamic variables is available in the online help in Thermo-Calc Console Mode. Use the command *INFORMATION* on page 5, with a subject keyword such as one of the following as argument: STATE VARIABLES, INTENSIVE VARIABLES, EXTENSIVE VARIABLES, DERIVED VARIABLES, CONDITIONS (for condition settings), AXIS-VARIABLES (for stepping/mapping variable settings).

Common Thermodynamic Variables

The tables below list common thermodynamic variables. Most of these can be used to define equilibrium conditions in the POLY module unless otherwise indicated.

The variables are divided into the following tables:

- Intensive variables
- Energy-related extensive variables for whole system or for a phase
- Compositional extensive variables (overall amount/size of components in whole system, or amount of a component in system or in a phase)
- Constitutional composition-related extensive variables (amount of a constituent/species on a sublattice site in a phase)
- Special quantities

How to Read the Tables

- The *Abbrev*. column shows the abbreviation that you use in the POLY module for referring to the variable. If the variable must be given any arguments, then these are given in parenthesis directly after the abbreviation. For most variables, this is also the abbreviation that you use to refer to the variable in other modules. However, some variables are referred to with a different abbreviation in the POST module. If this is the case, then this is noted at the bottom of the table.

- Variables that are specific to a species, in which case the species is specified as an argument (*sp* in the table below). If there are two or more sublattices in a given phase, then this argument is given as `constituent#sublattice`. This indicates a constituent on a specific sublattice or site in the phase (referred to by the # sign and a digit). For example, `y(BCC,C#2)` stands for the site fraction of the C species on the second sublattice site of the BCC_A2 solution phase.
- The *Unit* column show which unit that you can use to express the variable in the POLY module. This unit is always the SI unit. In other modules, state variables may be expressed in other units.
- The *Suffix* column shows the suffixes you can append to the variables. The R-suffix can be used with all compositional extensive state variables, but the suffix does not change the value of the variable. This is indicated by putting the `R` in parenthesis.

These types of variables are listed in the following topics:

- *Intensive Variables* below
- *Extensive Variables* on the next page
- *Special Quantities* on page 275
- *The u-Fraction Variable* on page 276
- *Suffixes* on page 276

Intensive Variables

V	Abbrev.	Unit	Descript.	Domain	Suffix
T	T[1]	K	Temperature	System	
P	P	Pa	Pressure	System	
μ	MU(comp)	J/mol	Chemical potential	Component	R
	MU(sp,ph)[2]			Species relative to a solution phase	R

V	Abbrev.	Unit	Descript.	Domain	Suffix
a	AC(comp)	N/A	Activity	Component	R
	AC(sp,ph)[2]			Species relative to a solution phase	R
	LNAC (comp)[3]		ln(Activity)	Component	R
	LNAC (sp,ph) [2]			Species relative to a solution phase	R

[1] When plotting in POST, besides T (Kelvin), you can also use T_C or T_F to plot temperature in °C or °F.

[2] Only for single-substitutional-lattice solution phases such as AQUEOUS solution and GASeous mixture phases and for interacting species on the substitutional sublattice of two-sublattice solution phases (such as the phases BCC_A2 and M6C phases).

[3] In natural logarithm (lnAC=MU/RT)

Extensive Variables

Energy-Related Extensive Variables for Whole System or for a Phase

See example 44 in the *Console Mode Examples Guide*for an example of using variables and functions.

V	Abbrev.	Unit	Descript.	Domain	Suffix
V	V	m3	Volume	System	M, W, V, R
	V(ph)[1]			Phase	M, W, F, R
	VP(ph) [2]			Phase	M, W, V, R
G	G	J	Gibbs energy	System	M, W, V, R
	G(ph) [1]			Phase	M, W, V, F, R
A	A	J	Helmholtz energy	System	M, W, V, R
	A(ph) [1]			Phase	M, W, V, F, R

V	Abbrev.	Unit	Descript.	Domain	Suffix
U	U	J	Internal energy	System	M, W, V, R
	U(ph) [1]			Phase	M, W, V, F, R
H	H	J	Enthalpy	System	M, W, V, R
	H(ph)			Phase	M, W, V, F
S	S	J/K	Entropy	System	M, W, V, R
	S(ph) [1]			Phase	M, W, V, F, R
Cp	HM.T[3]	J/mol/K	Heat capacity at constant pressure	System	R
	HM(ph).T[3]			Phase	R
Cv	HM.T[4]	J/mol/K	Heat capacity at constant volume	System	R
	HM(ph).T[4]			Phase	R
D	DG(ph) [5]	N/A	Driving force (thermodynamic factor)	Phase	Always use with M, W, V or F. R can also be used.

[1] For this variable, if the phase is unstable, then the variable gets a value of zero.

[2] Do not use VP(ph) as a condition in POLY. Instead, use CHANGE_STATUS PHASE <PHASE>=FIX <AMOUNT> where the fixed <AMOUNT> is roughly equal to VPF(ph), which cannot be directly evaluated or listed. If the phase is unstable, then VP(ph) and its normalized quantities get zero value.

[3] Use only if pressure has been set as a condition.

[4] Use only if volume has been set as a condition.

[5] Already divided by RT.

Compositional Extensive Variables

These are for the overall amount/size of components in whole system, or amount of a component in system or in a phase.

For NP(ph) or BP(ph), these cannot be used in the POLY module. Instead use CHANGE_STATUS PHASE <PHASE>=FIX <AMOUNT> where the fixed <AMOUNT> is roughly equal to NPF(ph) or BPF(ph), which cannot be directly evaluated or listed. If the phase is unstable, then NP(ph) or BP(ph) (and its normalized quantities) get zero value.

When plotting the following quantities in POST, refer to each as indicated:

- X(comp)>`Mole-Fraction <comp>` (or `M_F`)
- W(comp)>`Weight-Fraction <comp>` (or `W_F`)
- X%(comp)>`Mole-Percent <comp>` (or `M_P`)
- W%(comp)>`Weight-Percent <comp>` (or `W_P`).
- u-f(ph,comp)>`U-F`

V	Abbrev.	Unit	Descript.	Domain	Suffix	Comment
n	N	mole	Moles	All components	M, W, V, (R)	
	N(comp)			Component	M, W, V, (R)	
	N(ph,comp)			Component in a phase	M, W, V, (R)	
	NP(ph)[1]			Phase	M, W, V, (R)	See above
b	B	gram	Mass	All components	M, W, V, (R)	
	B(comp)			Component	M, W, V, (R)	
	B(ph,comp)			Component in a phase	M, W, V, (R)	
	BP(ph) [1]			Phase	M, W, V, (R)	See above
x	X(comp)	N/A	Mole fraction	Component	(R)	See above
	X(ph,comp)			Component in a phase	(R)	
w	W(comp)	N/A	Mass (weight) fraction	Component	(R)	See above
	W(ph,comp)			Component in a phase	(R)	
x%	X%(comp)[2]	N/A	Mole percent	Component	(R)	See above
w%	W%(comp)[2]	N/A	Mass (weight) percent	Component	(R)	See above
u	u-f (ph,comp)[1]	N/A	u-fraction	Component in a stable phase	(R)	See above

V	Abbrev.	Unit	Descript.	Domain	Suffix	Comment
in	IN(sp)[2]	mole	Input mole number	Phase species in the system	(R)	See above
im	IM(sp)[2]	gram	Input mass unit	Phase species in the system	(R)	See above

[1] Do not use in the POLY module.

[2] Only available in TQ-Interface and TC-API.

Constitutional Composition-Related Extensive Variable

This is the amount of a constituent/species on a sublattice site in a phase.

V	Abbrev.	Unit	Descript.	Domain
y	Y (ph,cons#sub)	N/A	Site fraction	Constituent on a sublattice site (denoted by # and a digit) in a phase

Special Quantities

V	Abbrev.	Unit	Descript.	Domain	Suffix
Q	QF(ph)[1]	N/A	Phase stability function	Phase	R Under a condition and system definition, QFR (ph) = QF(ph)
Tc	TC(ph) [2]	K	Curie temperature	Phase	R Under a condition and system definition, TCR (ph) = TC(ph)
Mb	BMAG(ph)[3]	N/A	Bohr magneton number	Phase	R Under a condition and system definition, BMAGR (ph) = BMAG(ph)

[1] Negative when phase composition is inside a spinodal, otherwise positive. Can be used to find out if an equilibrium is within the miscibility gap for a solution phase. Cannot be used as a condition.

[2] Calculated for phases with magnetic contributions in an equilibrium state. Cannot be used as a condition, but can be plotted in POST.

V	Abbrev.	Unit	Descript.	Domain	Suffix
[3] Calculated for phases with magnetic contributions in an equilibrium state.					

The u-Fraction Variable

The u-fraction is an important quantity in some single-point or stepping paraequilibrium calculations in the POLY module and in the DICTRA module. You can apply the variable in a paraequilibrium calculation to the substitutional matrix component, the substitutional alloying components or the interstitial component. The u-fraction is denoted as ui for the i-th system component *n* of a whole system or of a specific phase in various equilibrium states (such as full-equilibria, partial-equilibria, para-equilibria, local-equilibria).

Generally, the u-fraction is defined as:

$$\frac{x_i}{\sum_{j \in S} x_j}$$

The sum of *x*j in the denominator is calculated for the substitutional matrix element and all substitutional alloying element in the whole system or in a specific phase. The variable xi in the numerator stands for the mole-fraction of the i-th component in the whole system or in the specific phase.

Suffixes

Suffixes may be appended to all extensive variables and to some intensive variables. Some suffixes can be used to enter the value of variables in normalized form. These are referred to as normalizing suffixes. If the variable that you create with a normalizing suffix is based on a state variable that can be set as a condition in POLY, then the normalized variable can also be set as a condition in POLY.

The normalizing suffixes are M, W, V and F. There is also a reference state suffix `R`, which you can use if you want the value of a thermodynamic variable to be calculated with respect to a reference state that have previously set.

Normalizing Suffixes

When variables that express system and phase quantities are normalized, the following general rules are used:

- System quantities are normalized by the total system size (in terms of N, B or V).
- Phase quantities are normalized by the phase amount [in terms of NP(ph), BP(ph) or VP (ph)].

The normalized quantities of G(ph), A(ph), U(ph), H(ph), S(ph) and V(ph) are calculated according to the thermodynamic model used for the phase (e.g. GM(ph), AM(ph), UM(ph), HM(ph), SM(ph) and VM(ph)).

These quantities are calculated using the first derivatives of the Gibbs energy expressed for the phase with respect to the current composition of the system.

The tables describe the normalizing suffixes for the different state variables.

G, A, U, H, S and V of a Whole System

Suffix	Description	Example(s)
M (per mole)	First derivative of the variable with regard to the total system size in terms of N.	GM is the Gibbs energy per mole of the system (J/mol). GM = ∂G/∂N
W (per mass in gram)	First derivative of the variable with regard to the total system size in terms of B.	GW is the Gibbs energy per mass of the system (J/g). GW = ∂G/∂B
V (per volume in m^3)	First derivative of the variable with regard to the total system size in terms of V. VV does not have to be evaluated.	GV is the Gibbs energy per volume of the system (J/m3). GV = ∂G/∂V

G(ph), A(ph), U(ph), H(ph), S(ph) and V(ph) of a Phase

Suffix	Description	Example(s)
M (per mole)	First derivative of the variable with regard to the phase amount in terms of NP(ph).	GM is Gibbs energy of the phase per mole of the phase (J/mol). GM(ph) = ∂G(ph)/∂NP(ph)
W (per mass in gram)	First derivative of the variable with regard to the phase amount in terms of BP(ph).	GW is Gibbs energy of the phase per mass of the phase (J/mol). GW(ph) = ∂G(ph)/ ∂BP(ph)
V (per volume in m^3)	First derivative of the variable with regard to the phase amount in terms of VP(ph). VV(ph) does not have to be evaluated.	GV is Gibbs energy of the phase per volume of the phase (J/mol). GV(ph) = ∂G(ph)/∂VP(ph)
F (per mole formula unit)	First derivative of the variable with regard to the phase amount in terms of NP(ph) and NA (NA is the total atomic number in the phase formula)	GF(ph) is the Gibbs energy of the phase per formula unit of the phase (J/mol). GF(ph) = ∂G(ph)/∂NP(ph)*NA

DG(ph) of a Phase

Suffix	Description	Example(s)
M (per mole)	Theoretically, the first derivative of the variable with regard to the phase amount in terms of NP(ph). Since DG (ph) is not directly calculated, the second derivative of the Gibbs energy expressed for the phase in question with respect to the current compositions in the equilibrium state of the system is calculated instead. DGM(ph) cannot not be set as a condition since it is only calculated under a certain type of equilibrium state.	DGM(ph) is driving force for precipitation of the phase per mole of components. DGM(ph) = ∂DG(ph)/∂NP(ph)
W (per mass in gram)	Theoretically, the first derivative of the variable with regard to the phase amount in terms of BP(ph). Since DG (ph) is not directly calculated, the second derivative of the Gibbs energy expressed for the phase in question with respect to the current compositions in the equilibrium state of the system is calculated instead. DGW(ph) cannot not be set as a condition since it is only calculated under a certain type of equilibrium state.	DGW(ph) is driving force for precipitation of the phase per mass of components. DGW(ph) = ∂DG(ph)/∂BP(ph)
V (per volume in m^3)	Theoretically, the first derivative of the variable with regard to the phase amount in terms of VP(ph). Since DG (ph) is not directly calculated, the second derivative of the Gibbs energy expressed for the phase in question with respect to the current compositions in the equilibrium state of the system is calculated instead. DGV(ph) cannot not be set as a condition since it is only calculated under a certain type of equilibrium state.	DGV(ph) is driving force for precipitation of the phase per volume of components. DGV(ph) = ∂DG(ph)/∂VP(ph)
F (per mole formula unit)	Theoretically, the first derivative of the variable with regard to the phase amount in terms of NP(ph) and NA (NA is the total atomic number in the phase formula). Since DG(ph) is not directly calculated, the second derivative of the Gibbs energy expressed for the phase in question with respect to the current compositions in the equilibrium state of the system is calculated instead. DGF(ph) cannot not be set as a condition since it is only calculated under a certain type of equilibrium state.	DGF(ph) is driving force for precipitation of the phase per formula unit of components. DGF(ph) = ∂DG(ph)/∂NP (ph)*NA

N and B of a System

Suffix	Description	Example(s)
M (per mole)	First derivative of the variable with regard to the total system size in terms of N. NM does not have to be evaluated. BM cannot be set as a condition.	BM is mass (gram) of components per mole of the system (g/mol). BM = ∂B/∂N
W (per mass in gram)	First derivative of the variable with regard to the total system size in terms of B. BW does not have to be evaluated. NW cannot be set as a condition.	NW is mole number of components per mass of the system (mol/g). NW = ∂N/∂B
V (per volume in m^3)	First derivative of the variable with regard to the total system size in terms of V. BV is the density of the entire system.	NV is mole number of components per volume of the system (mol/m3). NV = ∂N/∂V

N (comp) and B(comp) of a Component in the System

Suffix	Description	Example(s)
M (per mole)	First derivative of the variable with regard to the total system size in terms of N. BM(c) cannot be set as a condition.	NM(comp is mole number of a component per mole of the system (mole fraction, that is, X (comp)). NM(comp) = ∂N(comp)/ ∂N
W (per mass in gram)	First derivative of the variable with regard to the total system size in terms of B. NW(c) cannot be set as a condition.	BW(comp) is mass (gram) of a component per mass of the system (mass fraction, that is, W (comp)). BW(comp) = ∂B(comp)/∂B
V (per volume in m^3)	First derivative of the variable with regard to the total system size in terms of V.	NV(comp) is mole number of a component per volume of the system (mol/m3). NV(comp) = ∂N(comp)/ ∂V

NP(ph), BP(ph) and VP(ph) of a Phase in the System

Suffix	Description	Example(s)
M (per mole)	BPM(ph) and VPM(ph): First derivative of the variable with regard to the phase amount in terms of NP(ph). NPM(ph): First derivative of the variable with regard to the total system size in terms of N.	BPM(ph) is mass (gram) of a phase per mole of the phase in the system (g/mol). BPM(ph) = ∂BP(ph)/∂NP(ph) NPM(ph) is mole number of a phase per mole of the system (mole fraction). NPM(ph) = ∂NP(ph)/∂N
W (per mass in gram)	NPW(ph) and VPW(ph): First derivative of the variable with regard to the phase amount in terms of BP(ph). BPW(ph): First derivative of the variable with regard to the total system size in terms of B.	VPW(ph) is volume (m3) of a phase per mass of the phase in the system (m3/g) VPW(ph) = ∂VP(ph)/∂BP(ph) BPW(ph) is mass (gram) of a phase per mass of the system (mass fraction) BPW(ph) = ∂BP(ph)/∂B
V (per volume in m^3)	NPV(ph) and BPV(ph): First derivative of the variable with regard to the phase amount in terms of VP(ph). VPV(ph): First derivative of the variable with regard to the total system size in terms of V.	NPV(ph) is mole number of a phase per volume of the phase in the system (mol/m3). NPV(ph) = ∂NP(ph)/∂VP(ph) VPV(ph) is volume (m3) of a phase per volume of the system (volume fraction). VPV(ph) = ∂VP(ph)/∂V

N(ph,comp) and B(ph,comp) of a Component in a Phase

Suffix	Description	Example(s)
M (per mole)	First derivative of the variable with regard to the phase amount in terms of NP(ph) of the phase.	NM(ph,comp)is mole number of a component per mole of a phase (mole fraction of a component in a phase, that is, X(ph, compt)). NM(ph,comp) = ∂N(ph,comp)/∂NP(ph)
W (per mass in gram)	First derivative of the variable with regard to the phase amount in terms of BP(ph) of the phase.	BW(ph,comp) is mass (gram) of a component per mass of a phase (mass fraction of a component in a phase, that is, W(ph,comp)). BW(ph,comp) = ∂B(ph,comp)/∂BP(ph)

Suffix	Description	Example(s)
V (per volume in m^3)	First derivative of variable with regard to the phase amount in terms of VP(ph) of the phase.	NV(ph,comp) is mole number of a component per volume of a phase (mol/m3). NV(ph,comp) = ∂N(ph,comp)/∂VP(ph)

The Reference State Suffix R

You can use the reference state suffix R for some thermodynamic variables to calculate their value with respect to a reference state that you have previously set for a system component with the *SET_ REFERENCE_STATE* on page 169 command in POLY (or in a response-driven modules such as the POURBAIX module). The value of energy-related variables that are used with the R suffix depends on the reference states of all the components in the defined system.

It is possible to use an R suffix on all compositional extensive state variables as well, but the value of the state variable is always the same, with or without the suffix.

If the reference state for a system component is the default reference state (the stable reference state (SER) which is defined in a Thermo-Calc database), then MUR(comp)= MU(comp), ACR(comp)= AC(comp) and LNACR(comp)= LNAC(comp).

In the case of some thermodynamic variables, you can also use the R suffix to express chemical potentials and activities of species relative to some single-substitutional-lattice solution phases (such as aqueous solution, gaseous mixture, metallic liquid solution, slag mixture or MO solid solution). These state variables are MU(sp,ph), MUR(sp,ph), AC(sp,ph), ACR(sp,ph), LNAC(sp,ph) and LNACR(sp,ph).

The reference states and standard states of various solution species are pre-defined for some solution phases in some databases. For all solution species in any solution model in any database, it is always the case that MUR(sp,ph)= MU(sp,ph), ACR(sp,ph)= AC(sp,ph) and LNACR(sp,ph)= LNAC(sp,ph).

Derived Variables and Partial Derivatives

Many derived variables can be obtained easily by using partial derivatives of state variables, such as heat capacity, thermal expansivity, isothermal compressibility, among others.

Derivatives of state variables can be defined or evaluated by putting a dot (.) between two state variables. The variable after the dot must be a condition that you have set when you defined your system. For example, the heat capacity (at constant pressure or volume) of a system is the partial derivative of the state variable enthalpy with respect to temperature. To evaluate this derivative, use the designation HM.T, where temperature is a state variable that you have set already.

Any partial derivative of a state variable with respect to another can be defined or evaluated as a derived variable by using the dot (.) notation.

Derived Variables Dot Notation Examples

The following table shows some examples:

Derived variable	Description
HM.T = ∂HM/∂T	Heat capacity for the system at either constant P if pressure is a condition or constant V if volume is a condition.
HM(ph).T = ∂HM(ph)/ ∂T	Heat capacity for a phase at either constant P if pressure is a condition or constant V if volume is a condition.
H.T = ∂H/∂T	Heat capacity for the system multiplied by total mole number of components, i.e., ∂H/∂T = Cp*N if pressure is a condition in a closed system or ∂H/∂T = Cv*N if volume is a condition in a closed system.
H(ph).T = ∂H(ph)/ ∂T	Heat capacity for a phase multiplied by the sum of NP(ph) and HM(ph)*∂NP(ph)/∂T, i.e., Cp(ph)*NP(ph)+HM(ph)*∂NP(ph)/∂T if pressure is a condition or Cv(ph)*NP(ph)+HM(ph)*∂NP (ph)/ ∂T if volume is a condition.
VM.T = ∂VM/∂T	Thermal expansivity of the system (already multiplied by the total molar volume), i.e., ∂VM/∂T = α*VM.
VM(ph).T = ∂VM(ph)/∂T	Thermal expansivity of a phase (already multiplied by its molar volume), i.e., ∂VM(ph)/ ∂T = α(ph)*VM (ph).
V.T = ∂V/∂T	Thermal expansivity of the system (already multiplied by the total volume), i.e., ∂V/∂T = α*V.
V(ph).T = ∂V(ph)/∂T	Thermal expansivity of a phase (already multiplied by the phase volume) plus the VM(ph)*∂NP(ph)/∂T term, i.e., ∂V (ph)/∂T = α(ph)*V(ph)= α(ph)*VM (ph)*NP(ph)+VM (ph)*∂NP (ph)/∂T.
VM.P = -∂VM/∂P	Isothermal compressibility of the system (already multiplied by the total molar volume), i.e., ∂VM/∂P = -κ*VM.
VM(ph).P = -∂VM(ph)/ ∂P	Isothermal compressibility of a phase (already multiplied by its molar volume), i.e., ∂VM(ph)/∂P = -κ(ph)*VM (ph).
V.P = -∂V/∂P	Isothermal compressibility of the system (already multiplied by the total volume), i.e., ∂V/∂P = -κ*V.
V(ph).P = -∂V(ph)/∂P	Isothermal compressibility of a phase (already multiplied by the phase volume) plus the VM(ph)*∂NP(ph)/∂P term, i.e., ∂V (ph)/∂P = -κ(ph)*V(ph) = -κ (ph)*VM (ph)*NP(ph)+VM (ph)*∂NP (ph)/∂P.
T.X(comp) = ∂T/∂X(comp)	Slope of a phase boundary on a T-X(comp) phase diagram with respect to mole fraction of the component in the system.

Derived variable	Description
T.W(comp) = ∂T/∂W(comp)	Slope of a phase boundary on a T-W(comp) phase diagram with respect to mass of the component in the system.
T.X(ph,comp) = ∂T/∂X(ph,comp)	Slope of a phase boundary on a T-X(ph,comp) phase diagram with respect to mole fraction of the component in the phase.
T.W(ph,comp) = ∂T/∂W(ph,comp)	Slope of a phase boundary on a T-W(ph,comp) phase diagram with respect to mole fraction of the component in the phase.
P.T = ∂P/∂T	Slope of a phase boundary on a P-T phase diagram (Note that the equilibrium with phase assemblage must be calculated first).

Defining New Derived Variables and Functions

You can define additional derived variables or functions based on a state variable by using *ENTER_ SYMBOL* on page 195. Any derived variable or function must have a unique name that starts with a letter and has a maximum number of eight characters. These characters can include both uppercase and lowercase letters, numerical digits and the underscore (_). No other characters are allowed.

The difference between new derived variables introduced with the ENTER_SYMBOL command and new functions introduced with the same command is as follows. Whenever Thermo-Calc calculates the value of a function (as part of an equilibrium calculation for example), all functions related to the defined system is evaluated. An entered derived variable, on the other hand, is only evaluated when it is defined and when it is re-evaluated with *EVALUATE_FUNCTIONS* on page 155.

Derived Variables for Aqueous Solutions

Standard state variables used for other phases can be directly applied to the aqueous solution phase. In addition, there are pre-defined derived variables for the aqueous solution phase in particular.

Examples of predefined derived variables for the aqueous solution phase are listed and briefly described in the following tables. Some derived variables for a gaseous mixture phase are also shown. This is because EOS (Equation of State) expressions, standard thermodynamic properties and transport properties of the pure solvent H2O are also expressions and properties of the pure gaseous species H2O.

For Aqueous Solution Phase

Name	Abbrev.	Units	Description	Comments
pH	PH	N/A	Acidity	of the aqueous solution phase pH = -log10(AC(H+)) = -log10(ACR(H+,AQ))
Eh	EH	V, mV	Hypothetical electric potential	of the aqueous solution phase Eh = u(ZE)/96485.309
Pe	PE	N/A	Logarithm of the hypothetical electron activity	of the aqueous solution phase pe = u(ZE)/(2.3025851*RT)
Ah	AH	kJ, kcal	Thermodynamic affinity per electron	of a redox couple wrt the standard hydrogen electrode in the aqueous solution phase Ah = u(ZE)
yw	YH2O	N/A	Mole fraction	of the solvent H2O in the aqueous solution phase YH2O = Y(AQ,H2O)
Nw	AH2O	mole	Mole number	of 1.0 kg of solvent H2O AH2O = 55.508435
Nsp	NSH2O	mole	Mole number, NS(AQ,H2O)	of the solvent (H2O) in the aqueous solution phase NS(AQ,H2O) = YH2O*NP (AQ)
	NS#		Mole number, NS(AQ,sp)	of a solute species in the aqueous solution phase NS(AQ,sp) = Y(AQ,sp)*NP (AQ)
M	ML#	mol/kg_ H2O	Molality, ML(AQ,sp)	of a solute species in the aqueous phase ML(AQ,sp) = Y (AQ,sp)*AH2O/YH2O

Name	Abbrev.	Units	Description	Comments
m*	TIM	equivalent molality	Total molality	of all solute species in the aqueous solution phase TIM = sum[ML(AQ,sp)]ions + sum[ML (AQ,sp)]complexes
mt	TIC#	equivalent molality	Total ionic concentration, TIC (AQ,sp)	of a cation I in the aqueous phase TIC(AQ,spI) = sum[ML (AQ,spJ)*V(spI-in-spJ)]
I	ISTR	N/A	Ionic strength	of the aqueous solution phase ISTR=1/2*sum[ML (AQ,sp)*Z(AQ,sp)**2]
γ	RCH2O	N/A	Activity coefficient, RC(H2O,AQ)	of the solvent (H2O) RC(H2O,AQ) = ACR (H2O,AQ)/YH2O
	RC#		Activity coefficient, RC(sp,AQ)	of a solute species RC(sp,AQ) = ACR(sp,AQ)/Y (AQ,sp)*YH2O
αi	AIH2O	N/A	Activity, AI(H2O,AQ)	of the solvent (H2O), AI(H2O,AQ) = ACR (H2O,AQ)
	AI#		Activity, AI(sp,AQ)	of a solute species related to the aqueous solution phase, AI(sp,AQ) = ACR (sp,AQ)*AH2O
	LogAI#		Log10 Activity, LogAI (H2O,AQ), LogAI(sp,AQ)	of the solvent or a solute species related to the aqueous solution phase in common logarithm, LogAI(H2O,AQ) = log10 [AI(H2O,AQ)]

Name	Abbrev.	Units	Description	Comments
				LogAI(sp,AQ) = log10[AI (sp,AQ)]
αw	AW	N/A	Activity	of H2O in the aqueous solution phase AW = ACR(H2O,AQ)
ϕ	OS	N/A	Osmotic coefficient	of aqueous solution phase OS = - 55.508435*lnAW/TIM
At1	AT1	equil_ mol/kg_ H2O	Titration alkalinity (definition 1)	of the aqueous solution phase Generally defined as the equivalent molality of carbonate and bicarbonate at the methyl orange endpoint (pH=4.5).
At2	AT2	equil_ mol/kg_ H2O	Titration alkalinity (definition 2)	of the aqueous solution phase Generally defined as the equivalent molality of carbonate and bicarbonate, plus sulfide, at the methyl orange endpoint (pH=4.5).

For Gaseous Mixture Phase

Name	Abbrev.	Units	Description	Comments
γ	RA#	N/A	Activity coefficient, RA(sp,GAS)	of a gaseous species in the gaseous mixture RA(sp,GAS) = function(Y,T,P)
γ*	RF#	N/A	Fugacity coefficient, RF(sp,GAS)	of a pure gaseous species under TP RF(sp,GAS) = function(T,P/V)
F	FUG#	pa, bar, psi	Fugacity, FUG(sp,GAS)	of a gaseous species in the gaseous mixture FUG(sp,GAS) = RA(sp,GAS)*RF(sp,GAS)*Y (GAS,sp)*P

Name	Abbrev.	Units	Description	Comments
ft	TFUG	pa, bar, psi	Total gas fugacity	of the gaseous mixture phase TFUG = sum[FUG(sp,GAS)]

Defining New Derived Variables for Aqueous Solutions

You can define many different additional derived variables and functions. For instance, the partition coefficient of a component between two phases is defined as follows:

```
Pc<name> = X(phase1,component)/X(phase2,component)
```

Another example is the activity coefficient of a component in the system. It is defined as follows:

```
Rc<name> = ACR(component)/X(component)
```

The activity coefficient of a species in a solution phase depends on which model definitions on the reference states that are used and it depends on the standard states for the species. If the standard state of a species is defined to be the same as the pure species in a substitutional phase, then the activity coefficient of the species can be calculated as:

```
RC<name> = ACR(species,phase)/Y(phase,species)
```

In some cases, there are more than one sublattice in the phase and the reference states for the phase species cannot be presented by the pure species, as in the case of C in FCC [(Fe)1(C,N,VA)1], where the model may end at fifty percent C and fifty percent Fe. In those cases, the chemical potential and activity of all phase constituents is not well defined and you cannot properly define the activity coefficients of the species in the phase.

For an aqueous solution phase, no matter what model is used, the reference state and standard state for both the solvent and solutes are defined in a special way in the Thermo-Calc software. The reference state for the solvent is set as pure water, the same as its standard state (according to Raoults's Law). The standard state for a solute species is set the hypothetical solute species, whilst its reference state is defined as the hypothetical state at unit molality (one mole dissolved into 1.0 kg of solvent) but in which the environment of each molecular is the same as at infinite dilution (according to Henry's Law). Under these definitions, the activity coefficients for solvent species can be calculated as follows:

```
RC<H2O> = ACR(H2O,aqueous)/ Y(aqueous,H2O)
```

The coefficients for the solute species can be calculated as follows:

```
RC<spec> = AI(species,aqueous)/ML(aqueous,species)
= ACR(species,aqueous)/ Y(aqueous,species)*Y(aqueous,H2O)
```

AI is here the activity of solute species calculated from the model, and ML is the molality of the species.

Thermodynamic Variables and Wildcard Characters

The ACR (species,phase) and the MUR (species,phase) quantities can only be appropriately and meaningfully applied to substitutional solution phases (such as GAS, AQUEOUS, LIQUID) and to solution phases without an interstitial sublattice set. This means that the wild card characters `*` and `$` normally don't work properly for the activity and/or chemical potential properties.

For example, do not use ACR(*,*) or MUR(*,*) for all the species relative to all phases, nor ACR(*,phase) or MUR(*,phase) for all the species relative to a specific phase, nor ACR(species,*) or MUR(species,*) for a specific species relative to all phases.

Instead, use specific names for species and (applicable) phase, as in the following examples:

```
show_value acr(fe+2,aqueous)Lists activity of the Fe+2 species in AQUEOUS solution
phase
```

```
show_value mur(c2,gas)Lists chemical potential of the C2 species in GAS mixture phase
```

```
set_axis_var y ac(o2,gas)Sets activity of the O2 species in GAS mixture phase as Y-
axis
```

Units of State Variables and Derived Variables

The units for the state variables in the Units column in the table in *User-Specified Units* below are the SI units that are used in the POLY module. However, in the POST module, you can both use pre-defined symbols and define your own symbols which are based on state variables and derived variables. By doing this, you can express the value of the thermodynamic variables in non-default units. The following topic briefly describes how you can create your own symbols and suggests expressions that you can use to convert default units into other standard units which are not default units in Thermo-Calc.

The topic *Examples of User-Defined Units* on the next page describes some variables of specific species in a phase and the units in which they are expressed. These variables are not used in POLY or POST, but in the GIBBS, DATA, PARROT and ED_EXP modules.

User-Specified Units

You do not always need to use the default units for variables. Using *ENTER_SYMBOL* on page 195, you can convert the value of variable from the default unit to your preferred unit. (Some response-driven modules also allow you to set conditions in non-default units.) User-defined symbols in the form of functions can be used for plotting in the POST module, but they cannot be used as conditions in POLY. User defined symbols in the form of variables can be used as conditions.

For example, if you want pressure to be plotted in bar rather than Pa, then you define the following function `PB`:

```
ENTER_SYMBOL FUNCTION PB = P/1E5 ;
```

If you wanted heat capacity of a system to be shown as in terms of J/mol/K, then you can define the following function Cp:

```
ENTER_SYMBOL FUNCTION CP = HM.T ;
```

If the variable you want to convert is a function of one or more derivatives, then you must enter the symbol as a variable rather as a function. For instance, a symbol expressing heat capacity of a system in a unit of cal/mol/K can be defined as the following variable `Cp2`:

```
ENTER_SYMBOL VARIABLE CP2 = HM.T/4.1858 ;
```

Also see *Examples of User-Defined Units* below for information about entering conventional functions and the associated units.

Examples of User-Defined Units

The following tables give examples of expressions you can use to enter symbols. This enables you to show the values of thermodynamic variables in a variety of units.

Units of State Variables and Derived Variables on the previous page and *User-Specified Units* on the previous page

You can also use thermodynamic variables with normalizing suffixes in the expression when you enter a symbol. However, the tables do not have examples of these expressions.

In the expressions these are defined as follows:

- `i` is for an auxiliary index or letter that corresponds to the component `c`
- `j` is for an auxiliary index or letter that corresponds to the phase `ph`, and
- `ij` stands for an auxiliary index or letter that corresponds to the component `c` in the phase `ph`.

Intensive and Extensive Variables of a Defined System

Quantity	Suggested name and expression	Unit
For intensive variables of a defined system		
Temperature	TempC = T-273.15	Celsius (C)
	TempF = 1.8*T-459.67	Fahrenheit (F)
	TempF = 1.8*TempC+32	
Pressure	PB = P/1E5	bars (bar)
	PKb = P/1E8	kilobars (kbar)
	PAtm = P/101325	atmosphere (atm)
	PSI = P/6894.76	pounds/sq. inch (psi)
	PIHg = P/3342.11	inches of Hg
	PTor = P/133.322	Tors (millimeters of Hg)

Quantity	Suggested name and expression	Unit
For extensive variables of a defined system		
Amount	BKg = 1E-3*B	kilograms (kg)
	BTon = 1E-6*B	tons
Volume	VDM = 1E-3*V	cubic decimeters (dm^3, l)
	VCM = 1E-6*V	cubic centimeters (cm^3)
	VMM = 1E-9*V	cubic millimeters (mm^3)
	VCI = 1.6387064E-5*V	cubic inches (in^3)
	VLi = 1E-3*V	liters (l)
Energy	GCal = G/4.1858	cal
	ACal = A/4.1858	cal
	UCal = U/4.1858	cal
	HCal = H/4.1858	cal
	SCal = S/4.1858	Cal/K
Heat Capacity	Cp2 = HM.T/4.1858	cal/mol/K
	Cp3 = HW.T/4.1858	cal/g/K
	Cp4 = HV.T/4.1858	cal/m^3/K
Thermal Expansion	A2 = VM.T*1E-6*VM	cm^3/mol/K
	A3 = VW.T*1E-6*VM	cm^3/g/K
Isothermal Compressibility	B2 = -VM.P*1E-9*VM	mm^3/mol/Pa
	B3 = -VW.P*1E-9*VM	mm^3/g/Pa

Intensive and Extensive Variables of a System Component

Quantity	Suggested name and expression	Unit
For intensive variables of a system component		
Chemical Potential	MUi = MU(c)/4.1858	cal/mol
	MURi = MUR(c)/4.1858	cal/mol

Quantity	Suggested name and expression	Unit
Special quantity for aqueous solution	pH = -LOG10(ACR(H+1))	dimensionless
	Eh = MUR(ZE)/RNF where RNF=96485.309	volts (V)
For extensive variables of a system component		
Amount	MFi = N(c)/N = X(c)	dimensionless
	MPi = N(c)/N*100	dimensionless
	WFi = B(c)/B = W(c)	dimensionless
	WPi = B(c)/B*100	dimensionless
	BKgi = 1E-3*B(c)	kilograms (kg)
	BToni = 1E-6*B(c)	tons

Extensive Variables of a Phase

Quantity	Suggested name and expression	Unit
Amount	MFj = NP(ph)/N	dimensionless
	MPj = NP(ph)/N*100	dimensionless
	WFj = BP(ph)/B	dimensionless
	WPj = BP(ph)/B*100	dimensionless
	BKgj = 1E-3*BP(ph)	kilograms (kg)
	BTonj = 1E-6*BP(ph)	tons
Volume	VCMj = 1E-3*V(ph)	cubic decimeters (dm^3, l)
	VCMj = 1E-6*V(ph)	cubic centimeters (cm^3)
	VMMj = 1E-9*V(ph)	cubic millimeters (mm^3)
Energy	GCalj = G(ph)/4.1858	cal
	ACalj = A(ph)/4.1858	cal
	UCalj = U(ph)/4.1858	cal
	HCalj = H(ph)/4.1858	cal
	SCalj = S(ph)/4.1858	Cal/K

Quantity	Suggested name and expression	Unit
Driving Force	DG2j = DGM(ph)/4.1858	cal/mol
	DG3j = DGW(ph)/4.1858	cal/g
	DG4j = DGV(ph)/4.1858	Cal/m^3
	DG5j = DGF(ph)/4.1858	cal/mole_formula_unit
Heat Capacity	Cp2j = HM(ph).T/4.1858	cal/mol/K
	Cp3j = HW(ph).T/4.1858	cal/g/K
	Cp4j = HV(ph)/.T/4.1858	cal/m^3/K
	Cp5j = HF(ph)/.T/4.1858	cal/mole_formula_unit/K
Thermal Expansion	A2j = VM(ph).T*1E-6*VM	cm^3/mol/K
	A3j = VW(ph).T*1E-6*VM	cm^3/g/K
	A4j = VF(ph).T*1E-6*VM	cm^3/mole_formula_unit/K
Isothermal Compressibility	B2j = -VM(ph).P*1E-9*VM	mm^3/mol/Pa
	B3j = -VW(ph).P*1E-9*VM	mm^3/g/Pa
	B4j = -VF(ph).P*1E-9*VM	mm^3/mole_formula_unit/Pa

Extensive Variables of a Component in a Phase

Quantity	Suggested name and expression	Unit
Amount	MFij = N(ph,c)/NP(ph) = X(ph,c)	dimensionless
	MPij = N(ph,c)/NP(ph)*100	dimensionless
	WFi = B(ph,c)/BP(ph) = W(ph,c)	dimensionless
	WPij = B(ph,c)/BP(ph)*100	dimensionless
	BKgij = 1E-3*B(ph,c)	kilograms (kg)
	BTonij = 1E-6*B(ph,c)	tons

Intensive and Extensive Variables of a Species in a Phase

Quantity	Suggested name and expression	Unit
For intensive variables of a species in a phase		
Chemical Potential	MUCalij = MU(sp,ph)/4.1858	cal/mol
	MUErgij = MU(sp,ph)*1E7	cal/mol
Fugacity	FAij = AC(sp,ph)*PATM	atmosphere (atm)
	FBij = AC(sp,ph)*PB	bars (bar)
	FCij = AC(sp,ph)*PKB	kilobars (kbar)
	FTij = AC(sp,ph)*PTOR	Tors (millimeters of Hg)
ln(Fugacity)	LnFAij = LNAC(sp,ph)+LN(PATM)	atmosphere (atm)
	LnFBij = LNAC(sp,ph)+LN(PB)	bars (bar)
	LnFCij = LNAC(sp,ph)+LN(PKB)	kilobars (kbar)
	LnFTij = LNAC(sp,ph)+LN(PTOR)	Tors (millimeters of Hg)
Special quantity for aqueous solution	AIij=ACR(FE+3,AQ)*AH2O	dimensionless
	RCij=ACR(FE+3,AQ)*YH2O/Y(AQ,FE+3) where AH2O=55.508435 and YH2O=Y(AQ,H2O)	dimensionless
	WRCalij = WR(AQ,sp)/4.1858 (WR(AQ,sp) is only valid for aqueous species in a system in which the AQUEOUS solution phase is considered)	cal/mol
For extensive variables of a species in a phase		
Special quantity for aqueous solution	MLij=Y(AQ,FE+3)*AH2O/YH2O	molality (m)
	ISTR=1*IS1+1*IS2+1*IS3 where AH2O=55.508435, YH2O=Y(AQ,H2O) and ISn=.5*MLi*Zi**2+.5*MLj*Zj**2+.5*MLk*Zk**2	molality (m)

Units for Variables of Specific Species in a Phase

In the GIBBS, DATA, PARROT and ED_EXP modules, there are quantities that describe various thermodynamic, physical, chemical and transport properties for a compound phase, and for a certain species or a given species combination in a specific solution phase. The following table describes the units for those variables.

These variables cannot be directly used in the POLY or POST modules.

Variables	Description and unit
G(ph,sp) G(ph,sp;0) G(ph,sp1;sp2;...;0)	Gibbs energy (J/mol) of a pure substance or end-member.
G(ph,sp1,sp2,...) L(ph,sp1,sp2,...) G(ph,sp1,sp2,...;0) L(ph,sp1,sp2,...;0) G (ph,sp1,sp2,...;...;0) L (ph,sp1,sp2,...;...;0)	Zero-order interaction parameter (J/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
G(ph,sp1,sp2,...;i) L(ph,sp1,sp2,...;i) G (ph,sp1,sp2,...;...;i) L (ph,sp1,sp2,...;...;i)	The ith -order interaction parameter (J/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
TC(ph,sp) TC(ph,sp;0) TC(ph,sp1;sp2;...;0)	Curie temperature (K) of a pure substance or end-member.
TC(ph,sp1,sp2,...) TC(ph,sp1,sp2,...;0) TC (ph,sp1,sp2,...;...;0)	Zero-order Curie-temperature contribution (K) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
TC(ph,sp1,sp2,...;i) TC (ph,sp1,sp2,...;...;i)	The ith -order Curie-temperature contribution (K) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.

Variables	Description and unit
BMAGN(ph,sp) BMAGN(ph,sp;0) BMAGN (ph,sp1;sp2;...;0)	Bohr magneton number (dimensionless) of a pure substance or end-member.
BMAGN (ph,sp1,sp2,...) BMAGN (ph,sp1,sp2,...;0) BMAGN (ph,...;sp1,sp2,...;0)	Zero-order Bohr magneton number (dimensionless) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
BMAGN (ph,sp1,sp2,...;i) BMAGN (ph,sp1,sp2,...;...;i)	The ith -order Bohr magneton number (dimensionless) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
V0(ph,sp) V0(ph,sp;0) V0(ph,sp1;sp2;...;0)	Molar volume (m^3) at 1 bar and reference T0 of a pure substance or end-member.
V0(ph,sp1,sp2,...) V0(ph,sp1,sp2,...;0) V0 (ph,sp1,sp2,...;...;0)	Zero-order composition-dependent molar volume (m^3) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
V0(ph,sp1,sp2,...;i) V0 (ph,sp1,sp2,...;...;i)	The ith-order composition-dependent molar volume (m^3) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VA(ph,sp) VA(ph,sp;0) VA(ph,sp1;sp2;...;0)	Integrated thermal expansivity (m^3/mol/K) of a pure substance or end-member.
VA(ph,sp1,sp2,...) VA(ph,sp1,sp2,...;0) VA (ph,sp1,sp2,...;...;0)	Zero-order composition-dependent thermal expansivity (m^3/mol/K) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.

Variables	Description and unit
VA(ph,sp1,sp2,...;i) VA (ph,sp1,sp2,...;...;i)	The ith -order composition-dependent thermal expansivity (m^3/mol/K) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VC(ph,sp) VB(ph,sp;0) VB(ph,sp1;sp2;...;0)	Isothermal compressibility (m^3/mol/Pa) of a pure substance or end-member.
VC(ph,sp1,sp2,...) VB(ph,sp1,sp2,...;0) VB (ph,sp1,sp2,...;...;0)	Zero-order composition-dependent isothermal compressibility (m^3/mol/Pa) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VC(ph,sp1,sp2,...;i) VB (ph,sp1,sp2,...;...;i)	The ith -order composition-dependent isothermal compressibility (m^3/mol/Pa) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VK(ph,sp) VB(ph,sp;0) VB(ph,sp1;sp2;...;0)	High-pressure fitting parameter (m^3/mol) of a pure substance or end-member.
VK(ph,sp1,sp2,...) VB(ph,sp1,sp2,...;0) VB (ph,sp1,sp2,...;...;0)	Zero-order composition-dependent high-pressure fitting parameter (m^3/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VK(ph,sp1,sp2,...;i) VB (ph,sp1,sp2,...;...;i)	The ith -order composition-dependent high-pressure fitting parameter (m^3/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
WR(ph,sp) WR(ph,sp;0)	Standard Born function (J/mol) of a specific aqueous species in the AQUEOUS solution phase (always with a single sublattice) under the reference-state temperature and pressure.

Data Optimization User Guide

Thermo-Calc Version 2016a

Introduction

In this section:

Introduction to Data Optimization

This user guide describes the basic idea behind data optimization, contains brief descriptions of the modules and file types that you work with while doing your optimization, as well as rough outline of overarching workflow that you typically follow when working with optimization.

This guide assumes that you are familiar with using the Thermo-Calc Console Mode. It also assumes that you understand the principles of thermodynamics and are familiar with the theoretical foundations of computational thermodynamics.

Also see the *Thermo-Calc Console Mode Command Reference* included with this documentation set, which has information about all the commands available in the Thermo-Calc Console Mode, including PARROT and ED_EXP. Note that the commands listed under PARROT and ED_EXP are only commands that are specific to PARROT and/or ED_EXP. These modules also contain some commands that are also available in POLY.

Example 36 in the *Console Mode Examples Guide*. It is a detailed example of assessment and optimization in Thermo-Calc.

Data Optimization

In a thermodynamic database, each phase in a system is characterised using a mathematical model of its Gibbs energy. The Gibbs energy of a phase depends on various state variables. It can be defined individually even in a heterogeneous system with many stable phases since the properties of one phase are completely independent of the properties of the other phases in the system. In most alloy systems, the thermodynamic properties of a phase can be modelled by expressing how the Gibbs energy depends on temperature and composition. State variables such as pressure, volume or entropy can also serve as parameters in the Gibbs energy expression. The Gibbs energy of the whole system is the sum of the products of the Gibbs energy of each phase multiplied by the amount of that phase.

How the Gibbs energy of the phase varies with various state variables is determined by a mathematical model and how various adjustable parameters of the model is set. By optimizing these parameters, you can calculate the thermodynamic properties of a system under various conditions (by calculating phase diagrams for example). The thermodynamic properties themselves are functions of temperature, pressure or composition. They include, among other properties, standard enthalpy of formation, entropy, heat capacity, molar volume, thermal expansivity, compressibility and Curie temperature for magnetic transformations.

Data optimization is about adjusting the model parameters so that calculated equilibria fit well with experimental data. An important part of a data optimization is therefore collecting and assessing available experimental and theoretical information about phase equilibria and thermochemical properties of a system.

When you optimize the parameters of the model of a phase, you put what is called optimizing variables into the Gibbs energy expressions of those parameters. During the optimization, the values of these variables are varied in order to find a fit between the calculated equilibria and the experimental data that you base the optimization on. Once you have settled on a Gibbs energy expression for a certain parameter of a phase of the element, then you must stick to this expression in all future optimizations involving the element. If you do not do this, then you must re-optimize all previous systems that involve this element when the expression is changed.

The Thermo-Calc software allows you to optimize parameters that characterize not only binary systems, but also ternary system and systems of even higher orders. You can even optimize parameters that characterise systems of different orders at the same time.

The Least-Squares Method

Data optimization in Thermo-Calc is based on the least-squares method for fitting values calculated on the basis of a model with observed quantities. The software is accordingly trying to find the optimizing variable values that lead the minimized sum of the squares of the differences between the calculated values and the observed quantities (that is, of the errors or residuals).

The least-squares method works best under the following conditions:

- The observed quantities have a Gaussian probability distribution.
- The observed quantities are only subject to random errors.
- The different observations (experiments) are uncorrelated
- The standard deviation of each observation can be estimated.
- The number of observations is large.
- The models used give precise predictions.

Of course, these conditions are usually not all met in a normal thermodynamic assessment. But even in non-ideal conditions, there is no known method that works better than the least-squares method.

The CALPHAD Approach

The data optimization functionality in Thermo-Calc works according to the CALPHAD approach to computational thermodynamics. This approach builds on the development of models that accurately represent thermodynamic properties for various phases. These models enable you to predict the thermodynamic properties of multicomponent systems-based data concerning binary and ternary subsystems. The predictions can take many factors into accounts, such as, for example, crystallography, type of bonding, order-disorder transitions and magnetic properties.

CALPHAD is originally an abbreviation of *CALculation of PHAse Diagrams*, but as the approach has expanded in scope, it now stands for *computer coupling of phase diagrams and thermochemistry*.

References

For in-depth information about CALPHAD as well as data optimization, see *Computational*

Thermodynamics – The Calphad Method by Hans Leo Lukas, Suzana G. Fries and Bo Sundman (Cambridge University Press, 2007). CALPHAD was given its first general description in *Computer Calculations of Phase Diagrams* by Larry Kaufman and Harold Bernstein (Academic Press, 1970). The book contains some case studies of optimizations using Thermo-Calc.

Information about the concept of 'lattice stability' (which is important for the development of multicomponent thermodynamic databases) can also be found in the paper *Hume-Rothery and Calphad Thermodynamics* by Larry Kaufman, published in *Calphad and Alloy Thermodynamics*, edited by Patrice E.A. Turchi, Antonios Gonis, Robert Shull (The Minerals, Metals & Materials Society, 2002), pp. 3-19.

Working with Data Optimization in Thermo-Calc

In this section:

Data Optimization in Thermo-Calc

The actual performance of the optimization of a system is primarily done in the PARROT module, and in its sub-module ED_EXP. The POLY and POST modules are used to calculate and plot property or phase diagrams based on the optimized variables. This allows you to visually assess how good a fit you have achieved between your calculated results and your experimental data.

The end result of a successful optimization is typically an updated Thermo-Calc database file (with filename extension *.TDB) or a new database file.

Data Optimization Workflow

When you carry out an optimization using Thermo-Calc, you typically follow a workflow as outlined below.

1. Collect experimental data about your system from various sources such as journal articles and reports.
2. Create a POP-file in which you enter the experimental data you have collected.
3. Create a setup macro file in which you define your system and the variables you want to optimize. A useful general procedure is to first find a minimum set of variables that allows you to calculate most of the experiments.
4. Run the setup macro file.
5. Compile your POP-file. This file often contains errors, which need to be corrected, and the file recompiled several times before the compilation is completed without errors.
6. In the ED_EXP module, check that the experiments in your POP-file reach equilibrium with reasonable results. If any experiments do not reach equilibrium with reasonable results, then try changing the starting values on equilibrium conditions (such as composition) to see whether the equilibrium can be computed after all. If this does not work, then you can temporarily exclude the experiment from the optimization. At a later stage in the optimization, when the optimizing variables have different values, you can check whether the experiments can be computed with reasonable results.
7. Back in the PARROT module, run the first optimization cycle and evaluate the feedback in the console. You can also plot diagrams in the POLY module that allow you to visually inspect the fit between calculated optimization results and the experimental data. If you are not satisfied with the fit, then enter the ED_EXP module again and adjust the weights of the experiments in a way that is likely to improve the fit and run another optimization cycle. Reiterate until you are satisfied with the fit.
8. Once you get the optimization stable and smooth with the minimum set of variables chosen in step 3 above, try using different sets of variables to see whether improvements are possible.

9. When you are satisfied with the fit between calculated results and experimental data with your final set of optimizing variables, update your setup file and POP-file. In the setup file, enter the calculated optimizing variable values as the variables' start values. In the POP-file, enter the final weights of the experiments. With these files updated, you can easily regenerate the parameter values of your optimized system. Having the files updated also makes it easier to optimize the system again in light of new data or new theoretical models.
10. Finally, update the database with your calculated results or create a new database with information about the system that you have optimized.

The PARROT Module

The PARROT module consists of a comprehensive subroutine package for data evaluation of thermodynamic model parameters from experimental data. The module has 99 variables that you can use for optimization and the module can handle a thousand experimental measurements in an optimization. However, there are limits on the simultaneous numbers of variables and experiments. These limits are listed at each optimization.

The PARROT module is part of the Thermo-Calc software. The module is not included in the software development kits TQ-Interface, TC-API or TC-Toolbox for MATLAB®.

The experimental data is entered and edited in a sub-module of PARROT: *The EDIT_EXPERIMENTS Module* below.

The EDIT_EXPERIMENTS Module

The EDIT_EXPERIMENTS module, or ED_EXP for short, is used for entering, evaluating and editing experimental data. To enter the module, go to *The PARROT Module* above and then issue the command *EDIT_EXPERIMENTS* on page 95.

Since ED_EXP uses the POLY module for doing equilibrium calculations of experimental data points, many POLY commands are also available in the ED_EXP module. However, the commands may work somewhat differently in the two modules. There are also unique ED_EXP commands which are not found in POLY.

When you execute ED_EXP commands, the PARROT workspace is modified. However, note that you must always first use *READ_WORKSPACES* on page 160 in ED_EXP before doing anything in the module. The command loads the experimental data that you are going to work with during the optimization. Furthermore, before leaving ED_EXP you must use *SAVE_WORKSPACES* on page 30 or the results of your work in the ED_EXP module is lost.

Data Optimization in Other Modules

All kinds of thermodynamic data, calculated equilibrium states or dynamic parameters are transferred back and forth between the PARROT and the GIBBS module as well between these and the POLY module. Whenever an optimization run is performed, PARROT calls the GIBBS module for stored system definition data and model parameters.

In Thermo-Calc, the GIBBS module (the Gibbs Energy System) handles the models of the various phases that can form in a multicomponent system. It stores thermodynamic data and performs various Gibbs energy equilibrium calculations. It contains subroutines to analytically calculate the first and second partial derivatives of integral Gibbs energy with respect to any set of variables. Many thermodynamic models for various types of substances are implemented in the module. Some commands that are available in the PARROT module are also available in the GIBBS module.

In the GIBBS module, parameters of the Gibbs energy models are referred to as *TP-functions*.

PARROT calls the POLY module for equilibrium calculations. The POLY module is used for calculating and storing complex heterogeneous equilibria.

Optimization File Types

These are the important files you work with when optimizing.

File type	Description
POP-file	The POP-file (Parrot OPtimization file) is the file that contains all the experimental data that you use for optimizing Gibbs energy values for your system. The POP-file is a plain text file that contains ED_EXP commands. By default, the file has the filename extension *.POP in Thermo-Calc.
Setup file	The setup file is a Thermo-Calc macro file (*.TCM). This file typically contains commands that define your system, opens the PARROT workspace (and associated GIBBS and POLY workspaces) and sets optimizing variables.
Other macro files	It is often useful to have other macro files than the setup file at hand while you are doing your optimization. For example, it is convenient to have a macro file that automatically plots phase diagrams or other diagrams based on the latest values of your optimizing variables. An optimization process typically involves many optimization cycles, so this operation needs to be done many times during a typical optimization.
Workspace files	The PARROT module has its own workspace with dynamic memories, similar to the workspaces of POLY and GIBBS. It is stored and updated in a PARROT workspace file with the filename extension *.PAR. The workspace file is a binary file whose format is hardware dependent. This means that the format is unique for each type of CPU. Hence, a workspace file saved on a computer with one type of CPU cannot be used on a computer with another type of CPU.

File type	Description
EXP-file	The EXP-file is a plain text file with information in the form of DATAPLOT Graphical Language commands. These commands can specify some or all of the data points in your POP-file. This allows you to plot these data points on top of your plotted optimization results (with the POST commands *APPEND_EXPERIMENTAL_DATA* on page 192 or *QUICK_EXPERIMENTAL_PLOT* on page 201). You can then visually assess the fit between these results and the experimental data. The PARROT workspace is not influenced in any way by the creation or use of EXP-files. Also see the *DATPLOT User Guide* included with this documentation set.

Optimization Workspaces

When writing the setup file, you enter a *CREATE_NEW_STORE_FILE* on page 94 command which creates a PARROT workspace file. Alternatively, select an existing PARROT workspace file with *SET_STORE_FILE* on page 116. The workspace file is automatically updated and saved with the latest optimization results (after each use of the *OPTIMIZE_VARIABLES* on page 109 command). You can also explicitly instruct Thermo-Calc to save the workspace using *SAVE_PARROT_WORKSPACES* on page 111. If you want to get rid of your latest changes and return to the state of your workspace when it was last saved, use *READ_PARROT_WORKSPACES* on page 109.

When the PARROT workspace is saved, parts of the POLY and GIBBS workspaces are also saved to the PARROT workspace file, along with the PARROT workspace itself. Using the POLY command *SAVE_WORKSPACES* on page 161 creates a new POLY3 file, nothing is saved to the PARROT workspace file.

If you do any POLY calculations based on data in the current workspace file, then a *.POLY3 file is created, which contains a copy of the current set of parameters. If you continue on your optimization and read the old *.POLY3 file, then the new set of parameters is overwritten with the old set. It is therefore recommended that you never read a *.POLY3 or *.GES5 file while you are doing an optimization.

Method for Optimization and Simulation

The PARROT module typically uses ordinary POLY minimization for equilibrium calculations. The global minimization technique that is used in POLY cannot be used because it automatically creates new composition sets which corrupt the PARROT workspace. Furthermore, the equilibrium definitions that you use as input for your optimization must each be attributed to a specific phase. Some of these definitions may specify some local or partial equilibrium state rather than a global minimum. However, it is possible to use the global minimization technique for some equilibrium calculations by using the ED_EXP command *ADVANCED_OPTIONS* on page 125 with the `Toggle_Alternate` option.

Optimization in PARROT is performed on the basis of a ‘maximum likelihood’ principle. According to this principle, the best fit between various calculated results and all the input of experimental data is found where the sum of the square of the weighted residuals is at its minimum.

Typically, the results of an optimization is considered to be better the fewer optimizing variables that are needed to get the same level of fit between computed results and experimental data. If you can get almost the same fit using eight instead of twelve parameters, then this should be considered to be a significant improvement. When more parameters are used, the values of the individual parameters tend to become unrealistic. This is particularly true when it comes to temperature-dependent parameters. However, relatively small differences in the number of parameters are often insignificant. For example, it does not typically matter whether you used 24 or 25 parameters to reach a certain degree of fit.

It is often difficult to compare the quality of different optimizations based on the number of optimizing variables that have been optimized. Since you may rely more or less heavily on different pieces of available experimental data (and do so with good reason), it is possible that two different users could end up with very different numbers of parameters for the same system. For example, suppose you optimize the Fe-Al system to incorporate it into an Al database, while someone else optimizes the same system to incorporate it into an Fe database. In this situation, the two of you would probably make different judgements about which experimental information your calculated results must have a good fit with.

Creating the POP-File

Before starting an optimization in Thermo-Calc, you must collect relevant experimental data about your system that is available in the research literature. This data is then entered in the POP-file. The experimental data points and measurements are entered in this plain text file in the form of ED_EXP commands.

The syntax of the POP-file is in principle independent of the models used for describing the phases in the system.

In this section:

Experimental Information

An optimization of thermodynamic model parameters should be based on a range of reliable experimental results that is as wide as possible, as well as on empirical or semi-empirical correlations and theoretical considerations. During an optimization in Thermo-Calc, many kinds of thermochemical and thermophysical data can be mixed and used together.

The information needed for an optimization represent measurements of thermodynamic parameters in the system at equilibrium. These can be measurements of any of the thermodynamic quantities that can be set as conditions in Thermo-Calc. The information may represent measurements of activities or enthalpies in single-phase regions for example, or solubilities or transformation temperatures in a multi-phase region.

For each equilibrium that is used in the optimization, at least one quantity must have been measured and you also need to know the conditions that must be set to compute the system’s equilibrium state. For example, consider a binary system in a single-phase region. Suppose that you have measured the temperature, pressure, composition and the chemical potential. Three of these quantities are necessary to specify the equilibrium state and the forth can be used as experimental information to model the phase.

If you have a two-phase region and know the temperature and pressure conditions, as well as which the stable phases are, then you could use the compositions of one or both phases as experimental data. The former information about the conditions is sufficient to determine the equilibrium. The experimental information can then be used to optimize the parameters that are used to characterise and model the system.

For examples how to enter various kinds of experimental data in the POP-file, see *Experiments and Experimental Data* on page 50.

Conflicting or Missing Information

You are likely to sometimes have inconsistent or conflicting experimental information. In principle, all available experimental data should still be entered into the POP-file, unless you have good reasons to exclude some information. For example, if you have reason to believe that the samples used in some experiments were not pure, then these experiments could be excluded from the POP-file.

For some systems, you have very little experimental data to go on. When this is the case, do not try to fit the available data uncritically with high accuracy without considering possible errors. You can look for data in systems with similar elements to get an idea about what the reasonable estimates of data that is lacking could be. Or you can use calculations from first principles or semi-empirical methods to get reasonable estimates that can be used as experimental data.

Invariant Equilibria Information

The most valuable experimental information for an optimization is information about stable invariant equilibria. An invariant equilibrium is such that neither pressure, temperature nor composition can be changed without there being a change in the set of phases present in the system. It is recommended that all information about invariant equilibria for a system is included in the POP-file. These equilibria need not all be based on explicit experimental measurements of thermodynamic properties.

Reasonable estimates from available experimental data are often useful to include in the POP-file. Such estimates, which should be based on the available experimental data, helps Thermo-Calc find a good set of start values on the optimization variables. However, at the end of the assessment, these estimated equilibria should be excluded and you should complete the optimization based only on the experimental information that is based on measurements.

With some experience from phase diagram evaluation, it is possible to make reasonable estimates of metastable invariant equilibria. Such estimates helps reduce the number of phases that are assessed simultaneously during the optimization. For example, you may assume that a certain intermediate phase does not form in a system. Then extrapolate the liquidus curves below the stable three-phase equilibria and finally, estimate temperatures and compositions of metastable three-phase equilibria between two other phases and the liquid.

Another useful technique is to extrapolate a liquidus line from a peritectic equilibrium to estimate the congruent melting temperature of a compound. This estimate may be more useful than the information about the peritectic equilibrium itself as the equilibrium involves only two phases.

Equilibria in Higher-Order Systems Information

When you optimize a binary system, the available experimental information can often be described equally well by very different sets of parameters. Information about ternary and higher order systems that include the components of the binary system are then often useful for optimizing the binary system (the same is true of information about quaternary system with respect to optimizing ternary systems). Often, it is only when you extrapolate the optimized parameters to a higher-order system that you can judge which of the sets of model parameters that is best. Sometimes information from several ternary systems is required to make a reliable judgement about which set of parameters best describes a particular binary system.

Heat Capacities Information

If you want to optimize your system based on data about heat capacities, then the corresponding parameters should first be optimized separately before you perform the general optimization of your system. The parameters should be kept fixed during the general optimization. This is because it is normally not possible to optimize heat capacity data together with other data in a meaningful way. If you do this, then it is very likely that the parameters related to the heat capacity are badly determined.

POP-File Syntax

It is recommended that you specify an experimental equilibrium as close as possible to actual experimental conditions. Ideally, there is information about the set of stable phases, temperature, pressure and some or all compositions for the equilibrium.

In addition, it is recommended that you keep the POP-file well-organised and extensively commented. This is especially important if someone else reassesses the system when new experimental information is available.

If you have too many equilibria defined in your file, or if you define equilibria with different sets of components in the same file, then you must divide the POP-file in *blocks* (using the *FLUSH_BUFFER* on page 41 command). You know whether you have defined too many equilibria for one block when you compile the file. A too large number of equilibria leads to an error message when the file is compiled.

If you use the *DEFINE_COMPONENTS* on page 143 command, then it must be the first command in the POP-file (or if you have several blocks, DEFINE_COMPONENTS can also be put as the first command after FLUSH_BUFFER). This is because the command automatically reinitiates the current workspace, so the effect of any commands placed before it are not saved.

The last command in the POP-file should always be *SAVE_WORKSPACES* on page 30.

Legal Commands in a POP-File

The following commands can be used in a POP-file.

Also see the *Thermo-Calc Console Mode Command Reference Guide*for information about all these commands.

- *ADVANCED_OPTIONS* on page 125
- *CHANGE_STATUS* on page 135
- *COMMENT* on page 40
- *CREATE_NEW_EQUILIBRIUM* on page 143
- *DEFINE_COMPONENTS* on page 143
- *ENTER_SYMBOL* on page 195
- *EVALUATE_FUNCTIONS* on page 155
- *EXPERIMENT* on page 24
- *EXPORT* on page 26
- *FLUSH_BUFFER* on page 41
- *IMPORT* on page 27
- *LABEL_DATA* on page 28
- *SAVE_WORKSPACE* on page 232
- *SET_ALL_START_VALUES* on page 162
- *SET_ALTERNATE_CONDITION* on page 30
- *SET_CONDITION* on page 165
- *SET_NUMERICAL_LIMITS* on page 168

- *SET_REFERENCE_STATE* on page 169
- *SET_START_VALUE* on page 171
- *SET_WEIGHT* on page 33
- *TABLE_HEAD, TABLE_VALUES and TABLE_END* on page 39

Entering Experimental Information

A POP-file in Thermo-Calc (*.POP) mainly consists of descriptions of equilibria, each equilibrium describing a data point that is based on an experiment from the existing literature on your system. An equilibrium is specified using ED_EXP commands.

The following is an example of a specification of an equilibrium with a single stable phase in the Au-Cu system. Equilibria with a single stable phase are often from experiments where enthalpies of mixing or chemical potentials have been measured.

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQUID=FIX 1
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563
SET_REFERENCE_STATE AU LIQ * 1E5
SET_REFERENCE_STATE CU LIQ * 1E5
EXPERIMENT HMR=-1520:200
COMMENT Measurement by Topor and Kleppa, Met trans 1984
```

Creating a New Equilibrium

The first command in an equilibrium specification is always *CREATE_NEW_EQUILIBRIUM* on page 143. This command takes two integers as arguments. The first integer should specify a unique identifier that can be used later to refer to the equilibrium in question. The second integer is an initialisation code 0, 1 or 2:

- Code 1: This means that all components are entered but all phases are suspended. This code is appropriate in most cases.
- Code 0: This means that all components and phases in the equilibrium are suspended initially. This code is needed if you are using experimental data about systems of different orders simultaneously, such as data about both binary and ternary systems for example.
- Code 2: This means that all components and phases are initially entered.

Example

For example, the first command in the example above creates an equilibrium data point with the unique identifier 1. All components of the system are entered but all phases are suspended:

```
CREATE_NEW_EQUILIBRIUM 1 1
```

Setting Equilibrium Conditions and its Uncertainty

All the equilibrium-related commands that are placed in between two *CREATE_NEW_EQUILIBRIUM* on page 143 commands are interpreted as specifying the conditions of the equilibrium created by the first command. The values of these conditions should be based on the conditions specified in the experiment that you base the equilibrium definition on.

Example

In the example, the *CHANGE_STATUS* on page 59 command specifies that the equilibrium is a single-phase equilibrium with the liquid phase.

```
CHANGE_STATUS PHASE LIQUID=FIX 1
```

Furthermore, the temperature is 1379 K, the pressure is 1 bar (1E5 Pa) and the mole fraction of Au is 0.0563.

```
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563
```

Sometimes you have reason to think that some information about an equilibrium condition may not be accurate. If this is the case, then you specify the degree of uncertainty for a condition with a colon after the condition quantity, directly followed by either an absolute value or a percentage. For example, if you have reason to doubt the reliability of the Au mole fraction measurement, then you could write:

```
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563:10%
```

This indicates that you think the Au mole fraction is between 0.0507 and 0.0619. PARROT calculates the equilibrium twice, once for 0.0507 and once for 0.0619, and then computes the change of each experimental value between these two equilibria. This change is used to modify the uncertainty specified for the experimental value.

Also see *Entering the Experimental Value and its Uncertainty* on the next page.

The possibility of specifying uncertainties on conditions may be useful when you have experimental data about a ternary system. In a binary system, a tie-line is determined by the two phases, the temperature and the pressure. Measurements of the compositions of both phases can then be used as experimental data. To specify the tie-line in a ternary two-phase equilibrium one of the four compositions must also be set as a condition. If the measurement of this composition has the same uncertainty as the other compositions, then you can assign this same degree of uncertainty both to the composition selected as a condition and the compositions selected as experimental data. Alternatively, the sample (overall) composition could be used as a condition and all four phase compositions could be used as experiment values. These phases must have the status *entered* (not fixed) since the relative amount of each is unknown.

Setting Reference States

If you want to specify a non-default reference state for one or several of your system components, then you do this with the *SET_REFERENCE_STATE* on page 169 command. In the following example, the reference state for the liquid phase is set to current temperature (*) and a pressure of 1E5 Pa.

Example

```
SET_REFERENCE_STATE AU LIQ * 1E5
SET_REFERENCE_STATE CU LIQ * 1E5
```

Note that in order for these reference states to be taken into account when the measurement of the enthalpy per mole of the system is entered with experiment, the R-suffix must be used.

Entering the Experimental Value and its Uncertainty

The *EXPERIMENT* on page 24 command is used to specify the quantity that the calculated results should be fitted to. When specifying this quantity, you both specify the measured quantity itself and your estimation of the uncertainty of this value.

Example

In the example, the EXPERIMENT command specifies the enthalpy per mole of the system, where this value is specified with respect to non-default reference states that you have set in the equilibrium specification:

```
EXPERIMENT HMR=-1520:200
```

The value before the colon is the quantity of the variable. The value after the colon is an assessment of the uncertainty of this quantity. A higher value means a greater uncertainty. In the example, the uncertainty has been specified with an absolute value but it could also be specified in percent of the quantity:

```
EXPERIMENT HMR=-1520:13%
```

Several experiments can be specified after the experiment command. Besides assigning values to conditions, it is also possible to specify that a certain quantity is greater than or less than a certain value. For instance, the following command would specify that the activity of the C-component is less than 0.01 and the mass fraction of the BCC phase of Ag is greater than 0.05:

```
EXPERIMENT ACR(C)<0.01:0.001 W(BCC,AG)>0.05:10%
```

Note that an experimental quantity can typically also be treated as an equilibrium condition, and vice versa. Which measured quantities that you treat as conditions and which quantities you treat as experimental data should be based on the accuracy of the different measurements. In most cases, the quantity that is based on the experimental technique with the lowest accuracy should be used as the experimental value.

Entering Many Equilibria as a Table

When one thermodynamic variable (e.g. heat capacity) has been measured as a function of another (e.g. temperature), then experimental data is often presented as a table. By using the TABLE_HEAD, TABLE_VALUES and TABLE_END commands, you can enter such experimental information into the POP-file in the form of a table. These three commands should always be used together and must be entered in the right order. See *TABLE_HEAD, TABLE_VALUES and TABLE_END* on page 39.

The TABLE_HEAD command is followed by a number. This number identifies the table as a whole, but also uniquely identifies the first equilibrium that is included in the table. This numeric identifier is incremented by one for each additional equilibrium in the table. Each row in the table represents one equilibrium. Thus, the equilibrium defined by the first row in the table above (on the line immediately below TABLE_VALUE) is uniquely identified with the number 1, the equilibrium defined by the second row by number 2, and so on until the last equilibrium on the sixth row, which is identified by the number 6.

Since the unique identifiers of the equilibria in the table are automatically generated from the unique numerical identifier that you give to the table, the unique numerical identifier that is normally given to the *CREATE_NEW_EQUILIBRIUM* on page 143 command is replaced by two at signs (@@).

At least one of the equilibria conditions specified with the *SET_CONDITION* on page 165 command must be a condition whose value is set by the numbers in one of the table columns. To assign the numbers in a column to a condition, use the syntax SET_CONDITION `<condition>=@<column number>`. For instance, in the example above, the mole fraction of X(ALO3/2) is for each equilibrium given by the number in the first column of the table.

At least one of the measured variables that are entered with the experiment command must be a variable whose value is set by the numbers in a table column. To assign the numbers in a column to a measured variable, use the syntax EXPERIMENT `<variable>=@<column number>`. In the example above, the measured temperature for each equilibrium is given by the number in the second table column.

The actual table should be entered between TABLE_VALUES and TABLE_END. Separate the columns with tabs or any number of spaces. The columns do not have to contain numbers, but can also contain other types of values, such as phase names for example. If a column contains text with spaces or special characters, then the text must be surrounded by double quotes (`“ABC DEF&ghi”`, for example). The table may have columns that are not used, but each row must have exactly the same number of columns.

Example

The following block of commands exemplifies how a table can be entered into a POP-file:

```
TABLE_HEAD 1

CREATE_NEW_EQUILIBRIUM @@ 1

CHANGE_STATUS PHASE TETR=FIX 0

CHANGE_STATUS PHASE ION=ENT 1

SET_CONDITION P=101325 N=1 ACR(O2,GAS)=.21 X(ALO3/2)=@1

EXPERIMENT T=@2:5

TABLE_VALUE

0.46082476

0.57142380

0.66822276

0.7496 2176

0.50382426

0.63132249

TABLE_END
```

Setting Site Fraction Start Values

In some systems, you have to set non-default composition start values on some experiments in order to get them to converge properly during optimization. This may be important if your system has a miscibility gap for example, in which case several equilibria with different compositions exist at certain temperatures. In such cases, you have to make sure it is specified in the POP-file which side of the miscibility gap the equilibrium is on. The calculated equilibrium could otherwise end up on the wrong side, resulting in a large error.

In the POP-file, you can set the site fraction of a constituent on an equilibrium with *SET_START_VALUE* on page 171.

Example

For example, the following command sets the composition start value of VA constituent of the FCC_A1 phase to 0.99:

```
SET_START_VALUE Y(FCC_A1,VA#2)=0.99
```

Setting the mole fraction for a component in a phase is not as reliable as setting the site fraction.

Additional equilibria entered after the last SET_START_VALUE command are automatically given a similar start value if the following command appears previously in the POP-file:

```
SET_ALL_START_VALUES Y
```

If start values are not set automatically for phase constituents, then you may have to set the composition start value for each equilibrium (or table of equilibria) separately.

Commenting About an Equilibrium

The COMMENT command inserts a comment about the equilibrium that is saved to the PARROT workspace file when the POP-file is compiled. The comment must be entered on one line.

Comments can also be entered immediately after a dollar sign ($), but these are not saved to the PARROT workspace. Instead, these comments are ignored by Thermo-Calc when the POP-file is compiled.

Simultaneous Use of Binary and Ternary Experiments

You can use experimental information about binary and ternary systems (as well as systems of even higher order) in the same POP-file and optimization. To do this, you must inform the PARROT module that not all components of the ternary system (or system of even higher order) should be considered. This allows you, for example, to use experimental information about a binary equilibrium that exists within a ternary system.

Use *CHANGE_STATUS* on page 59 with the keyword component and which components that should be entered. You should also give the initialization code 0 (rather than 1) to the *CREATE_NEW_EQUILIBRIUM* on page 143 command. This code indicates that all components must be entered.

Example

For example, the following set of ED_EXP commands characterises a binary (A-B) three-phase equilibrium (FCC-BCC-LIQ) in a ternary system (A-B-C). It can be used in a POP-file that otherwise only contain experimental information about the ternary system (A-B-C):

```
CREATE_NEW_EQUILIBRIUM 1 0

CHANGE_STATUS COMPONENT A B = ENTERED
```

```
CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1
SET_CONDITION P=1E5
EXPERIMENT T=1177:10
COMMENT from A-B
```

Using Stability Conditions

Early in the optimization when parameter values are not so good it is often useful to set the driving force for precipitation of a phase (per mole of components) to make sure that it appears where it should. The DGM condition is very useful for making sure that phases appear where they should. It can be removed when the optimization becomes stable.

The DGM condition is also useful for suppressing phases that appear where they should not appear. Do this by setting the value and the uncertainty to something reasonable. For example, the driving force for precipitation of the BCC phase could be set as follows:

```
DGM(BCC)<-0.1:0.1
```

This produces an error even before BCC becomes stable and the weight of the experiment in question can be adjusted as needed. If you specify that the driving force for the BCC phase should be below 0 (DGM(BCC)<0:1E-4 for example), then the optimization converges extremely slowly (if at all). With such a sharp error condition, calculating the equilibrium is like finding the minimum point on a lawn that gently slopes towards a rock wall. The optimization keeps bouncing into the wall.

Another useful stability condition is the phase stability function, abbreviated QF. This can be used to specify that a phase is outside the miscibility gap for a solution phase. If QF(phase) is negative, then the phase is inside the miscibility gap; if QF(phase) is positive, then it is outside the miscibility gap.

Grouping Equilibria with Labels

If you have several equilibria describing experiments that you want to be able to treat collectively, you can give all of the equilibria one and the same label.

Use LABEL to give an equilibrium a label. The label must start with the letter A and can only be up to four characters long. For example, you can insert the following command in several equilibria specifications:

```
LABEL ALH
```

During the optimization, you could use this label to, for example, set all the equilibria that have it to the same weight in the ED_EXP module.

Dividing the POP-File into Several Blocks

For two different reasons, you may have to divide your POP-file into different blocks:

- PARROT uses a buffer for storing compiled experimental data. If there is not enough memory in this buffer for storing all the experiments from the POP-file, then the POP-file can be divided into several blocks. When PARROT encounters the end of a POP-file block, it saves the experimental data that has been compiled to PARROT workspace, clears out the memory buffer and then proceeds to read the next block in the POP-file. This goes on until the experimental data from all the blocks have been saved to the PARROT workspace.

- If you want to put equilibria with different sets of components in one and the same POP-file, then you must divide the file up into blocks. Each block should only contain equilibria that all have the same set of components.

To divide the equilibria in the POP-file into different blocks, enter the *FLUSH_BUFFER* on page 41 command between two equilibria. This command marks the beginning of new block. When the command is encountered during the compilation of the POP-file, the compilation of the current block is terminated, the equilibria saved to the PARROT workspace file and a new block is initiated. That the buffer is reinitiated means that all functions and constants that were entered in the previous block must be entered again in new block in order to be used.

When editing your experimental data in the ED_EXP module, you can select which block to load and edit using *READ_WORKSPACES* on page 160 `<block number>`. The block before the first instance of FLUSH_BUFFER is block number 1, the block after the first instance and before the second instance is block number 2, and so on.

The Setup File

In this section:

Creating the Setup File

The second step is to create the setup file. This is an ordinary Thermo-Calc macro file (*.TCM). Typically, your setup file should contain the following:

- A system definition.
- A section where you enter model parameters and optimizing variables.
- A GO PARROT command.
- A section where initial values are assigned to the optimizing variables for the first optimization cycle.
- A *CREATE_NEW_STORE_FILE* on page 94 `<filename.par>` which creates the PARROT workspace file onto which the results of your optimization are continuously saved and updated. By default this workspace file has the same filename as the setup file, but with the *.PAR filename extension.
- A *COMPILE_EXPERIMENTS* on page 92 `<filename.pop>` command. Since you often encounter syntax errors when you first try to compile a POP-file, it may be useful to execute this command from the console rather than as part of the setup macro file.
- Normally, you also have a *SET_INTERACTIVE* on page 8 at the end of the setup file. This command returns control over Thermo-Calc to the console.

It is possible to interactively execute all these commands one at a time in the console. It may be instructive to enter the commands that would normally go into the setup file interactively while using the on-line help while you are learning to use the PARROT module.

Defining the System

If you have an existing database with information about your system, then you can write the system definition part of your setup file in two different ways.

- If you have an unencrypted user database, then you can define your system by entering DATA module commands directly in the setup file. Use the DATA commands that you normally use to define a system when doing equilibrium calculations in Thermo-Calc Console Mode.
- Define your system by entering DATA module commands directly in the console, then generate a macro file with the GIBBS command *LIST_DATA* on page 77 `<name of setup file>.tcm p` . This command saves all the data that has been loaded in the GIBBS workspace as a macro file. This file can then be modified as required into an optimization setup file.

Creating Additional Composition Sets of a Phase

If you need several composition sets of a phase, then these should be created with the GIBBS command *AMEND_PHASE_DESCRIPTION* on page 52 `<phase name> composition_sets <composition set number>` in the setup file.

If you add composition sets of a phase after the POP-file has been compiled, the data structure in the workspace is corrupted and you have to recompile the POP-file.

Entering the Optimizing Variables

When you have retrieved the data about your system for the setup file, you must manually enter the model parameters that you want to optimize. This is done using the GIBBS command *ENTER_PARAMETER* on page 99, but when you specify the Gibbs energy expression for the parameter, you enter variables that can be given various values into the expression. It is the values of these variables (and thus, the Gibbs energy of the parameter) that you are optimizing.

How you in detail specify the optimizing variables in your set up file depends on which phase you are trying to optimize and what thermodynamic model that is used in the optimization.

Also see *Thermodynamic Models* on page 59 for a range of examples.

Entering Optimizing Variables in Parameter Specifications

There are ninety-nine predefined variables in the PARROT module which you can use in your optimization. These variables are referred to as `V1`, `V2`, `V3`, … `V98` and `V 99`. When the GIBBS workspace is initialised, all optimizing variables are set to a fixed numerical value of zero. When you do an optimization, there is always some variable for which you want to evaluate the best value, but you can assign fixed values to optimizing variables too. It is often useful to enter all the variables that you can conceivably want to optimize, and then you can interactively select and change which variable to focus at various stages of the optimization.

The setup file excerpts that follow show how optimizing variables are entered. However, before the optimizing variables are introduced in this example, the following command enters an ionic liquid solution phase (a phase for which you have some experimental information):

```
ENTER_PHASE IONIC_LIQ  Y,
AL+3,ZR+4 ; ALO2-1,O-2 ; N N
```

This *ENTER_PHASE* on page 69 command enters a phase called IONIC_LIQ into the GIBBS workspace. This phase is of the type ionic liquid solution, which is specified by type code `Y`. Such a phase is modelled by the *Ionic Two Sublattice Liquid Model*. The next two arguments specify the two sublattices, in this case the cations AL+3,ZR+4 and the anions ALO2-1,O-2. Components that interact in a sublattice are separated by a comma. The first `N` at the end means that no additional constituents are added to the phase. The second `N` means that a list of possible parameters for the phase should not be shown in the console.

There are model parameters that describe the phase which you do not want to optimize, in which case the Gibbs energy expression of the parameter does not contain any optimizing variables.

Example 1

For example, the following parameters of the IONIC_LIQ phase do not contain any optimizing variables:

```
ENTER_PARAMETER G(IONIC_LIQ,AL+3:ALO2-1;0) 298.15 +2*GAL2O3_L; 6000 N

ENTER_PARAMETER G(IONIC_LIQ,AL+3:O-2;0) 298.15 +GAL2O3_L+400000; 6000
N

ENTER_PARAMETER G(IONIC_LIQ,ZR+4:O-2;0) 298.15 +2*GZRO2L; 6000 N
```

These parameters define the Gibbs energy expressions for three constituents in the constituent array, namely AL+3:ALO2-1, AL+3:O-2 and ZR+4:O-2. The names that occur in these expressions—that is, GAL2O3_L and GZRO2L—are the names of complex functions of temperature that are entered in the GIBBS workspace with the *ENTER_SYMBOL* on page 72 command.

The numbers that frame the Gibbs energy expressions specify the temperature range within which the expression should be applied. In the above, the expressions are applied in a temperature range from 298.15 K to 6000 K.

Example 2

In this example, suppose that you want to optimize the model parameter for the ZR+4:ALO2-1 combination, then you could enter the variables V1 and V2 in the Gibbs energy expression for that parameter in the following way:

```
ENTER_PARAMETER G(IONIC_LIQ,ZR+4:ALO2-1;0) 298.15 +GZRO2L +2*GAL2O3_L
+V1+V2*T; 6000 N
```

You normally have different optimizing variables in different parameters, but in some cases you may want several parameters to vary together, in which case you can use the same variable in the Gibbs energy expressions of several parameters.

Entering Interaction Parameters

For some systems, you have to consider interaction parameters in order to take excess Gibbs energy into account and correctly model the phase. By default, the software uses a Redlich-Kister expression for the excess Gibbs energy. The coefficient is entered after the constituent array when you use the *ENTER_PARAMETER* on page 99 command. In the last section, all the parameters were entered with a coefficient of zero (`0`).

Example

To take account of the excess Gibbs energy, you might enter the following:

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:ALO2-1,O-2;0) 298.15  +V11+V12*T;
6000 N
```

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:ALO2-1,O-2;1) 298.15  +V13+V14*T;
6000 N
```

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:ALO2-1,O-2;2) 298.15  +V15+V16*T;
6000 N
```

Parameters with interaction coefficients are referred to as *interaction parameters*. The regular parameter has a coefficient of 0. The subregular parameter has a coefficient of 1. It is possible to interaction coefficients up to a value of 9, but you are advised not to use more than three coefficients (that is, with coefficient values of 0, 1 and 2).

Initializing Optimizing Variables in PARROT

Variables that have been entered into Gibbs energy expressions for model parameters in the setup file has to be initialized and given start values before they are optimized.

You have to estimate what start values to give to the optimizing variables based on your knowledge of the system or of relevantly similar systems. There are no general rules for what start values your optimizing variables should have. If you are not able to make any informed estimation about what start values to use, then you may not be able to find a good fit between your experimental data and your calculated results. In this case, it is often best to start the optimization in the so-called alternate mode. In alternate mode, the PARROT module helps you find some reasonable start values for your optimizing variables.

Also see *About Alternate Mode* on page 43.

Even if you are starting your optimization in alternate mode, you still have to initialize the optimizing variables though (you can all give them a start value of 0).

To initialize an optimizing variable and to give it a start value, use either *SET_OPTIMIZING_VARIABLE* on page 114 or *SET_SCALED_VARIABLE* on page 115. Using SET-OPTIMZING_VARIABLE results in a scaling factor equal to the current value of the parameter unless the current value is zero, in which case the scaling factor is 1000. This is typically appropriate for enthalpic parameters, but not necessarily for other parameters. The scaling factor is the factor by which the optimizing variable value is varied during the optimization.

Example

For example, if you want to optimize the variables V1, V2, V11, V13 and V17, then you could enter the following in the setup file:

```
SET_OPTIMIZING_VARIABLE 1 +43000

SET_OPTIMIZING_VARIABLE 2 -35.2

SET_OPTIMIZING_VARIABLE 11 +72000

SET_OPTIMIZING_VARIABLE 13 +27000

SET_OPTIMIZING_VARIABLE 17 -100000
```

The value given to each optimizing variable (+43000 for variable V1 for example) is also that variable's initial scaling factor. If you want to set the initial scaling factor of a variable to a value that is lower or higher than its start value, then use SET-SCALED_VARIABLE `<start value><scaling factor><min value><max value>`.

Creating the Workspace File

In the setup file, after the system definition and the entered optimizing variables, use the PARROT command *CREATE_NEW_STORE_FILE* on page 94 to create and save a PARROT workspace file for the system defined in the setup file. For example, the following command saves all current data on system definitions (elements, species, components, phases), symbol definitions (constants, variables, functions, tables) and parameters that are stored in the workspace used by GIBBS, POLY and PARROT into a file called AL2O3-ZRO2.PAR:

```
CREATE_NEW_STORE_FILE AL2O3-ZRO2
```

The experimental information in the POP-file is not saved to this PARROT workspace file.

Compiling the POP-file

At the end of the setup file, use the command *COMPILE_EXPERIMENTS* on page 92 followed by the name of your POP-file to compile the data in that file and save it to the current PARROT workspace file. Note that you must first have opened or created a new PARROT workspace file before you compile your POP-file. The other argument to the command can typically be given default values with comma signs (,):

```
COMPILE_EXPERIMENTS AL2O3-ZRO2.POP,,,
```

If Thermo-Calc encounters errors in the syntax of the POP-file, then error messages are displayed in the console. Normally, the compilation also stops when syntax errors are encountered. If this happens, then you must correct the errors and recompile the file. It is useful to have two console windows open when you do this. That way, you can use one console for editing the experimental data and another console for compiling and reading any further error messages.

Returning Control to the Console

Put the command *SET_INTERACTIVE* on page 8 at the end of the setup file if you want to return control to the console when the commands in the setup file have all been executed.

Optimizing in PARROT

In this section:

Optimizing the System in PARROT

When you have created a POP-file, a setup macro file and (optionally) EXP-files, you should start doing optimization runs interactively in the console. This process can be divided into a number of steps (such as running optimization, evaluating optimization results, modifying weights and models and adding new information), but usually, you have to go through many optimization cycles before achieving a satisfactory fit between calculation results and experimental data.

Typically, it is not obvious when an optimization is finished. You must exercise your judgement about when the fit you have achieved is good enough.

It is possible to interactively change almost everything in the initial setup macro file and in the POP-file. For example, you can add more parameters to be optimized and modify or add more experimental information.

Changes that Require POP-file Recompilation

Some changes that you do interactively in the PARROT module or the ED_EXP sub-module destroys the data structure in the workspace. For example, if you add more composition sets to a phase, then the number of phases actually change and the links break between the experimental data and the thermodynamic models. Note that if the global minimization technique is used in POLY, then new composition sets that corrupts the PARROT workspace may be created automatically. If this happens, then you have to compile the POP-file again.

If you recompile the POP-file, then the experimental equilibria are calculated with the default start values.

Optimization Workflow

Normally, the workflow during an optimization is the following:

1. In the PARROT module, run the setup macro file using *MACRO_FILE_OPEN* on page 5.
2. Go to ED_EXP with *EDIT_EXPERIMENTS* on page 95, use *READ_WORKSPACES* on page 160 to load your experimental data and then execute *COMPUTE_ALL_EQUILIBRIA* on page 23.
3. If an equilibrium cannot be computed, then this either means that the start values (site fractions, for example) were inappropriate or that the equilibrium cannot be calculated with the current set of parameters. If the latter, then the equilibrium may have to be excluded from the optimization. However, you should make an effort to exclude the possibility that the reason is just that you do not have the appropriate start values for the equilibrium.
4. When each equilibrium is either successfully computed or excluded from the calculation, go back to the PARROT module and run the *OPTIMIZE_VARIABLES* on page 109 command. Give the number of optimization iterations you want the Thermo-Calc

program to carry out as an argument to the command. This number specifies how many different sets of values for the optimizing variables that PARROT tries to fit with the experimental data. It is recommended that you run `0` iterations first to see the discrepancy between the experimental data and the initial calculated results. The current workspace file is automatically updated with the last set of optimized variables and calculated results.

5. List and evaluate the result of the optimization using *LIST_RESULT* on page 104. Typically, what is of interest in the results are the relative standard deviation of the optimizing variables and the error values in the list of calculated experimental data results. If the relative standard deviation for an optimizing variable is very large, this means that the variable doesn't have any effect on the computed results and that the variable should not be used (you can make PARROT ignore the variable by setting its value to `0` with *SET_FIX_VARIABLE* on page 112). If the relative standard deviation is close to zero for a variable, then this might indicate that you need to use more variables. When it comes to errors in the calculated equilibrium results, these can be decreased by lowering the weight that PARROT gives the experiment in the calculation (the more weight, the more the experiment affects the calculated results).
6. Use *OPTIMIZE_VARIABLES* on page 109 or *CONTINUE_OPTIMIZATION* on page 94 to carry out a new optimization cycle. Go back to step 5 above and continue to do optimization cycles until you judge that there is a good enough fit between your calculated results and your experimental data. It is recommended that you do not only rely on feedback from *LIST_RESULT* on page 104 but also frequently plot various diagrams and compare them visually to some of your experimental data.

Ensuring Computation of all Equilibria

When you have run the setup macro file, you should start by trying to compute all the equilibria that were created when your POP-file was compiled. To do this, you must first go to the ED_EXP submodule, and load the experimental data from the current PARROT workspace file with the *READ_WORKSPACES* on page 160`<block #>` command (normally, the block number is `1`).

When you have loaded all equilibria, you should try to compute them all using the ED_EXP command *COMPUTE_ALL_EQUILIBRIA* on page 23. A list of calculated equilibrium results are shown in the console, but the software is not able to reach an equilibrium for an experiment, then the program cancels the calculations and displays an error message.

The following list shows an example output where the first two experiments in the POP-file have been successfully computed, but an equilibrium is not reached for the third experiment. No other equilibria are computed, since the calculations are aborted when the error occurs.

```
Eq  Lab  Iter Weight Temp   Exp  Fix phases or comments
  1 A1    22   1.    2575.3      IONIC_ FLUORI TETR
  2 A1    18   1.    2156.9      IONIC_ CORUND TETR
```

```
*** ERROR  1614 IN QTHISS
*** CONDITIONS CAN NOT BE FULLFILLED
```

When this happens, try to find out why the equilibrium calculation fail. Are the condition start values that you have given the equilibrium inappropriate for example? If you have a similar equilibrium that has already been successfully computed, then you can select the equilibrium that does not work, use the ED_EXP command *SET_ALL_START_VALUES* on page 162 and accept all the suggestions for the site fractions. Alternatively, use *TRANSFER_START_VALUE* on page 34`<equilibrium number>` to make the currently selected equilibrium inherit the start values of the indicated equilibrium. Run *COMPUTE_ EQUILIBRIUM* on page 139 again to see if the equilibrium can be computed with the inherited start value.

If you cannot find an explanation of why the equilibrium cannot be computed, then it may be that the experiment is faulty. If so, then this experiment should be excluded from the optimization by being given a weight of 0. You should then runCOMPUTE_EQUILIBRIUM again to make sure that all equilibria can be computed. However, experiments that fail to reach equilibria could succeed in doing so if the optimizing variables have other values. It is therefore sometimes worth trying to include experiments later in the optimization process (when the optimizing variables are relatively close to their final values), even if they had to be excluded at its start. In some cases, it might also be appropriate to change the error tolerance for in an experiment (using the ED_EXP command *EXPERIMENT* on page 24).

You must save the PARROT workspace with *SAVE_WORKSPACES* on page 30 before you exit the ED_EXP module.

Setting Weights

To change the weight of an experiment, first enter the ED_EXP module from PARROT using *EDIT_ EXPERIMENTS* on page 95. Then use the *SET_WEIGHT* on page 33 command to set the weight for a specific equilibrium (equilibrium number `3` for example), for a range of equilibria (the equilibria `10-19`) or all the equilibria that have a certain label (such as all the equilibria with the label `A1`). To save the new weight(s), use *SAVE_WORKSPACES* on page 30.

By default, all the experimental information in your POP-file is treated equally: each experiment has a weight of `1`. However, experiments that you excluded because they did not reach equilibrium now have a weight of `0`. The higher the weight of an experiment is, the more Thermo-Calc tries to ensure that the computed result fits the condition measured in that experiment.

The contribution of an experiment’s error to the sum of errors is multiplied by the square of the weight set for that experiment. In other words, if you want to cut the error of an experiment with weight 1.0 by half, then you should use a weight of 0.7. This makes the experiment’s error 0.49. Setting the weight to 1.4 instead makes the error twice as large (1.96).

Note that if you use *READ_WORKSPACES* on page 160 after you have changed weights on the experiments, your changes are lost unless you have saved those changes. In the ED_EXP module, you can get a list of all the experiments in the console if you first read the experimental data from the workspace file (with READ_WORKSPACES) and then use *LIST_ALL_EQUILIBRIA* on page 28.

Optimizing and Evaluating Results

When you have ensured that all the experimental data points can reach equilibrium, go back to the PARROT module and use *OPTIMIZE_VARIABLES* on page 109 <number of iterations> with 0 iterations. Present the result of this initial optimization by using *LIST_RESULT* on page 104.

When evaluating the optimization results, the most important information is found in the list of experiments at the end of the listed results. The rightmost column shows the difference between the value of the experimental data and the value computed by Thermo-Calc in the last optimization cycle.

The following shows part of list of experiments in the output of a LIST_RESULT command:

```
106 ACR(B)=0.34                 0.3127      2.89E-02 -2.7282E-02 -0.9444
107 ACR(B)=0.23                 0.2085      2.89E-02 -2.1522E-02 -0.7450
108 ACR(B)=0.12                 0.1042      2.89E-02 -1.5761E-02 -0.5455
110 HMR(LIQUID)=-1964           3.6380E-12  5.00E+02  1964.        3.928
111 HMR(LIQUID)=-3500            0.000      5.00E+02  3500.        7.000    *
112 HMR(LIQUID)=-4588            0.000      5.00E+02  4588.        9.176    *
113 HMR(LIQUID)=-5239           7.2760E-12  5.00E+02  5239.        10.48    *
114 HMR(LIQUID)=-5454          -3.6380E-12  5.00E+02  5454.        10.91    *
115 HMR(LIQUID)=-5233           1.8190E-12  5.00E+02  5233.        10.47    *
116 HMR(LIQUID)=-4575          -1.8190E-12  5.00E+02  4575.        9.150    *
117 HMR(LIQUID)=-3481          -1.8190E-12  5.00E+02  3481.        6.962    *
118 HMR(LIQUID)=-1950            0.000      5.00E+02  1950.        3.900

PARROT:
```

The experiments for which the optimization has not found a good fit are marked with an asterisk (*) or a hash sign (#) in the rightmost column. An asterisk in this column indicates that the experiment has an error that is larger than the error tolerance and a hash sign indicates that the error is much larger. The error tolerance depends on the degree of accuracy set for that particular piece of experimental information. The error values are presented in the next to rightmost column.

Note that if you are optimizing in *alternate mode*, then you usually do not need to change the weights for the experiments. Instead, you can simply run OPTIMIZE_VARIABLES again, specifying, say, thirty iterations, evaluate the results again, and continue in that way until you are satisfied with the results.

Also see *About Alternate Mode* on page 43.

The Critical Set of Experiments

The set of weighted experiments that you end up with as a result of your optimization is referred to as the *critical set* of experiments. When you work to determine this set and the weights of the experiments, the following factors should be reflected in both the selection of experiments and the weights of those experiments:

- The reliability of the experimental technique.
- The extent of agreement/disagreement between independent measurements of the same quantity.
- The extent of agreement/disagreement between data obtained with different experimental methods.

Continuing the Optimization and Resetting Variables

To continue the optimization after you have used *OPTIMIZE_VARIABLES* on page 109, you can either use this command and enter `<Number of iterations>` again or *CONTINUE_OPTIMIZATION* on page 94 `<Number of iterations>`.

If you use *CONTINUE_OPTIMIZATION* on page 94, then PARROT continues the optimization using the same Hessian matrix. If the optimization is going well but you run out of iterations, then continuing with the same Hessian matrix may be useful.

As before, use *LIST_RESULT* on page 104 to inspect the fit between the calculated and the experimental values for your experiments. If necessary, adjust the weights of the experiments again in the ED_EXP module. You may even discover that you have to exclude additional experiments from the optimization.

Besides looking at the rightmost column in the list of experiments (that is produced when you use LIST_RESULT), it is also useful to look at the *sum of squares* of the errors of all the variables. This is shown in the output of OPTIMIZE_VARIABLES and CONTINUE_OPTIMIZATION. This sum should be as low as possible.

Example Output

The following shows part of the output that you get when using *LIST_RESULT* on page 104:

```
 == OPTIMIZING VARIABLES ==

 AVAILABLE VARIABLES ARE V1  TO V00

 VAR.   VALUE             START VALUE       SCALING FACTOR    REL.STAND.DEV
 V1      6.94641846E+05    4.30000000E+04    4.30000000E+04    1.42299558E+00
 V2     -3.32619842E+02   -3.52000000E+01   -3.52000000E+01    7.91369573E-01
 V11     8.37422071E+04    7.20000000E+04    7.20000000E+04    2.02212692E-01
 V13     2.65253341E+04    2.70000000E+04    2.70000000E+04    1.45201828E-01
 V17    -3.66637428E+05   -1.00000000E+05   -1.00000000E+05    8.04279315E-01

 NUMBER OF OPTIMIZING VARIABLES :   5
 ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO
 THE SUM OF SQUARES HAS CHANGED FROM   3.50196879E+05 TO   3.49096720E+05
 DEGREES OF FREEDOM   14.  REDUCED SUM OF SQUARES   2.49354800E+04
```

In the example you can see that the sum of squares have decreased slightly. (The final sum of squares that you get after an optimization run is also presented directly in the output from OPTIMIZE_VARIABLES and CONTINUE_OPTIMIZATION.) As long as the sum of squares figure is decreasing, the optimization is working as it should.

When your calculated results have been improving for a while, that is, when the sum of squares has been decreasing, and you are confident that you are making definitive progress, then you should use the PARROT command *RESCALE_VARIABLES* on page 110 to reset the starting values of the optimizing variables to their current values. When you do this, the start values and scaling factors of the optimizing variables (shown in the table's third column in the preceding screen shot), are all set to the current values (shown in the table's second column).

If your optimization is not going well, then you may want to reset the values of your optimization variables to their current start values. To do this, use *RECOVER_VARIABLES* on page 110.

After rescaling the variables, continue the optimization using OPTIMIZE_VARIABLES again. Cycle through calls of the commands OPTIMIZE_VARIABLES/CONTINUE_OPTIMIZATION, CONTINUE_OPTIMIZATION, RESCALE_VARIABLES and OPTIMIZE_VARIABLES again until your set of optimizing variables does not seem to improve anymore. Even when it seems to you that you have reached this point, it is recommended that you use OPTIMIZE_VARIABLES/CONTINUE_OPTIMIZATION a few extra times to make sure that the variable values cannot be further improved, that is, that the sum of squares of the errors of all the variables cannot be reduced further. If the final solution is repeatedly calculated after the same number of iterations and reaching the same result, then it is generally reasonable to accept the current variable values.

Updating the Set of Optimizing Variables

Besides evaluating the fit between the calculated results and the experimental data, you should also ensure that you have the right number of optimizing variables. If you have too many or too few optimizing variables, then you may not be able to use your model to accurately extrapolate thermodynamic properties in a wide range of temperature, pressure and composition conditions.

With more optimization parameters, the sum of errors usually decrease, but the parameters also tend to become less precisely determined. A measure of this precision is the Relative Standard Deviation (RSD) of each optimizing variable. However, the RSD is only significant if you have used *RESCALE_VARIABLES* on page 110 followed by an optimization which converges and the values of the variables do not change much. The RSD tells you how much the parameter can be changed in either a positive or in a negative direction without changing the reduced sum of errors with more than one unit.

Example

A large RSD value thus means that the parameter has not been determined well.

```
== OPTIMIZING VARIABLES ==

AVAILABLE VARIABLES ARE V1  TO V00

VAR.   VALUE              START VALUE       SCALING FACTOR    REL.STAND.DEV
V1      2.01820948E+04     2.01874462E+04    2.01874462E+04    2.60519742E-02
V2     -2.90936164E+01    -2.90969472E+01   -2.90969472E+01    1.41576516E-02
V11    -2.18127453E+04
V12     1.55559524E+01
V15     2.36701148E+04     2.37258719E+04    2.37258719E+04    1.00528931E-01
V16    -7.56541897E+00    -7.64417807E+00   -7.64417807E+00    3.17218507E-01
V17     3.00342429E+03     3.02274918E+03    3.02274918E+03    2.60675550E-01
V19     2.20133190E+04     2.14534347E+04    2.14534347E+04    6.92923849E-01
V20    -6.72498276E+00    -6.31334932E+00   -6.31334932E+00    1.79242248E+00

NUMBER OF OPTIMIZING VARIABLES :   7
ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO
THE SUM OF SQUARES HAS CHANGED FROM   1.51360340E-01 TO   1.44420287E-01
DEGREES OF FREEDOM   18.  REDUCED SUM OF SQUARES   8.02334928E-03
```

If such a significant RSD has a value larger than 1, then this typically means that you are using too many optimizing variables. However, the RSD values also depend on the weighting of the experiments, so it is sometimes possible to reduce the RSD by changing weights.

The RSD should not be too low either. If one or more of them is very close to 0, then this suggests that you are trying to optimize the system with too few optimizing variables. You can add optimizing variables interactively using *ENTER_PARAMETER* on page 99 in GIBBS. However, it is recommended that you add a sufficient number of variables in the setup file from the start so that do not need to do this. You can fix variables to a value of 0 until you discover that they are needed.

Besides the RSD, you should also keep an eye on the variable values themselves (shown in the VALUE column, next to the leftmost column in the table shown in the preceding screenshot). If the value of a variable is in the order of 1E5 or higher and this variable has been put in the temperature-independent part of an interaction parameter’s Gibbs energy expression, then you may not have the right weights on your experiments or you are using too many optimizing variables. This is also typically the case if a variable with a value in the order of 10 or more (per mole atoms) is in the temperature-dependent part of the Gibbs energy expression of an interaction parameter. Such a high parameter value may lead to inverted miscibility gaps or re-stabilisation of a phase at high temperature. It is important to keep an eye on this since the temperature-independent and the temperature-dependent interaction parameter often varies together. If the temperature-dependent interaction parameter cannot be reliably determined, you may have to fix its value.

Note that even if you have found a good enough fit between calculated and experimental values after an optimization run, a parameter value may suddenly start to change by several orders of magnitude when you use *OPTIMIZE_VARIABLES* on page 109 is used again. If this happens, then careful reconsideration of the weighting of the experiments of the set of optimizing variables is required. However, when the optimization has converged and the parameter values remain stable after repeated use of OPTIMZE_ VARIALBES, this should usually not happen.

Reducing the Number of Optimizing Variables

If you have one or more optimizing variables with RSD larger than 1, then you should remove one or more of the variables by setting them to a value of 0, or alternately, set them to a reliably estimated value (estimated based on, for example, semi-empirical methods).

To fix an optimizing variable to certain value, use the PARROT command *SET_FIX_VARIABLE* on page 112 `<variable number> <value>`.

Plotting Intermediate Results

It is typically not sufficient to look at the plain text output of *LIST_RESULT* on page 104 to reliably judge how the optimization is progressing. Plotting a phase or property diagram of the system you are optimizing using the POLY and POST modules is often very useful.

It is recommended that you create a macro file for plotting the diagram since you probably want to repeatedly plot the diagram as you progress with the optimization. This macro file can then be called directly from the PARROT module with the *MACRO_FILE_OPEN* on page 5 command.

Superimposing Experimental Data from EXP-files

If you create one or several EXP-files based on the experiments in your POP-file, you can visually compare the fit your plot and the experimental data you are optimizing the system against. An EXP-file contains data points specified in the DATAPLOT format. You could make an EXP-file for each type of data that you are using in the optimization.

Use the POST command *APPEND_EXPERIMENTAL_DATA* on page 192 to superimpose the data points in an EXP-file on a phase or property diagram that you have plotted. For example, to put the experimental data points from a file entitled DP.EXP onto a plotted diagram, you could enter the following:

```
APPEND_EXPERIMENTAL_DATA Y DP.EXP 0 1
```

Such a command call tells Thermo-Calc to superimpose the data points in dataset 1 of DP.EXP onto the plot in the Console Results window, but not to impose any information about axis scaling, labels and other information which is normally found in the prologue of the DP.EXP file.

The POST command *QUICK_EXPERIMENTAL_PLOT* on page 201 can also be used to superimpose data points on a plotted phase or property diagram.

Also see the *DATPLOT User Guide* included with this documentation set.

Finalizing the Optimization

In this section:

Finishing the Optimization

When to stop optimizing is a matter of judgement. It is rarely the case that you finish optimizing a system with the feeling that it cannot be improved.

The final step is to update your files with the final results. Besides updating your database file (or alternatively, creating a new database file), it is advised that you also update your setup and POP-files with the final weights and optimizing variable values.

Rounding off Optimizing Variable Values

When you save the optimizing variable values, it is important that you round off the values correctly. When you have a metallic system, it is usually best to keep whatever number of decimal digits that you need to make less than one J/mol difference at 1000 K. When you have an aqueous system or a system that you are optimizing based on data from very different temperature ranges, then this is not always the only appropriate guideline. However, rounding off the values in such a way that it gives a difference that is larger than one J/mol may lead to differences in the phase diagram.

Another approach you can take when rounding off variable values is by progressively fixing the value of more and more of the variables. This allows you to round-off all but one of the variables values so that each only has a few significant digits. Variables whose values have been rounded off in this way are easier to handle compared with those that that are rounded off so that they continue to have many significant (non-zero) digits.

To round off the variable values according to this second approach, first use *SET_FIX_VARIABLE* on page 112 to set the variable(s) with the highest RSD to a rounded off value (such as, say, 0.4). When you then re-optimize using *OPTIMIZE_VARIABLES* on page 109, the sum or errors changes. However, you should get almost exactly the same sum of errors as before after you rescale the variable values using *RESCALE_VARIABLES* on page 110. (If the sum of errors is different, then this means that the variable with the highest RSD was not rounded off in a good way.) If you successfully round off the first variable value, then continue to round of the optimizing variable which now has the highest RSD. Continue doing this until you have fixed all but one of the optimizing variables to their rounded-off values. The final sum of errors that you get after having fixed the variable values should not deviate significantly from the initial sum of errors you had before starting to round-off the variable values.

Updating the Database File

To update the database file (with filename extension TDB) open it in a text editor and enter or update each of the parameters that you have optimized.

For example, suppose you have the following command in your setup file:

```
ENTER_PARAMETER L(IONIC_LIQ,AL+3,ZR+4:O-2;0) 298.15  +V17; 6000 N
```

If you then end up with a value for optimizing variable V17 of -100000, then you should enter the following under the IONIC_LIQ phase in the database file:

```
PARAM L(IONIC_LIQ,AL+3,ZR+4:O-2;0) 298.15 -100000; 6000 N REF !
```

If you often need to round-off the optimizing variable value in some way, see *Rounding off Optimizing Variable Values* on the previous page.

When you have updated or entered the parameters that you optimized, save the database file.

Creating a Database File

In some circumstances you may want to save all the information about your system into a new database file. To do this, go to the GIBBS module and use LIST_DATA `<Filename> N`. This creates a new user database file that contains all the information about the system that is in the workspace.

The N-argument tells the *LIST_DATA* on page 77 command to save the output into database file rather than some other type of file, such as a macro file for example. Note that the output database file that you get from LIST_DATA have to undergo some manual editing before it can be used as proper database file.

Updating the Setup File and the POP-file

It may be useful for you to be able to recreate your optimization from scratch. It is therefore recommended that you do the following when you are done with the optimization:

- Update your POP-file so that it contains the weights of your critical set of experiments. Use *SET_WEIGHT* on page 33 to set the weights directly in the POP-file. You may also have to use *SET_START_VALUE* on page 171 for some conditions in the POP-file. Otherwise, you may not be able to compute the equilibria with *COMPUTE_ALL_EQUILIBRIA* on page 23.
- Copy the final values of your optimizing variables and set them as start values of the variables in the setup macro file (with *SET_FIX_VARIABLE* on page 112).

It is important that you round off the values of the optimizing variables correctly. See *Rounding off Optimizing Variable Values* on the previous page. When you have done this, you can run the setup macro file to recreate the final result of the optimization.

Alternate Mode

In this section:

About Alternate Mode

To turn Alternate Mode ON or OFF, use the PARROT command *SET_ALTERNATE_MODE* on page 111.

When you have one or more multi-phase equilibria (that is, with the fixed phase-status) in your POP-file and the optimizing values lack appropriate starting values (such as the default value of 0 for example), then Thermo-Calc may not be able to compute the equilibria for the experiments. In these cases, a common tangent cannot be found, that is, a state where the chemical potentials for all components are the same in all phases cannot be found. A common tangent may not exist or it be found at the wrong composition or temperature given the initial set of model parameters.

If you are not able to estimate reasonable starting values for the optimizing variables, then this problem can be overcome by initially running your optimization in alternate mode. The alternate mode in PARROT module should help you find reasonable starting values for the optimizing variables. Note that you are only supposed to use alternate mode in the beginning of the optimization. Once you have achieved a reasonable fit between calculated results and experimental data, turn off alternate mode and calculate all equilibria again in the normal mode.

In alternate mode, PARROT modifies multi-phase equilibria so that the chemical potentials for each stable phases is calculated separately. For example, consider the following experimental equilibrium:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQ FCC=FIX 1
SET_CONDITION T=1000 P=1E5
EXPERIMENT X(LIQ,B)=.2:.01 X(FCC,B)=.1
```

When PARROT optimizes in alternate mode, the computation of this equilibrium is done by way of the following two calculations:

```
CHANGE_STATUS PHASE LIQ=FIX 1
SET_CONDITION T=1000 P=1E5 X(LIQ,B)=.2
CHANGE_STATUS PHASE FCC=FIX 1
SET_CONDITION T=1000 P=1E5 X(FCC,B)=.1
```

The difference between the chemical potentials in every pair of phases is calculated and PARROT tries to adjust the parameters of the phases so that these differences are minimized. In other words, PARROT tries to make the chemical potentials equal in all phases. For PARROT to be able to do this, some additional information about the experimental multi-phase equilibria is required, at least for all invariant equilibria.

Optimizing in Alternate Mode

When you are optimizing in the alternate mode, you usually do not need to enter the ED_EXP module and manually change the weights of experiments that the computed results have not been fitted well with. Instead, after using *OPTIMIZE_VARIABLES* on page 109 initially with no iterations, and listing the result with *LIST_RESULT* on page 104, simply run OPTIMIZE_VARIABLES again specifying, say, thirty iterations. Continue to optimize until your optimization seems to be going in the right direction, then set the optimizing variables' starting values to their current values use *RESCALE_VARIABLES* on page 110 before you turn alternate mode off by answering N at the *SET_ALTERNATE_MODE* on page 111 prompt.

Preparing the POP-file for Alternate Mode

Information in your POP-file that concerns equilibria with ENTERED or DORMANT phases is ignored when you optimize in alternate mode.

To use the alternate mode, ensure that there is enough information about each phase in a given equilibrium. In addition to the temperature and pressure of the system, you should specify the composition of all phases in a given equilibrium. This is done with, *SET_CONDITION* on page 165, *EXPERIMENT* on page 24or *SET_ALTERNATE_CONDITION* on page 30.

If both compositions of a binary tie-line have been measured, then you could specify this in the POP-file in the following way:

```
CREATE_NEW_EQUILIBRIUM 1 1

CHANGE_STATUS PHASE FCC BCC=FIX 1

SET_CONDITION P=1E5 T=1000

EXPERIMENT X(BCC,B)=.2:.01 X(FCC,B)=.3:.01
```

This equilibrium could be calculated with alternate mode without any modification. The compositions specified with the experiment command is used as condition when the thermodynamic properties of each phase are calculated.

Examples of SET_ALTERNATE_CONDITION

The following topics demonstrate how *SET_ALTERNATE_CONDITION* on page 30 is used to prepare different kinds of equilibria for optimization in alternate mode.

Specifying the Composition of One Side of a Binary Tie-Line

If only one side of the tie-line has been measured, then you must provide an estimate of the composition of the other phase. You can make such an estimate with SET_ALTERNATE_CONDITION. This command has no effect unless alternate mode is turned on. The equilibrium described could thus be specified in the following way:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE FCC BCC=FIX 1
SET_CONDITION P=1E5 T=1000
EXPERIMENT X(BCC,B)=.2:.01
SET_ALTERNATE_CONDITION X(FCC,B)=.3
```

When the alternate mode is turned on, the composition given by the SET_ALTERNATE_CONDITION command is used for the FCC phase. The properties of the BCC phase is calculated using the composition specified by experiment as a condition.

Specifying the Compositions of a Three-Phase Equilibrium

If you have a three-phase equilibrium in your POP-file, then one of its compositions can be specified as experimentally determined, while two can be provided as alternate conditions:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1
SET_CONDITION P=1E5
EXPERIMENT T=912:5 X(LIQ,B)=0.2:.02
SET_ALTERNATE_CONDITION X(FCC,B)=0.1 X(BCC,B)=.4
```

Specifying the Composition of a Stoichiometric Phase

When you specify the composition of stoichiometric phase with SET_ALTERNATE_CONDITION, the value must be given with at least seven decimal digits, as in the following example:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQ A2B=FIX 1
SET_CONDITION P=1E5 X(LIQ,B)=0.2
EXPERIMENT T=992:5
SET_ALTERNATE_CONDITION X(A2B,B)=.6666667
```

Entering Other Experiments in the POP with Alternate Mode

The following table shows some additional examples of how to enter different kinds of experiments in the POP-file when you are using the alternate mode.

Example	**ED_EXP commands**
Two-phase equilibrium: The melting temperature of an Au-Cu alloy	CHANGE_STATUS PHASE LIQUID FCC=FIX 1 SET_CONDITION X (FCC,CU)=0.14 P=1E5 EXPERIMENT T=970:2 SET_ALTERNATE_CONDITION X(LIQUID,CU)=0.16
Invariant equilibrium experiment: A three-phase equilibrium in a binary system	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC BCC LIQUID=FIX 1 SET_CONDITION P=1E5 EXPERIMENT T=912:5 SET_ALTERNATE_CONDITION X(FCC,B)=0.1SET_ ALTERNATE_CONDITION X(BCC,B)=0.4SET_ALTERNATE_CONDITION X (LIQ,B)=0.2
Ternary system, with two compositions measured (both with uncertainty 0.02)	CREATE_NEW_EQUILIBRUM 1 1 CHANGE_STATUS PHASE FCC BCC=FIX 1 SET_CONDITION T=1273 P=1E5 SET_CONDITION X(FCC,B)=0.1:0.02 EXPERIMENT X(FCC,C)=0.12:.02 SET_ALTERNATE_CONDITION X (BCC,B)=0.17 SET_ALTERNATE_CONDITION X(BCC,C)=0.07

Troubleshooting

During an optimization, you are likely to run into many different problems and challenges. This topic gives some general guidelines that can help avoid common mistakes and problems.

In this section:

Useful Guidelines

The following rules of thumb are generally good to adhere to when you are planning and executing an optimization:

- Do not use data from thermochemical tables of unknown origin.
- Use experimentally determined properties; avoid converted quantities.
- Carefully estimate the accuracy of the experiments.
- Correct systematic errors (by adjusting a temperature scale for example).
- Use negative information such as, for example, the information that a phase should not be stable in a composition or temperature region.
- With a hundred activity measurements in a system but only ten composition points from the phase diagram, it is often appropriate to decrease the weights on the activity experiments.
- If you initially exclude any intermediate phases to only optimize the liquid and the terminal or end-member phases for the pure components, then it is often useful to compute a metastable phase diagram with just these phases. In the metastable phase diagram, the metastable solubility lines should not have any strange kinks or turns. Such a diagram is often also useful to compute at later stages in the optimization.
- When you have fitted the liquid phase and some solution phases reasonably well, then you can fix the optimizing variables that characterise these phases. You can then go on and optimize the variables for the intermediate phases.
- Phases with miscibility gaps are always difficult to optimize. Try to keep control of the miscibility gaps by using real or estimated experimental information. (Note that it is not possible to calculate the top of a miscibility gap as a single equilibrium.)
- Phases with order/disorder transformation are often difficult to optimize. It may be difficult to find the right start values on the optimizing variables. The TABULATION module (see *TABULATION_REACTION Commands* on page 245) can be useful sometimes. Properties for specified site fractions can be calculated in the TABULATION module. This is not possible in the POLY module since this module only calculates for equilibrium site fractions. The ordered state may also disappear during the optimization of these phases. If this happens, you may be able to add an experiment that controls the state of order.
- Only use the alternate mode to find an initial set of model parameters that can then be make it possible to calculate the experimental equilibria in the normal mode.
- When you have achieved a satisfactory fit between calculated results and experimental data with an appropriate number of optimization variables, perform a couple of final optimization runs to make sure that your results are stable.

- Make sure that the values of the optimizing variables keep within a reasonable range. If a variable starts to change several orders of magnitudes, then you must consider whether you have the right weights on the experiments and whether you are using too many optimizing variables.

Excluding and Including the Correct Equilibria

Use as few experiments as possible initially to get a reasonable overall fit. It is generally best to first focus on the invariant equilibria, as well as metastable states that can be estimated by excluding some phases. Crucially, any experiments for which the computed results are clearly faulty should be excluded from the final critical set, while all important invariant equilibria are calculated and included in this critical set.

If you cannot compute some invariant equilibria with intermediate phases, then it may be best to first exclude those phases from the optimization and only optimize the liquid and the most important solution phases. After you have obtained reasonable results for these phases, you can enter the intermediate phases back in again. With the variables for the optimized liquid and solid phases set fixed, you can finally optimize the intermediate phases.

Conflicting Data

It is important that you do not include conflicting sets of data at the same time during an optimization. After all, if they are genuinely conflicting, then at least one of the sets must be incorrect. If you have conflicting sets of data, then only use one of these sets at a time together with the rest of the data. Often, the optimization makes it clear which dataset coheres best with the other information that is available about the system.

Sometimes, conflicting data cannot be detected directly. For example, you may have activity data which is inconsistent with solubility data from the phase diagram. This would be indicated by large errors in the fit when both the activity data and the solubility data are included. To find these inconsistencies, let alone correct the errors, you may thus have to optimize with some datasets excluded.

Experiments and Experimental Data

These topics contain examples of different kinds of equilibria in different kinds of systems. They are intended to help you figure out how to use various kinds of experimental data when creating POP-files. The topics that follow contain examples how to enter various kinds of experimental data into a POP-file.

In this section:

Phase Diagram Data

P-T-X phase diagram data for binary and ternary systems, or systems of even higher order can be used in the POP-file. This includes data of the following types:

- Data about the temperature- and composition-dependence of various properties (obtained from differential thermal analysis (DTA) measurements for example), including enthalpy, lattice parameter, dilatometric length, electronic conductivity and magnetic susceptibility;
- Data about reaction types (invariant/monovariant) and phase relations obtained from qualitative or quantitative metallographic studies;
- Data about phase amounts or compositions obtained from microprobe measurements (using X-ray spectroscopy or transmission electron microscopy (TEM));
- Data about the positions or directions of two-phase tie-lines and three-phase equilibria for ternary systems.

Example

The following table shows a POP-file excerpt that demonstrates how some kinds of phase diagram data can be entered:

Example	ED_EXP commands

Solidus (melting temperature) of an Au-Cu alloy	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE LIQUID FCC=FIX 1`
	`SET_CONDITION X(FCC,CU)=0.14 P=1E5`
	`EXPERIMENT T=970:2`

Three-phase equilibrium in a binary A-B system	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE FCC BCC LIQUID=FIX 1`
	`SET_CONDITION P=1E5`
	`EXPERIMENT T=912:5`

Congruent transformation in a binary A-B system	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE BCC LIQUID=FIX 1`
	`SET_CONDITION P=1E5 X(BCC,B)-X(LIQ,B)=0`
	`EXPERIMENT T=1213:10`

Tie-line in a ternary A-B-C system	An uncertainty factor is set not only for the experimental measurement, but also for the X(FCC,B) condition. The factor is 0.02. `CREATE_NEW_EQUILIBRUM 1 1`

Example	ED_EXP commands
	CHANGE_STATUS PHASE FCC BCC=FIX 1 SET_CONDITION T=1273 P=1E5 X(FCC,B)=0.1:0.02 EXPERIMENT X(FCC,C)=0.12:.02

Data for Individual Compounds

You can use data about individual components in the POP-file. Such data includes activity, heat capacity, heat content, entropy, Gibbs energy of phase transformations, heat of transition and heat of melting.

Examples

Example	ED_EXP commands
Carbon activity data in the fcc phase	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1=FIX 1 CHANGE_STATUS PHASE GRAPHITE=D SET_REFERENCE_STATE C GRAPHITE,,,, SET_CONDITION P=101325 T=1273 X(MN)=0.03 SET_CONDITION X(C)=0.03 EXPERIMENT ACR(C)=0.29:5%
Ln(activity coefficient for O)	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE IONIC_LIQ=FIX 1 CHANGE_STATUS PHASE GAS=D SET_CONDITION X(O)=0.02 T=1523 P=101325 SET_REFERENCE_STATE O GAS,,,, ENTER_SYMBOL FUNCTION LNFO=LOG(ACR(O)/X(O)); EXPERIMENT LNFO=-1.5:0.01
Entropy at 298.15 K of Cu2O	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE CU2O=FIX 1 CHANGE_STATUS PHASE CUO=FIX 0 SET_CONDITION P=101325 T=298.15 EXPERIMENT S=92.36:1
Gibbs energy of formation of NiAl2O4	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 CHANGE_STATUS PHASE FCC O2GAS=DORM

Example	ED_EXP commands
	SET_REFERENCE_STATE NI FCC,,,, SET_REFERENCE_STATE AL FCC,,,, SET_REFERENCE_STATE O O2GAS,,,, SET_CONDITION P=101325 T=1000 N(NI)=1 N(AL)=2 SET_CONDITION N(O)=4 EXPERIMENT GM=-298911:5%
Heat capacity of MgFe2O4	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(FE)=2 N(MG)=1 N(O)=4 SET_CONDITION T=800 ENTER_SYMBOL FUNCTION CP=H.T; EXPERIMENT CP=207:5%
Heat content (H-H298) for MnFe2O4	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 T=298.15 N(MN)=1 N(FE)=2 SET_CONDITION N(O)=4 ENTER_SYMBOL VAR H298=H; ENTER_SYMBOL FUN HMNFE2O4=H-H298; CREATE_NEW_EQUILIBRIUM 2 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(MN)=1 N(FE)=2 N(O)=4 SET_CONDITION T=400 EXPERIMENT HMNFE2O4=16610:5%

Single Phase Mixing Enthalpies or Partial Enthalpies

You can use calorimetric data for mixing in the POP-file. This data could be the result of, for example, drop calorimetry or scanning calorimetry. This includes data about, for example, enthalpy of mixing of liquids, series of mixing-enthalpy and about partial enthalpy. Information about enthalpies of mixing can also be derived from theoretical principles using special quasi-random structures (SQSs).

Example

Example	ED_EXP commands

Enthalpy of mixing in the liquid state of an Au-Cu alloy	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE LIQUID=FIX 1`
	`SET_CONDITION T=1379 P=1E5`
	`SET_CONDITION X(LIQUID,AU)=0.0563`
	`SET_REFERENCE_STATE AU LIQ * 1E5`
	`SET_REFERENCE_STATE CU LIQ * 1E5`
	`EXPERIMENT HMR=-1520:200`

Enthalpies of Formation and Enthalpies of Reactions

You can use calorimetric data for phase transformations in the POP-file. This data could be the result of, for example, direct-reaction calorimetry, solution calorimetry or combustion calorimetry. Information about enthalpies of formation can also be derived from theoretical principles.

Examples

Example	ED_EXP commands

Enthalpy of melting of the eutectic in the Al2O3-ZrO2 binary (J/g)	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE CORUNDUM ZRO2_TETR=FIX 0`
	`CHANGE_STATUS PHASE IONIC_LIQ=ENT 1`
	`SET_CONDITION P=101325 N=1 LNAC(O)=-80`
	`SET_REFERENCE_STATE ZRO2 TETR,,,,,,`
	`SET_REFERENCE_STATE AL2O3 CORUND,,,,,,`
	`EXPERIMENT HWR=1080:90`

Enthalpy of formation of o-CaZrO3 from the component oxides	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE ZRO2_MONO HALITE CAZRO3_O=ENT 1`
	`SET_CONDITION T=298.15 P=101325 N(CAO)=.5`
	`SET_CONDITION N(ZRO2)=.5 LNAC(O)=-20`
	`SET_REFERENCE_STATE CAO HALITE,,,,`
	`SET_REFERENCE_STATE ZRO2 ZRO2_MONO,,,,`
	`EXPERIMENT HMR(CAZRO3_O)=-15960:5%`

Chemical Potentials via EMF Measurements

You can use chemical potentials and activities/activity coefficients from electromagnetic field (EMF) measurements in the POP-file.

Example

Example	ED_EXP commands

Emf data on Cu(s)/Cu2O (s)	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE FCC CU2O=FIX 1`
	`CHANGE_STATUS PHASE O2GAS=DORM`
	`SET_CONDITION P=101325 T=1000`
	`SET_REFERENCE_STATE O O2GAS,, 100000`
	`EXPERIMENT MUR(O)=-95387:1000`

Driving Force for Metastable Phases

You can use chemical driving forces for metastable phases in the POP-file.

Examples

Example	ED_EXP commands

To prevent BCC_A2 to be stable	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE FCC_A1 M7C3 M23C6=FIX 1`
	`CHANGE_STATUS PHASE BCC_A2=D`
	`SET_CONDITION P=101325 T=1473`
	`EXPERIMENT DGM(BCC)<-0.1:0.1`

To promote CEMENTITE to be stable	`CHANGE_STATUS PHASE BCC M7C3 GRAPHITE=FIX 1`
	`CHANGE_STATUS PHASE CEMENTITE=DORMANT`
	`SET_CONDITION P=P0 T=973`
	`EXPERIMENT DGM(CEM)>0.1:0.01`

Pressure-Temperature-Volume EOS Data

You can use Pressure-Temperature-Volume Equation of State (EOS) data in the POP-file. Such data includes, for example, molar volume, density, thermal expansion and bulk-modulus/compressibility.

Examples

Example	ED_EXP commands

Volume of liquid Pt	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE LIQUID=ENT 1`
	`SET_CONDITION P=101325 N=1 T=1873`
	`EXPERIMENT VM=1.00758E-5:5%`

Thermal expansion	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE FCC_A1=ENT 1`
	`SET_CONDITION P=101325 N=1 T=400`
	`ENTER_SYMBOL FUNCTION ALPHA=VM.T/VM/3;`
	`EXPERIMENT ALPHA=2.35E-5:5%`

Crystal Structure, Lattice Parameters and Site-Occupancy

You use information about defects, ordering or site occupancy, as well as information about crystal structures in the form of lattice parameter values. This information may come from measurements made by X-ray, neutron diffraction, Mössbauer spectroscopy or perturbed angular-correlation.

Examples

Example	ED_EXP commands
Lattice parameter for fcc	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1=ENT 1 SET_CONDITION P=101325 N=1 T=298.15 SET_CONDITION X(CR)=0.05 ENTER_SYMBOL FUNCTION LPFCC=((4*VM/6.02214179E23)**(1/3))*1E10; EXPERIMENT LPFCC=4.02:5%
Lattice parameter for bcc	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE BCC_A2=ENT 1 SET_CONDITION P=101325 N=1 T=298.15 SET_CONDITION X(AL)=0.08 ENTER_SYMBOL FUNCTION LPBCC=((2*VM/6.02214179E23)**(1/3))*1E10; EXPERIMENT LPBCC=2.90:5%
Site-occupancy, degree of inversion for spinel	CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(MG)=1 N(AL)=2 N(O)=4 SET_CONDITION T=1073 EXPERIMENT Y(SPINEL,AL+3#1)=0.31:5%

Magnetism and Other Atomistic Properties

You can use data about atomistic properties such as magnetism and Curie temperatures in the POP-file.

Example

Example	ED_EXP commands

Curie temperature of MnxFe3-xO4, x=1.26	`CREATE_NEW_EQUILIBRIUM 1 1`
	`CHANGE_STATUS PHASE SPINEL=ENT 1`
	`SET_CONDITION  T=520 P=101325 N(O)=4`
	`SET_CONDITION N(FE)+N(MN)=3 N(MN)=1.26`
	`EXPERIMENT TC(SPINEL)=520:5`

Data About Systems of Different Orders

You might want to use data about systems of different orders in the same optimization, such as data about a binary equilibrium with components A and B in a ternary system with components A, B and C, together with data about equilibria in the binary system with components A and B. To do this, you must first suspend all components in the ternary system (A, B and C), and then set the two components you want binary information on (A and B) to status `ENTERED`.

Suspending all components is done by giving the initialization code 0 to the *CREATE_NEW_EQUILIBRIUM* on page 143 command. The components about which you want to use binary information must then be entered again using the *CHANGE_STATUS* on page 59 command.

Example

Example	ED_EXP commands

A binary (A-B) three-phase equilibrium (FCC-BCC-LIQ) in a ternary system (A-B-C)	`CREATE_NEW_EQUILIBRIUM 1 0`
	`CHANGE_STATUS COMPONENT A B = ENTERED`
	`CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1`
	`SET_CONDITION P=1E5`
	`EXPERIMENT T=1177:10`

Thermodynamic Models

The most common models used in assessments are described briefly in these topics. Examples of how to write the phase definitions in the setup-file are also given for each of these models. In addition to the models described here, a large number of thermodynamic models for various phases in different states have been implemented.

In this section:

Gas

The gaseous mixture phase is usually treated as a substitutional phase without sublattice. In the gaseous mixture phase, there are usually molecules formed, and the number of constituents of the gas phase is often much larger than the number of elements. These complex constituents have to be defined as species. The ideal gas model may handle a gas phase under low pressures (and low temperatures). This implies that the P-V-T relations and thermodynamic properties of pure gaseous species are calculated as for ideal gas, and that there is interaction between gaseous species in the mixture. In a defined system, there may only exist one gaseous mixture phase. Type code `G` is used for the gas phase:

```
ENTER_PHASE GAS G, 1 H,H1O1,H1O2,H2,H2O1,H2O2,O,O2,O3; N N
```

Compound Energy Formalism

If the atoms are sufficiently different in size, electronegativity or charge, they may prefer different types of sites in the lattice of crystalline solids. In some cases, a solute atom may even occupy interstitial sites between the normal lattice sites. All such phenomena are treated in Thermo-Calc by the sublattice concept. The Compound Energy Formalism (CEF) has proved to be the most general formalism for many different types of solution phases. It is therefore by default applied to various solid solution phases in the Thermo-Calc software package. CEF can also take ionic constraints (charged cation/anion species) into account. For a comprehensive description on the CEF, please see Hillert (2001), Frisk and Selleby (2001), Sundman and Ågren (1981).

Since the CEF is used by default, no special notation is needed in the phase definition, except for ionic phases, where type code `I` is given.

Examples

Examples of phase definitions in a setup-file using the CEF:

```
ENTER_PHASE SPINEL I, 4 1 2 2 4
     FE+2,FE+3; FE+2,FE+3,VA; FE+2,VA; O-2; N N
ENTER_PHASE SIGMA, 3 10 4 16
     AL,CO,CR,FE,NB; AL,CO,CR,FE,NB; AL,CO,CR,FE,NB; N N
ENTER_PHASE FCC_A1, 2 1 1
     CR,FE,MN; C,N,VA; N N
ENTER_PHASE BCC_A2, 2 1 3
     CR,FE,MN,VA; C,N,VA; N N
```

Substitutional Liquid

The liquid phase is normally (but not always) modelled as a substitutional phase with no sublattice, which is occupied by two or more constituents. The constituents are normally the elements, but molecules or associates can also be used. Type code `L` is used for the liquid phase.

Example

```
ENTER_PHASE LIQUID L, 1
```

```
        AL,C,CR,FE,MO,NI,SI ;  N N
    ENTER_PHASE LIQUID L, 1
        C,FE,FEO,FEO3/2,NI,NIO ;  N N
```

Ionic Two-Sublattice Liquid Model

Within the framework of the CEF, the ionic two-sublattice liquid (I2SL) model was developed to be used when there is a tendency for ionization in the liquid, which happens in liquid oxides and sulphides for example. The same model can be used both for metallic and oxide melts. At low levels of oxygen, the model becomes equivalent to a substitutional solution model between metallic atoms. Two sublattices are assumed, one containing charged cations and one containing charged anions, neutrals and vacancies. The complication for an ionic liquid is that the numbers of sites on the cation sublattice and anion sublattice vary with the composition to preserve electroneutrality. For a comprehensive description on the I2SL model, see Hillert et al. (1985), Sundman (1991). In these references, the possible excess parameters are also discussed. Type code `Y` is used for the I2SL model.

Example

Examples of phase definitions in a setup-file:

```
    ENTER_PHASE LIQUID Y,
        AL+3,CR+2,FE+2,NI+2; VA; N N
    ENTER_PHASE IONIC_LIQ Y,
        CA+2,MG+2; O-2; N N
    ENTER_PHASE IONIC_LIQ Y,
        AL+3,CA+2,FE+2,MG+2,SI+4;
        ALO2-1,O-2,SIO4-4,VA,FEO3/2,SIO2; N N
```

Models for Ordered Phases

Some solution phases have chemical order-disorder transformations: for example, the disordered FCC_ A1 phase can transform to the ordered $L1_2$ or $L1_0$ structure where the atoms of different kinds occupy different sublattices. In some cases, it may be advantageous and easy to describe such ordered structures as completely different phases; but in some other cases, like the B2-ordered BCC structure in Al-Fe, the disordered and ordered structures must be described as one singe phase because the transformation is of second order along a line in temperature and composition.

The ordering can easily be described by the CEF. For instance, an ordered FCC phase with only $L1_2$ or $L1_0$ ordering can be efficiently handled by the so-called Two Substitutional-Sublattice Ordering Model (2SL model), while an ordered FCC with both $L1_2$ and $L1_0$ ordering must be described with four sublattices using the so-called Four Substitutional-Sublattice Ordering Model (4SL model). Please note that in both these models, all the substitutional constituents must enter into these sublattices. In addition, you may also have one sublattice for interstitials.

Thermo-Calc supports a feature of splitting the parameters of a chemically ordered phase onto two different phase descriptions. This is very convenient. In this way, you can have one phase for all parameters describing the disordered state (the configuration independent state) and in the other phase only those parameters needed to describe the remaining part (the configuration dependent part), that is, the Gibbs energy contribution due to the ordering transformation. This has been implemented in such a way that the contribution from the configuration dependent part is zero when the solution phase is disordered. The optimization of the disordered phase is independent of the order-disorder transformation; the ordering energy is an add-on to the disordered part. This is particularly useful in higher order systems, as an ordered phase can dissolve several elements that have no particular contribution to the actual ordering. The phases Gibbs energy add together by giving this command in the setup-file: *AMEND_PHASE_DESCRIPTION* on page 52 `<ordered phase> disordered_part <disordered phase>`

Since PARROT does not automatically create new composition sets, it is necessary to create composition sets for both the ordered and disordered phases in the setup-file: `Amend_phase_description <phase> composition_sets <new highest set number>,,,,`

Example

Example of phase definitions for the corresponding disordered and ordered parts in a setup-file:

```
ENTER_PHASE FCC_A1, 2 1 1
     AL,CR,FE,NI; VA; N N
ENTER_PHASE FCC_L12, 3 .75 .25 1
     AL,CR,FE,NI ; AL,CR,FE,NI; VA; N N
AMEND_PHASE_DESCRIPTION FCC_L12 DIS_PART FCC_A1
```

4SL Model for FCC, HCP and BCC

Ordered FCC, HCP and BCC solution phases handled by the 4SL model requires four sublattices for substitutional ordering and can additionally have an interstitial sublattice. A 4SL model is especially useful for modelling different kinds of ordered phases that are based on the same disordered phase, such as $L1_0$ and $L1_2$ based on FCC_A1, and B2, $D0_3$, $L2_1$ and B32 based on BCC_A2 etc.

For ordered FCC or HCP phases, these four substitutional sublattices represent four corners of the regular tetrahedron on these lattices, all of which are the nearest neighbours. These corners are equivalent lattice points, thus all G parameters for each end-member with the same elements, but distributed on different sites, must be identical. It should be emphasized that the end-member energy here represents the ordering energy rather than the formation energy of the compound.

The constraints on the parameters in the 4SL model can be derived based on the symmetry of the lattice:

$G_{A:B:B:B}=G_{B:A:B:B}=G_{B:B:A:B}=G_{B:B:B:A}$

$G_{A:A:B:B}=G_{A:B:A:B}=G_{A:B:B:A}=G_{B:A:A:B}=G_{B:A:B:A}=G_{B:B:A:A}$

$G_{B:A:A:A}=G_{A:B:A:A}=G_{A:A:B:A}=G_{A:A:A:B}$

$L_{A,B:*:*:*}=L_{*:A,B:*:*}=L_{*:*:A,B:*}=L_{*:*:*:A,B}$

The asterisk * means that the interaction parameter is independent of the occupation of that sublattice. For the disordered phase to be completely disordered, i.e. that all site fractions are equal on all four sublattices, all constraints must be correct.

For ordered BCC phases, the situation is a bit more complicated, as the four sublattice ordering phase represents an irregular tetrahedron with two pairs of sites that are next nearest neighbours. Thus, for an A-B binary solution phase, with A located on two sublattice sites and B on the other two, the end-member described by $G_{A:A:B:B}$ has four nearest neighbour bonds between A and B atoms, whereas the end-member described by $G_{A:B:A:B}$ has two nearest neighbour bonds between A and B atoms and two next nearest neighbour bonds. Many parameters thus have a relation:

$G_{A:B:B:B}=G_{B:A:B:B}=G_{B:B:A:B}=G_{B:B:B:A}$

$G_{B:A:A:A}=G_{A:B:A:A}=G_{A:A:B:A}=G_{A:A:A:B}$

$G_{A:A:B:B}=G_{B:B:A:A}$

$G_{A:B:A:B}=G_{A:B:B:A}=G_{B:A:A:B}=G_{B:A:B:A}$

$L_{A,B:*:*:*}=L_{*:A,B:*:*}=L_{*:*:A,B:*}=L_{*:*:*:A,B}$

The $G_{A:B:B:B}$ term represents $D0_3$, $G_{A:A:B:B}$ B2 and the $G_{A:B:A:B}$ B32 ordering. There are also two kinds of reciprocal interaction parameters:

$L_{A,B:A,B:*:*;0...9}$

$L_{A,B:*:A,B:*;0...9}$

Kusoffsky et al. (2001) have shown the influence of different parameters available in the 4SL model applied to fcc ordering. The possibility to use the 4SL model to BCC ordering has been studied by Sundman et al. (2009) in the Al-Fe system.

2SL Model for FCC, HCP and BCC

The reason for using a 2SL model instead of a 4SL model is that the calculations are faster, but on the other hand it can only model one kind of ordered phase(s). For instance, it is not possible to model FCC with both $L1_0$ and $L2_1$, or BCC with $D0_3$ and B2, etc.

In order to get the fully disordered phase to form, constraints are needed as for the 4SL model. For the symmetric phases, e.g. B2 and $L1_0$, the relation between parameters are:

$G_{A:B}=G_{B:A}$

$L_{A,B:*}=L_{*:A,B}$

For a 2SL asymmetric model including, but not limited to, $L1_2$, more constraints are needed. This model has many complicated relations between the parameters. The relation between the parameters for the 2SL $L1_2$ model can be derived from a 4SL model. The conversion from 4SL model parameters to 2SL parameters for $L1_2$ can be found in Dupin et al. (2001) and for $L1_0$ in Yuan et al. (2012). Higher-order systems with an $L1_2$ phase modelled with two sublattices require a lot of ternary and some quaternary interaction parameters in order to make the disordered state stable. These parameters have been derived by e.g. Dupin (1995), also found in Kusoffsky et al. (2001).

Many publications exist on how to model order/disorder transformations using a two-sublattice model, see e.g. Dupin, Ansara (1999), De Keyzer et al. (2009).

Magnetic Ordering

The magnetic contribution to the Gibbs energy is given by a model proposed by Inden (1975) and adapted by Hillert and Jarl (1978), and is described by:

```
G=RTln(b+1)f(t)
```

where $t=T/T_C$.

The values of the Curie temperature, T_C, and Bohr magneton number, b, at a certain composition are then used in an optimization procedure to calculate the contribution to Gibbs energy due to magnetic ordering. Curie temperature and Bohr magneton number are given in the command *ENTER_PARAMETER* on page 99 for phases that undergo a magnetic transition:

```
AMEND_PHASE_DESCRIPTION BCC_A2 MAGNETIC -1.0 0.4

ENTER_PARAMETER TC(BCC_A2,FE:VA;0) 298.15 +V1; 6000 N

ENTER_PARAMETER BMAGN(BCC_A2,FE:VA;0) 298.15 +V2; 6000 N
```

The arguments -1 and 0.4 specify the values of parameters in the magnetic model: -1 is the antiferromagnetic factor and 0.4 is the structure factor. Usually the antiferromagnetic factor =-1 for bcc and =-3 for fcc (and other non-bcc phases). The structure factor corresponds to the short range order fraction of the enthalpy. This factor is 0.4 for bcc and 0.28 for non-bcc phases.

Molar Volumes and High Pressure

Including volumes in a thermodynamic database enable calculations of volume changes, volume fraction, lattice mismatch in cubic structures and phase diagrams at increased pressure. Pressure independent volumes can be used up to about 1 GPa. At pressures above 1 GPa the pressure dependence of the volumes must be included by using a proper equation of state (EOS). An EOS has been developed by Lu et al. (2005), based on the EOS from Jacobs and Oonk (2000). Four parameters are used in the implementation of the model in Thermo-Calc:

- `V0`: Volume at 1 bar and at reference temperature T_0
- `VA`: Integrated thermal expansivity
- `VK`: Isothermal compressibility at 1 bar
- `VC`: Parameter to fit high pressure data

All parameters can be composition dependent in the same way as all other model parameters in Thermo-Calc. Experimental information about the temperature, pressure and composition dependence of the volume can be assessed using these model parameters. To evaluate V0, VA and VK experimental data at 1 bar only are needed.

Example

Examples on modelling molar volume in a setup-file:

```
ENTER_PARAMETER V0(BCC_A2,FE:VA;0) 298.15 +V1; 6000 N
ENTER_PARAMETER VA(BCC_A2,FE:VA;0) 298.15 +V2*T+V3*T**2; 6000 N
ENTER_PARAMETER VC(BCC_A2,FE:VA;0) 298.15 +V4+V5*T; 6000 N
ENTER_PARAMETER VK(BCC_A2,FE:VA;0) 298.15 +V6+V7*T; 6000 N
ENTER_PARAMETER V0(FCC_A1,CR,FE:VA;0) 298.15 +V8; 6000 N
ENTER_PARAMETER VA(FCC_A1,CR,FE:VA;0) 298.15 +V9*(T-298.15); 6000 N
```

Excess Models

Unless specifically amended, the excess Gibbs energy terms are modelled by the Redlich-Kister-Muggianu Model. Other excess models for the Gibbs energy are also implemented in Thermo-Calc, such as e.g. Kohler and Toop-Kohler extrapolation models.

Thermo-Calc is not restricted to binary interaction parameters. Ternary, quaternary and higher-order interaction parameters can also be used if there is enough information to evaluate these from experimental data. The ternary interaction parameters are identified by the degrees of the so-called ternary L parameter (i.e. 0, 1 and 2) in the command *ENTER_PARAMETER* on page 99. If the parameter is composition-independent, you only need a single interaction parameter with a degree of zero, i.e. only the ${}^{0}L$ term. If there is a ternary composition-dependent interaction parameter, you must enter all three parameters (0,1 and 2). If such a ternary interaction parameter should be zero, then assign it a function which is zero. If the three interaction parameters have the same value, then that is identical to having a composition-independent ternary parameter.

Example

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;1) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;2) 298.15 +1000; 6000 N
```

is identical to

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
```

This also means that

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;1) 298.15 +ZERO#; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;2) 298.15 +ZERO#; 6000 N
```

is NOT identical to

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
```

Database Manager User Guide

Version 2016a

Thermo-Calc Software

Introduction to the Database Manager Guide

In this section:

Thermodynamic and Kinetic/Mobility Databases

This guide describes how to define and construct a thermodynamic or kinetic database. It includes details about the data structures and formats of the Thermo-Calc databases as well as the Diffusion Module (DICTRA) database-extensions.

The Thermo-Calc databases are created and maintained in the Thermo-Calc Database Format (TDB) which is an international standard for the CALPHAD-type thermodynamic calculations and kinetic simulations. This document gives a full description of the TDB format (with the Diffusion Module (DICTRA) database extension) and is a comprehensive guide about how to create and manage a Thermo-Calc (and Diffusion Module (DICTRA)) database.

Usually, a Thermo-Calc or Diffusion Module (DICTRA) database is constructed as a single text file with an extension of TDB, for example, `PAQS2setup.TDB`. For some databases (especially large databases used with older versions of Thermo-Calc or Diffusion Module (DICTRA)) a database may consist of one primary-definition file (`***setup.TDB`) and several additional-definition TDB files (such as `***param.TDB`, `***funct.TDB`, `***refer.TDB`, etc. These are built for a variety of defined parameters, functions, references and so forth, and are linked to the `***setup.TDB` file through the correct use of the *TYPE_DEFINITION* on page 18 and/or *FTP_FILE* on page 21 commands inside the `***setup.TDB` file.

With Thermo-Calc, Precipitation Module (TC-PRISMA) and Thermo-Calc Software Development Kits (SDKs), you can code everything into a single *****setup.TDB** file for a database at any scale.

Initialization of the DATA Module

The following explains how a database initiation file (or a database directory file) is constructed to work with the DATA module. When you start, DATA looks for this file that has information about the available predefined databases.

The *database initiation file* is called:

- Windows: `TC_INITD.TDB` (or `TC_INITD`)
- Linux/UNIX: `initd.tdb`

You can have your own file, but it is more efficient to define an environment variable with the default name as above, and then during installation, translate it to a common initiation file name. The common file is automatically copied by the Thermo-Calc installation script to the following directories:

- Windows: `\DATA\` under the directory defined by the `TCPATH` parameter
- Linux/UNIX:`/data/` under the directory defined by the `TC_DATA` parameter.

The database manager can find the file on a local computer (independent installations) or a connected server (server installations).

When editing or modifying the definitions of initiation parameters (including short-names, paths and subdirectories, database definition file names, and instructive database descriptions) for the available databases in the database initiation file a specific format is used with the fields:

- The first field gives the abbreviated name for the database (to a maximum of five characters)
- The second field specifies a path and filename (maximum 78 characters) containing the database path definition and the actual database definition (*setup*) file name, where the extension is either `TDB` (which stands for original textual TDB file) or `TDC` (which is encrypted from the original TDB file). There are differences between platforms with respect to entering parameters as listed in the table.
- The third field (maximum 60 characters) details an optional full name (and version/subversion numbers) for the brief database description.

Initiation Parameter Examples

Parameter	Windows	Linux/Unix
The `setup` file name must consist of a main part and an extension of a maximum of three characters:	For example, `SSOL5SETUP.TDB`	For example, `ssol5setup.tdb`
Database path definition	Use a back-slash `\`	Use a forward-slash `/`
All databases should be normally located in subdirectories that are under the directory defined by the:	`TCPATH` parameter or under its subdirectory `\DATA\`	`TC_DATA` parameter or under its subdirectory `/data/`

Each entry record (i.e. for a single database) must finish with an exclamation mark !. There should be no spaces or commas within the first and second fields since these characters are taken as field separators, while spaces or commas can be used in the third field.

You can write each database entry on one or two lines (up to the!), it is recommended that the third field is added as follows:

- Windows: on the same line as the first and second fields.
- Linux/UNIX: on the following line.

The following are examples of an initiation file defining two databases named SSOL6 and TCFE8.

Windows

```
SSOL6 TCPATH\DATA\SSOL6\SSOL6SETUP.TDC  SGTE Solutions Database
version 6.0 !

TCFE8 TCPATH\DATA\TCFE8\TCFE8SETUP.TDC  TCS Steels/Fe-Alloys Database
version 8.0 !
```

Linux/UNIX

```
ssol5 TC_DATA/data/ssol5/ssol5setup.tdc

     SGTE Solutions Database version 5.1  !

Tcfe8 TC_DATA/data/tcfe8/tcfe8setup.tdc

     TCS Steels/Fe-Alloys Database version 8.0 !
```

The Database Definition File and Keywords

In this section:

Database Definition File Syntax

The database definition (*.TDB) file (normally named `***setup.TDB`) consists of a set of keyword codes each followed by one or several parameters (arguments).

- A complete keyword entry must end with an exclamation mark (!).
- A single keyword entry can be up to 2000 characters long. However, the maximum length of a line in an *.TDB file is 78 characters--it may be necessary to continue the keyword parameter (arguments) on several lines. The ! must be at the end of the last line.
- It is recommended to always have at least one empty space at the beginning of each continuation line for the keyword parameters (arguments); otherwise, the DATA module can misunderstand the parameters (or arguments), or issue an error message.
- The keyword and its various parameters (arguments) are separated by a space or a comma.
- A dollar sign ($) in the first position of the line indicates that the line is a comment line, which is ignored by the DATA module.

When the database is selected the DATA module only reads the database definition file (***setup.TDB) once, from beginning to end. The DATA module checks continuously when reading the definition file. This implies that (nearly) everything must be declared or defined before it is used in any other way.

For example, if the GRAPHITE phase is to be included in the database definition, the element C (carbon) and the phase GRAPHITE must be defined before declaring that carbon dissolves in graphite. This definition order is necessary to build the internal data structure acceptable by the DATA module (during its consistency checking).

This topic gives a description of the available keywords and the appropriate arguments. A basic knowledge of the Gibbs Energy System (GIBBS) module is assumed.

This syntax is used:

```
KEYWORD [arg.1]*# [arg.2]*## {optional arg.3}!
```

The keywords are written in full length but can be abbreviated as long as the abbreviation is unique. A keyword may have syntax consisting of several arguments and optional arguments. The number, # or ##, in the notation, [...]*# or [...]*##, indicates an argument with a maximum length of # ASCII characters.

> Arguments within square brackets `[...]` must always be given, but are optional when enclosed in curly brackets `{...}`.

ELEMENT

```
ELEMENT [element name]*2 [ref. state]*24 [mass] [H298] [S298] !
```

The *element name* (maximum two characters) is the one found in the periodic chart but there are no naming restrictions. However, the GIBBS module only recognizes UPPER-case element names (if the `Upper Case Mode` is selected by the GIBBS command *REINITIATE* on page 83), which means that lower-case (if defined in a database) is automatically converted to UPPER-case by the DATA/GIBBS module.

The elements are automatically entered as species using the same names of the elements. If, for example, the species corresponding to FE needs to be named FE1, you can define the species as FE1, which results in an element named FE and a species named FE1. Vacancies (VA) and electrons (denoted either as /- in gaseous, liquid or solid phases, or ZE in an aqueous solution phase), need to be entered as special elements for correct handling by the DATA module.

The *reference (ref.) state* (maximum 24 characters) is the stable phase (at 298.15 K and 1 bar) that should contain this element and be used as the reference state for all element thermodynamic data. The *mass*, given in gram per mole, is used in various calculation programs and should always be given the correct value. *H298* and *S298* denote the enthalpy and entropy difference between 0 and 298.15 K for the element in SI units. If these are unknown, the values can be set to zero. All this information (reference state, H298 and S298) define the SER (Stable Element Reference state).

Examples

```
ELEMENT /- ELECTRON_GAS       0.0   0.0   0.0 !
ELEMENT VA VACUUM             0.0   0.0   0.0 !
ELEMENT ZE UNIT_CHARGE        0.0000000001  0.0  0.0 !
ELEMENT AL FCC_A1             26.98154 4577.296 28.3215 !
ELEMENT C  GRAPHITE           12.011  1054.0    5.74 !
ELEMENT FE BCC_A2             55.847  4489     27.28 !
ELEMENT O  1/2_MOLE_O2(G)     15.9994 4341    102.5158 !
ELEMENT TI HCP_A3             47.88   4810     30.648 !
ELEMENT ZR HCP_A3             91.224  5566.27  39.181 !
ELEMENT ZY DUMMY               1         1         1 !
```

SPECIES

```
SPECIES [species name]*24 [stoichiometric formula] !
```

This keyword defines species in the data structure. Every *species name* (maximum 24 characters) must be unique. The species are built from the predefined set of elements in the stoichiometric formula. If an undefined element is referenced, DATA displays an error message and the data structure is probably damaged.

The species names do not have to be the same as the stoichiometry formula, although in general this is recommended. The elements are automatically entered as species using the same names of the elements.

You can define a species name as a mixture of UPPER-case and lower-case letters in a database, but the DATA module automatically converts all lower-case to UPPER-case because the GIBBS module only recognizes UPPER-case species names.

When naming the species in a database, use special characters (such as +, -, _, / and .) in species names. Avoid using other special characters (such as `(` and `)`).

The *stoichiometric formula* is written with a simplified chemical notation, in which the chemical elements should always be given in UPPER-case and in any preferred order, and the stoichiometric coefficients are written in either real numerical factors or integer digits.

It is important that the numerical factor of `1` is not left out. Subgroups are not allowed in a stoichiometry formula; however, while specifying the stoichiometry formula for a specific species in a database, you can specify it in a way that some elements (always together with the corresponding partial stoichiometric coefficients) are repeated (as in the examples below).

Examples

```
SPECIES AL2O3       AL2O3 !
SPECIES Silica      SI1O2 !
SPECIES NaSb_6OH    NA1SB1O6H6 !
SPECIES FE+2         FE/+2 !
SPECIES SB-3         SB/-3 !
SPECIES AlCl2/3             AL.33333CL.666667 !
SPECIES AL1CL1H2O2          AL1CL1H2O2 !
SPECIES AlCl3_3H2O          AL1CL3H6O3 !
SPECIES AlO2H2Cl.H6O3       AL1O2H2CL1H6O3 !
SPECIES AlCl2-OH.3H2O       AL1CL2O1H1H6O3 !
SPECIES AlCl2OH.3Water      AL1O1H1CL2H6O3 !
```

PHASE

```
PHASE [phase name]*24 [data-type code]*8 [numb. subl.] [sites in
subl. 1] [sites in subl. 2] etc... {auxiliary text string} !
```

This keyword defines a phase and its properties (except for what species are allowed to enter it and for its parameters).

The *phase name* (maximum 24 characters) must be unique; otherwise the DATA module sees it as an attempt to redefine a previously defined phase. This causes DATA to display an error message and ignore the rest of the line. A phase name can be suffixed by an underscore (_and letters to identify the physical state(s) or structure type(s) of the phase.

Examples of Recommended Suffixes

Suffix	Definition
ABC_S	The ABC phase in solid state.
ABC_S2	The ABC phase in solid state 2.
ABC_S3	The ABC phase in solid state 3.
ABC_LT	The ABC phase in solid state at low temperatures.
ABC_HT	The ABC phase in solid state at high temperatures.
ABC_L	The ABC phase in liquid state.
ABC_LIQ	The ABC phase in liquid state.
FCC_A1	The FCC phase in disordered structure type A1.
FCC_L12	The FCC phase in ordered structure type L12.

The *phase name* can also be attached with a colon sign (:) and a letter for a legal GIBBS phase-type code (e.g. IONIC_LIQ:Y and GAS:G).

GIBBS Phase-type Codes

Code	Definition
G	Bit set for a gaseous mixture phase.
A	Bit set for an aqueous solution phase.
Y	Bit set for an ionic liquid solution phase (specially treated by the Ionic Two-Sublattice Liquid Model).
L	Bit set for a liquid solution phase (but not A (aqueous) or Y (ionic liquid)).
I	Bit set for a phase with charged species (but not G (gaseous), A (aqueous) or Y (ionic liquid)).
F	Bit set for an ordered FCC or HCP solution phase using the *Four Substitutional-Sublattice Ordering Model* (additionally, such a phase can also have interstitial sublattices).
B	Bit set for an ordered BCC solution phase using the *Four Substitutional-Sublattice Ordering Model* (additionally, such a phase can also have interstitial sublattices).

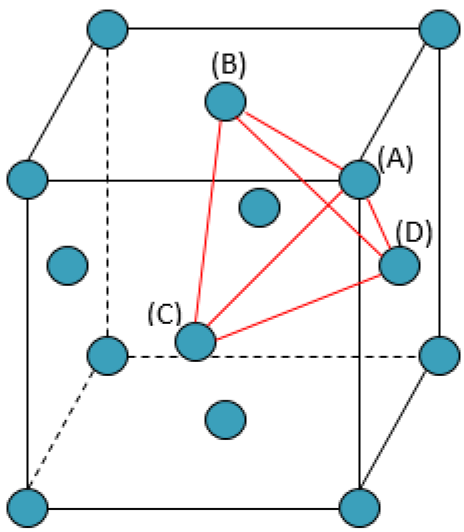
Other invalid characters (e.g. M or P) are eventually treated, together with the colon (:) as a part of a phase name.

A G phase (gaseous mixture) or an A phase (aqueous solution) is usually treated as a substitutional phase without sublattice, and an L phase (ordinary liquid solution) is normally (but not always) modelled as a substitutional phase without sublattice, too.

For ordered FCC or HCP phases, these four substitutional sublattices represent four corners of the regular tetrahedron on these lattices, all of which are the nearest neighbours, as shown.

FCC Unit Cell Example

An FCC unit cell with the lattice positions indicated that correspond to the G (FCC,A:B:C:D) end member. All lattice positions are equivalent for a four substitutional-sublattice ordering model.



A *Normal 4-Sublattice Model* requires that all the G parameters for each of the end-members with the same elements but distributed on different sites be given separately. However, as these corners are identical lattice points, the phase-type option F means that the G parameters need be given only once. The possible permutations are handled automatically.

Additional Clarification

An A–B binary solution phase (with the element A located on one sublattice site and B on the other three sublattice sites) treated by the Normal 4-Sublattice Model has to have four G parameters for four end-members, i.e.

- G(phase,A:B:B:B)
- G(phase,B:A:B:B)

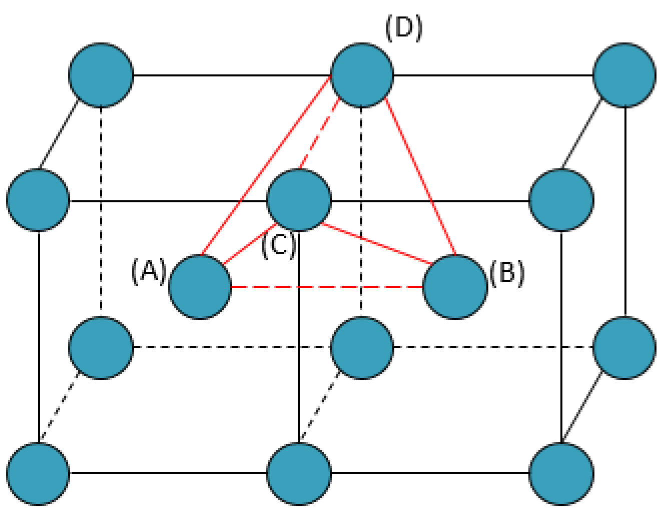
- G(phase,B:B:A:B, and
- G(phase,B:B:B:A)

This is because in the general case these `G` parameters can be different from each other. But for the FCC and HCP orderings, they are identical and thus all `G` parameters of such end-members need to be given only once, and the possible permutations are then automatically handled by the GIBBS module. Also, only one of the identical permutations is listed; in this example, `G(phase,A:B:B:B)` where it is alphabetically the first in the list of permutations. This significantly simplifies the usage of this model (*Four Substitutional-Sublattice Ordering Model*) in multicomponent alloys.

For ordered BCC phases, the phase-type option `B` means the same thing but it is more complicated since the 4-substitutional-sublattice ordering phase represents an irregular tetrahedron with two pairs of sites that are next nearest neighbours as shown:

BCC Unit Cell Example

Two BCC unit cells with the lattice positions indicated that correspond to the G (BCC,A:B:C:D,0) end member. Lattice positions (A) and (B) are equivalent, as are lattice positions (C) and (B) for a four substitutional-sublattice ordering model.



For an end member described by the parameter `G (phase,A:B:C:D)` A and B are next nearest neighbours, as are C and D. And the nearest neighbours of A (or B) are C and D. Thus, for an `A–B` binary solution phase (with the element A located on two sublattice sites and `B` on two sublattice sites) treated by the Normal 4-Sublattice Model, the end-member described by the `G(phase,A:A:B:B)` term has four nearest neighbour bonds between A and B atoms, whereas the end-member described by the `G (phase,A:B:A:B)` term has two nearest neighbour bonds between A and B atoms and three next nearest neighbour bonds.

The first end-member (described by the `G(phase,A:A:B:B)` term) represents B2-ordering and the second (described by the `G(phase,A:B:A:B)` term) stands for B32-ordering. There are two permutations of the `G(phase,A:A:B:B)` term and four permutations of the `G(phase,A:B:A:B)` term, automatically conducted in the *Four Substitutional-Sublattice Ordering Model*. If you enter the unary, binary, ternary and quaternary parameters you are dealing with 1, 6, 21, 55 parameters for BCC:B, 1, 5, 15, 35 parameters for FCC:F and 1, 16, 81, 256 parameters for phases without F/B.

An additional feature with the phase-type options `F` and `B` is that a composition set that represents the solution phase has a suffix (indicating what ordering the phase has) that is automatically added to its phase name in some listings of equilibrium calculations (when performing either single-point or stepping or mapping calculations, and when plotting the calculated property diagrams or phase diagrams).

Such suffix indications can be:

Solution Phase	**Suffix for Disordered Phase**	**Suffix for Ordered Phase**
`FCC PHASE`	`FCC_A1`	`FCC_L12`
		`FCC_L10`
`BCC PHASE`	`BCC_A2`	`BCC_B2`
		`BCC_B32`
		`BCC_D03`
		`BCC_L21`
`HCP PHASE`	`HCP_A3`	`HCP_D019`
		`HCP_B19`

If you want to convert an existing database TDB-file to use the F/B feature, add the phase-type code to the corresponding phase name in the PHASE and CONSTITUENT commands in the TDB-file, then when running the LIST_DATA command in the GES module. The created database file is in this less verbose format.

The *data-type code* consists of 1 to 8 characters where each character must stand for an action, which is to be coupled to this phase. The keyword *TYPE_DEFINITION* on page 18, described below, must be used in the current database to specify what action should be taken by DATA for each character code.

The data entries `[numb. subl.] [sites in subl. 1] [sites in subl. 2]` etc., specify the total number of sublattices (always as an integer digit) and the sites (i.e. stoichiometric coefficients) of each of the sublattices (given in either integer digits or real numerical factors) for the phase.

Optionally, an *auxiliary text string* (maximum 78 characters) can be given after the last [sites in sublattice #] but before the exclamation mark `!`. This string displays in connection with the phase name in some listings within the DATA module.

Examples

```
PHASE GAS:G  % 1 1.0 !
```

```
PHASE LIQUID:L %ZCDQ  2  1.0  1.0

> Metallic liquid solution, modelled by CEF Model. !

PHASE IONIC-LIQ:Y %ZCDQ  2  1.0  1.0

> Ionic liquid solution, modelled by Ionic Two-Sublattice Model. !

PHASE SPINEL:I    %ZA   4   1  2  2  4

> Complex Spinel Solution, by CEF model with ionic constraints. !

PHASE M23C6  % 3 20.0 3.0 6.0 !

PHASE FCC_A1    %&A  2   1 1

> Disordered FCC phase; also as MX carbides/nitrides. !

PHASE FCC_L10   %&AX  3  0.75 0.25 1

> Ordered FCC phase, modelled by 2-Sublattice Model for Ordering. !

PHASE FCC_L12:F %&AX  5  0.25 0.25 0.25 0.25 1.0

> Ordered FCC phase, modelled by 4-Sublattice Model for Ordering. !

PHASE AQUEOUS:A %HIJMR 1 1.0

> Aqueous Solution: using the Complete Revised HKF Model. !
```

CONSTITUENT

```
CONSTITUENT [phase name]*24 [constituent description]*2000 !
```

This keyword (and the *ADD_CONSTITUENT* on the next page keyword for large solution phase) defines the phase-constitution as a list of constituents (for a substitutional phase with no sublattice) or of constituent arrays (for a sublattice phase).

The *phase name* (maximum 24 characters) must be a predefined phase (i.e. already through the *PHASE* on page 9 keyword).

It is important that if a phase bears a legal phase-type (among G, A, Y, L, I, F and B) in its phase definition (already by the PHASE keyword; such as GAS:G, SLAG:L, LIQUID:L, IONIC_LIQ:Y, SPINEL:I, FCC_L12:F, HCP_D021:F, BCC_B2:B, AQUEOUS:A), such a valid phase-type code must also always be attached to the phase name in the CONSTITUENT keyword (and the ADD_CONSTITUENT keyword).

Specifying the phase name in UPPER-case is recommended. You can define a phase name as a mixture of UPPER-case and lower-case letters in a database, but the DATA module automatically converts all lower-case to UPPER-case because the GIBBS module only recognizes UPPER-case phase names.

The *constituent description* (maximum 2000 characters) is a list of the species that enter a phase. The list starts with a colon (`:`), indicating the start of the sub-list of species for the first sublattice, and different sublattices are separated by colons. The complete sequence ends with a final colon.

Optionally, each sublattice may specify which species are considered to be major constituents. This is done by adding a percent sign (`%`) directly to the species name. The start values on the site fractions of the major constituents should sum to 0.99 on a specific sublattice. Thus, the minor constituents (i.e. those without a `%`) add up to 0.01. A maximum of 2000 characters can be coded in the constituent description, continuing in sequent lines. If the phase has a constituent description longer than 2000 characters, the rest can be coded in one or several ADD_CONSTITUENT keywords.

Examples

```
CONSTITUENT BCC_A2 :FE

CONSTITUENT IONIC-LIQ:Y :FE+2 : SB-3: !

CONSTITUENT M23C6 :CR FE :FE CR W MO : C: !

CONSTITUENT AQUEOUS:A :H2O% AG+1 AGF AGCL AGCL2-1 AGI3-2 AGSO4-1
AGC2H4+1

   AGN2H6+1 AGC2N2-1 AGC2H4NO2 AL+3 ALF3 ALO2-1, ... : !

CONSTITUENT SPINEL:I  : AL+3% CR+3 FE+2% FE+3 MG+2% NI+2

                     : AL+3% CA+2 CR+3 FE+2 FE+3 MG+2% NI+2 VA

                     : FE+2 MG+2 VA%

                     : N-3 O-2% :!
```

ADD_CONSTITUENT

```
ADD_CONSTITUENT [phase name]*24 [constituent description]*2000 !
```

This keyword adds more constituents to a phase that already has some constituents. Its syntax is the same as for the *CONSTITUENT* on the previous page keyword. This keyword can be used several times, if the phase is very large, e.g. a gaseous mixture or a complex aqueous solution. This is useful when there are so many constituents in a phase that the 2000 characters available for the constituent description list is not enough.

Constituents are not necessary on all sublattices. In the second example below, no addition is made to the first sublattice.

Examples

```
ADD_CONSTITUENT GAS :S1 S2 S3 ... : !

ADD_CONSTITUENT IM-PHASE : :CR:W ... : !

ADD_CONSTITUENT AQUEOUS:A :CUCL+1 CUCL2 CUCL2-1 CUCL3-2 CUOH+1 CUO2H2
CUO3H3-1
```

```
    CUO4H4-2 CU2OH+3 CU2O2H2+2 CU3O4H4+2 NIO2H2 NIO3H3-1 NIO4H4-2
NI2OH+3
    NI4O4H4+4 ZNOH+1 ZNO2H2 ZNO3H3-1 ZNO4H4-2 ... : !
```

COMPOUND_PHASE

```
COMPOUND_PHASE [phase name]*24 [data-type code]*8 [constituent] !
```

The keyword is a compact way to simultaneously define a species, a *compound phase* (maximum 24 characters) and its phase-constituent. It is useful for stoichiometric phases with constant compositions. The species name and stoichiometric formula must be identical, i.e. being the given *constituent*. The phase has this species as its only constituent. This keyword allows the database definition file for a large substance database to be more compact; it is a combination of the *SPECIES* on page 8, *PHASE* on page 9 and *CONSTITUENT* on page 14 keywords.

Examples

```
COMPOUND_PHASE AL2O3 % AL2O3 !
COMPOUND_PHASE MAGNETITE %MF FE3O4 !
COMPOUND_PHASE QUARTZ % SIO2 !
```

ALLOTROPIC_PHASE

```
ALLOTROPIC_PHASE [phase name]*24 [data-type code]*8 [constituent] !
```

This keyword does the same as the *COMPOUND_PHASE* above keyword for entering an allotropic phase (maximum 24 characters), but does not enter the constituent as a species to the data structure. Use this if the species is already defined.

Examples

```
ALLOTROPIC_PHASE BETHA-AL2O3 % AL2O3 !
ALLOTROPIC_PHASE CRISTOBALITE % SIO2 !
ALLOTROPIC_PHASE TRIDYMITE % SIO2 !
```

TEMPERATURE_LIMITS

```
TEMPERATURE_LIMITS [lower limit] [upper limit] !
```

This keyword sets the default upper and lower temperature limits used by the GIBBS module for Gibbs energy parameters and functions. It can be used only once in one database definition file and all its sequential files.

Example

```
TEMPERATURE_LIMITS 500.0 1800.0 !
```

DEFINE_SYSTEM_DEFAULT

```
DEFINE_SYSTEM_DEFAULT [keyword] {G-ref. type index} !
```

This keyword sets the default value to ELEMENT or SPECIES in the DATA command *DEFINE_SYSTEM* on page 15 in the *Thermo-Calc Console Mode Command Reference*.

For a substance database, it can be appropriate to have ELEMENT as a default value whereas a large solution database can benefit from having SPECIES as a default value. A proper default value is useful for beginners. An advanced user is more likely to use the DATA commands DEFINE_ELEMENT and DEFINE_ SPECIES to override the default value.

{G-ref. type index} is an integer indicating the reference state type for an element when entering and listing data in the GIBBS module. The following lists legal numbers and the corresponding meaning (the reference state type for an element):

Number	Definition
1	symbol: G
2	symbol: H298
3	symbol: H0

Example

```
DEFINE_SYSTEM_DEFAULT element 2 !
```

DEFAULT_COMMAND

```
DEFAULT_COMMAND [secondary keyword and parameters] !
```

This keyword specifies commands to be executed by the DATA module at database initialization. The syntax of the available command is currently not the same as the user available DATA commands but the actions are similar. The available *secondary keyword and parameters* in syntax are:

```
DEFINE_SYSTEM_ELEMENT [element names]
DEFINE_SYSTEM_SPECIES [species names]
DEFINE_SYSTEM_CONSTITUENT [phase] [sublattice] [species]
REJECT_SYSTEM_ELEMENT [element names]
REJECT_SYSTEM_SPECIES [species names]
REJECT_SYSTEM_CONSTITUENT [phase] [sublattice] [species]
REJECT_PHASE [phase names]
RESTORE_PHASE [phase names]
```

Examples

```
DEFAULT_COMMAND DEFINE_SYSTEM_ELEMENT FE VA !
DEFAULT_COMMAND REJECT_SYSTEM_CONSTITUENT LIQUID 2 C !
```

```
DEFAULT_COMMAND REJECT_PHASE LIQUID !
DEFAULT_COMMAND RESTOR_PHASE GAS !
```

DATABASE_INFORMATION

```
DATABASE_INFORMATION [text]*10000 !
```

This keyword defines a text for the detailed description of the current database. The text can be listed with the DATA command DATBASE_INFORMATION. An apostrophe (`'`) can be used in the text to indicate a new line; and two apostrophes, (`''`) can be used in the text to indicate a new line plus an empty line.

The continuous *text* length (each line with max 78 characters) is 10,000 characters.

Example

```
DATABASE_INFORMATION This is the XXX-Alloy Solution Database '
      in the A-B-C-D-...... System. '
   Developed by TCS, released in May 2001. ''
... more ... !
```

TYPE_DEFINITION

```
TYPE_DEFINITION [data-type code]*1 [secondary keyword with
parameters] !
```

This keyword couples phases to an action performed by the DATA module when the DATA command *GET_DATA* on page 15 is executed.

Secondary Keywords

The secondary keywords and associated parameters in syntax are:

```
SEQ [filename]
RND# [filename]
GES [valid GIBBS command with parameters]
POLY3 [valid POLY command with parameters]
TDB [valid DATA command with parameters]
IF [conditional statement] THEN [keyword with parameters]
AFTER [valid GIBBS command with parameters]
```

The secondary keyword `SEQ` specifies a sequential file that stores parameters belonging to the phases using the associated data type code (which is defined by this keyword). A special case where the filename is given as an asterisk (`*`) implies that the database definition file also acts as a sequential data storage file. This case makes it possible to have a single file for a small database, which is especially suited for personal databases.

The secondary keyword `RND` should be concatenated with a positive integer `#` to indicate the type of the random file. Currently, there are these types of random files:

- `RND0`, the default, is used for complete Gibbs energy expressions (G0 parameters), where the search field is the unabbreviated parameter name.
- `RND1` is designated for functions, where the function name is used as the search field.
- `RND2` is reserved for binary interaction parameters, where its search field is also the unabbreviated parameter name without any interaction order notation.

Ternary and higher order interaction parameters must be specified on a sequential file. Moreover, the internal structures of these random files are subject to changes with different versions of DATA, and with implementations of DATA on various computer systems. For more information, see the FORTRAN program TDBSORT, which is available from Thermo-Calc Software AB.

The secondary keywords `GES`, `POLY3`, or `TDB` specifies a modification of, or addition to, phases having the associated data type code, such as magnetic contribution, another excess model, or any other valid GIBBS/POLY/DATA command that applies to a certain phase. By implementing this as a call to the interactive GIBBS/POLY/DATA module, flexibility is achieved. If a new type of addition is implemented in a GIBBS/POLY/DATA module, it can be immediately used in the database definition file without reprogramming the DATA module.

In several examples below, the use of the at (`@`) sign indicates any phase to which the relevant type definition (e.g. `A`, `B`, `4`, or `E`) applies.

The secondary keywords `IF` and `THEN` allow specification of a conditional statement structured with respect to the phase constitution that controls the execution of a following type-definition (keyword with parameters) string. See the last four examples.

The secondary keyword `AFTER` is similar to the GES keyword except the defined GIBBS command is executed after all parameters are entered. The reason for this is that the command has no effect unless there is a parameter. Following the `AFTER` keyword, a GIBBS command must be given and it is executed after entering the parameters of the phase.

The data-type code (always as one string) can be any normal or special character, e.g. `0`, `5`, `A`, `F`, `M`, `%`, `&`, and so forth, and is referred in the definition keywords *PHASE* on page 9, *COMPOUND_PHASE* on page 16 and *ALLOTROPIC_PHASE* on page 16 for various phases.

A phase can have several ADDITIONAL parts of different types (that are enforced by certain TYPE_DEFINITIONS which call the GIBBS command AMEND_PHASE_DESCRIPTION for describing various contributions to Gibbs energy). However, the DATA/GIBBS module automatically deletes any early-defined ADDITIONAL part(s) of the same type.

The DATA module can selectively retrieve functions that are necessary for a defined system from a database that has functions stored in its `setup` file or `SEQ` sequential function file, while all other functions irrelevant for the defined system are ignored and are not saved in associated GIBBS and POLY workspaces.

Examples

```
TYPE_DEF % SEQ  TCPATH\DATA\[DATABASE]\PARAMETERS.TDB !
TYPE_DEF I SEQ  TCPATH\DATA\[DATABASE]\INTERACTION-PARAMS.TDB !
TYPE_DEF G RND0 TCPATH\DATA\[DATABASE]\GZERO-PARAMS.TDB !
TYPE_DEF F RND1 TCPATH\DATA\[DATABASE]\FUNCTIONS.TDB !
TYPE_DEF & RND2 TCPATH\DATA\[DATABASE]\BINARY-INTERACTIONS.TDB !
TYPE_DEF A GES AM_PH_DES @ MAGNETIC -1 0.40 !
TYPE_DEF B GES AM_PH_DES @ MAGNETIC -3 0.28 !
TYPE_DEF 4 GES AM_PH_DES @ EXCESS_MODEL REDLICH-KISTER_KOHLER !
TYPE_DEF 5 GES AM_PH_DES AQUEOUS EXCESS_MODEL HKF !
TYPE_DEF 6 GES AM_PH_DES AQUEOUS HKF_ELECTROSTATIC !
TYPE_DEF 7 GES AM_PH_DES AQUEOUS STATUS 02084000,,, !
TYPE_DEF 8 GES AM_PH_DES AQUEOUS MAJOR_CONST 1 H2O !
TYPE_DEF E AFTER AM_PH_DES LIQUID EXCESS MIXED-EXCESS
                            A B LEGENDRE   C A POLYNOM ,,,, !
TYPE_DEF T AFTER AM_PH_DES LIQUID TERN-EXT TOOP-KOHLER  B A C ,,,, !
TYPE_DEF Q AFTER AM_PH_DES LIQUID TERN-EXT KOHLER FE CR NI !
TYPE_DEF C IF (PD AND PT AND SN)  THEN TDB RESTORE_PHASE BCT_A5 !
TYPE_DEF D IF (PD AND (PT OR SN)) THEN TDB REJECT_PHASE BCC_A2 !
TYPE_DEF E IF ((NB OR TI OR V) AND (C OR N)) THEN
          GES AM_PH_DES @ COMP_SET ,, CR NB TI V: C N: !
TYPE_DEF F IF (ALO3/2 OR CRO3/2 OR FEO OR MNO OR SIO2) THEN
          GES AM_PH_DES LIQUID COMP_SET ,, ALN%,ALO3/2%,CRO3/2%,
             FEO%,FEO3/2,MNO3/2%,MNS%,SIO2%,TIO2% : !
TYPE_DEF R GES AM_PH_DES FE_LIQUID FRACTION_LIMITS  Fe 0 0.6
         Ag 0 0.01  Al 0 0.05  Ca 0 0.05  Co 0 0.01  Cr 0 0.01
         Cu 0 0.02  Mg 0 0.05  Mn 0 0.05  Mo 0 0.05  Nb 0 0.05
         Ni 0 0.05  Pb 0 0.05  Si 0 0.10  Sn 0 0.02  Ti 0 0.05
         U  0 0.01  V  0 0.02  W  0 0.02  Zr 0 0.03
         B  0 0.01  C  0 0.01  H  0 0.01  N  0 0.01  O  0 0.01
```

```
P  0 0.01  S  0 0.01  !
```

Ordered Phase Restores Disorder Contribution

When an ordered phase with disordered contribution is restored, the disordered contribution is automatically restored as well. You can also restore the disordered phase *without* restoring the ordered phase.

However the DIS_PART TYPE_DEFINITION must always be set on the ordered phase. It is important that the disordered phase is declared *before* the ordered phase. Otherwise it causes an error when DIS_PART TYPE_DEFINITION is executed and then either the ordered or disordered phase is rejected.

Example

```
TYPE_DEFINITION & GES A_P_D FCC_A1 MAGNETIC  -3.0    2.80000E-01 !
PHASE FCC_A1  %&  2 1   1 !
    CONSTITUENT FCC_A1  :CR,NI% : C%,VA :  !
$ THIS PHASE HAS A DISORDERED CONTRIBUTION FROM FCC_A1
TYPE_DEFINITION ' GES AMEND_PHASE_DESCRIPTION FCC_L12 DIS_PART FCC_
A1,,,!
TYPE_DEFINITION ( GES A_P_D FCC_L12 MAGNETIC  -3.0    2.80000E-01 !
PHASE FCC_L12  %'(  3 .75   .25   1 !
    CONSTITUENT FCC_L12  :CR,NI% : CR,NI% : C,VA% :  !
```

For example a GES command such as the following only affects phases defined above or at the phase with this TYPE_DEFINITION in the TDB file:

```
TYPE_DEFINITION ( GES A_P_D FCC_L12 DIS_PART FCC_A1,,, !
```

For example a TDB command such as the following only affects phases defined *below* the phase with this TYPE_DEFINITION in the TDB file:

```
TYPE_DEFINITION ( TDB RESTORE_PHASE FCC_A1 !
```

FTP_FILE

```
FTP_FILE [filename] !
```

FTP_FILE is a special function random file and the function names correspond to the record numbers where these record names and the functions are stored. The FTP_FILE decreases search time for the associated database in the DATA module. The file is used for large substance databases along with a SEQ sequential or RND0 random file for storage of G0 parameters referring the functions named `FxxxxT` that are stored in the FTP file. The integer number `xxxx` is a search code used by DATA when such files are processed.

No modification of this file type is allowed.

Example

```
FTP_FILE TCPATH\DATA\[DATABASE]\FTP-FILE.DATA !
```

FUNCTION

```
FUNCTION [function name]*8 [lowest temp. limit]
[expression 2]; [upper temp. limit 2] Y
[expression 1]; [upper temp. limit 1] Y
[expression 3]; [upper temp. limit 2] Y
   .......... ; ..... Y
[expression n-1]; [upper temp. limit n-1] Y
[expression n]; [upper temp. limit n] N {Ref. Index} !
```

GIBBS can use predefined functions in the expression (TP-Function) of a Gibbs energy parameter or in other functions. This is often used when several parameters (or functions) have a common sub-expression, such as for metastable modifications of elements. This keyword can appear in both files for database definition and sequential storage, but not in FTP files. A valid *function name* can have up to 8 characters.

A function always starts with a lowest temperature limit of its applicability, followed by one or more (up to 10) expressions (TP-Functions) that are coded as mathematical relations of constants, functions of stable variables (T and P) and other entered functions (normally with a # suffix, e.g. +3*GHSERAL#).

Also see the *ENTER_PARAMETER* on page 99 in the *Thermo-Calc Console Mode Command Reference*.

The expression is a FORTRAN-like expression and operators +, -, *, = and ** can be used (** only with integer powers). Unary-functions LN or LOG (both for natural logarithm) and EXP (for exponential) can also be used. Each expression (TP-Function) should end with a semicolon (;), and be followed by its upper applicable temperature limit and a continuation indicator (Y to continue with the next expression or N to end the function's expression). If there is no continuation after a specific expression (TP-Function), the reference index can optionally be given after the N indicator.

A complete/valid function entry can be written in several continuation lines if the function's expression (TP-Function) is too long or if there is more than one applicable expression (TP-Function), as the maximum length of each line is 78 characters.

It is recommended to always have at least one empty space at the beginning of each continuation line. Otherwise, the DATA module may misunderstand the expression or issue some error messages when reading the function entry. Avoid entering functions like the one below.

```
FUNCTION GHSERXY  298.15
-1000+1058*T-38.9*T*LOG(T)+GFUNXY#; 6000 N !
```

Such a function is read by the DATA module as

```
1000+1058*T-38.9*T*LOG(T)+GFUNXY#
```

rather than

```
-1000+1058*T-38.9*T*LOG(T)+GFUNXY#
```

This is because the DATA module concatenates all lines and removes extra spaces before trying to enter the function in the GIBBS workspace. Thus, the – sign is taken as a delimiter between 298.15 and 1000, and the function incorrectly becomes:

```
FUNCTION GHSERXY 298.15 1000+1058*T-38.9*T*LOG(T)+GFUNXY#; 6000 N !
```

Avoid this mistake by giving at least one empty space as the first character of a new line, such as

```
FUNCTION GHSERXY  298.15
 -1000+1058*T-38.9*T*LOG(T)+GFUNXY#; 6000 N !
```

which is read correctly as

```
FUNCTION GHSERXY  298.15 -1000+1058*T-38.9*T*LOG(T)+GFUNXY#; 6000 N !
```

The lowest-temperature limit (in Kelvin) for the applicability of the (first) TP-Function in a function is normally set by default as 298.15 K, in most cases. However, you can set another limit when it is applicable (according to experimental data and assessments).

An upper-temperature limit (in Kelvin; followed by a `Y` or `N` sign) for the applicability of each TP-Function in a function must be given after the semicolon (`;`) immediately following the specific TP-Function. The highest-temperature limit (in Kelvin) for the applicability of the current function is always followed by the `N` sign. If a negative number is given as the lowest-temperature limit, it assumes there are breakpoints in pressure for this function. In these cases, it is interpreted as the lowest-pressure limit (in Pascal), and the other limits in the current function is also taken as pressure limit values (in Pascal).

The temperature/pressure limits for the functions are checked during calculations. An indicator is set if the actual temperature/pressure condition is below the lowest temperature/pressure limit or above the highest temperature/pressure limit. In these cases, an extrapolation is done using the TP-Function valid in the nearest temperature/pressure range.

The optional reference index *{Ref. Index}* is an integer number indicating where to find the particular function in a special reference file. The references are listed when doing the GET_DATA command in the

DATA module. They can also be listed in the GIBBS module with the command LIST_DATA with the option R.

For accounting for the reference indices, also see the keyword *REFERENCE_FILE* on page 35.

The reference index field can also be an abbreviation (such as REF:250, REF_002, or REF-SGTE) which denotes the original reference. In this case, the reference cannot be obtained when issuing the DATA command GET_DATA or the GIBBS command LIST_DATA (with the option R).

However, the references directly coded in the database definition file (***setup.TDB) starting with a letter can be shown when issuing the DATA command GET_DATA or the GIBBS command LIST_DATA (with the option N or R). Normally, such references must be located after the LIST_OF_REFERENCE keyword. It is recommended to use reference code names such as REF001, REF018, etc. The reference list, which is generated by the GIBBS command *LIST_DATA* on page 77 <file> with the N or R option, is also possible to be directly read by the DATA module.

The DATA module can selectively retrieve functions which are necessary for a defined system from a database that has functions stored in its setup file or SEQ sequential function file, while all other functions irrelevant for the defined system are ignored and are not saved in associated GIBBS and POLY workspaces. Previously, this can only be done for large databases that have functions stored in RND1 random or FTP function files.

Examples

```
FUNCTION GFREE  298.15  1000+GFUNXY#; 6000 N !

FUNCTION GFUNXY 298.15 -1000+200*T+30*T*LOG(T); 6000 N 505 !

FUNCTION G0_CAO 298.15 -663538.11+352.67749*T-57.7533*T*LN(T)

        +5.3895E-03*T**2-8.879385E-07*T**3+575530*T**(-1);

     1400.00 Y -625196.99+78.896993*T-20.40145*T*LN(T)

        -1.112923E-02*T**2+5.1896733E-07*T**3-6917350*T**(-1);

     2900.00 Y -499226.55-490.37695*T+51.95912*T*LN(T)

       -2.961051E-02*T**2+1.4033905E-06*T**3-48114685*T**(-1);

     3172.00 Y -587711.89+375.04117-62.76*T*LN(T);

     6000.00 N REF020 !
```

PARAMETER

```
PARAMETER [GIBBS parameter name] [lowest temp. limit]

[expression 1]; [upper temp. limit 1] Y

[expression 2]; [upper temp. limit 2] Y

[expression 3]; [upper temp. limit 2] Y

   .......... ; ..... Y
```

```
[expression n-1]; [upper temp. limit n-1] Y
[expression n]; [upper temp. limit n] N {Ref. Index} !
```

This keyword can appear in both files for database definition and sequential storage, but not in FTP files. After the keyword, a valid *GIBBS parameter name* should be given.

It is used to define standard Gibbs energies (i.e. the `G` parameters for Gibbs energy of formations) of all valid end-members of various stoichiometric and solution phases, and excess Gibbs energies (i.e. the `L` parameters for Gibbs energy of interactions) of all binary, ternary, quaternary or higher-order interactions in various solution phases; both standard Gibbs energies and excess energies can also have parameters for contributions from PT-depended volume variations (i.e. the `V0`, `VA`, `VB`, `VC` and `VK` parameters for molar volume, thermal expansivity, bulk modulus, isothermal compressibility and high-pressure fitting parameter), magnetic ordering (i.e. the `TC` and `BM` parameters for Curie temperature and Bohr magneton number) and hypothetical electrostatic interactions (i.e. `BM` parameter for Born functions ωPr,Tr of aqueous solute species).

The general form of a parameter is:

```
<identifier>(<phase>, <constituent array>; <digit>) <xxx>
<expression> <yyy> <keyword Y or N> <zzz> !
```

General Form

Name	Description
identifier	The parameter type.
phase	The phase name (maximum 24 characters).
constituent array	The specific constituent array in the phase.
digit	The degree of composition-dependent interaction contribution (an integer number from `0` through `9`), that is only for excess energy (`L`), Curie temperature (`TC`) and Bohr magneton number (`BMAGN`), as well as for volume-related parameters (`V0` or `VA` or `VB` or `VC` or `VK`); if it is valued as zero, or if it is for the standard Gibbs energy (`G`) for which the degree is always zero, it can be omitted.
expression	The mathematical relation to describe the parameter.
xxx and yyy	The low and high temperature limits respectively for the applicable temperature range of the parameter expression
keyword Y or N	The indicator on if there is continuation for the parameter expression or not
zzz	The reference index/number for the assessment of this parameter;
!	The exclamation mark is used to indicate that the current parameter definition is ended

GIBBS Parameter Name

The GIBBS parameter name has a general form of:

```
<identifier>(<phase>,<constituent array>;<digit>)
```

Name Examples

Parameter	Definition
`G(GAS,C1O2)`	The Gibbs energy of formation of a CO2 molecule in gas.
`G(FCC,FE:VA)`	The Gibbs energy of formation of fcc Fe with interstitials.
`L(LIQ,Fe,Cr;0)`	The regular solution parameter for Fe and Cr in liquid.
`L(LIQ,Fe,Cr;1)`	The sub-regular solution parameter.
`TC(BCC,Fe:Va)`	The Curie temperature of bcc Fe.
`BMAGN(BCC,Fe:Va)`	The Bohr magneton number parameter of bcc Fe. '

The GIBBS parameter name consists of several parts. The first is a *type-identifier*. The following type-identifiers are legal:

Type-Identifier	Definition
G	Standard energy parameter (Gibbs energy of formation)
L	Excess energy parameter (Gibbs energy of interaction)
TC	Curie temperature for magnetic ordering
BMAGN or BM	Bohr magneton number for magnetic ordering (or Born function ωPr,Tr for aqueous solute species).
V0	Molar volume at 298.15 K and 1 bar (a numeric value only)
VA	Integrated thermal expansivity $\int_{298.15}^{T} \alpha(T)dT$
VB	Bulk modulus at 1 bar
VC	Isothermal compressibility
VK	High-pressure fitting parameter

You can also use `G` for interaction parameters; and on output list (performed by the GIBBS command LIST_PARAMETER or LIST_PHASE_DATA) the type-identifier `L` is always used for interaction parameters. Note that the type-identifier `BM` is also used for Born functions ωPr,Tr of aqueous solute species.

The identifier must be followed by an opening parenthesis, a phase name, a comma and a constituent array. Optionally, the constituent array can be followed by a semicolon and a digit. The parameter name is terminated by a closing parenthesis.

It is important that if a phase bears a legal phase-type (among `G`, `A`, `Y`, `L`, `I`, `F` and `B`) in its phase definition (already by the PHASE keyword; such as `GAS:G`, `LIQUID:L`, `SLAG:L`, `IONIC_LIQ:Y`, `SPINEL:I`, `FCC_L12:F`, `HCP_D021:F`, `BCC_B2:B`, `AQUEOUS:A`), such a valid phase-type code should not be attached to the phase name in the PARAMETER keyword.

Specifying the phase name in UPPER-case is recommended. You can define a phase name as a mixture of UPPER-case and lower-case letters in a database, but the DATA module automatically converts all lower-case to UPPER-case because the GIBBS module only recognizes UPPER-case phase names.

The *constituent array* consists of a list of constituent names. Interaction parameters have two or more constituents from the same sublattice separated by a comma. If the phase has sublattices, at least one constituent in each sublattice must be specified. The constituents in different sublattices must be given in sublattice order and are separated by a colon.

After the component array, a sub-index digit can be specified after a semicolon. This digit must be in the range 0 to 9. The interpretation of the sub-index depends on the excess energy model used for the phase. If no semicolon and digit are given, the sub-index value is assumed to be as zero.

The excess energy parameters, e.g. the regular/subregular (binary) parameter or ternary parameters, are multiplied with two or more fractions of the constituents from the same sublattice of the solution phase. These additional constituents must be given as interacting constituents.

Be careful about the sign of odd terms, for example, `L(BCC, B, A:VA;1)` is treated as `L(BCC,A,B:VA;1)`, i.e. it is always put into alphabetical order.

Solution phases with sublattices may have interacting constituents in each sublattice.

You can use an asterisk (`*`) to denote that the excess interaction parameter is independent of the constituents of a specific sublattice. For example, `L(FCC_L12,AL,NI:*)` means that the interaction parameter is for the binary interaction between constituents AL and NI on the first sublattice in the FCC_L12 solution phase, while it is independent of all constituents on the second sublattice. A interaction parameter in the list of constituents is always added to the Gibbs energy and the asterisk (`*`) is calculated with the term of [`1-∑y(specified constituents)`], which implies that in an A-B binary system the following three `L` parameters are identical (but in higher-order systems, they are different):

- `L(phase,A,B)` is multiplied with `X(A)*X(B)`
- `L(phase,A,*)` is multiplied with `X(A)*(1-X(A))`
- `L(phase,B,*)` is multiplied with `X(B)*(1-X(B))`

A parameter always starts with a lowest temperature limit of its applicability, followed by one or more (up to 10) expressions (TP-Functions) coded as mathematical relations of constants, functions of stable variables (`T` and `P`) and entered functions (normally with a # suffix, e.g. `+3*GSERAL#`).

The expression is a FORTRAN-like expression and operators `+`, `-`, `*`, `=` and `**` can be used (`**` only with integer powers). Unary-functions `LN` or `LOG` (both for natural logarithm) and `EXP` (for exponential) can also be used. Each expression (`TP-Function`) should ends with a semicolon (`;`) and be followed by its upper applicable temperature limit and a continuation indicator (`Y` to continue with the next expression, or `N` to end the parameter's expression). If there is no continuation after a specific expression (TP-Function), the reference index can be optionally given after the `N` indicator.

A complete/valid parameter entry can be written in several continuation lines if the parameter's expression (TP-Function) is too long or if there is more than one applicable expression (TP-Function), as the maximum length of each line is 78 characters.

It is recommended to always have at least one empty space at the beginning of each continuation line. Avoid entering parameters such as:

```
PARAMETR G(LIQUID,A,B) 298.15
-2000+4568*T+2*GFUNAB#; 6000 N !
```

Such a parameter is read by the DATA module as `2000+4568*T+2*GFUNAB#`, rather than as `-2000 +4568*T+2*GFUNAB#`.

Avoid this mistake by giving at least one empty space as the first character of a new line, such as

```
PARAMETR G(LIQUID,A,B) 298.15
-2000+4568*T+2*GFUNAB#; 6000 N !
```

The lowest-temperature limit (in Kelvin) for the applicability of the (first) TP-Function in a parameter is normally set by default as 298.15 K, in most cases; however, you can set another limit when it is applicable (according to experimental data and assessments). An upper-temperature limit (in Kelvin; followed by a `Y` or `N` sign) for the applicability of each TP-Function in a parameter must be given after the semicolon (`;`) immediately following the specific TP-Function; and the highest-temperature limit (in Kelvin) for the applicability of the current parameter is always followed by the `N` sign. If a negative number is given as the lowest-temperature limit, it is assumed that there are breakpoints in pressure for this parameter. In such cases, it is interpreted as the lowest-pressure limit (in Pascal), and the other limits in the current parameter are also taken as pressure limit values (in Pascal).

The temperature/pressure limits for the parameters are checked during calculations. An indicator is set if the actual temperature/pressure condition is below the lowest temperature/pressure limit or above the highest temperature/pressure limit. In these cases, an extrapolation is done using the TP-Function valid in the nearest temperature/pressure range.

The optional reference index *{Ref. Index}* is an integer number indicating where to find the particular parameter in a special reference file. The references are listed when doing the GET_DATA command in the DATA module. These can also be listed in the GIBBS module with the command LIST_DATA and the option `R` or `N`.

For accounting the reference indices, also see the keyword *REFERENCE_FILE* on page 35.

The reference index field can also be an abbreviation (such as `REF:250`, `REF_002`, or `REF-SGTE`) denoting the original reference. In this case, the reference cannot be obtained when issuing the DATA command GET_DATA or the GIBBS command LIST_DATA (with the option `R` or `N`).

However, the references directly coded in the database definition file (`***setup.TDB`) that starts with a letter can be shown when issuing the DATA command `GET_DATA` or the GIBBS command LIST_DATA (with the option `R` or `N`). Normally, such references must be located after the `LIST_OF_REFERENCE` keyword. It is recommended to use reference code names such as `REF001`, `REF018`, etc. The reference list, which is generated by the GIBBS command LIST_DATA `<file>` with the `N` option, is thus also possible to be directly read by the DATA module.

Examples

```
PARAMETER G(BCC,FE:VA) 298.15 1000+200*T+...; 6000 N 91DIN !

PARAMETER TC(BCC,FE:VA)    298.15 +1043; 6000 N 91DIN !

PARAMETER BMAGN(BCC,FE:VA) 298.15 +2.22; 6000 N 91DIN !

PARAMETER G(SIGMA,FE:CR:CR;0) 298.15 1000+200*T+...; 6000 N 101 !

PARAMETER G(LIQUID,AL;0)  298.15  +11005.553-11.840873*T

        +7.9401E-20*T**7+GHSERAL#;

933.60 Y +10481.974-11.252014*T+1.234264E+28*T**(-9)+GHSERAL#;

   2900.00 N  REF:283 !

PARAMETER G(BCC_A2,PB:C)   298.15 UN_ASS#; 300 N REF:0 !

PARAMETER G(BCC_A2,NI:C;0) 298.15 +GHSERNI#+3*GHSERCC#

                                    +400000-100*T; 6000 N REF071 !

PARAMETER G(BCC_A2,MN:VA)  298.15 +GMNBCC#; 6000 N REF285 !

PARAMETER BM(AQUEOUS,OH-1) 298.15 +Z0002PW0#; 1600 N 155 !

PARAMETER L(BCC,FE,CO:VA;0) 298.15 1000+200*T+...; 6000 N !

PARAMETER L(BCC,FE,CO:VA;1) 298.15 1000+200*T+...; 6000 N !

PARAMETER L(BCC,FE,CO:VA;2) 298.15 1000+200*T+...; 6000 N !

PARAM    TC(BCC_A2,CO,MO:VA;0) 298.15  -3700;    6000 N R454 !

PARAM    TC(BCC_A2,CO,MO:VA;1) 298.15  +2300;    6000 N R454 !

PARAM BMAGN(BCC_A2,CO,MO:VA;0) 298.15 -3.445;    6000 N R454 !

PARAM V0(BCC_A2,CR,FE:VA;0) 298.15 +ZERO#; 6000 N REF06V !

PARAM V0(BCC_A2,CR,FE:VA;1) 298.15 -1.10524097E-7; 6000 N REF06V !

PARAM V0(BCC_A2,CR,FE:VA;2) 298.15 +1.40024130E-7; 6000 N REF06V !

PARAM VA(BCC_A2,CR,FE:VA;0) 298.15 -6.49444634E-6*DELTAT#; 6000 N
REF06V !

PARAM VA(BCC_A2,CR,FE:VA;1) 298.15 +2.91269321E-5*DELTAT#; 6000 N
REF06V !
```

OPTIONS

```
OPTIONS /[alloy name]([composition limitations for all alloying
elements]) !
```

This keyword defines an alloy in a database. An alloy has a name, a major component and a number of alloying elements. The purpose for defining an alloy is to be able to tell you about applicable composition limits of the current database in applications to that particular type of alloy. It is possible to have several alloys in the same database. The alloys are given after the keyword in the database.

The alloy name must be proceeded by a slash (/) and terminated by the opening parenthesis, with no spaces are allowed in between.

- The alloy name is maximum 8 characters.
- After the parenthesis, follows the major element and a parenthesis with its minimum mass and minimum mole percent given inside.
- Then, the alloying element names, each with its maximum mass and mole percent are given within parenthesis.
- There must a space between definitions for each alloying element.
- The alloy definition is terminated by a closing parenthesis, and the whole OPTIONS keyword by the exclamation mark (!).

Example

```
OPTIONS /SSteel(Fe(60,60) CR(30,30) NI(15,15) SI(1,1) N(.1,1)) !
```

TABLE

It is recommended to always have at least one empty space at the beginning of each continuation line. Otherwise, the DATA module may issue some error messages when reading the table entry.

```
TABLE [name]*8 [start temp] [end temp] [delta temp] [table values] !
```

This keyword can appear in both files for database definition and sequential storage, but not in FTP files. It makes a table of Gibbs energy as a function of temperature where the values are given from the start temperature to the end temperature, at a step of the delta temperature.

Example

```
TABLE DEMTAB 1000.0 1500.0 100.0 -2912.9008 -2834.2416 -2755.5824
              -2677.7600 -2600.7744 -2524.2072 !
```

ASSESSED_SYSTEMS

```
ASSESSED_SYSTEM [descriptions on special treatments for specific
assessed systems]*8000 !
```

This keyword can be included in the database definition file (the `***setup.TDB` file). A maximum of 8000 characters after the keyword (to describe some special options when the DATA, GIBBS and POLY models deal with the existing systems with assessed data) is allowed until the exclamation mark (!).

Several ASSESSED_SYSTEMS keywords can be used in the same TDB file in order to have more lists of assessed systems. There is no limit to the number of the entries in the same TDB file.

The assessed systems in the database, and the special treatment options, are typed after the keyword. The elements (always in UPPER CASE) in each assessed system must be in alphabetical order and be separated by a hyphen, such as `C-FE` for the Fe-C binary system, and `C-CR-FE` for the Fe-Cr-C ternary system. A space must be between each assessed system. Information on assessed binary, ternary or higher-order systems may also be given in this way.

A ternary system like C-CR-FE does not imply that the binary C-CR, C-FE and CR-FE are assessed. There is no way to indicate partially assessed systems.

- There is a field to give some descriptive information for a specific system, with various options about how to:
- Reject or restore phase(s) from the current database in the DATA module;
- Set major constituent(s) in the first composition set and to set a second composition set for a specific phase available in the current database in the GIBBS module; *and*
- Calculate this specific system in the POLY module.
- This is the facility used by the BIN (binary phase diagrams) and TERN (ternary phase diagrams) modules in the Thermo-Calc software/database package.

The descriptive information must immediately follow the specific system name, and must be enclosed within parenthesis `(` and `)`, and the left parenthesis must follow directly after the system, such as :

```
AL-NI(TDB +L12 ;G5 C—S:L12/NI:AL:VA ;P3 STP:.8/1200/1 STP:.2/600/1)
```

The syntax TDB means that the commands to the DATA module proceed, and +L12 in the example means that the phase called L12 should be restored (it has been rejected by default).

The directive `;G5` means that the following are commands to the GIBBS module. `C_S:` means creating a second composition set, after the colon follows the phase name and after the slash the major constituents.

After the directive `;P3` follows commands to the POLY module. `STP:` means setting a start point with the value of the X-axis first (composition for the second element in a binary system), the slash separates the Y-axis value (temperature), and possibly one or more directions (`-1`, `1`, `-2` or `2`).

A summary of the allowed syntax is:

TDB accepts

- `+phase` and `-phase` for restore/reject.
- `;G5` accepts
- `MAJ:phase/constituent-array` for major constituents of the first composition set.
- `C_S:phase/constituent-array` for a second composition set.

- `;P3` accepts
- `TMM:lt/ht` for the low-/high-temperature limits (`lt` and `ht`; for instance `TMM:500/4000`) suitable for calculating phase diagrams and property diagrams of a binary system.
- `*` for a default start point which is set as:
- for a binary system: at the composition `X(2`nd `element)=.1234`, temperature `T=1100 K` and with the default directions; or
- for a ternary system: at the compositions `X(2`nd `element)=.1234` and `X(3`rd `element)=.1234` and with the default directions.
- `STP:x/t/d1/d2/d3` for a specific start point in a binary system which is set as at the composition `X(2`nd `element)=x` and temperature `T=t` (in K), and with the directions `d1`, `d2` and/or `d3`.
- `STP:x1/x2/d1/d2/d3` for a specific start point in a ternary system which is set as at the compositions `X(2`nd `element)=x1` and `X(3`rd `element)=x2` and with the directions `d1`, `d2` and/or `d3`.
- The direction(s) can be defined as `-1`, `1`, `-2` or `2`. If no direction is specified, all default directions are used (meaning no `ADD` command is enforced in the POLY module).
- If only one start point is specified, the direction(s) may be omitted; if more than one start points are specified, at least one direction for each start point must be given for all start points.

Each entry for a specific binary or ternary sub-system can be written in one or several lines (each line with 78 characters).

Other Examples

```
ASSESSED_SYSTEMS
AL-NI(TDB +L12 +BCC_B2 ;G5 C_S:L12/NI:AL:VA
       ;P3 STP:.8/1000/1 STP:.45/700/1 STP:.7/700/1)
AL-PB(TDB -HCP -BCC
       ;G5 MAJ:LIQ/AL MAJ:FCC/AL:VA C—S:LIQ/PB C—S:FCC/PB:VA ;P3 *)
CR-FE(;G5 C—S:BCC/CR:VA ;P3 STP:.6/1200/1/-2/2)
AG-CU(;G5 MAJ:FCC/AG:VA C_S:FCC/CU:VA ;P3 STP:.3/1000)
C-NB(;P3 STP:.9/1100/1)
C-SI(;P3 *)
CO-CR(;G5 MAJ:FCC/CO:VA C_S:FCC/CR:VA ;P3 STP:.1/1100)
CR-FE(TDB -HCP ;G5 C_S:BCC/CR:VA ;P3 STP:.6/1200/1/-2/2)
```

```
CR-NI(;P3 *)
CR-W(;G5 MAJ:BCC/W:VA C_S:BCC/CR:VA
       ;P3 TMM:500/4000 STP:.3/700/1 STP:.3/1800/2 )
CU-FE(TDB -HCP ;G5 MAJ:LIQ/CU MAJ:FCC/FE:VA C_S:FCC/CU:VA
       ;P3 STP:.9/1400)
FE-N(TDB +FE4N ;P3 *)
FE-O(TDB -LIQUID +IONIC ;G5 C_S:ION_LIQ/FE+2:O-2 MAJ:ION_LIQ/FE+2:VA
      ;P3 STP:.2/2000/1 )
FE-S(TDB -LIQUID +IONIC ;G5 C_S:ION_LIQ/FE+2:S MAJ:ION_LIQ/FE+2:S-2
;P3 *)
AL-MG-SI(;P3 *)
C-CR-FE ;G5 MAJ:BCC/FE:VA C_S:BCC/CR:VA ;P3 *) !
```

The semicolon (`;`) is a part of the `;G5` and `;P3` directives. A long descriptive information can be written in more than one line, such as for the `AL-NI`, `AL-PB`, `CU-FE` and `FE-O` systems shown above.

The directive `;P3 *` is needed if the default start point should be used. If there is no `P3` directive, the BIN or TERN module generates some 20 different start points in order to cover all possible compositions and temperatures (for a binary system) or all possible compositions (for a ternary system under any specific temperature).

REFERENCE_FILE

```
REFERENCE_FILE [file name] !
```

This keyword takes a reference file name as its argument. This reference file (that must be edited in a restrictive way and be saved as a blocked file with a fixed line-length for each line in the file, as described below) contains a complete list of the references for the various parameters (and sometimes functions) in the database. The file must have a fixed record structure: each reference entry with one or several records, and each record with exactly 78 characters written in one single blocked line; and no empty space is allowed at the beginning of all lines in the file. If there are more than one record entered for a reference entry, all the continuation lines must start with an ampersand (`&`). The line number of the first record for a specific reference entry is then accounted as the unique integer for that specific reference, which is referred when a parameter or function calls this integer as the optional *{Ref. Index}*.

Also see keyword *PARAMETER* on page 24 or *FUNCTION* on page 22 about specifying a reference index.

Example

```
/-1<G>            T.C.R.A.S. Class 1
```

```
AG1.64TE1          THERMODATA 01/93

&28/01/93

&SILVER 1.64-TELLURIDE. Solid Standard State.

AG1                HULTGREN SELECTED VAL.       SGTE **

&AT.WEIGHT 107.870,STANDARD STATE:CODATA KEY VALUE.MPT=1234.93K. &--
U.D. 30/10/85.

AG1<G>             T.C.R.A.S  Class: 1

AG1/+1<G>          T.C.R.A.S  Class: 1

AG1BR1             N.P.L.                       SGTE **

&Tfusion uncertain and heat vaporization estimated.

AG1BR1<G>          THERMODATA 01/93

&28/01/93

&Gaseous Standard State.

AG1BR1O3           BARIN & KNACKE.SUPPL.REF:62,* SGTE **

&AGO3BR   SILVER OXYTRIBROMIDE
```

In the example, the unique integers of related references (for assessed elements, species, phases, interactions, etc.) are:

```
1/-1<G>

2AG1.64TE1

5 AG1

8AG1<G>

9AG1/+1<G>

10AG1BR1

12AG1BR1<G>

15AG1BR1O3
```

LIST_OF_REFERENCE

```
LIST_OF_REFERENCE

NUMBER SOURCE

  [REFxxx]  '[Detailed reference]'

..........

.......... !
```

This keyword starts a reference list that is directly coded in the database definition file (`***setup.TDB`). Its argument begins on the following line, and normally has an explanation line (NUMBER SOURCE) that is followed by various reference codes. Each reference code may occupy one or more lines (each line with maximum 78 characters), but must have a reference code name (that starts with a letter) and the detailed reference information (that is written within two single-quotation marks, `''`). It is recommended to use reference code names such as `REF001`, `REF018`, etc. A maximum of 400,000 characters after the keyword is allowed until the exclamation mark (`!`).

Such a reference list can be shown when issuing the DATA command *GET_DATA* on page 15 or the GIBBS command *LIST_DATA* on page 77 (with the `R` option). The reference list, which is generated by the GIBBS command LIST_DATA `<file>` with the `N` option, has this structure, and is thus possible to be directly read by the DATA module.

Example

```
LIST_OF_REFERENCES
NUMBER  SOURCE
REF283 'Alan Dinsdale, SGTE Data for Pure Elements,
          Calphad Vol 15(1991) p 317-425,
          also in NPL Report DMA(A)195 Rev. August 1990'
REF224 'P-Y Chevalier, Thermochimica Acta, 130 (1988) p 33-41; AG-SI'
!
```

ADD_REFERENCE

```
ADD_REFERENCE
  [REFxxx]  '[Detailed reference]'
..........
.......... !
```

Use this keyword to have an unlimited number of references. The use of this is the same as that of *LIST_OF_REFERENCE* on the previous page, and is used after it. Several `ADD_REFERENCES` sections can be used in a single database if there are too many references.

It is recommended to always start with a so-called *empty reference* (such as the `DUMP0` reference in the following example) as the first entry in each of the `ADD_REFERENCE` sections to make sure that all references are listed appropriately as retrieving data by issuing the *GET_DATA* on page 15 command.

Example

```
ADD_REFERENCE
   DUMP0  'Empty reference 0'
   REF4 'J-O Andersson, B. Sundman, CALPHAD Vol 11, (1987), p 83-92
          TRITA 0270 (1986); CR-FE'
   REF5 'J-O Andersson, Met. Trans A, Vol 19A, (1988) p 627-636
          TRITA 0207 (1986); C-CR-FE'
```

```
   REFS 'Pingfang Shi (2006), TCS PTERN Public Ternary Alloys
Database,
          v1.2; Modified L0(BCC,Fe,C) and L0(BCC,Cr,C) parameters
          at high temperatures.'
   ... ...
!
ADD_REFERENCE
   DUMP1  'Empty reference 1'
   REF275 'A. Fernandez Guillermet, Z. Metallkde. Vol 79(1988) p.524-
536,
          TRITA-MAC 362 (1988); C-CO-NI AND C-CO-FE-NI'
   REF393   'K. Frisk, Metall. Trans. Vol 21A (1990) p 2477-2488,
          TRITA 0409 (1989); CR-FE-N'
   REF1096  'P. Gustafson, Metall. Trans. 19A(1988) p 2547-2554,
          TRITA-MAC 348, (1987); C-CR-FE-W'
   ... ...
!
```

CASE and ENDCASE

```
CASE [ELEMENT/SPECIE/PHASE] !
  IF (boolean algebra on element, species or phase names) THEN
       [GIBBS/POLY/DATA command] !
ENDCASE !
```

This keyword takes as its argument, a definition on which type of the following Boolean algebra operates. A simple Boolean algebra using `AND` and `OR` with a maximum of four levels of parentheses works. The `CASE` construction must end with the `ENDCASE` keyword. This makes it possible to have additional GIBBS or POLY commands executed depending on the user selection of elements, species or phases. The DATA commands that can be given as secondary keyword to *DEFAULT_COMMAND* on page 17 can also be executed.

Example 1

```
CASE ELEMENT !
   IF((CR OR TI OR V) AND N)
   THEN GES AM_PH_DES @ C_S ,, CR MO TI V:C N: !
ENDCASE !
```

Example 2

```
CASE ELEMENT !
```

```
   IF(O) THEN TDB DEFINE_SYSTEM_ELEMENT /- !
ENDCASE !
```

Example 3

```
CASE ELEMENT !
   IF(AL AND FE)
   THEN TDB DEF_SYS_ELEMENT VA !
ENDCASE !
```

VERSION_DATA

```
VERSION_DATE [string]*78 !
```

The string is denoted as the version/revision date and database manager of the database. Nowadays, this directive is used mainly for the purpose of keeping the development and revision history mostly updated.

Example

```
VERSION_DATE Last update and adjustment: Database Manager, 2015-09-25
!
```

Working with the Mobility Databases

In this section:

About the Diffusion Module (DICTRA) and the Databases

A Diffusion Module (DICTRA) installation shares directories with Thermo-Calc, including the `\DATA\` (or `/data/`) area and the database initiation file (`TC_INITD.TDB` or `initd.tdb`).

When you want to switch or add self-generated databases/data-sets or any new database, use the predefined database list in the database initiation file as part of the Thermo-Calc installation packages.

You can add comment lines in the database initiation file, which must start with a $ sign. These lines are ignored by the DATA module. This is also applicable if the database manager or user wants to temporarily disable a database in the predefined database list. If there are too many databases in the \DATA\ area that the DATA module cannot handle properly, the $ sign can be used to temporarily comment about the uncommon databases. However, this does not often happen.

If Thermo-Calc is used with a Windows platform, such additional database initiation files can be located at any directory of any driver, on either a local computer or connected server. The NEW_DICTORY_FILE command displays an **Open** window to access a database initiation file if the file name or its path is not given on the same line of the NEW_DICTORY_FILE command, or if it is incomplete or incorrect, so that the path (in the Look in box) and database initiation file name (in the File name box) can be appropriately selected. However, if Thermo-Calc is run on a Linux/UNIX platform, these files must be located in the current working directory (where Thermo-Calc is started).

In an additional database initiation file, the first database entry may need to have the same path definition structure as in the ordinary database initiation file. Copy the entry line(s) for common databases from the original file to a database initiation file. The entries of the additional databases follow. Similar to the standard databases predefined in the database initiation file, all databases should normally be located in subdirectories under the directory as defined by the `TCPATH` parameter (Windows) or the `TC_DATA` parameter (Linux/UNIX), or under its subdirectory `\DATA\` or `/data/`.

Windows Database Initiation File Example

For a Windows platform, the following example is an additional database initiation file, called `MYINITD1.TDB`:

```
$

$ DATABASES TCC (Additional TCC Databases)

PURE5 TCPATH\DATA\PURE5\PURE5SETU.TDB   SGTE Pure Elements Database,
version 5 !

AD1   TCPATH\DATA\ADD1\AD1SETUP.TDB     TCS ADD1 Solution Database,
version 1 !

AD2   TCPATH\ADDDATA\ADD2\AD2SETUP.TDB  TCS ADD2 Solution Database !

$AD2o  TCPATH\ADDDATA\ADD2old\AD2SETUP.TDB  TCS ADD2 Database (old) !

AD3   TCPATH\DATA\NEWDATA\ADD3\AD3SETUP.TDB  TCS ADD3 Solution
Database !
```

```
AD4   TCPATH\DATA\NEWDATA\MYPROJ1\ADD4\AD4SETUP.TDB
      MYPROJECT1 ADD4 Solution Database !
$
$ DATABASES DIC (Additional DICTRA Databases)
DCAD1 TCPATH\DICDATA\DCADD1\DCAD1SET.TDB
          TCS DCADD1 Mobility Database !
```

Extensions to Database Definition File Syntax

Software packages for simulation of diffusional phase transformations, such as the Diffusion Module (DICTRA) and Precipitation Module (TC-PRISMA), need both thermodynamic data and kinetic data (i.e. diffusivities or mobilities). Naturally, the handling and storage of kinetic data also benefits from the use of some kind of database management. Thus, the TDB database definition file syntax has been extended to incorporate some new keywords needed for storing kinetic data - *PARAMETER* on page 24, *DIFFUSION* on the next page, and *ZERO_VOLUME_SPECIES* on the next page.

PARAMETER

```
PARAMETER [special GIBBS parameter name] [low temp. limit]
[expression 1]; [upper temp. limit 1] Y
[expression 2]; [upper temp. limit 2] Y
[expression 3]; [upper temp. limit 2] Y
   .......... ; ..... Y
[expression n-1]; [upper temp. limit n-1] Y
[expression n]; [upper temp. limit n] N {Ref. Index} !
```

This keyword allows you to enter all types of normal GIBBS parameters for thermodynamic data as well as five special extensions suitable for kinetic data used in the Diffusion Module (DICTRA).

Valid extensions to special GIBBS parameter names are:

- `MQ`: Activation enthalpy for mobility equation.
- `MF`: Pre-exponential factor for mobility equation.
- `DQ`: Activation enthalpy for diffusivity equation.
- `DF`: Pre-exponential factor for diffusivity equation.
- `VS`: Volume counted per mole of volume carrying species.

Examples

```
PARAMETER MQ(BCC,FE:VA) 298.15 1000+200*T+...; 6000 N !
PARAMETER MF(BCC,CO:VA) 298.15 1000+200*T+...; 6000 N !
PARAMETER DQ(FCC,FE:VA) 298.15 1043+...; 6000 N 10 !
```

```
PARAMETER DF(FCC,CR:C ) 298.15 1000+200*T+...; 6000 N 10 !
PARAMETER VS(FCC) 298.15 1000+200*T+...; 6000 N 11 !
```

DIFFUSION

```
DIFFUSION [model keyword] [phase name] [additional parameter(s)] !
```

This keyword specifies the type of diffusion model to use for a phase if the default model is not desired. The default model calculates the full diffusion matrix. A diffusivity is calculated from the different mobilities and the thermodynamic factors. The former ones are calculated as:

```
M = exp (∑MF/RT) exp (∑MQ/RT) / RT
```

where ∑ stands for a weighted summation of the different MF's and MQ's plus possibly a Redlich-Kister term.

Valid Model Keywords

Keyword	Description
NONE	No diffusion in this phase
DILUTE	Constitution list of dependent species in each sublattice must be given as an additional parameter. Only the diagonal terms in the diffusion matrix are calculated. D = exp (∑ DF/RT) exp (∑DQ/RT).
SIMPLE	Constitution list of dependent species in each sublattice must be given as additional parameter. Only the diagonal terms in the diffusion matrix are calculated. D = ∑ DF + ∑ DQ.
MAGNETIC	The so-called ALPHA and ALPHA2 parameters must be given as additional parameters. ALPHA is for the substitutional magnetic model and ALPHA2 for the interstitial one. By appending an & sign and a species name after the alpha keyword one can supply individual values for the different species. The full diffusion matrix is calculated.

Examples

```
DIFFUSION NONE SIGMA !
DIFFUSION DILUTE CEMENTITE : FE : C : !
DIFFUSION MAGNETIC BCC_A2 ALPHA=0.3 ALPHA2&C=1.8 ALPHA2&N=0.6 !
```

ZERO_VOLUME_SPECIES

```
ZERO_VOLUME_SPECIES [list of species] !
```

In the Diffusion Module (DICTRA), the assumption that the volume is carried by the substitutional elements only is applied. The interstitial elements are assumed to have zero molar volumes. This keyword uses a list of which species are to be considered as zero volume ones for an argument.

Example

```
ZERO_VOLUME_SPECIES VA C N !
```

Database Definition File Examples

In this section:

Example 1: A Steel Database

```
TEMP-LIM 500.0 2000.0 !
$
$ELEMENT, NAME, REF.STATE, ATOMIC-MASS, H0, S0 !
ELEMENT VA VACUUM            0.0    0.0 0.0 !
ELEMENT C  GRAPHITE         12.011  0.0 0.0 !
ELEMENT V  BCC              50.9415 0.0 0.0 !
ELEMENT CR BCC-PARAMAGNETIC 51.996  0.0 0.0 !
ELEMENT FE FCC-PARAMAGNETIC 55.847  0.0 0.0 !
ELEMENT NI FCC-PARAMAGNETIC 58.69   0.0 0.0 !
ELEMENT MO BCC              95.94   0.0 0.0 !
ELEMENT W  BCC             183.85   0.0 0.0 !
$
$PHASE, NAME, TYPE, NR-OF-SUBL, SITES-IN-EACH-SUBL. !
PHASE BCC       B1M  2 1.0 3.0 !
PHASE FCC       F2M  2 1.0 1.0 !
PHASE HCP       0    2 2.0 1.0 !
PHASE LIQUID    3    2 1.0 1.0 !
PHASE CEMENTITE 4    2 3.0 1.0 !
PHASE M23C6     4    2 23.0 6.0 !
PHASE M7C3      4    2 7.0 3.0 !
PHASE M6C       4    4 2.0 2.0 2.0 1.0 !
PHASE SIGMA     0    3 10.0 4.0 16.0 !
PHASE MU-PHASE  0    3 7.0 2.0 4.0 !
PHASE R-PHASE   0    3 27.0 14.0 12.0 !
PHASE GRAPHITE  4    1 1.0 !
$
$CONSTITUENT, PHASE-NAME : CONSTITUENTS !
CONSTITUENT BCC :V CR FE NI MO W:VA C: !
CONSTITUENT FCC :V CR FE NI MO W:VA C: !
CONSTITUENT HCP :CR FE NI:VA C N: !
CONSTITUENT LIQUID :C V CR FE NI MO W VA:VA C: !
CONSTITUENT CEMENTITE :CR FE:C: !
CONSTITUENT M23C6 :CR FE:C: !
CONSTITUENT M7C3 :CR FE:C: !
CONSTITUENT M6C :FE:W:FE W:C: !
```

```
CONSTITUENT SIGMA :FE:V CR MO:FE V CR MO: !
CONSTITUENT MU-PHASE :FE:MO W:FE MO W: !
CONSTITUENT R-PHASE :FE:MO :FE MO: !
CONSTITUENT GRAPHITE :C: !
$
$TYPE_DEFINITIIONS:
TYPE-DEFINITION 0 SEQ TCPATH\DATA\METDATA\TC-THEREST.TDB !
TYPE-DEFINITION 1 SEQ TCPATH\DATA\METDATA\TC-BCC.TDB !
TYPE-DEFINITION 2 SEQ TCPATH\DATA\METDATA\TC-FCC.TDB !
TYPE-DEFINITION 3 SEQ TCPATH\DATA\METDATA\TC-LIQUID.TDB !
TYPE-DEFINITION 4 SEQ TCPATH\DATA\METDATA\TC-CARBIDES.TDB !
TYPE-DEFINITION M SEQ TCPATH\DATA\METDATA\TC-CURIE-BOHR.TDB !
TYPE-DEFINITION B GES AM-PH BCC MAGNETIC -1 .4 !
TYPE-DEFINITION F GES AM-PH FCC MAGNETIC -3 .28 !
$
$DEFAULT_COMMANDS:
DEFAULT-COMMAND DEF_ELEMENT VA !
DEFAULT-COMMAND REJ_SYS-CONST LIQUID 1 VA !
$DATABASE_INFORMATION:
DATABASE-INFO The following binary and ternary systems are available: '
  FE-CR-NI by Hertzman'
  FE-MO       Fernandez'
  FE-CR-C     Andersson'
  FE-W-C      Gustafson'
  FE-W        Andersson & Gustafson' !
```

Example 2: A Custom Database for the Sb-Sn Binary System

```
$
$ELEMENT, NAME, REF.STATE, ATOMIC-MASS, H0, S0 !
ELEM VA VACUUM              0.0    0.0 0.0 !
ELEM MG HCP(A3)            24.305 0.0 0.0 !
ELEM SB RHOMBOHEDRAL(A7)  121.75   0.0 0.0 !
ELEM SN BCT(A5)           118.69   0.0 0.0 !
ELEM /- ELECTRON-GAS        0      0   0 !
$
$SPECIES, NAME, STOICHIOMETRIC-FORMULA !
SPECIE MG1 MG1!
```

```
SPECIE MG2 MG2!
SPECIE MG2+ MG/+2!
SPECIE SB1 SB1!
SPECIE SB2 SB2!
SPECIE SB4 SB4!
SPECIE SB3- SB/-3!
SPECIE SB5- SB/-5!
SPECIE SN1 SN1!
SPECIE SN4- SN/-4!
$
$PHASE, NAME, TYPE, NR-OF-SUBL, SITES-IN-EACH-SUBL. !
PHASE BCT        Z 1 1.0!
PHASE HCP        Z 1 1.0!
PHASE RHOMBO     Z 1 1.0!
PHASE GAS:G      Z 1 1.0!
PHASE LIQUID:L   Z 1 1.0!
PHASE IONICLIQ:Y Z 2 1 1!
PHASE SPLIQ:Y    Z 2 1 1!
PHASE BMG3SB2:I  Z 2 3 2!
PHASE AMG3SB2:I  Z 2 3 2!
PHASE MG2SN:I    Z 2 2 1!
PHASE SBSN       Z 2 1 1!
PHASE SB2SN3     Z 2 2 3!
$
$CONSTITUENT, PHASE-NAME : CONSTITUENTS !
CONSTITUENT RHOMBO :SB SN:!
CONSTITUENT HCP :MG SN:!
CONSTITUENT BCT :SB SN:!
CONSTITUENT GAS:G :MG1 MG2 SB1 SB2 SB4 SN1:
  > Gas phase, using the Ideal EOS and Mixing Model. !
CONSTITUENT LIQUID:L :SB SN:!
CONSTITUENT IONICLIQ:Y :MG2+:SB SB3- SN SN4- VA:
  > This is the Ionic Liquid Solution Phase. !
CONSTITUENT SPLIQ:Y :MG2+:SB SB3- SN SN4- VA:!
CONSTITUENT BMG3SB2:I :MG2+:SB3- SB5- VA SN4-:!
CONSTITUENT AMG3SB2:I :MG2+:SB3- VA SN4-:!
```

```
CONSTITUENT MG2SN:I :MG2+ VA:SB3- SN4-:!
CONSTITUENT SBSN :SB SN:SB SN:!
CONSTITUENT SB2SN3 :SB:SN:!
$
$DEFAULT_COMMANDS:
DEFAULT-COM DEF-ELEM VA /-!
DEFAULT-COM REJ-PHASE LIQUID!
DEFAULT-COM REJ-PHASE SPLIQ!
$
$TYPE_DEFINITIIONS:
TYPE-DEFINITION Z SEQ * !
$
$DATABASE_INFORMATION:
DATABASE_INFO The Sb-Sn system with isentropic temperatures!
$
$VERSION_DATE:
VERSION_DATE Last update 1986-05-18 11:39:49 !
$
$
$ HERE COMES THE THERMODYNAMIC DATA (expressed in functions & parameters):
$
FUNCTION MGLIQUID 298.15 -4630.90976+192.994374*T-34.0888057*T*LOG(T)
   -36544605.6*T**(-2); 6000 N!
$
FUNCTION MGSOLID 298.15 -8367.34+143.677876*T-26.1849785*T*LOG(T)
   +4.858E-4*T**2-1.393669E-6*T**3+78950*T**(-1);
    923.00 Y -13804.4772 +202.909445*T-34.0888057*T*LOG(T)
   -3.65446056E7*T**(-2) +1.06753982E28*T**(-9); 6000 N!
$
FUNCTION SBLIQUID 298.15 9071.98+146.800*T-31.38*T*LOG(T)
   -2.441646E8*T**(-2); 6000 N!
$
.....
..... <more>
$
FUNCTION LFCT 298.15 -17325.6+5.03600*T; 6000 N!
```

```
FUNCTION GFCTSBSN 298.15 LFCT+SBSOLID+SNSOLID+2948.291+3721.286;
   6000 N!
FUNCTION ISB 298.15 15000; 6000 N!
FUNCTION ISN 298.15 47199.9-95.6270*T; 6000 N!
$
.....
..... <more>
$
PARAMETER G(RHOMBO,SB;0) 298.15 SBSOLID; 6000 N!
PARAMETER G(RHOMBO,SN;0) 298.15 2035+SNSOLID; 6000 N!
$
PARAMETER G(HCP,MG;0) 298.15 MGSOLID; 6000 N!
PARAMETER G(HCP,SN;0) 298.15 32000+SNSOLID; 6000 N!
PARAMETER G(HCP,MG,SN;0) 298.15 -69566-9.23183*T; 6000 N!
$
PARAMETER G(BCT,SN;0) 298.15 SNSOLID; 6000 N!
PARAMETER G(BCT,SB;0) 298.15 1000+SBSOLID; 6000 N!
PARAMETER G(BCT,SB,SN;0) 298.15 0.5*ISB+0.5*ISN; 6000 N!
PARAMETER G(BCT,SB,SN;1) 298.15 0.5*ISB-0.5*ISN; 6000 N!
$
PARAMETER G(IONICLIQ,MG2+:SB3-;0) 298.15 -204389-4.98506*T
   -2.75637E9*T**(-2)+3*MGLIQUID+2*SBLIQUID; 6000 N!
PARAMETER G(IONICLIQ,MG2+:SN4-;0) 298.15 -98639.5+881.073*T
   -174.523*T*LOG(T)-1.79808E9*T**(-2); 6000 N!
PARAMETER G(IONICLIQ,MG2+:SB;0) 298.15 SBLIQUID; 6000 N!
$
......
...... <more>
$
```

Example 3: A Public Database for the Fe-Cr-C Ternary System

```
$
$   Revision history:
$       Created as PDEMO  by Pingfang Shi  on 2004-10-05
$       Renamed to DFeCrC by Pingfang Shi  on 2006-10-25
$
$ FURTHER MODIFICATIONS:
```

```
$ =====================
$ pfs: /20041005 (PDEMO)
$       * Retrieved all definitions from PTERN for the Fe-Cr-C ternary !
$
$ pfs: /20061025 (DFeCrC)
$       * Ignore ELEMENT /- definition.
$       * Modify the L(BCC_A2,FE:C,VA;0) [and L(BCC_A2,CR:C,VA;0)]
$         parameters, in order to avoid BCC-appearance (Fe-C)
$         at temperatures higher than 3900 K.
$           $L(BCC_A2,CR:C,VA;0) 298.15 -190*T;           6000 N REF1 !
$            L(BCC_A2,CR:C,VA;0) 298.15 -190*T; 3000 Y 0; 6000 N REF1 !
$           $L(BCC_A2,FE:C,VA;0) 298.15 -190*T;           6000 N REF3 !
$            L(BCC_A2,FE:C,VA;0) 298.15 -190*T; 3900 Y 0; 6000 N REF3 !
$       * Add Cr into HCP_A3 phase, and
$         add all necessary G0/L parameters (according to SSOL2/TCFE3):
$           G(HCP_A3,CR:VA;0)
$           TC(HCP_A3,CR:VA;0)
$           BMAGN(HCP_A3,CR:VA;0)
$           G(HCP_A3,FE:C;0)
$           G(HCP_A3,CR:C;0)
$           L(HCP_A3,CR:C,VA;0)
$           L(HCP_A3,CR,FE:VA;0)
$       * Add "TMM:300/3000" limits to all three binary joins in the
$           ASSESSED_SYSTEM section.
$       * Change the reference indices, e.g., 101 to REF1, in order to
$           always get the reference list correctly.
$       * Re-arrange database format slightly.
$       * Updated the DATABASE_INFO section.
$-----------------------------------------------------------------------
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ DATABASE INFORMATION:
$***********************************************************************
$
DATABASE_INFO                                  DFeCrC '
                TCS Demo Fe-Cr-C Alloy Solutions Database'
```

```
          (only for demonstration purposes in TCC-Demo/TCW-Demo)'
      ************************************************************ '
                         (Version 1.2, Oct. 2006) ''
          Copyright @ 2004-2012: Thermo-Calc Software AB, Sweden ''
The DFeCrC database (previously called PDEMO) is the TCS Demo Fe-Cr-C '
Alloy Solutions Database, which is especially designed for primarily '
demonstrating the uses of: '
  * TCC-Demo software: BIN, TERN, SCHEIL and some other modules; '
  * TCW-Demo software: '
     "Binary" module calculations of phase diagrams and property diagrams'
                 in available binary systems; '
     "Ternary" module calculations of phase diagrams (e.g., isothermal'
                 sections, monovariant lines involving liquid, liquidus'
                 surface projections) and property diagrams in available'
                 ternary systems; '
     "Scheil" module simulations of alloy solidifications; and/or '
     "Equilibrium" calculation routines for phase diagrams and property'
                 diagrams in available binary or ternary systems. '
It can also be used for other types of calculations/simulations in both'
TCC and TCW software, and in their application programming interfaces.''
It includes critically-assessed data for all possible three binary joins'
(i.e. Cr-C, Fe-Cr and Fe-Cr) and one complete ternary alloy solutions '
within the Fe-Cr-C system. However, due to the fact that experimental'
data for the Fe-C and Cr-C binary joins and Fe-Cr-C ternary system at '
extremely-high temperatures are not sufficient, the available assessed'
data can not be applied to temperature conditions higher than 4000 K.''
Included thermodynamic data are available for various stoichiometric '
and solution phases, e.g. liquid mixture, various alloy solutions and '
intermetallic phases. But the gaseous mixture phase is excluded in this'
DEMO version.''
The DFeCrC database can be used not only in calculating various types '
of phase diagrams (binary phase diagrams, ternary isothermal sections,'
ternary monovariant lines involving liquid, ternary liquidus surface '
projections, ternary isopleth sections, etc.) and property diagrams '
(the easiest ways are through the BIN and TERN modules), but also in '
simulating alloy solidification processes (with the SCHEIL module) of'
```

```
Fe-based or Cr-based alloys (but limited within the Fe-Cr-C ternary '
system in this DEMO version). Many unique features of the TCC/TCW '
software can be demonstrated using this and other specially-designed '
DEMO databases. ''
However, this DFeCrC database (similar to other DEMO and/or PUBLIC TC '
databases) is provided within the TCC-Demo/TCW-Demo software only for '
the purposes of demonstration, testing and evaluation. For R&D projects'
and teaching activities, you are highly encouraged to only use the FULL'
versions of the TCC and/or TCW software together with some commercial '
databases that are appropriate for your specific applications. Please '
contact us for all kinds of details. ''
Release History: Version 1.1 with minor improvements,  Oct. 2004 '
                 Version 1.2 with minor modifications, Oct. 2006 ''
Edited by: Dr. Pingfang Shi (Thermo-Calc Software, Oct. 2006). ''
!

$ ------------------------------------------------------------------------
$VERSION_DATE Last update and adjustment: Pingfang Shi, 2004-10-05 !
VERSION_DATE Last update and adjustment: Pingfang Shi, 2006-10-25 !

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ Definition of Elements in the Database System:
$-------------------------------------------------------------------------
$ELEM NAME STABLE_ELEMENT_REF         ATOMIC MASS  H298-H0     S298 !
$-------------------------------------------------------------------------
$ELEMENT /-   ELECTRON_GAS            0.0000E+00  0.0000E+00  0.0000E+00!
ELEMENT VA   VACUUM                  0.0000E+00  0.0000E+00  0.0000E+00!
ELEMENT C    GRAPHITE                1.2011E+01  1.0540E+03  5.7400E+00!
ELEMENT CR   BCC_A2                  5.1996E+01  4.0500E+03  2.3560E+01!
ELEMENT FE   BCC_A2                  5.5847E+01  4.4890E+03  2.7280E+01!

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ TYPE_DEFINITIONS for data inclusions:
$-------------------------------------------------------------------------
```

```
TYPE_DEFINITION % SEQ *!

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$  TYPE_DEFINITIONS for phase descriptions:
$-------------------------------------------------------------------------
$... For magnetic contributions:
TYPE_DEFINITION A GES AMEND_PHASE_DES @ MAGNETIC  -1.0  4.00000E-01 !
TYPE_DEFINITION B GES AMEND_PHASE_DES @ MAGNETIC  -3.0  2.80000E-01 !

$... For Cr/Fe-C in FCC phase and Cr/Fe-C in HCP phase:
TYPE_DEFINITION C IF(C) THEN GES AMEND_PHASE_DES @ C_S ,, :C: !

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ Default Commands:
$-------------------------------------------------------------------------
$DEFINE_SYSTEM_DEFAULT SPECIES 2 !
DEFINE_SYSTEM_DEFAULT ELEMENT 2 !

DEFAULT_COMMAND DEF_SYS_ELEMENT VA !

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ FUNCTIONS for various phases:
$-------------------------------------------------------------------------
$
FUNCT GHSERCC  298.15 -17368.441+170.73*T-24.3*T*LN(T)
     -4.723E-04*T**2+2562600*T**(-1)-2.643E+08*T**(-2)
     +1.2E+10*T**(-3); 6000 N!
FUNCT GHSERCR  298.15 -8856.94+157.48*T-26.908*T*LN(T)
     +.00189435*T**2-1.47721E-06*T**3+139250*T**(-1);  2180 Y
      -34869.344+344.18*T-50*T*LN(T)-2.88526E+32*T**(-9); 6000 N!
FUNCT GHSERFE  298.15 +1225.7+124.134*T-23.5143*T*LN(T)
     -.00439752*T**2-5.8927E-08*T**3+77359*T**(-1);  1811.00 Y
     -25383.581+299.31255*T-46*T*LN(T)+2.29603E+31*T**(-9); 6000 N!
```

```
FUNCT GFELIQ   298.15 +GHSERFE#+12040.17-6.55843*T
     -3.6751551E-21*T**7;  1811 Y
      -10839.7+291.302*T-46*T*LN(T);  6000 N!
FUNCT GFEFCC   298.15 +GHSERFE#-1462.4+8.282*T
     -1.15*T*LN(T)+6.4E-04*T**2;  1811 Y
      -27098.266+300.25256*T-46*T*LN(T)+2.78854E+31*T**(-9);  6000 N!
FUNCT GFECEM   298.15 +GPCEM1#-10745+706.04*T-120.6*T*LN(T); 6000 N!
FUNCT GFEM23C6  298.15 +7.666667*GFECEM#-1.666667*GHSERCC#
     +66920-40*T; 6000 N!

FUNCT GCRFCC   298.15 +GHSERCR#+7284+.163*T;  6000 N!
FUNCT GCRM23C6 298.15 -521983+3622.24*T-620.965*T*LN(T)
     -.126431*T**2;   6000 N!
FUNCT GCRM3C2  298.15 -100823.8+530.66989*T-89.6694*T*LN(T)
     -.0301188*T**2;  6000 N!
FUNCT GCRM7C3  298.15 -201690+1103.128*T-190.177*T*LN(T)
     -.0578207*T**2;  6000 N!

FUNCT GPCLIQ   298.15 +YCLIQ#*EXP(ZCLIQ#);    6000 N!
FUNCT ACLIQ    298.15 +2.32E-05*T+2.85E-09*T**2;  6000 N!
FUNCT BCLIQ    298.15 +1+3.2E-10*P;               6000 N!
FUNCT CCLIQ    298.15 1.6E-10;                    6000 N!
FUNCT DCLIQ    298.15 +1*LN(BCLIQ#);              6000 N!
FUNCT ECLIQ    298.15 +1*LN(CCLIQ#);              6000 N!
FUNCT VCLIQ    298.15 +7.626E-06*EXP(ACLIQ#);     6000 N!
FUNCT XCLIQ    298.15 +1*EXP(.5*DCLIQ#)-1;        6000 N!
FUNCT YCLIQ    298.15 +VCLIQ#*EXP(-ECLIQ#);       6000 N!
FUNCT ZCLIQ    298.15 +1*LN(XCLIQ#);              6000 N!

FUNCT GPCGRA   298.15 +YCGRA#*EXP(ZCGRA#);    6000 N!
FUNCT ACGRA    298.15 +2.32E-05*T+2.85E-09*T**2;  6000 N!
FUNCT BCGRA    298.15 +1+3.6E-10*P;               6000 N!
FUNCT CCGRA    298.15 3.3E-10;                    6000 N!
FUNCT DCGRA    298.15 +1*LN(BCGRA#);              6000 N!
FUNCT ECGRA    298.15 +1*LN(CCGRA#);              6000 N!
```

```
FUNCT VCGRA    298.15 +5.259E-06*EXP(ACGRA#);     6000 N!
FUNCT XCGRA    298.15 +1*EXP(.9166667*DCGRA#)-1;  6000 N!
FUNCT YCGRA    298.15 +VCGRA#*EXP(-ECGRA#);       6000 N!
FUNCT ZCGRA    298.15 +1*LN(XCGRA#);              6000 N!

FUNCT GPCFCC   298.15 +YCFCC#*EXP(ZFEFCC#);   6000 N!
FUNCT ACFCC    298.15 +1.44E-04*T;              6000 N!
FUNCT VCFCC    298.15 +1.031E-05*EXP(ACFCC#);   6000 N!
FUNCT YCFCC    298.15 +VCFCC#*EXP(-EFEFCC#);    6000 N!

FUNCT GPCRLIQ  298.15 +YCRLIQ#*EXP(ZCRLIQ#);    6000 N!
FUNCT ACRLIQ   298.15 +1.7E-05*T+9.2E-09*T**2;  6000 N!
FUNCT BCRLIQ   298.15 +1+4.65E-11*P;            6000 N!
FUNCT CCRLIQ   298.15 3.72E-11;                 6000 N!
FUNCT DCRLIQ   298.15 +1*LN(BCRLIQ#);           6000 N!
FUNCT ECRLIQ   298.15 +1*LN(CCRLIQ#);           6000 N!
FUNCT VCRLIQ   298.15 +7.653E-06*EXP(ACRLIQ#);  6000 N!
FUNCT XCRLIQ   298.15 +1*EXP(.8*DCRLIQ#)-1;     6000 N!
FUNCT YCRLIQ   298.15 +VCRLIQ#*EXP(-ECRLIQ#);   6000 N!
FUNCT ZCRLIQ   298.15 +1*LN(XCRLIQ#);           6000 N!

FUNCT GPCRBCC  298.15 +YCRBCC#*EXP(ZCRBCC#);  6000 N!
FUNCT ACRBCC   298.15 +1.7E-05*T+9.2E-09*T**2;  6000 N!
FUNCT BCRBCC   298.15 +1+2.6E-11*P;             6000 N!
FUNCT CCRBCC   298.15 2.08E-11;                 6000 N!
FUNCT DCRBCC   298.15 +1*LN(BCRBCC#);           6000 N!
FUNCT ECRBCC   298.15 +1*LN(CCRBCC#);           6000 N!
FUNCT VCRBCC   298.15 +7.188E-06*EXP(ACRBCC#);  6000 N!
FUNCT XCRBCC   298.15 +1*EXP(.8*DCRBCC#)-1;     6000 N!
FUNCT YCRBCC   298.15 +VCRBCC#*EXP(-ECRBCC#);   6000 N!
FUNCT ZCRBCC   298.15 +1*LN(XCRBCC#);           6000 N!

FUNCT GPFELIQ  298.15 +YFELIQ#*EXP(ZFELIQ#);  6000 N!
FUNCT AFELIQ   298.15 +1.135E-04*T;                           6000 N!
FUNCT BFELIQ   298.15 +1+4.98009787E-12*P+3.20078924E-14*T*P; 6000 N!
FUNCT CFELIQ   298.15 +4.22534787E-12+2.71569924E-14*T;       6000 N!
```

```
FUNCT DFELIQ   298.15 +1*LN(BFELIQ#);                           6000 N!
FUNCT EFELIQ   298.15 +1*LN(CFELIQ#);                           6000 N!
FUNCT VFELIQ   298.15 +6.46677E-06*EXP(AFELIQ#);                6000 N!
FUNCT XFELIQ   298.15 +1*EXP(.8484467*DFELIQ#)-1;               6000 N!
FUNCT YFELIQ   298.15 +VFELIQ#*EXP(-EFELIQ#);                   6000 N!
FUNCT ZFELIQ   298.15 +1*LN(XFELIQ#);                           6000 N!

FUNCT GPFEFCC  298.15 +YFEFCC#*EXP(ZFEFCC#);  6000 N!
FUNCT AFEFCC   298.15 +7.3097E-05*T;                            6000 N!
FUNCT BFEFCC   298.15 +1+3.25236341E-11*P+3.36607808E-16*T*P; 6000 N!
FUNCT CFEFCC   298.15 +2.62285341E-11+2.71455808E-16*T;       6000 N!
FUNCT DFEFCC   298.15 +1*LN(BFEFCC#);                           6000 N!
FUNCT EFEFCC   298.15 +1*LN(CFEFCC#);                           6000 N!
FUNCT VFEFCC   298.15 +6.688726E-06*EXP(AFEFCC#);               6000 N!
FUNCT XFEFCC   298.15 +1*EXP(.8064454*DFEFCC#)-1;               6000 N!
FUNCT YFEFCC   298.15 +VFEFCC#*EXP(-EFEFCC#);                   6000 N!
FUNCT ZFEFCC   298.15 +1*LN(XFEFCC#);                           6000 N!

FUNCT GPFEBCC  298.15 +YFEBCC#*EXP(ZFEBCC#);  6000 N!
FUNCT AFEBCC   298.15 +2.3987E-05*T+1.2845E-08*T**2;            6000 N!
FUNCT BFEBCC   298.15 +1+2.80599565E-11*P+3.06481523E-16*T*P; 6000 N!
FUNCT CFEBCC   298.15 +2.20949565E-11+2.41329523E-16*T;       6000 N!
FUNCT DFEBCC   298.15 +1*LN(BFEBCC#);                           6000 N!
FUNCT EFEBCC   298.15 +1*LN(CFEBCC#);                           6000 N!
FUNCT VFEBCC   298.15 +7.042095E-06*EXP(AFEBCC#);               6000 N!
FUNCT XFEBCC   298.15 +1*EXP(.7874195*DFEBCC#)-1;               6000 N!
FUNCT YFEBCC   298.15 +VFEBCC#*EXP(-EFEBCC#);                   6000 N!
FUNCT ZFEBCC   298.15 +1*LN(XFEBCC#);                           6000 N!

FUNCT GPFEHCP  298.15 +YFEHCP#*EXP(ZFEHCP#);  6000 N!
FUNCT AFEHCP 298.15 +7.3646E-5*T;                             6000 N!
FUNCT BFEHCP 298.15 +1+32.5236341E-12*P+3.36607808E-16*P*T; 6000 N!
FUNCT CFEHCP 298.15 +26.2285341E-12+2.71455808E-16*T;       6000 N!
FUNCT DFEHCP 298.15 +LOG(BFEHCP#);                            6000 N!
FUNCT EFEHCP 298.15 +LOG(CFEHCP#);                            6000 N!
FUNCT VFEHCP 298.15 +6.59121E-6*EXP(AFEHCP#);                 6000 N!
```

```
FUNCT XFEHCP 298.15 +EXP(0.8064454*DFEHCP#)-1;             6000 N!
FUNCT YFEHCP 298.15 +VFEHCP#*EXP(-1*EFEHCP#);              6000 N!
FUNCT ZFEHCP 298.15 +LOG(XFEHCP#);                         6000 N!

FUNCT GPCEM1   298.15 +VCEM1#*P;     6000 N!
FUNCT ACEM1    298.15 -1.36E-05*T+4E-08*T**2;    6000 N!
FUNCT VCEM1    298.15 +2.339E-05*EXP(ACEM1#);    6000 N!

FUNCT GPSIG1   298.15 +1.09E-04*P;        6000 N!
FUNCT GPSIG2   298.15 +1.117E-04*P;       6000 N!

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ Define the various Phase and their Constituents, and
$ Assign parameters to phases:
$------------------------------------------------------------------------
$PHASE  NAME:TYPE  MARKCODE  #SUBL  SITES_IN_EACH_SUBL. !
$------------------------------------------------------------------------

$***** LIQUID (Solution) ************************************************
PHASE LIQUID:L %  1  1.0
  > This is metallic liquid solution phase, with C species !
CONST LIQUID:L  : C,CR,FE :  !

PARAM G(LIQUID,C;0)  298.15 +GHSERCC#+GPCLIQ#
                               +117369-24.63*T; 6000 N REF0 !
PARAM G(LIQUID,CR;0)  298.15 +GHSERCR#+GPCRLIQ#
                  +24339.955-11.420225*T+2.37615E-21*T**7;  2180 Y
                              +GHSERCR#+GPCRLIQ#
                  +18409.36-8.563683*T+2.88526E+32*T**(-9); 6000 N REF0 !
PARAM G(LIQUID,FE;0)  298.15 +GFELIQ#+GPFELIQ#;  6000 N REF0 !

PARAM L(LIQUID,C,CR;0)  298.15 -90526-25.9116*T; 6000 N REF1 !
PARAM L(LIQUID,C,CR;1)  298.15 +80000;           6000 N REF1 !
PARAM L(LIQUID,C,CR;2)  298.15 +80000;           6000 N REF1 !
PARAM L(LIQUID,C,FE;0)  298.15 -124320+28.5*T;  6000 N REF3 !
```

```
PARAM L(LIQUID,C,FE;1)  298.15  +19300;         6000 N REF3 !
PARAM L(LIQUID,C,FE;2)  298.15  +49260-19*T;    6000 N REF3 !
PARAM L(LIQUID,CR,FE;0)  298.15 -14550+6.65*T;   6000 N REF4 !
PARAM L(LIQUID,C,CR,FE;0)  298.15 -496063;  6000 N REF2 !
PARAM L(LIQUID,C,CR,FE;1)  298.15  +57990;  6000 N REF2 !
PARAM L(LIQUID,C,CR,FE;2)  298.15  +61404;  6000 N REF2 !

$***** FCC_A1 (Solution) ************************************************
$PHASE FCC_A1  %BC  2 1   1 !
$ Note the C TYPE_DEF for 2nd FCC composition-set (MC) is not necessary
$    for the (Fe,Cr)-C system.
PHASE FCC_A1  %B   2 1   1 !
CONST FCC_A1  : CR,FE%
               : C,VA% :  !

PARAM G(FCC_A1,CR:VA;0)  298.15 +GCRFCC#+GPCRBCC#;  6000 N REF0 !
PARAM TC(FCC_A1,CR:VA;0)     298.15 -1109;  6000 N REF0 !
PARAM BMAGN(FCC_A1,CR:VA;0)  298.15 -2.46;  6000 N REF0 !
PARAM G(FCC_A1,FE:VA;0)  298.15 +GFEFCC#+GPFEFCC#;  6000 N REF0 !
PARAM TC(FCC_A1,FE:VA;0)     298.15 -201;  6000 N REF0 !
PARAM BMAGN(FCC_A1,FE:VA;0)  298.15 -2.1;  6000 N REF0 !

PARAM G(FCC_A1,CR:C;0)  298.15 +GHSERCR#+GHSERCC#
                                  +1200-1.94*T;     6000 N REF2 !
PARAM G(FCC_A1,FE:C;0)  298.15 +GFEFCC#+GHSERCC#+GPCFCC#
                                  +77207-15.877*T;  6000 N REF3 !
PARAM TC(FCC_A1,FE:C;0)     298.15 -201;  6000 N REF3 !
PARAM BMAGN(FCC_A1,FE:C;0)  298.15 -2.1;  6000 N REF3 !

PARAM L(FCC_A1,CR:C,VA;0)  298.15 -11977+6.8194*T; 6000 N REF2 !
PARAM L(FCC_A1,FE:C,VA;0)  298.15 -34671;          6000 N REF3 !
PARAM L(FCC_A1,CR,FE:VA;0) 298.15 +10833-7.477*T;  6000 N REF4 !
PARAM L(FCC_A1,CR,FE:VA;1) 298.15  +1410;          6000 N REF4 !
PARAM L(FCC_A1,CR,FE:C;0)  298.15 -74319+3.2353*T; 6000 N REF2 !

$***** BCC_A2 (Solution) ************************************************
```

```
PHASE BCC_A2  %A  2 1   3 !
CONST BCC_A2  : CR%,FE%
              : C,VA% :  !

PARAM G(BCC_A2,CR:VA;0)  298.15 +GHSERCR#+GPCRBCC#;  6000 N REF0 !
PARAM TC(BCC_A2,CR:VA;0)     298.15 -311.5;  6000 N REF0 !
PARAM BMAGN(BCC_A2,CR:VA;0)  298.15   -.01;  6000 N REF0 !
PARAM G(BCC_A2,FE:VA;0)  298.15 +GHSERFE#+GPFEBCC#;  6000 N REF0 !
PARAM TC(BCC_A2,FE:VA;0)     298.15 1043;  6000 N REF0 !
PARAM BMAGN(BCC_A2,FE:VA;0)  298.15 2.22;  6000 N REF0 !

PARAM G(BCC_A2,CR:C;0)  298.15 +GHSERCR#+3*GHSERCC#+GPCRBCC#+3*GPCGRA#
                               +416000;           6000 N REF1 !
PARAM TC(BCC_A2,CR:C;0)     298.15 -311.5;  6000 N REF1 !
PARAM BMAGN(BCC_A2,CR:C;0)  298.15  -.008;  6000 N REF1 !
PARAM G(BCC_A2,FE:C;0)  298.15 +GHSERFE#+3*GHSERCC#+GPFEBCC#+3*GPCGRA#
                               +322050+75.667*T;   6000 N REF3 !
PARAM TC(BCC_A2,FE:C;0)     298.15 1043;  6000 N REF3 !
PARAM BMAGN(BCC_A2,FE:C;0)  298.15 2.22;  6000 N REF3 !

$ PF-20061025: Modify the L(BCC_A2,FE:C,VA;0) [and L(BCC_A2,CR:C,VA;0)]
$         parametera, in order to avoid BCC-appearance (Fe-C)
$         at temperatures higher than 3900 K.
$PARAM L(BCC_A2,CR:C,VA;0) 298.15 -190*T;           6000 N REF1 !
PARAM L(BCC_A2,CR:C,VA;0) 298.15 -190*T; 3000 Y 0; 6000 N REFS !
$PARAM L(BCC_A2,FE:C,VA;0) 298.15 -190*T;           6000 N REF3 !
PARAM L(BCC_A2,FE:C,VA;0) 298.15 -190*T; 3900 Y 0; 6000 N REFS !
$
PARAM L(BCC_A2,CR,FE:VA;0) 298.15 +20500-9.68*T;   6000 N REF4 !
PARAM TC(BCC_A2,CR,FE:VA;0)     298.15 1650;  6000 N REF4 !
PARAM TC(BCC_A2,CR,FE:VA;1)     298.15  550;  6000 N REF4 !
PARAM BMAGN(BCC_A2,CR,FE:VA;0)  298.15 -.85;  6000 N REF4 !

PARAM L(BCC_A2,CR,FE:C;0)  298.15 -1250000+667.7*T; 6000 N REF2 !
PARAM TC(BCC_A2,CR,FE:C;0)    298.15 1650;  6000 N REF5 !
PARAM TC(BCC_A2,CR,FE:C;1)    298.15  550;  6000 N REF5 !
```

```
PARAM BMAGN(BCC_A2,CR,FE:C;0) 298.15 -.85;  6000 N REF5 !

$***** HCP_A3 (Solution) ***************************************************
$PHASE HCP_A3  %BC  2 1   .5 !
$ Note the C TYPE_DEF for 2nd HCP composition-set (M2C) is not necessary
$    for the (Fe,Cr)-C system.
PHASE HCP_A3  %B   2 1   .5 !
CONST HCP_A3  : CR,FE,
               : VA%,C :  !

PARAM G(HCP_A3,CR:VA;0) 298.15 +GHSERCR#+4438;           6000 N REF1 !
PARAM TC(HCP_A3,CR:VA;0)    298.15 -1109;  6000 N REF1 !
PARAM BMAGN(HCP_A3,CR:VA;0) 298.15 -2.46;  6000 N REF1 !
PARAM G(HCP_A3,FE:VA;0) 298.15 +GHSERFE#+GPFEHCP#
               -3705.78+12.591*T-1.15*T*LN(T)+6.4E-04*T**2; 1811 Y
                                +GHSERFE#+GPFEHCP#
               -3957.199+5.24951*T+4.9251E+30*T**(-9);      6000 N REF0 !
PARAM G(HCP_A3,CR:C;0) 298.15 +GHSERCR#+.5*GHSERCC#
               -18504+9.4173*T-2.4997*T*LN(T)+.001386*T**2; 6000 N REF1 !
PARAM G(HCP_A3,FE:C;0) 298.15 +GFEFCC#+.5*GHSERCC#+GPCFCC#
                                 +52905-11.9075*T;          6000 N REF3 !

PARAM L(HCP_A3,CR:C,VA;0)  298.15  +4165;  6000 N REF1 !
PARAM L(HCP_A3,FE:C,VA;0)  298.15 -22126;  6000 N REF3 !
$PARAM L(HCP_A3,FE:C,VA;0)  298.15 -17335;  6000 N TCFE3 !
PARAM L(HCP_A3,CR,FE:VA;0) 298.15 +10833-7.477*T; 6000 N REF4 !

$***** SIGMA (Solution) ****************************************************
PHASE SIGMA  %  3 8   4   18 !
CONST SIGMA  : FE
              : CR
              : CR,FE :  !

PARAM G(SIGMA,FE:CR:CR;0)  298.15 +8*GFEFCC#+22*GHSERCR#
                                +92300-95.96*T+GPSIG1#;   6000 N REF4 !
PARAM G(SIGMA,FE:CR:FE;0)  298.15 +8*GFEFCC#+4*GHSERCR#+18*GHSERFE#
```

```
                              +117300-95.96*T+GPSIG2#;  6000 N REF4 !

$***** CEMENTITE (Solution) *********************************************
PHASE CEMENTITE  %  2 3   1 !
CONST CEMENTITE  : CR,FE%
                  : C :  !

PARAM G(CEMENTITE,CR:C;0)  298.15 +3*GHSERCR#+GHSERCC#
                                   -48000-9.2888*T; 6000 N REF2 !
PARAM G(CEMENTITE,FE:C;0)  298.15 +GFECEM#;          6000 N REF3 !

PARAM L(CEMENTITE,CR,FE:C;0)  298.15 +25278-17.5*T;  6000 N REF2 !

$***** M3C2 (Compound) *************************************************
PHASE M3C2  %  2 3   2 !
CONST M3C2  : CR
             : C :  !

PARAM G(M3C2,CR:C;0)  298.15 +GCRM3C2#;  6000 N REF2 !

$***** M7C3 (Solution) *************************************************
PHASE M7C3  %  2 7   3 !
CONST M7C3  : CR%,FE
             : C :  !

PARAM G(M7C3,CR:C;0)  298.15 +GCRM7C3#;          6000 N REF2 !
PARAM G(M7C3,FE:C;0)  298.15 +7*GHSERFE#+3*GHSERCC#
                              +75000-48.2168*T; 6000 N REF2 !

PARAM L(M7C3,CR,FE:C;0)  298.15 -4520-10*T;  6000 N REF2 !

$***** M23C6 (Solution) ************************************************
PHASE M23C6  %  3 20   3   6 !
CONST M23C6  : CR%,FE%
              : CR%,FE%
              : C :  !
```

```
PARAM G(M23C6,CR:CR:C;0)  298.15 +GCRM23C6#;           6000 N REF5 !
PARAM G(M23C6,FE:CR:C;0)  298.15 +.1304348*GCRM23C6#
                                   +.8695652*GFEM23C6#; 6000 N REF5 !
PARAM G(M23C6,CR:FE:C;0)  298.15 +.8695652*GCRM23C6#
                                   +.1304348*GFEM23C6#; 6000 N REF5 !
PARAM G(M23C6,FE:FE:C;0)  298.15 +GFEM23C6#;           6000 N REF5 !

PARAM L(M23C6,CR,FE:CR:C;0) 298.15 -205342+141.6667*T; 6000 N REF2 !
PARAM L(M23C6,CR,FE:FE:C;0) 298.15 -205342+141.6667*T; 6000 N REF2 !

$***** GRAPHITE (Compound) ******************************************
PHASE GRAPHITE  %  1  1.0  !
CONST GRAPHITE  : C :  !

PARAM G(GRAPHITE,C;0)  298.15 +GHSERCC#+GPCGRA#; 6000 N REF0 !

$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$ ASSESSED_SYSTEMS information:
$    *  necessary for the BIN and TERN modules in TCC
$                     the Binary Phase Diagram module in TCW
$                 and the Equilibrium calculation routine in TCW
$-----------------------------------------------------------------------
ASSESSED_SYSTEM
$... Binary ...
  C-FE(;P3 TMM:300/4000 *)
  C-CR(;P3 TMM:300/4000 *)
  CR-FE(;G5 MAJ:BCC/FE:VA C_S:BCC/CR:VA
        ;P3 TMM:300/3000 STP:.6/1200/1/-2/2)
$ CR-FE(TDB -HCP ;G5 MAJ:BCC/FE:VA C_S:BCC/CR:VA
$       ;P3 TMM:300/3000 STP:.6/1200/1/-2/2)
$
$... Ternary ...
  C-CR-FE(;G5 MAJ:BCC/FE:VA C_S:BCC/CR:VA
          ;P3 STP:.4/.5/1 STP:0.1/0.5/1)
```

```
!


$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$ LIST OF REFERENCE:
$
$------------------------------------------------------------------------
LIST_OF_REFERENCES
NUMBER  SOURCE
REF0 'Alan Dinsdale, SGTE Data for Pure Elements, Calphad Vol 15(1991)
       p 317-425, also in NPL Report DMA(A)195 Rev. August 1990'
REF1 'J-O Andersson, Calphad Vol 11 (1987) p 271-276, TRITA 0314; C-CR'
REF2 'Byeong-Joo Lee, unpublished revision (1991); C-Cr-Fe-Ni'
REF3 'P. Gustafson, Scan. J. Metall. vol 14, (1985) p 259-267
       TRITA 0237 (1984); C-FE'
REF4 'J-O Andersson, B. Sundman, CALPHAD Vol 11, (1987), p 83-92
       TRITA 0270 (1986); CR-FE'
REF5 'J-O Andersson, Met. Trans A, Vol 19A, (1988) p 627-636
       TRITA 0207 (1986); C-CR-FE'
REFS 'Pingfang Shi (2006), TCS PTERN Public Ternary Alloys Database,
       v1.2; Modified L0(BCC,Fe,C) and L0(BCC,Cr,C) parameters
       at high temperatures.'
!
```

The Database Checker

In this section:

About the Database Checker

The Thermo-Calc software package includes a program to check that the syntax of unencrypted database files is correct. The program applies the syntax rules set out in the *Database Manager User Guide* and reports errors and issues warnings. This program is intended for advanced users who develop and manage databases.

Thermo-Calc accepts deviations from these syntax rules. This means that a database can work even if the Database Checker reports errors and warnings. For example, an error is reported if an abbreviated phase name is found, but phase name abbreviations are accepted by Thermo-Calc and its add-on modules.

The executable **DatabaseChecker** file is found in the Thermo-Calc home directory. The program can also be launched by selecting **Tools→ Database Checker** from the main menu.

Database Checker Revision History below for releases notes that may assist database developers.

Database Checker Revision History

The DatabaseChecker application is standalone and need not be installed separately, just download the containing zip file, open it and follow the instructions in the README.TXT file. The program requires that Java is installed on the target machine. Please note that large amounts of RAM may be required when checking large databases.

Release Notes Version 1.4

- The DatabaseChecker user interface has been modified to allow checking of several database files in one batch. Instead of selecting an individual database file, the user now selects a directory that contains databases. The application then shows a tree structure with all database files in the chosen directory and its subdirectories. The user may then select any number of these files for processing.
- A check of parameters whose interaction order is 3 has been introduced. For such parameters, there must either exist one parameter with the interaction degree 0 or three parameters with the interaction degrees 0, 1 and 2.
- To facilitate optimization of databases, the tool now generates information messages about functions and parameters that will always return 0.

Release Notes, Version 1.3.2

- The tool now detects conflicts between identical parameters even if one of them have been defined with the G prefix and the other with the L prefix.

Release Notes, Version 1.3

A bug prevented integers bigger than 32 bits from being parsed. This has been corrected.

The tool now checks that the species and number of sublattices used in parameter definitions match the phase definition and generates an error message for every mismatch. Ionic liquid parameters are currently exempted from this check as not to break the widespread use of defining only one constituent for ionic liquid parameters.

Type definitions and case statements are now parsed for errors. However, only a subset of all possible GES, POLY and TDB commands are accepted and all other commands will result in an error message. The accepted commands are:

- GES AMEND_PHASE_DESCRIPTION
- COMPOSITION_SETS
- MAJOR_CONSTITUENTS
- DISORDERED_PART
- MAGNETIC_ORDERING
- STATUS_BITS
- DEFAULT_STABLE
- FRACTION_LIMITS
- EXCESS_MODEL

- REDLICH-KISTER_MUGGIANU
- HKF
- HKF_ELECTROSTATIC
- DEBYE_HUCKEL
- GES CHANGE_STATUS
- TDB DEFINE_SYSTEM_ELEMENTS
- TDB RESTORE_PHASES
- TDB REJECT_PHASES

Some improvements to warning and error messages have been made to facilitate finding errors.

Release Notes, Version 1.2

- The tool now uses parallel processing of database files on multi-core systems. This significantly improves performance when large databases are checked.
- It is no longer required that inter-dependent database statements are defined in a specific order in the database file (e.g., species need not any longer be defined before the elements on which they depend). However, for the sake of readability, it is still advised that the database statements occur in a natural order in the file.
- Because of the parallel processing of database statements, multiple definition errors (cases where one name has been used to define more than one entity in the database e.g, two identically named species) may now be generated for any of the violating occurences and not only for the last occurence in the file.

Release Notes, Version 1.1

- Phase names were not properly recognized in parameter definitions if there was one or more whitespace characters between the opening parenthesis and the phase name. This has been corrected.
- The error message Could not parse parameter, <species name> has been given without a sublattice number but is defined in more than one sublattice is now displayed only for phases where the general diffusion model is used. Note: this implementation is temporary and may be subject to changes later.
- The meaning of the information message The phase <phase name> expects the following parameters: [<parameter names>] was unclear to users. It is now longer shown for the parameter names T and P. For all other names, the message is now displayed as an error message with the following text: The phase <phase name> contains references to the following undefined functions: [<function names>].

- The tool now looks for function names that are abbreviations of other function names and warns about this, since abbreviated names may cause problems during the execution of macro files that have been auto-generated from GES.
- The tools also warns for functions whose names are in the format V<integer>, since this may cause conflicts with names that are used by PARROT.
- Parameter names are now shown in a format that is less cryptic and more resembles the format used in the database. However, the format of diffusion parameters may still differ somewhat from what is used in the database.

Release Notes, Version 1.0

- The tool parses a TDB file and produces a log with error, warning and information messages about the state of the file
- The message log can be exported and saved as a separate file for later analysis
- The tool is based on the database command definitions given in TC_Database_ManagersGuide and DICTRA27_UsersGuide. All commands in these documents are supported. Some undocumented DIFFUSION commands are not supported.
- Databases that span several files are correctly handled, also blocked files
- The tool is in some aspects more strict than required by the guides (see Additional Information below)
- The tool performs the following checks:
- Syntax control of database statements: ERROR
- Line size: ERROR or WARNING depending on context
- Multiply defined elements, species or phases: ERROR
- Multiply defined functions and parameters: ERROR
- Multiply defined assessed system: ERROR
- Multiply defined references: ERROR
- References entries that are never used: WARNING
- References to non-existent type definitions: ERROR
- Type definitions that are never used: WARNING
- Unused functions and parameters: WARNING
- Functions and parameters with undefined references: WARNING
- References to undefined elements, species or phases: ERROR

- Dependencies to external names from a phase: INFORMATION. (All parameters of a phase are evaluated with respect to their dependencies to other parameters and functions and all unresolved names are listed. Typically, this should only yield dependencies on T and possibly P)

Additional Information

In addition to the command definitions given in the user guides listed above, the following rules are also applied:

- Names are case-sensitive, meaning that e.g., the defined element MN cannot be referred to under the name Mn;
- Phase names cannot be abbreviated. When referred to in parameter descriptions their full names must be used.

DATAPLOT User Guide

Thermo-Calc Version 2016a

Introduction to the DATAPLOT User Guide

In this section:

Using this Guide

The *DATAPLOT User Guide* gives all the details on the graphical language DATAPLOT that a phase diagram or property diagram calculated by Thermo-Calc or a kinetic profile simulated by the Diffusion Module (DICTRA) can be defined.

This guide

- Discusses the important features of the DATAPLOT language.
- Describes the commands to define PROLOGUE and DATASET.
- Several examples and the standard codes for formatting are provided.

A topic introduce the method of formatting DIGLIB symbols in LaTeX documents, for the purpose of necessarily/appropriately referring to the corresponding LaTeX symbols (closest to those DIGLIB symbols which have been plotted on a diagram using the DATAPLOT Graphical Language) in the texts of LaTeX documents for publications/reports.

About the DATAPLOT Graphical Language

In order to obtain graphical output of any numerical data and informative strings, a graphical language called DATAPLOT was developed in connection with the graphical software DIGLIB. Using this graphical language, you can store information in a normal text file (*.EXP), which can be plotted as graphical symbols, lines, texts or Greek letters on any plot device support by DIGLIB.

You can generate and plot DATAPLOT (*.EXP) files together with various calculation and/or experimental results from, for example, phase and property diagrams calculated with the Thermo-Calc software composition profiles or other diagrams simulated by the Diffusion Module (DICTRA).

An EXP file automatically generated by the POST-processor (using the *MAKE_EXPERIMENTAL_DATAFILE* on page 199 command) in the Thermo-Calc software is a DATAPLOT (*.EXP) file and may contain all types of legal DATAPLOT commands and parameters. With a text editor, you can modify or add some DATAPLOT commands and related parameters in an existing EXP file. This is useful when appending experimental information to calculated/simulated plots, and when specifying user-defined texts, symbols, colors, fonts, filled patterns, diagram types, diagram sizes, symbol/character sizes, titles, special characters, etc.

DATAPLOT Examples

To help you prepare your EXP (experimental) files to use for your plotted diagrams generated by Thermo-Calc simulations, there are examples of EXP files using the DATAPLOT graphical language standard, which are normally independent of software versions.

Using DATAPLOT Graphical Language

In this section:

DATAPLOT File Structure

A DATAPLOT file is a normal text file with the extension `EXP` that can be created with a text editor or by a program. The file must contain one or more `DATASETS` and possibly also one or more `PROLOGUES`. Each `PROLOGUE/DATASET` is an entity that can be individually selected for plotting.

A PROLOGUE/DATASET is identified by a unique positive number in the file. A PROLOGUE normally contains various DATASET commands for defining information about axis scaling, axis text, axis length, title and so on. A PROLOGUE is terminated by another PROLOGUE or by the first DATASET. This means that all PROLOGUES must be placed at the beginning of the file, before the first DATASET.

A DATASET normally contains DATASET commands that are associated to separate data points, as well as with one or more `BLOCKS` of data (calculated or experimental). A DATASET is terminated by another DATASET command or the end of file.

DATAPLOT Language Syntax

The DATAPLOT language consists of commands with or without parameters:

```
COMMAND {parameter(s)}
```

The basic graphical command consists of an X/Y coordinate pair and a Graphical Operation Code (`GOC`). With other commands, the interpretation of this basic command can be modified. There are separate commands to draw a polygon or a function and ways to get texts in different fonts.

To edit a DATAPLOT file, the graphical commands can be abbreviated.

A command (with parameters) must not exceed 80 characters. If it is too long (normally as writing necessary codes in a command's parameters for a complex expression), two or more lines can be edited.

Coordinate Systems

The DATAPLOT language accepts coordinates in three different coordinate systems, which are called *word*, *virtual* and *normalized*.

- The *word* coordinates are selected to represent any kind of data and be of "any" magnitude.
- The *virtual* coordinate system uses centimeters as units. However, the actual size of one unit is dependent on the implementation of the device driver in DIGLIB. It is not recommended to use this coordinate system if different output devices are used for preliminary and final plots.
- The *normalized* coordinate system goes from zero to one. When plotting, you interactively scale each axis by selecting the minimum and maximum word coordinates on the axis. In the normalized coordinate system, the minimum axis value is represented by

zero and the maximum by one. DIGLIB draws a square between the four points (of the X and Y axes) that are determined by the coordinates zero and one in the normalized coordinates.

It is also possible to draw triangular plots as described below. However, in most places, it only references square diagrams. All data points within the minimum and maximum word coordinates are plotted inside this square. DIGLIB also writes tic marks and corresponding word values at such tic marks.

It may be convenient to use normalized coordinates to draw boxes and texts. You can give normalized coordinates outside zero and one if you want to write texts outside the area enclosed by the square. To ensure proper operation outside the normalized box, the clipping must be turned off.

Graphical Operation Codes

The Graphical Operation Code (`GOC`) determines how the coordinates are interpreted and what is done at the point determined by the coordinates. For an individual data point, its GOC codes must be given. For each data *BLOCK* on page 19, you define a default GOC that is used for the whole BLOCK, unless a GOC is explicitly defined for a specific point.

- If `S` is omitted, no symbol is plotted. A number selecting the symbol can optionally follow the character `S`. The DIGLIB software determines which symbol the number represents. See *Plotting a Symbol* on page 9.
- If any of the other characters are omitted, the default defined for the BLOCK is used. Absolute values imply the current coordinates; relative values mean that these are added to the current coordinates. The GOC must be written without any space between various characters, but the order for the characters is irrelevant.
- If no `GOC` is defined for a BLOCK, the system default is `MWA`, (Move Word Absolute), i.e., move to the given point that is interpreted as absolute word coordinates. `GOC=DNA` means draw a line from the current point to the new one interpreting the coordinates as normalized. After each draw or move operation, the so-called current point is the new point.
- The *TEXT* on page 27 following the `'` command is interpreted and expanded by the DIGLIB software, if it includes any ^ operator or *STRING* on page 25 names using the ~ descriptor. To get PostScript outputs, the TEXT must be edited by the LaTeX Text Formatting Program instead (see *LaTeX Formatting Codes* on page 12).

GOC Character Definitions

The GOC is a combination of the following characters:

Character	Meaning
`W`, `V` or `N`	To use word (`W`), virtual (`V`) or normalized (`N`) coordinates. `W` is the default.
`A` or `R`	XY are absolute (A) or relative (R) values. `A` is the default.
`M` or `D`	To perform a move (M) or draw (D) operation to XY. `M` is the default.
`B`	To apply “soft” splines on a line drawn between the coordinate pairs (used only on BLOCK data).
`S`	To plot default symbol at XY.
`S#`	To change default symbol to No # symbol and plot it at XY.
`‘TEXT`	To plot `TEXT` at XY. It must be the last operation in a GOC code. The TEXT following the ’ command is interpreted and expanded by the DIGLIB software, if it includes any ^ operator or STRING names using the ~ descriptor.

Tables or Blocks

In many cases, you have (calculated or experimental) data in the form of tables, and want to plot one or several columns as X-axis and one or more columns as Y-axis. A single table can contain many sets of data records of the same kind of information for a specific data *BLOCK* on page 19. There can be many data BLOCKS in a *DATASET* on page 19.

By enclosing these tables in a data BLOCK, you can select which column(s) is the X-axis and which is the Y-axis. For example,

```
BLOCK X=C1; Y1=C3; Y2=C2; GOC=C4,DAW
```

Where the X-axis values are in column 1, the Y-axis values are in columns 2 and 3, and any GOC codes are in column 4. The default GOC for this BLOCK is Draw Absolute Word. The GOC code inside the table is only necessary if the default GOC is not applicable.

A more elaborate use of the table is shown in this example:

```
BLOCK X=C3; Y=1E3/(C3+273); GOC=C8,MAWS1
```

Where the X-axis values are in column 3, the Y-axis values are in column 1 with some additional calculations applied. Any GOC is in column 8, and the default GOC is Move Absolute Word and plot the symbol no. 1.

Columns in a table must be separated by one or more space characters and do not have to be justified.

It is possible to have tables with mixed text and numbers, but each word followed by a space is counted as one column. The columns used for plotting also must be numerical. An example of a legal line in a table is

```
298.15 This_is_the_second_column 11.5 This_is_the_fourth_column
```

> A line in the table must not exceed 80 characters. A BLOCK must be terminated by a line with the *BLOCKEND* on page 20 command.

Drawing a Polygon

Normally, each point is written on a separate line. But in order to draw a line in a more compact way, use the command *DRAWLINE* on page 21. DRAWLINE is followed by a couple of X/Y number pairs. The X/Y pairs must be separated by a space, and there must be a comma sign between the X and Y values. DRAWLINE makes a move operation to the first pair of X/Y coordinates, and then draws a line among all pairs up to the last one. All pairs must fit on one line of 80 characters, but there can be several consequent DRAWLINE commands.

Drawing an Analytical Function

Use a *BLOCK* on page 19 command to set an axis to a function. It is not necessary to use a value from any column in order to compute the function value to be plotted. Use the *FUNCTION* on page 27 command to plot a function with an even increment of the independent variable.

Painting of an Enclosed Area

Use the *PAINT* on page 27 command to paint or fill an area in a specified pattern in the plot. Available patterns are determined by the DIGLIB software. A related command to PAINT is *PCFUNCTION* on page 27.

Writing a Text

Use the *TEXT* on page 27 command to write a text at the current point. You can write a text at any X/Y pair by appending a single quote followed by the text on the same line. For example:

```
1.1 1.0 NAM'This is a text
```

writes `This is a text` at the normalized coordinates (`1.1, 1.0`).

Use the command *FONT* on page 24 to select the font used for the text, and *CHARSIZE* on page 22 to select the size of the characters. Set this size of the symbols with *SYMBOLSIZE* on page 22.

If a text or a single character should be of a different font than all the other text, or to use subscripts or superscripts in a text, use the ^ operators or STRING command to create the text. The command *STRING* on page 25 stores the text in a specified variable that includes all text-formatting information defined by the DIGLIB software. See *Example 3 – Using Strings and Various Line Types* on page 32 and *Example 5 – Use Included Files for Predefined Symbols* on page 34.

However, if a graphical output is done on a PostScript device using the PostScript hardcopy fonts, special text formatting codes as *LaTeX Formatting Codes* on page 12 should be used and the STRING formatting syntax is then not valid.

Plotting a Symbol

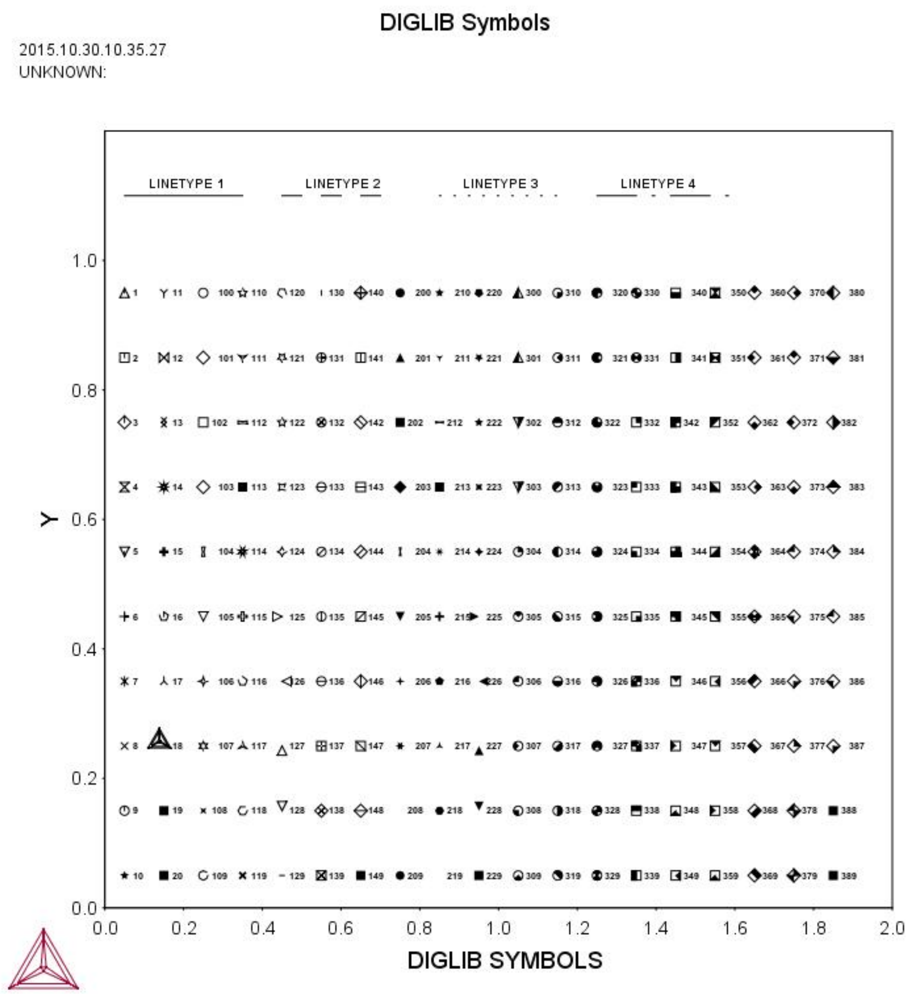
As described in *Graphical Operation Codes* on page 6, a GOC code in a data BLOCK may contain an `S` option to plot a symbol for an X/Y pair or the same symbols for the data BLOCK. A number selecting the symbol can optionally follow the character `S`. You can plot a symbol at any current X/Y position by appending a quote specified by the symbol number in the GOC code (e.g., `1.1 1.0 MANS5'This is a text`).

You can also insert DIGLIB symbols into LaTeX-edited documents.

DIGLIB symbols available in Thermo-Calc

The image summarizes all the standard symbols available in the DIGLIB software. A default symbol is the current symbol in the run of the software (it is usually the No 1 symbol if the POST-processor is switched on for the first time). S (i.e., # is not specified) means that the current symbol is plotted. S0 (i.e., #=0) means that no symbol is plotted.

All the DIGLIB symbols work properly and give good output results for the PostScript format.



Other Commands

Use the *LINETYPE* on page 21 command to define line types as solid, long dashed, short dashed or dotted. If you are using a color device, use the *COLOR* on page 22 command to change color of the lines. On some black and white devices, colors are simulated with different width and dashing of the lines.

- Use the *CLIP* on page 20 command to change the default where all data outside the normalized coordinates zero and one are not plotted.
- When plotting symbols representing various experimental data, it is important that the symbols are centered around the coordinate values.
- Use the *ATTRIBUTE* on page 20 command to change the coordinates where text is displayed. The default is the lower left corner of the first character in the text.
- Use the *INCLUDE* on page 28 command to create libraries with texts and include these in similar plots.

- The dollar sign $ as the first character of a line stands for a comment character, which means the whole line is ignored when plotting.

Interactive Plotting

The DATAPLOT file is read into the POST module in the workspace with *APPEND_EXPERIMENTAL_DATA* on page 192 or *QUICK_EXPERIMENTAL_PLOT* on page 201. These commands ask for the name of the DATAPLOT file and also which PROLOGUE(S) and DATASET(S) are to be plotted.

See the *Thermo-Calc Command Mode Reference Guide* for details about these commands.

By giving the PROLOGUE/DATASET number as `-1`, you get a list of the available PROLOGUES/DATASETS in the file. Note that if `DATASET 0` is in a DATAPLOT file, its data is always used even if other DATASETS are chosen.

Formatting DIGLAB Symbols in LaTeX Documents

When using the LaTeX editor, you can use the DIGLIB symbols in texts and in figures. This then refers to the corresponding LaTeX symbols (closest to those DIGLIB symbols which are plotted on a plot using the DATAPLOT Graphical Language) in the texts of LaTeX documents for publications/reports.

This topic shows how to generate some DIGLIB symbols in texts, through the attached LaTeX source file and its converted jpg file.

DIGLIB_Sym.tex

```
\documentclass[dvips,12pt]{article}
\textwidth 165mm
\textheight 225mm
\oddsidemargin 1mm
\evensidemargin 1mm
\topmargin 1mm
%%\usepackage{amssymb}
%% next replace amssymb and to get udtimes
\usepackage[utopia]{mathdesign}
\usepackage{rotating}
\usepackage[latin1]{inputenc}
\usepackage{graphics}
\usepackage{graphicx,subfigure}% with figures
%\usepackage[draft]{graphicx}% without figures
\usepackage{subfigure}% with figures
\topmargin 1mm
\oddsidemargin 1mm
```

```
\evensidemargin 1mm
\begin{document}
{\Large \bf Diglib symbols and their corresponding LaTeX symbols}
\vspace{5mm}
```

The information below gives the closest corresponding LaTeX symbol. All symbols (except +) must be generated in math mode. Most of these require the package `amssymb`, i.e., you need a directive `usepackage{amssymb}` in the preamble. Two of the symbols require the more extensive `mathdesign` that can be included with `usepackage[utopia]{mathdesign}`.

```
\vspace{5mm}
{\Large
\begin{tabular}{llll}
Diglib & Latex & Latex name & Note\\
1& $\vartriangle$& $\backslash$vartriangle & amssymb \\
2& $\square$& $\backslash$square\\
3& {\Huge $\diamond$} & $\backslash$diamond & size $\backslash$Large \\
4& $\udtimes$& $\backslash$udtimes & mathdesign \\
5& $\triangledown$& $\backslash$triangledown & amssymb\\ 6 & +& normal + \\
7& $\ast$& $\backslash$ast & amssymb\\
8& $\times$& $\backslash$times \\
9& {\Huge $\circ$}& $\backslash$circ & size $\backslash$Large \\
10& {\Huge $\star$}& $\backslash$star & size $\backslash$Large, amssymb\\
11& $\curlyvee$& $\backslash$curlyvee & amssymb\\
12& $\Join$& $\backslash$Join\\
13&& - & nothing similar, overlapping $> <$\\
14&& - & nothing similar, 10-edged star \\
15& $\maltese$& $\backslash$maltese& mathdesign \\
16&& - & nothing similar, a pentagon \\
17& $\curlywedge$& $\backslash$curlywedge & amssymb\\
\end{tabular}}
\end{document}
```

LaTeX Formatting Codes

For a list of supported LaTeX commands, see: http://www2.ph.ed.ac.uk/snuggletex/documentation/math-mode.html

By default when using the LaTeX command, you are in math mode and Thermo-Calc automatically interprets the entered text as math. The above link uses the dollar sign ($) in all the examples but this is NOT required. For example, if you enter `$E=mc^2$` this is displayed including the $ instead of the equation.

As of Thermo-Calc 2015a the LTEXT text formatting is replaced by LaTeX. For existing users who may have plots and files that use the LTEXT format, and especially when using Console Mode, the following examples show the command changes from LTEXT and its LaTeX equivalent:

LTEXT vs LaTeX Commands for Labels

Symbol or text format	LTEXT	LaTeX
Subscripts in an axis text	S-A-TEXT Y N Mole-fraction Al^DO2O^DO3	S-A-TEXT Y N \latex Mole-fraction Al_2O_3
Label with subscript	add .13 .15 n c-ZrO^DO2$	add .13 .15 n \latex c-ZrO_2
Greek symbol	ADD .05 .50 N ^GRd$	ADD .05 .50 N \latex \delta

PROLOGUE Commands

In this section:

PROLOGUE

PROLOGUE is the beginning of a number of consequent lines of user-defined diagram layout commands. The lines are displayed with the `optional text` when using the `-1` option in the POST-processor as prompted for the PROLOGUE number with *APPEND_EXPERIMENTAL_DATA* on page 192 or *QUICK_EXPERIMENTAL_PLOT* on page 201.

Syntax	`PROLOGUE # optional text`
	`#` is an unsigned integer identifying the PROLOGUE.

XSCALE

XSCALE sets the scaling in word coordinates of the X-axis.

Syntax	`XSCALE min max`
	`min` and `max` are real numbers.

YSCALE

YSCALE sets the scaling in word coordinates of the Y-axis.

Syntax	`YSCALE min max`
	`min` and `max` are real numbers.

XTEXT

XTEXT sets the X-axis text.

Syntax	`XTEXT text`
	`text` is an arbitrary text string that may contain text-formatting codes.

YTEXT

YTEXT sets the Y-axis text.

Syntax	`Y TEXT text`
	`text` is an arbitrary text string that may contain text-formatting codes.

XTYPE

XTYPE sets the X-axis type as linear (default), logarithmic or inverse.

Syntax	`XTYPE type`
	`type` is a character string reading LIN, LOG or INV.

YTYPE

YTYPE sets the Y-axis type as linear (default), logarithmic or inverse.

Syntax	`Y TYPE type`
	`type` is a character string reading LIN, LOG or INV.

XLENGTH

XLENGTH sets the X-axis length to approximately # centimeters.

Syntax	`XLENGTH #`
	`#` is a positive real number (the approximate X-axis length in centimeters).

YLENGTH

YLENGTH sets the Y-axis length to approximately # centimeters.

Syntax	`Y LENGTH #`
	`#` is a positive real number (the approximate Y-axis length in centimeters).

TIC_TYPE

TIC_TYPE sets the relative length of the tic marks.

Syntax	`TIC_TYPE #`
	`#` is a real number. Default value is 1. Negative number gives tics on the inside of the diagram frame. 0 gives no tics.

TITLE

TITLE sets the title text string to be printed above the diagram.

Syntax	`TITLE text`
	`text` is an arbitrary text string that may contain text-formatting codes

DIAGRAM_TYPE

DIAGRAM_TYPE sets the diagram type to square (the default) or triangular.

Syntax	`DIAGRAM_TYPE type plot_3rd_axis clip_along_third-axis`
	`type` is a character string reading SQUARE (default) or TRIANGULAR. If `type` reads TRIANGULAR, then two additional parameters are given: `plot_3rd_axis` and `clip_along_third-axis` that are characters strings reading YES or NO.

DATASET Commands

In this section:

DATASET

DATASET indicates the beginning of a number of consequent lines comprising a set of user-defined data. The DATASET lines are displayed on the terminal along with the `optional text` when using the `-1` option in the POST-processor as prompted for the DATASET number in the *APPEND_EXPERIMENTAL_DATA* on page 192 or *QUICK_EXPERIMENTAL_PLOT* on page 201 command.

Syntax	`DATASET # optional text`
	`#` is an unsigned integer identifying this set of data.

BLOCK

BLOCK defines how the following numeric data block shall be interpreted. The definitions of X and Y coordinates may also be expressed as a function of the column values, making it possible to perform transformations.

Syntax	`BLOCK X&=C#; ...; Y&=C#; ...; GOC=C#,@@@...`
	`&'` are optional unsigned integers that make it possible to plot several (maximum 9) X- or Y- axis columns. `#'` are unsigned integers identifying the column numbers. The column number # in "GOC=C#" is the location of any possible GOC codes in the current data BLOCK; @@@ stands for the default Graphical Operation Code (GOC) for the current BLOCK. The GOC code inside the current table is only necessary if the current default GOC is not applicable.

GOC Characters

The GOC is a combination of the following characters:

Character	Meaning
`W`, `V` or `N`	To use word (`W`), virtual (`V`) or normalized (`N`) coordinates. `W` is the default.
`A` or `R`	XY are absolute (A) or relative (R) values. `A` is the default.
`M` or `D`	To perform a move (M) or draw (D) operation to XY. `M` is the default.
`B`	To apply "soft" splines on a line drawn between the coordinate pairs (used only on BLOCK data).
`S`	To plot default symbol at XY.
`S#`	To change default symbol to No # symbol and plot it at XY.

Character	Meaning
`'TEXT`	To plot `TEXT` at XY. It must be the last operation in a GOC code. The TEXT following the ' command is interpreted and expanded by the DIGLIB software, if it includes any ^ operator or STRING names using the ~ descriptor.

BLOCKEND

Syntax	BLOCKEND
	BLOCKEND terminates the local definition of the graphical operation code defined by the earlier BLOCK command.

DATAPOINT

DATAPOINT is not a DATASET command, but the basic DATAPLOT command (see synopsis below) performs an action at the current point determined by the specified X/Y- coordinates. A DATASET may contain various data points, in addition to one or more data *BLOCK* on the previous pages. Such data points are separated and independent on each other.

Syntax	`X Y GOC`

	X and Y are unsigned real numbers identifying the X/Y-coordinates for the current data point. GOC stands for Graphical Operation Code (GOC) for the current point. Legal GOC characters.
	Example
	`0.7 0.95 N'Example 6`
	`0.5 0.08 MNA'E^FS18^SQ(^SK^FS10A+5#8*C#^FS10 -!a^FS18)^FS11+B^DIa#b#$`

CLIP

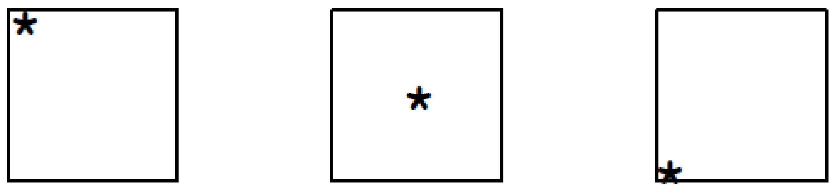
CLIP turns clipping on or off. If it is OFF, it allows output outside the ordinary plot area defined by normalized coordinates zero and one.

Syntax	`CLIP CLP`
	`clp` is a character string reading ON or OFF.

ATTRIBUTE

ATTRIBUTE specifies where the current XY position is in the character or symbol plotbox.

Syntax	ATTRIBUTE
Prompts	`attribute`
	`attribute` may be TOP, CENTER or BOTTOM. ATTRIBUTE CENTER is default for symbols; ATTRIBUTE BOTTOM is default for characters.



TOP CENTER BOTTOM

LINETYPE

LINETYPE redefines the current linetype in the plot.

Syntax	`LINETYPE #`
	`#` must be an unsigned integer. Legal linetypes are (for both normal graphical outputs and for PostScript formats): • Solid (default) • Long dashed • Short dashed • Dotted

DRAWLINE

DRAWLINE draws a line starting at (x1,y1) to (xn,yn) through (x2,y2) → (x(n-1), y(n-1)).

Syntax	`DRAWLINE x1,y1 x2,y2 .... xn,yn`

	`x` and `y` may be reals or integers of any value. This is the same as connecting all points in a table:
	`x1 y1 M`
	`x2 y2 D`
	`... ... D`
	`xn yn D`

CHARSIZE

CHARSIZE redefines the default size of the characters in the plot. The character size has an initial default value, which may vary with the current font setting and the output device (the plot format).

Syntax	`CHARSIZE $$`
	`$$` must be an unsigned real.

SYMBOLSIZE

SYMBOLSIZE redefines the current symbol size setting. The symbol size has an initial default value.

Syntax	`SYMBOLSIZE $$`
	`$$` must be an unsigned real.

GLOBALSIZE

GLOBALSIZE redefines the default global size of the plot. The global size has an initial default value.

Syntax	`GLOBALSIZE $$`
	`$$` must be an unsigned real.

COLOR

COLOR redefines the current color setting. There are 22 legal color codes (from 0 to 21).

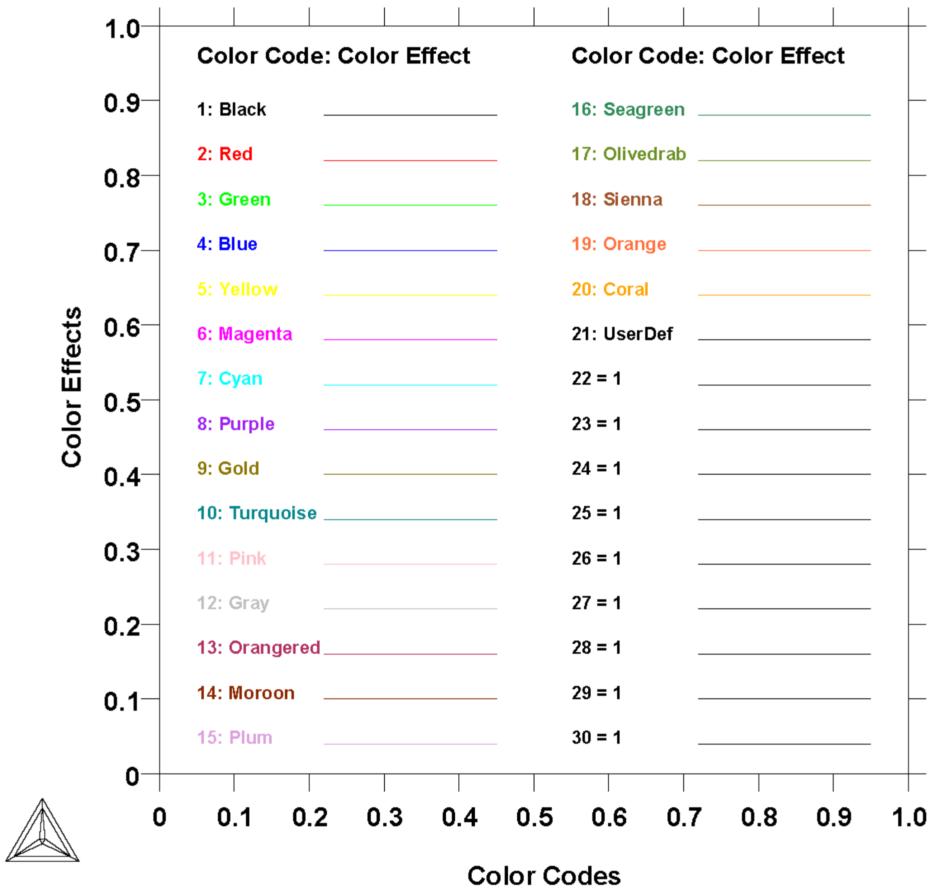
Syntax	`COLOR code`
	`code` is an unsigned integer number (from 0 to 21) identifying the color, or a character string specifying the color.

DIGLIB Color Codes

Code	Color	Equivalent Character
0	Background	Invisible
1	Black	Normal
2	Red	Very_Thick
3	Green	Thin
4	Blue	Thick

Code	Color	Equivalent Character
5	Yellow	Very_Thin
6	Magenta	Dashed
7	Cyan	Dotted
8	Purple	
9	Gold	
10	Turquoise	
11	Pink	
12	Gray	
13	Orangered	
14	Maroon	
15	Plum	
16	Seagreen	
17	Olivedrab	
18	Sienna	
19	Orange	
20	Coral	
21	UserDef	
22 and higher	Any digit larger than 21 has the color effect as 1 (black).	

DIGLIB Color Effects



FONT

FONT redefines the default font setting in the POST-processor.

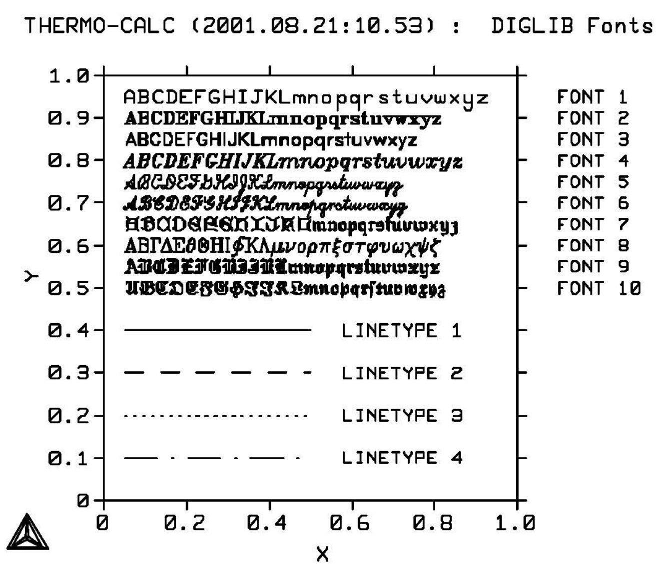
Syntax	`FONT #`
	# is an unsigned integer.

Font Numbers

Number	Font Name
1	Cartographic Roman (the default)
2	Bold Roman script
3	Bold Roman
4	Bold italic
5	Script

Number	Font Name
6	Bold script
7	UNCIAL
8	Bold Greek
9	Gothic English
10	Gothic Greek

Examples of all the DIGLIB fonts and line types



STRING

STRING defines a string containing a text and operation codes (e.g., to change the default font settings).

Syntax	`STRING name text_with_each_character_in_ ^S#^G^F#^U#^D#^R#^L#^N`
	`name` is a valid alphanumeric name (variable) to represent the text including all text formatting codes. It is highly recommended that all the characters in the name are in CAPITAL CASE (e.g. ACA2SO4); otherwise, the defined STRING can be incorrectly plotted. `text_with_each_character_in_^S#^G^F#^U#^D#^R#^L#^N` means the text is coded with each of its characters (and/or numeric numbers) that are formatted with various operators (`^S#`, `^G`, `^F#`, `^U#`, `^D#`, `^R#`, `^L#` and/or `^N`). `#'` are unsigned integers. `^` is the caret character and does not mean a control character. Between an `#` (in an operator) and a numeric number (as a part of the text), there must always be a comma sign (`,`) or the number is not plotted in the text, because the `#` with the number is interpreted as another incorrect #. In some of the commands, `#=0` resets the option to previous (or default) value.

String operation codes in the DIGLIB software

For the PostScript hardcopy fonts, these operation codes are not valid; see *LaTeX Formatting Codes* on page 12:

Operator	Operation
`^S#`	Set character size to size #
`^G`	Set font to Greek
`^F#`	Set font to font number #
`^U#`	Move up # units
`^D#`	Move down # units
`^R#`	Move right # units
`^L#`	Move left # units
`^N`	Do not move, remain at current

Examples

```
STRING Alpha1 ^Ga^F0^D0^S8,1^S0^U0
```

is set to α_1

```
STRING M23C6 M^D0^S8,23^S0^U0C^D0^S8,6^S0^U0
```

is set to $M_{23}C_6$

```
STRING ACA2CO3 ^Ga^F0^D0^S8Ca^D0^S4,2^S0^U0^S8CO^D0^S4,3^S0^U0
```

is set to Ca_2CO_3

TEXT

TEXT outputs, at the current position, the text following the keyword TEXT or the text in the string `string_name` that is defined with *STRING* on page 25.

Syntax	`TEXT text` or `~string_name`
	It is possible to mix ^ operators and previously defined string names using the ~ operator.

FUNCTION

FUNCTION defines and plots a user-defined function.

Syntax	`FUNCTION Y=f(X); start end number_of_XY_pairs; GOC;` `FUNCTION X=f(Y); start end number_of_XY_pairs; GOC;`
	`f(X)` or `f(Y)` are legal mathematical functions of X or Y, understandable by a FORTRAN program. `start` and `end` are unsigned real numbers, and `number_of_XY_pairs` an unsigned integer. `GOC` is a legal graphical operation code as defined with *BLOCK* on page 19.

PCFUNCTION

PCFUNCTION appends a user-defined function to the current path. It is used with *PAINT* below.

Syntax	`PCFUNCTION Y=f(X); start end number_of_XY_pairs; GOC;` `PCFUNCTION X=f(Y); start end number_of_XY_pairs; GOC;`
	`f(X)` or `f(Y)` are legal mathematical functions of X or Y, understandable by a FORTRAN program. `start` and `end` are unsigned real numbers, and `number_of_XY_pairs` an unsigned integer. `GOC` is a legal graphical operation code as defined with *BLOCK* on page 19.

PAINT

PAINT paints the area enclosed by the current path in the current pattern. The current path starts at the last MOVETO given and includes all subsequent DRAWs. This command only works for the PostScript format (as graphical files or on printed hardcopy).

Also see *PCFUNCTION* above.

Syntax	`PAINT <code> <video> <mode>`
	<> denotes optional parameters. To set a new current pattern, supply any or all of the optional parameters. • `<code>` is a single letter 0-9, A-Z or a-t (if `<code>=t`, also supply a space and a number in the range 0.00-1.00). The default is `0`. • `<video>` is a string reading NORMAL (the default) or INVERSE. • `<mode>` is a string reading TRANSPARENT (the default) or OPAQUE.

INCLUDE

INCLUDE adds a file into the current input stream See *Example 5 – Use Included Files for Predefined Symbols* on page 34.

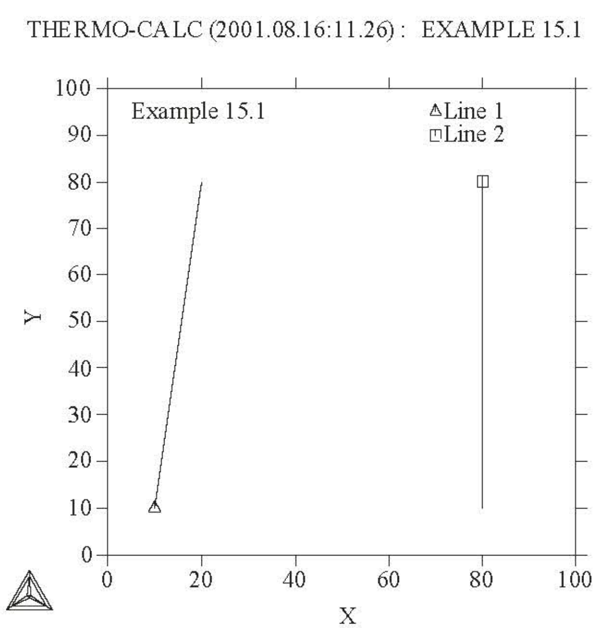
Syntax	`INCLUDE filename`
	`filename` is a legal filename (with its correct path) for the operation system.

Examples of DATAPLOT Files and Outputs

In this section:

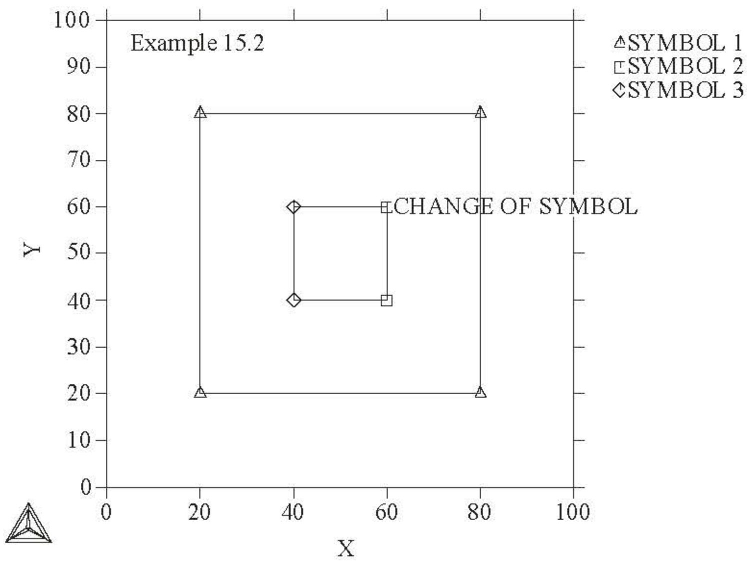
Example 1 – Draw Lines and Symbols

```
$DATAPLOT Example 1
PROLOG 1 EXAMPLE 10<X<100, 0<Y<100
XSCALE0.00000100
YSCALE0.00000100
XTYPE LINEAR YTYPE LINEAR
XLENGTH11.5000
YLENGTH11.5000
TITLEEXAMPLE 1 XTEXT X
YTEXTY

DATASET 1 Two lines started with two symbols
ATTRIBUTE CENTER
0.050.95 N'Example 1
0.70.95 NS'Line 1
0.70.90 NS2'Line 2 1010S1
2080D
8080S2
8010D
5060
```



Example 2 – Draw Polygons and Symbols

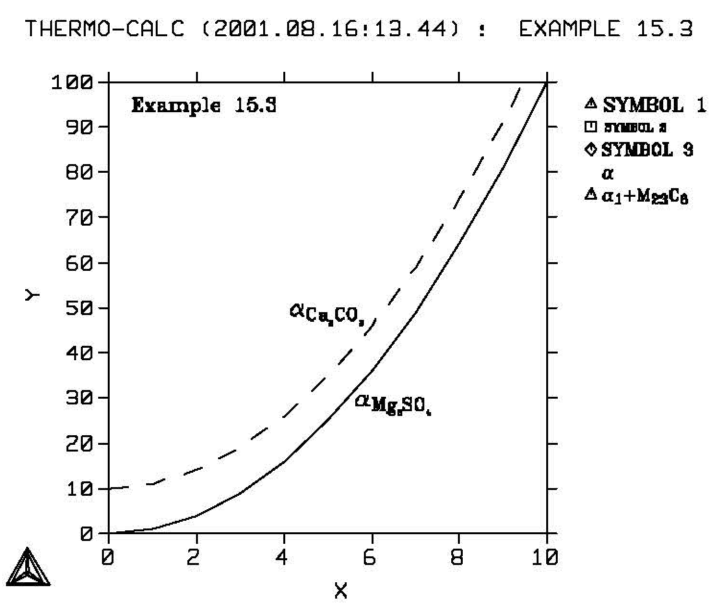
```
$DATAPLOT Example 2

PROLOG 2 EXAMPLE 20<X<100, 0<Y<100

XSCALE0.00000100

YSCALE0.00000100

XTYPE LINEAR YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000

TITLEEXAMPLE 2 XTEXT X

YTEXTY


DATASET 2 Two ploygons with three types of symbols ATTRIBUTE CENTER

CLIP OFF

0.050.95 N'Example 2

1.10.95 NS1'SYMBOL 1

1.10.90 NS2'SYMBOL 2

1.10.85 NS3'SYMBOL 3

BLOCK X=C1; Y=C2; GOC=C3,DSWA

4040M

4060

6060S2'CHANGE OF SYMBOL

6040

4040S0

BLOCKEND

BLOCK X=C1*100; Y=C2*100; GOC=C3,DSWA 0.20.2 MS1

0.20.8

0.80.8

0.80.2

0.20.2

BLOCKEND
```



Example 3 – Using Strings and Various Line Types

```
$DATAPLOT Example 3

PROLOG 3 EXAMPLE 3  0<X<10, 0<Y<100

XSCALE0.0000010

YSCALE0.00000100

XTYPE LINEAR

YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000

TITLE EXAMPLE 3 XTEXT X

YTEXT Y

DATASET 3 Draw curves; plot formatted texts and symbols

$Define some strings: STRING BCC ^Ga^F0

STRING BCC1 ^Ga^F0^D0^S8,1^S0^U0

STRING M23C6 M^D0^S8,23^S0^U0C^D0^S8,6^S0^U0

STRING ACA2CO3 ^Ga^F0^D0^S8Ca^D0^S4,2^S0^U0^S8CO^D0^S4,3^S0^U0 STRING AMG2SO4
^Ga^F0^D0^S8Mg^D0^S4,2^S0^U0^S8SO^D0^S4,4^S0^U0

$ Note: if as PostScript output:

$STRING BCC !a

$STRING BCC1 !a^do1$

$STRING M23C6 M^do23$C^do6$

$STRING ACA2CO3 !a^doCa^do2$^doCO^do3$

$STRING AMG2SO4 !a^doMg^do2$^doSO^do4$ ATTRIBUTE CENTER
```

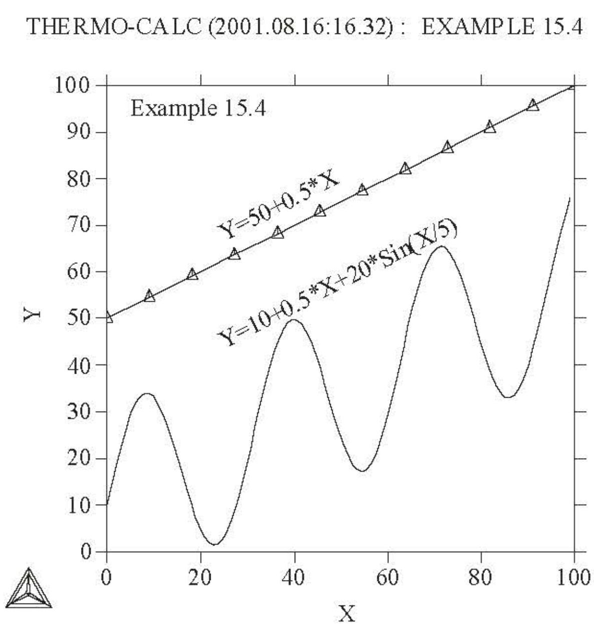
```
CLIP OFF FONT 2
0.05  0.95 N'Example 3
1.10.95 NS1'SYMBOL 1 CHARSIZE 0.2
1.10.90 NS2' SYMBOL 2 CHARSIZE 0.3
1.10.85 NS3'SYMBOL 3 1.10.80 N' ~BCC
1.10.75 NS1'~BCC1+~M23C6
SYMBOLSIZE 0.4
CHARSIZE 0.4
0.41  0.50 N'~ACA2CO3
0.56 0.30 N'~AMG2SO4 CLIP ON
LINETYPE 1
BLOCK X=C1; Y=C1*C1; GOC=C2, DWA
1 2 3  4  5  6  7  8  9           10
BLOCKEND LINETYPE 2
BLOCK X=C1; Y=C1*C1+10; GOC=C2, DWA
0 M 1 2 3  4  5  6  7  8  9           10
BLOCKEND
0 M
```



Example 4 – Draw Curves Defined by Functions

```
$DATAPLOT Example 4
PROLOG 4 EXAMPLE 40<X<100, 0<Y<100
```

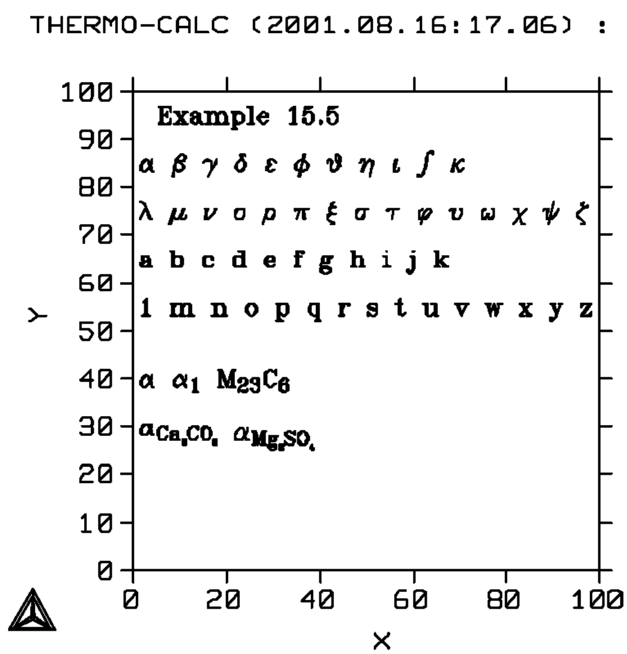
```
XSCALE0.00000100

YSCALE0.00000100

XTYPE LINEAR YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000

TITLEEXAMPLE 4 XTEXT X

YTEXTY

DATASET 4 Plot two functions as lines: ATTRIBUTE CENTER

0.050.95 N'Example 4

$ Draw two lines defined by FUNCTIONS:

FUNCTION Y=10+0.5*X+20*Sin(X/5); 0 100 100; DWA;

FUNCTION Y=50+0.5*X; 0 100 10; DS1WA;

$ Write funtions beside the lines:

$Note the real rotation angle (27 degree) can be seen

$only on the PostScript hardcopy! 0.250.68 N'^RO27Y=50+0.5*X

0.250.45 N'^RO27Y=10+0.5*X+20*Sin(X/5)
```



Example 5 – Use Included Files for Predefined Symbols

```
$DATAPLOT Example 5

PROLOG 5 EXAMPLE 50<X<100, 0<Y<100

XSCALE0.00000100

YSCALE0.00000100
```

```
XTYPE LINEAR YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000

TITLEEXAMPLE 5 XTEXT X

YTEXTY

DATASET 5 Write characters in various fonts, and defined symbols FONT 2

INCLUDE <DATAPLOT-EXAMPLE-PATH>INCLUDE.EXP ATTRIBUTE CENTER

0.050.95 N'Example 5 CLIP OFF

0.010.85 N'~TEST1A

0.010.75 N'~TEST1B

0.010.65 N'~TEST2A

0.010.55 N'~TEST2B

0.010.40 N'~BCC ~BCC1 ~M23C6

0.010.30 N'~ACA2CO3 ~AMG2SO4
```



The INCLUDE.EXP file has the following content:

```
STRING TEST1A ^Ga b c d e f g h i j k^F0

STRING TEST1B ^Gl m n o p q r s t u v w x y z^F0 STRING TEST2Aa b c d e f g h i j k

STRING TEST2B1 m n o p q r s t u v w x y z STRING BCC ^Ga^F0

STRING BCC1 ^Ga^F0^D0^S8,1^S0^U0

STRING BCC2 ^Ga^F0^D0^S8,2^S0^U0 STRING FCC ^Gc^F0
```

```
STRING FCC1 ^Gc^F0^D0^S8,1^S0^U0 STRING FCC2 ^Gc^F0^D0^S8,2^S0^U0

STRING L Liquid STRING SIGMA ^Gs^F0 STRING MU ^Gm^F0 STRING LAVES ^Gl^F0 STRING CHI
^Gx^F0 STRING KSI ^Gr^F0

STRING MCETA ^S12^Gh^F0^S0 STRING M2CM^D0^S8,2^S0^U0C STRING M2CT M^D0^S10,2^S0^U0C
STRING M6CM^D0^S8,6^S0^U0C

STRING M23C6 M^D0^S8,23^S0^U0C^D0^S8,6^S0^U0 STRING M7C3
M^D0^S8,7^S0^U0C^D0^S8,3^S0^U0 STRING M3C2 M^D0^S8,3^S0^U0C^D0^S8,2^S0^U0 STRING MC1-X
MC^D0^S8,1-x^S0^U0

STRING XC x^D0^S7,C^S0^U0 STRING XFE x^D0^S7,Fe^S0^U0 STRING XMO x^D0^S7,Mn^S0^U0
STRING XW x^D0^S7,W^S0^U0 STRING UW ~XW/(~XMO+~XW)

STRING ac a^D0^S7,C^S0^U0 STRING ACC a^D0^S7,C^S0^U0

STRING ACA2CO3 ^Ga^F0^D0^S8Ca^D0^S4,2^S0^U0^S8CO^D0^S4,3^S0^U0 STRING AMG2SO4
^Ga^F0^D0^S8Mg^D0^S4,2^S0^U0^S8SO^D0^S4,4^S0^U0

$ Note: if as PostScript output:

$STRING TEST1A !a !b !c !d !e !f !g !h !i !j !k^fo27

$STRING TEST1B !l !m !n !o !p !q !r !s !t !u !v !w !x !y !z^fo27

$STRING TEST2Aa b c d e f g h i j k

$STRING TEST2B1 m n o p q r s t u v w x y z

$STRING BCC !a

$STRING BCC1 !a^do1$

$STRING BCC2 !a^do2$

$STRING FCC !c

$STRING FCC1 !c^do1$

$STRING FCC2 !c^do2$

$STRING L Liquid

$STRING SIGMA !s

$STRING MU !m

$STRING LAVES !l

$STRING CHI !x

$STRING KSI !r

$STRING MCETA !h

$STRING M2CM^do2$C

$STRING M2CT M^do2$C

$STRING M6CM^do6$C

$STRING M23C6 M^do23$C^do6$

$STRING M7C3 M^do7$C^do3$

$STRING M3C2 M^do3$C^do2$

$STRING MC1-X MC^do1-x$
```

```
$STRING XC x^doC$

$STRING XFE x^doFe$

$STRING XMO x^doMo$

$STRING XW x^doW$

$STRING UW ~XW/(~XMO+~XW)

$STRING ac a^doC$

$STRING ACC a^doC$

$STRING ACA2CO3 !a^doCa^do2$^doCO^do3$

$STRING AMG2SO4 !a^doMg^do2$^doSO^do4$
```

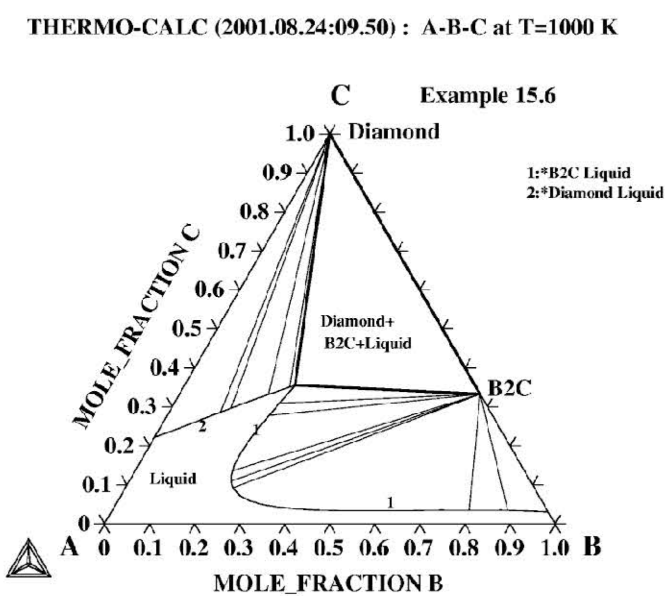
Example 6 – Plot Triangular Diagrams for Ternary Systems

```
$DATAPLOT Example 6

PROLOG 6 EXAMPLE 6  0<X<0.969224, 0<Y<1.00000

XSCALE0.000000.969224

YSCALE0.307492E-011.00000

XTYPE LINEAR YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000 TITLE A-B-C at T=1000 K XTEXT MOLE_FRACTION B YTEXT  MOLE_FRACTION C

DIAGRAM_TYPE TRIANGULAR YES YES

DATASET 6 Plot a ternary phase diagram CLIP OFF

0.70 0.95 N'Example 6 0.85  0.30 N'B2C

0.54 0.87 N'Diamond CHARSIZE 0.25

1.4E+01  1.10E+01 MVA'1:*B2C Liquid

1.4E+01  1.05E+01 MVA'2:*Diamond Liquid

0.10  0.10 N'Liquid

0.48  0.45 N'Diamond+

0.48 0.40 N' B2C+Liquid 5.80E-01  5.40E-02 MWA' 1

1.90E-01  2.40E-01 MWA' 1

0.65E-01  2.50E-01 MWA' 2

CHARSIZE 0.45

-0.10  -0.05 N'A

1.06  -0.05 N'B

0.500.95 N'C

CHARSIZE 0.35

$$ Calculated A-B-C Phase Equilibrium Data:

$ PHASE REGION FOR:

$F0 LIQUID
```

```
$E DIAMOND_A4
$F0 B2C
$ INVARIANT EQUILIBRIUM COLOR2
BLOCK X=C1; Y=C2;GOC=C3,WAD; 2.4555855989E-013.5568857193E-01M
0.0000000000E+009.9999523163E-01
2.4555855989E-013.5568857193E-01M
6.6666668653E-013.3333334327E-01
0.0000000000E+009.9999523163E-01M
6.6666668653E-013.3333334327E-01 COLOR1
BLOCKEND
$ PHASE REGION FOR:
$E LIQUID
$F0 B2C
BLOCK X=C1; Y=C2;GOC=C3,WAD;
$ PLOTTED COLUMNS ARE : X(LIQUID,B) and X(LIQUID,C)
2.2030337155E-011.2340000272E-01M
2.2632879019E-011.1058768630E-01
2.3371633887E-019.9345825613E-02
2.4253317714E-018.9345827699E-02
2.6429468393E-017.2744041681E-02
2.8429466486E-016.2814079225E-02
2.9617273808E-015.8319382370E-02
3.2811737061E-014.9470417202E-02
3.6353862286E-014.3130427599E-02
3.9895987511E-013.8979098201E-02
4.5209178329E-013.5266116261E-02
5.2293431759E-013.3152002841E-02
6.1148744822E-013.3077053726E-02
6.4690870047E-013.3490389585E-02
6.8232995272E-013.4017231315E-02
7.3546189070E-013.4814555198E-02
7.5317251682E-013.5033416003E-02
8.0630439520E-013.5373892635E-02
8.5943627357E-013.4983776510E-02
9.1256815195E-013.3575300127E-02
9.6747112274E-013.0857827514E-02
```

```
9.6922445297E-013.0749246478E-02

2.2030337155E-011.2340000272E-01M

2.1294665337E-011.5308913589E-01

2.1171525121E-011.8851040304E-01

2.1532440186E-012.2393165529E-01

2.2180187702E-012.5935292244E-01

2.2992117703E-012.9477417469E-01

2.3888295889E-013.3019542694E-01

2.4555855989E-013.5568857193E-01

$ PLOTTED COLUMNS ARE : X(B2C,B) and X(B2C,C) 6.6666668653E-013.3333334327E-01M

6.6666668653E-013.3333334327E-01

$ TIELINES COLOR3

6.6666668653E-013.3333334327E-01M

8.7775242329E-013.4625384957E-02

6.6666668653E-013.3333334327E-01M

7.9250496626E-013.5342670977E-02

6.6666668653E-013.3333334327E-01M

2.4555824697E-013.5568737984E-01

6.6666668653E-013.3333334327E-01M

2.3944084346E-019.2542596161E-02

6.6666668653E-013.3333334327E-01M

2.3359020054E-013.0954307318E-01

6.6666668653E-013.3333334327E-01M

2.2585247457E-012.7766343951E-01

6.6666668653E-013.3333334327E-01M

2.1618695557E-011.3621240854E-01

2.2632879019E-011.1058768630E-01M

6.6666668653E-013.3333334327E-01 COLOR1

BLOCKEND

$ PHASE REGION FOR:

$F0 DIAMOND_A4

$E B2C

BLOCK X=C1; Y=C2;GOC=C3,WAD;

$ PLOTTED COLUMNS ARE : X(DIAMOND_A4,B) and X(DIAMOND_A4,C)
0.0000000000E+009.9999523163E-01M

0.0000000000E+009.9999976158E-01 BLOCKEND
```

```
$ PHASE REGION FOR:
$E LIQUID
$F0 DIAMOND_A4
BLOCK X=C1; Y=C2;GOC=C3,WAD;
$ PLOTTED COLUMNS ARE : X(LIQUID,B) and X(LIQUID,C)
2.45558E-01  3.55688E-01 M
2.02635E-01 3.34830E-01
1.62439E-01 3.13753E-01
1.22439E-01 2.91531E-01
8.24390E-02 2.68542E-01
4.24390E-02 2.45480E-01
2.43905E-03 2.23138E-01
2.49999E-07 2.21816E-01
$ TIELINES COLOR3
0.00000E+00  9.99995E-01 M
2.35291E-01 3.50859E-01
0.00000E+00  9.99994E-01 M
1.98265E-01 3.32609E-01
0.00000E+00  9.99994E-01 M
1.32400E-01 2.97160E-01
0.00000E+00  9.99993E-01 M
1.14399E-01 2.86953E-01 BLOCKEND
```



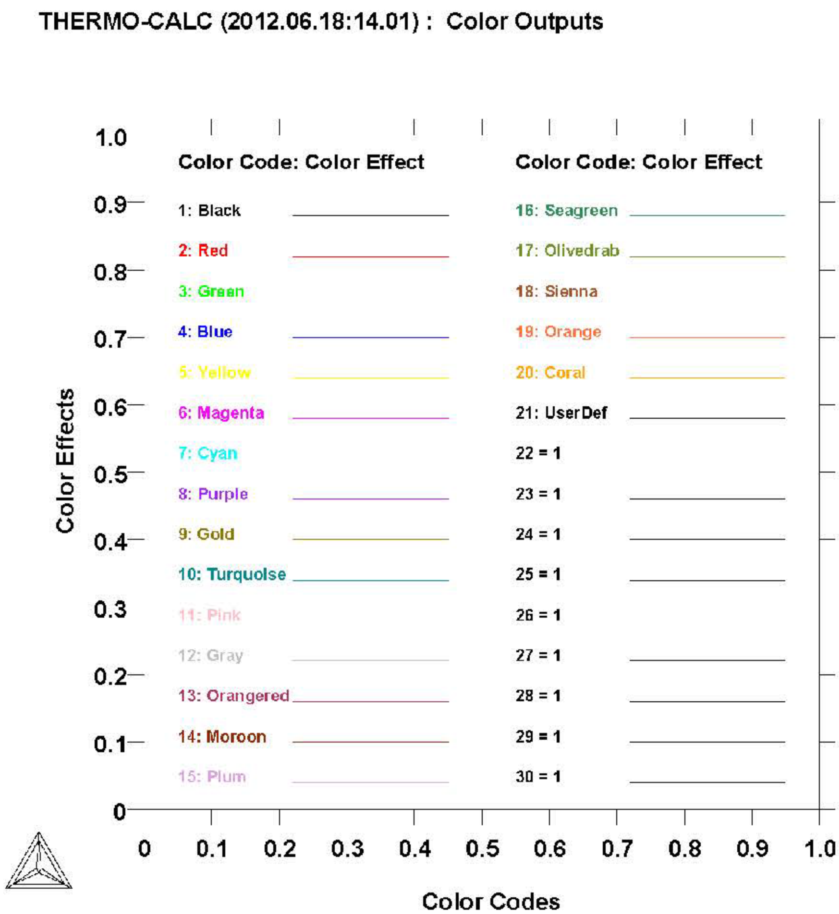
Example 7 – Color Codes and Effects

```
$DATAPLOT Color Codes & Color Effects

PROLOG 10 Color Codes  0<X<1.0, 0<Y<1.0

XSCALE0.01.0

YSCALE0.01.0

XTYPE LINEAR

YTYPE LINEAR

XLENGTH11.5000

YLENGTH11.5000

TITLE Color Outputs XTEXT Color Codes YTEXT  Color Effects

DATASET 10 Various Color Codes for Color Outputs: CHARSIZE 0.3

LINETYPE 1

BLOCK X=C1; Y=C2;GOC=C3,WAD;

0.05  0.95 MNA'Color Code: Color Effect

0.55 0.95 MNA'Color Code: Color Effect CHARSIZE 0.25

COLOR 1

0.05 0.88 MNA'1: Black CLIP ON

0.22  0.88 M

0.45 0.88

CLIP OFF COLOR 2

0.05 0.82 MNA'2: Red CLIP ON

0.22  0.82 M

0.45 0.82

CLIP OFF COLOR 3

0.05 0.76 MNA'3: Green CLIP ON

0.22  0.76 M

0.45 0.76

CLIP OFF COLOR 4

0.05 0.70 MNA'4: Blue CLIP ON

0.22  0.70 M

0.45 0.70

CLIP OFF COLOR 5

0.05 0.64 MNA'5: Yellow CLIP ON

0.22  0.64 M

0.45 0.64

CLIP OFF COLOR 6
```

```
0.05 0.58 MNA'6: Magenta CLIP ON
0.22  0.58 M
0.45 0.58
CLIP OFF COLOR 7
0.05 0.52 MNA'7: Cyan CLIP ON
0.22  0.52 M
0.45 0.52
CLIP OFF COLOR 8
0.05 0.46 MNA'8: Purple CLIP ON
0.22  0.46 M
0.45 0.46
CLIP OFF COLOR 9
0.05 0.40 MNA'9: Gold CLIP ON
0.22  0.40 M
0.45 0.40
CLIP OFF COLOR 10
0.05 0.34 MNA'10: Turquoise CLIP ON
0.22  0.34 M
0.45 0.34
CLIP OFF COLOR 11
0.05 0.28 MNA'11: Pink CLIP ON
0.22  0.28 M
0.45 0.28
CLIP OFF COLOR 12
0.05 0.22 MNA'12: Gray CLIP ON
0.22  0.22 M
0.45 0.22
CLIP OFF COLOR 13
0.05 0.16 MNA'13: Orangered CLIP ON
0.22  0.16 M
0.45 0.16
CLIP OFF COLOR 14
0.05 0.10 MNA'14: Moroon CLIP ON
0.22  0.10 M
0.45 0.10
CLIP OFF COLOR 15
```

```
0.05 0.04 MNA'15: Plum CLIP ON
0.22  0.04 M
0.45 0.04
CLIP OFF COLOR 16
0.55 0.88 MNA'16: Seagreen CLIP ON
0.72  0.88 M
0.95 0.88
CLIP OFF COLOR 17
0.55 0.82 MNA'17: Olivedrab CLIP ON
0.72  0.82 M
0.95 0.82
CLIP OFF COLOR 18
0.55 0.76 MNA'18: Sienna CLIP ON
0.72  0.76 M
0.95 0.76
CLIP OFF COLOR 19
0.55 0.70 MNA'19: Orange CLIP ON
0.72  0.70 M
0.95 0.70
CLIP OFF COLOR 20
0.55 0.64 MNA'20: Coral CLIP ON
0.72  0.64 M
0.95 0.64
CLIP OFF COLOR 21
0.55 0.58 MNA'21: UserDef CLIP ON
0.72  0.58 M
0.95 0.58
CLIP OFF COLOR 22
0.55 0.52 MNA'22 = 1 CLIP ON
0.72  0.52 M
0.95 0.52
CLIP OFF COLOR 23
0.55 0.46 MNA'23 = 1 CLIP ON
0.72  0.46 M
0.95 0.46
CLIP OFF
```

```
BLOCKEND
```


THERMO-CALC (2012.06.18:14.01) : Color Outputs
Color Code: Color Effect
1: Black
2: Red
3: Green
4: Blue
5: Yellow
6: Magenta
7: Cyan
8: Purple
9: Gold
10: Turquoise
11: Pink
12: Gray
13: Orangered
14: Moroon
15: Plum
Color Code: Color Effect
16: Seagreen
17: Olivedrab
18: Sienna
19: Orange
20: Coral
21: UserDef
22 = 1
23 = 1
24 = 1
25 = 1
26 = 1
27 = 1
28 = 1
29 = 1
30 = 1
Color Effects
Color Codes
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0

Index

M

N

O

S