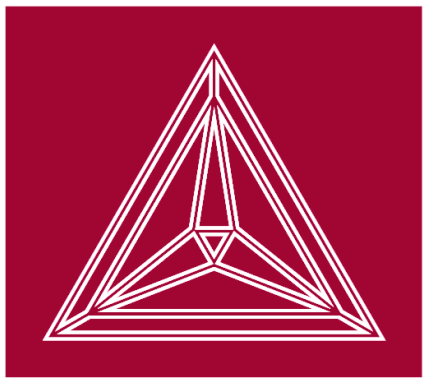
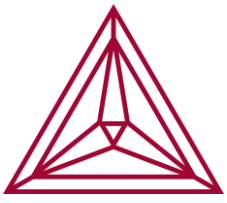


Thermo-Calc Console Mode Command Reference

Version 4.1



© 1995-2014 Foundation of Computational Thermodynamics

Stockholm, Sweden

Thermo-Calc Console Mode Command Reference

Contents

1 Introduction 7
2 General commands 7
Question mark (“?”) 7
Double question marks (“??”) 7
BACK 7
EXIT 8
GOTO_MODULE 8
HELP 8
INFORMATION 8
MACRO_FILE_OPEN 9
SET_INTERACTIVE 12
3 DATABASE_RETRIEVAL commands 12
AMEND_SELECTION 12
APPEND_DATABASE 13
DATABASE_INFORMATION 14
DEFINE_ELEMENTS 14
DEFINE_SPECIES 15
DEFINE_SYSTEM 15
GET_DATA 16
LIST_DATABASE 16
LIST_SYSTEM 17
NEW_DIRECTORY_FILE 17
REJECT 18
RESTORE 20
SET_AUTO_APPEND_DATABASE 21
SWITCH_DATABASE 22
4 EDIT_EXPERIMENTS commands 23
COMPUTE_ALL_EQUILIBRIA 24
EXPERIMENT 25
EXPORT 26
IMPORT 26
LABEL_DATA 27
LIST_ALL_EQUILIBRIA 28
MAKE_POP_FILE 28
READ (READ_WORKSPACE = READ_BLOCK) 28

RESTORE_ALL_WEIGHTS....29
SET_ALTERNATE_CONDITION....29
SET_WEIGHT....31
STORE_ALL_WEIGHTS....32
TRANSFER_START_VALUE....33
5 GIBBS commands....33
ADD_COMMENT....33
AMEND_ELEMENT_DATA....33
AMEND_PARAMETER....34
AMEND_PHASE_DESCRIPTION....37
AMEND_SYMBOL....52
CHANGE_STATUS....54
DELETE....54
ENTER_ELEMENT....55
ENTER_PARAMETER....56
ENTER_PHASE....63
ENTER_SPECIES....68
ENTER_SYMBOL....68
LIST_CONSTITUENT....73
LIST_DATA....73
LIST_PARAMETER....74
LIST_PHASE_DATA....77
LIST_STATUS....77
LIST_SYMBOL....79
READ_GES_WORKSPACE....80
REINITIATE....80
SAVE_GES_WORKSPACE....81
SET_R_AND_P_NORM....81
6 PARROT commands....82
AMEND_PARAMETER....82
AMEND_SYMBOL....84
COMPILE_EXPERIMENTS....87
CONTINUE_OPTIMIZATION....88
CREATE_NEW_STORE_FILE....89
EDIT_EXPERIMENTS....89
ENTER_PARAMETER....89
LIST_ALL_VARIABLES....96
LIST_CONDITIONS....97
LIST_PARAMETER....97
LIST_PHASE_DATA (from GIBBS)....100

LIST_RESULT 100
LIST_STORE_FILE 104
LIST_SYMBOL_IN_GES 104
OPTIMIZE_VARIABLES 104
READ_PARROT_WORKSPACES 105
RECOVER_VARIABLES 105
REINITIATE 105
RESCALE_VARIABLES 105
SAVE_PARROT_WORKSPACES 106
SET_ALTERNATE_MODE 106
SET_EXTERNAL_PARAMETER 106
SET_FIX_VARIABLE 107
SET_OPTIMIZING_CONDITION 108
SET_OPTIMIZING_VARIABLE 109
SET_OUTPUT_LEVELS 109
SET_SCALED_VARIABLE 110
SET_STORE_FILE 111
7 POLY commands 112
ADD_INITIAL_EQUILIBRIUM 112
ADVANCED_OPTIONS 114
AMEND_STORED_EQUILIBRIA 129
CHANGE_STATUS 131
COMPUTE_EQUILIBRIUM 136
COMPUTE_TRANSITION 138
CREATE_NEW_EQUILIBRIUM 140
DEFINE_COMPONENTS 141
DEFINE_DIAGRAM 141
DEFINE_MATERIAL 144
DELETE_INITIAL_EQUILIBRIUM 149
DELETE_SYMBOL 149
ENTER_SYMBOL 149
EVALUATE_FUNCTIONS 152
LIST_AXIS_VARIABLE 152
LIST_CONDITIONS 152
LIST_EQUILIBRIUM 153
LIST_INITIAL_EQUILIBRIA 153
LIST_STATUS 153
LIST_SYMBOLS 154
LOAD_INITIAL_EQUILIBRIUM 154
MAP 155

POST ... 157
READ_WORKSPACES ... 158
RECOVER_START_VALUES ... 159
REINITIATE_MODULE ... 159
RESET_CONDITION ... 159
SAVE_WORKSPACES ... 160
SELECT_EQUILIBRIUM ... 161
SET_ALL_START_VALUES ... 161
SET_AXIS_VARIABLE ... 163
SET_CONDITION ... 164
SET_INPUT_AMOUNTS ... 167
SET_NUMERICAL_LIMITS ... 167
SET_REFERENCE_STATE ... 169
SET_START_CONSTITUTION ... 170
SHOW_VALUE ... 171
STEP_WITH_OPTIONS ... 172
TABULATE ... 188
8 POST commands ... 188
ADD_LABEL_TEXT ... 188
APPEND_EXPERIMENTAL_DATA ... 190
DUMP_DIAGRAM ... 191
ENTER_SYMBOL ... 191
FIND_LINE ... 194
LIST_DATA_TABLE ... 194
LIST_PLOT_SETTINGS ... 195
LIST_SYMBOLS ... 195
MAKE_EXPERIMENTAL_DATAFILE ... 195
MODIFY_LABEL_TEXT ... 195
PLOT_DIAGRAM ... 196
PRINT_DIAGRAM ... 197
QUICK_EXPERIMENTAL_PLOT ... 197
REINITIATE_PLOT_SETTINGS ... 198
RESTORE_PHASE_IN_PLOT ... 198
SELECT_PLOT ... 198
SET_AXIS_LENGTH (not supported in Thermo-Calc 3.1) ... 199
SET_AXIS_PLOT_STATUS (not supported in Thermo-Calc 3.1) ... 199
SET_AXIS_TEXT_STATUS ... 199
SET_AXIS_TYPE ... 199
SET_COLOR (not supported in Thermo-Calc 3.1) ... 200
SET_CORNER_TEXT ... 201

SET_DIAGRAM_AXIS 202
SET_DIAGRAM_TYPE 204
SET_FONT (not supported in Thermo-Calc 3.1) 205
SET_INTERACTIVE_MODE 205
SET_LABEL_CURVE_OPTION 205
SET_PLOT_FORMAT 206
SET_PLOT_OPTIONS 207
SET_PLOT_SIZE (not supported in Thermo-Calc 3.1) 209
SET_PREFIX_SCALING 209
SET_RASTER_STATUS 209
SET_REFERENCE_STATE 210
SET_SCALING_STATUS 210
SET_TIC_TYPE 211
SET_TIELINE_STATUS 211
SET_TITLE 211
SET_TRUE_MANUAL_SCALING (not supported in Thermo-Calc 3.1) 212
SUSPEND_PHASE_IN_PLOT 212
TABULATE 212
9 REACTOR commands 213
AMEND_RECORD 213
CREATE_DIVIDERS 213
CREATE_PIPES 213
CREATE_PROBE 214
CREATE_STAGE_BOXES 214
EXECUTE_POLY3_COMMAND 215
LIST_DESIGN 215
LIST_RECORDS 215
READ_WORKSPACE 216
SAVE_WORKSPACE 216
SHOW_PROBE 217
START_SIMULATION 217
10 SYSTEM_UTILITIES commands 219
ABOUT 219
CLOSE_FILE 219
DISPLAY_LICENSE_INFO 219
HP_CALCULATOR 219
OPEN_FILE 220
SET_COMMAND_UNITS 220
SET_ECHO 221
SET_INTERACTIVE_MODE 221

SET_LOG_FILE 221
SET_PLOT_ENVIRONMENT 222
SET_TC_OPTIONS 222
SET_TERMINAL 223
STOP_ON_ERROR 224
11 TABULATION_REACTION commands 224
ENTER_FUNCTION 224
ENTER_REACTION 226
LIST_SUBSTANCES 227
SET_ENERGY_UNIT 228
SET_PLOT_FORMAT 229
SWITCH_DATABASE 229
TABULATE_DERIVATIVES 231
TABULATE_REACTION 234
TABULATE_SUBSTANCE 237
12 Appendix – Commands in the Experimental Data files (*.POP, *.DOP) 243
TABLE_HEAD, TABLE_VALUES and TABLE_END 245
COMMENT 246
FLUSH_BUFFER 247
GRAPHICS_PLOT 247

1 Introduction

1.1 Target group

This command reference is aimed at users of the Thermo-Calc program who use it in the console mode. It may also be useful for users of DICTRA. However, the commands available in DICTRA module are described in the *DICTRA Command Reference*.

1.2 Purpose

Describes the function and syntax of the various commands that are available in the various modules of the Thermo-Calc program.

1.3 Content

Commands that are available in all or most modules are described in the first section, in section 2. Each section then covers the commands of a section. The sections are presented in alphabetical order according to the module names. In each section, the commands of that module are also presented in alphabetical order according to the command names.

An Appendix describes commands used in POP- and DOP-files. These files contain experimental data used for data optimisation in Thermo-Calc (POP) and DICTRA (DOP).

2 General commands

Question mark (“?”)

Description: Executing this command will provide you with on-line help information regarding 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: ?

Double question marks (“??”)

Description: For some commands and prompts, more extensive on-line help information will be presented in response to this command.

Syntax: ??

BACK

Description: This command switches control back to the most recent module. See also the `GOTO_MODULE` command. Going from the POST module (post-processor), `BACK` goes only to the TAB or POLY module (from where the POST module entered).

Syntax: `BACK`

EXIT

Description: This command terminates the program and returns to the operating system. Unless a SAVE command has been executed (in either the GIBBS, POLY, PARROT or REACTOR module), all data and results will be lost.

Syntax: EXIT

GOTO_MODULE

Description: This command switches between modules. The name of the desired module must also be typed. In order to obtain a list of available modules, press the <RETURN> key.

Syntax: GOTO_MODULE <MODULE NAME>

Prompt: MODULE NAME: <MODULE NAME>

Options: module name -- the name of the module to subsequently open.

Notes: Pressing the <RETURN> key without typing a specific and having specified unique module name (or typing a "?" mark) in its ensuring prompt MODULE NAME: will list all the available Thermo-Calc (Console Mode) modules.

HELP

Description: This command lists the available commands or gives an explanation of a specified command.

Syntax: HELP <COMMAND NAME>

Prompt: COMMAND: <COMMAND NAME>

Options: command name -- the name of the command (one of the REACTOR-module commands) to obtain help.

Notes: Pressing the <RETURN> key without typing a command name will list all the available REACTOR commands.

Specifying a unique command will print an explanation of that command on the screen (usually the same text as found in this User's Guide). Typing a command abbreviation which is not unique will list all matching commands. The desired command information can be obtained by typing a unique abbreviation or the complete command name.

INFORMATION

Description: Basic information about various topics related to the module you are in can be obtained with this command.

Syntax: INFORMATION

Prompt: `which subject /purpose/: <topic name>`

Specify a subject (or its abbreviation as long as it is unique) on which information should be given, from the available topics. A list can be seen if typing a question mark "?".

MACRO_FILE_OPEN

Description: The macro file functionality is an extraordinary convenient and easier way of pre-defining sequences of various legal Thermo-Calc (and DICTRA) commands stored in a so-called macro file (which is a simple textual file normally with the default extension "TCM" for the Thermo-Calc Console Mode or "DCM" for the DICTRA software) and then executing all of them simply by this MACRO_FILE_OPEN command (preceded by the macro file name). This command can be operated 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).

This is extremely useful when the same/similar calculations are made often with just some small changes [in terms of system definitions, data manipulations, conditions (for single points, stepping or mapping calculations), plotting settings, *etc.*). One good case for applying this feature is when calculating desired 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 command is used and a *.LOG file name is given before any other SYS/DATA/TAB/GIBBS/POLY/POST/PARROT/ED-EXP command or any special-module command (*e.g.,* BIN, TERN, POTENTIAL, SCHEIL, POURBAIX, REACTION). Such an *.LOG file generated from the Thermo-Calc (Console Mode) software is a simple textual file, and by using any simple textual editor (such as Notepad, Wordpad, PFE, Emacs, vi, *etc.*) it can be further edited: *e.g.,* taking away unnecessary command lines, modifying some commands, settings and definitions, adding some pausing points, adding some helpful commenting lines began with "@@" signs, etc. Then it can be saved as a macro file with the standard file extension "TCM".

An experienced user can also directly and speedily write/edit an appropriate macro file for desired calculations/simulations, using any simple textual editor outside the Thermo-Calc Console Mode (and DICTRA) program.

All Thermo-Calc and DICTRA 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 one can use comment-lines (for describing the problems and for explaining on various commands and inputs/outputs), which should always start with the "@@" signs in the beginning of each comment-line. Such comment-lines will provide a great assistant in easily documenting the macro file, while they will not be considered as command lines and thus will not affect the proceeding of all the normal commands in various 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 will be ignored, and the line started with @) will also be ignored.

An interesting facility of a macro file is to allow the user to have some interactions at some desired 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 which are requested by a certain command. The macro will temporally stop at the "@?" sign, prompt on screen the text given after "@?", and wait for the user specified argument/parameter/value. The Thermo-Calc (Console Mode) software will then utilize the user specified argument/parameter/value as the input(s) for the associated command. For an example, you can have the following way for inputting the values of lower and higher temperature limits for the second axis-variable:

```
GO POLY-3
SET-AXIS-VAR 2 T  @?Low-temperature-limit:
@?High-temperature-limit:
```

You can have macro-variables that are denoted by the signs of @#n (for definition) and ##n (for usage); and you can have up to 9 variables inside a single macro file. Such a macro-variable can be assigned with its desired value by e.g.:

```
@#3First-element?
```

This will write the text (note that the text describing the expected user specification/input should be written as a continuous string without any empty space) after the "@#3" sign as prompt on screen and wait for user's specification. The input will be assigned to the macro-variable ##3, which can then be directly called in different parts within the current macro file.

For instance, the content of the macro-variable ##3 will be inserted at "##3" in the following command:

```
DEFINE-SYSTEM ##3
```

You can also use this in more complicated commands, such as the following:

```
SET AXIS VAR 1 x(##3) 0 1,,,
```

This command will set the mole fraction of macro variable 3 as axis 1.

A macro file can have any number of pauses at the "**@&**" signs, for the purposes of checking the details/results of executing certain commands when running the macro file. However, one may also prevent the Thermo-Calc (Console Mode) software from temporarily stopping at any pause by simply 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 maximum 5 nested levels, i.e., a macro file can call another macro file, and if one sub-level macro is terminated by the SET_INTERACTIVE command it will be resumed at next command in the previous macro. If it is terminated by "end-of-file", the Thermo-Calc (Console Mode) software will be aborted.

This nice feature can be utilized for many different purposes, especially when a user is performing alloy design which may require many (say hundreds) of calculations/simulations on similar material system/processes (specified in many different but appropriately-documented macro files which are organized in up-to-5 levels) during a certain period of time (*e.g.,* in an evening), the user can simply run the main macro (on the top level) at a certain time (*e.g.,* before leaving office) and afterwards (*e.g.,* next morning) the user can systematically and efficiently check/compare/analysis the results (which have been saved as various graphical files, and/or EXP/TXT/XLS/... files).

It is worth to mention that: by simply adding the SYS command SET_ECHO at the very beginning of a macro file [or of the primary macro file on the top level if any sub-level(s) of macro files are used], it is very convenient and extremely useful to automatically show up on screen the complete/detailed meaning of various commands in all the sequential operations in the Thermo-Calc (Console Mode) (and DICTRA) software, that are enforced according to the macro file(s).

Syntax: MACRO_FILE_OPEN <NAME OF A MACRO FILE *.TCM>

An Open window will pop up on the screen for specifying the filename of the macro file (*.TCM) with the macro command, so that the path (in the Look in box) and file name (in the File name box) can be appropriately specified, as shown in Figure 7-2, for choosing an appropriate macro file for a desired Thermo-Calc (Console Mode) calculation.

The file type *.TCM for Thermo-Calc (Console Mode) calculations and *.DCM for DICTRA simulations, as in the Files of type box, cannot be changed. By pressing the Open button, the program proceeds with executing various Thermo-Calc (Console Mode) commands.

If the macro file contains some commands for setting *.LOG files, saving or reading GIBBS, POLY or PARROT workspaces, switching user databases, compiling experiments (from existing *.POP files), creating new *.PAR files, appending *.EXP files, plotting or dumping diagrams, etc., a corresponding window (e.g., Save, Open, Print, etc.) will pop up on screen. If desired by the user (and if the user knows exactly what is doing at a certain point in a macro file), such popped-up 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 is supposed to plot graphs on screen, but the command SET_PLOT_FORMAT has been 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. For an extensive example on using macro features for the TAB-Module, please refer to the TCEX12.TCM file.

SET_INTERACTIVE

Description: This command 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.

Syntax: SET_INTERACTIVE

3 DATABASE_RETRIEVAL commands

AMEND_SELECTION

Description: This command can be utilized after the elements or species or the system have been defined (with the command DEFINE_ELEMENTS or DEFINE_SPECIES or DEFINE_SYSTEM). The subsequent prompts allow changes to the pre-defined system. By answering Y (Yes) or N (No), each of the previously selected elements, species or phases can be accepted or rejected (this is not the case for the constituents or the entire system).

Syntax: AMEND_SELECTION [KEYWORD]

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

Description: This command appends 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.

The APPEND_DATABASE command will also enter 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 will be replaced with the values retrieved from the appending database.

The APPEND_DATABASE command is similar to the SWITCH_DATABASE command, but does not reinitialize the DATA module and GIBBS workspace. Therefore, it is equivalent to the SWITCH_DATABASE USER command sequence. For further information, see SWITCH_DATABASE.

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 the NEW_DIRECTORY_FILE command, are listed by pressing enter without giving any argument. The user can also supply his own database by first giving the argument `USER` and then the database name and, if it is not located in the current working directory, its correct path.

After this command executed for the first time, the `TDB_XYZ:` prompt (`XYZ` stands for the name of the primarily switched database) is changed to the `APP:` prompt for proceeding further commands related to appending data.

Syntax: APPEND_DATABASE <ADDITIONAL DATABASE NAME>

Prompt: Use one of these databases

```
...
PURE   = SGTE pure element database
...
SSUB   = SGTE substance database 1997
...
USER    = user defined database
DATABASE NAME /XYZ/: <additional database name>
```

Options: `additional database name` -- the name of an existing database or a `USER` database definition file (`***setup.TDB`) that corresponds to the appending database.

Notes: 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 will be popped up for opening an appropriate database-setup. The user can then open the chosen USER database, or may cancel such an open session; in the latter case, the program will list all predefined databases in the installation area, and the user can then select one of them 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) that contains the setup definitions of the USER database, and its correct path, must be provided.

In Linux, the filename of a USER database, or one of the predefined names, can be used under the ensuing prompt:

FILENAME: 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 primarily switched database.

After this command, those commands for defining system (in terms of either elements or species), rejecting/restoring phases or species, and retrieving data must be repeated; however, their second keywords and parameter values might have to be different from the previous ones.

DATABASE_INFORMATION

Description: 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.

Syntax: DATABASE_INFORMATION

DEFINE_ELEMENTS

Description: This command can be used to define the system in terms of elements. All possible species that can be formed by the given elements will be retrieved from the database. The different 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, will be defined into the system. Note that since the TCC version N, up to 40 elements can be defined into a single system (previously 20 elements).

Syntax: DEFINE_ELEMENTS <ELEMENT1, ELEMENT2, …>

Options: `element&` -- a list of elements to be defined into the system

Notes: When appending database(s), this command, or the command DEFINE_SPECIES or DEFINE_SYSTEM, must be repeated with the same or similar elements as defined in the first switched database. See detail in the command DEFINE_SYSTEM.

DEFINE_SPECIES

Description: This command can be used to define the system in terms of species. Only those species that are given will be 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, will be defined into the system. Up to 1000 species can be defined into a single system.

Syntax: DEFINE_SPECIES <SPECIES1, SPECIES2, …>

Options: `species&` -- a list of species to be defined into the system

Notes: When appending database(s), this command, or the command DEFINE_ELEMENTS or DEFINE_SYSTEM, must be repeated with the same or similar elements as defined in the first switched database. See detail in the command DEFINE_SYSTEM.

DEFINE_SYSTEM

Description: This command defines the system in terms of either elements (equivalent to DEFINE_ELEMENT) or species (equivalent to DEFINE_SPECIES). 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, will be defined into the system.

Syntax: DEFINE_SYSTEM <ELEMENT/SPECIES1, ELEMENT/SPECIES2, …>

Prompt: ELEMENTS: <element1, element2, …>, or

SPECIES: <species1, species2, …>

Options: `SPECIES` or `ELEMENTS` (default keyword) -- asked only in certain cases

`element&` -- a list of elements for the defining system should be given

species& -- a list of species for the defining system should be given

Notes: When appending database(s), this command, or the command DEFINE_ELEMENTS or DEFINE_SPECIES, must be repeated with the same or similar elements as defined in the first switched database. However, the user should be aware of that different databases might contain different elements and have different species definitions, so that he/she should avoid defining elements/species that are missing in the appending database(s). Otherwise, the program will indicate such missing elements/species, and ignore them in subsequent steps. But additional elements/species, and of course additional phases, that are 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.

GET_DATA

Description: This command enters the defined system's elements, species, phases, and the connected parameters obtained from either the primarily 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.

Syntax: GET_DATA

Notes: Only after executing this command is it possible to 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.

LIST_DATABASE

Description: This command lists all elements, species, phases or phase constituents in the present database.

Syntax: LIST_DATABASE [KEYWORD]

Options: keyword -- One of the keywords ELEMENTS, SPECIES, PHASES or CONSTITUENT must be used to indicate what is to be listed

ELEMENTS -- all available elements, their 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 their 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 the following, 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 that those interaction parameters are, by default, set to zero, which are originated from binary systems not included in the database.

LIST_SYSTEM

Description: This command lists all elements, species, phases or phase constituents in the defined system. It is functional only after a system has been defined.

Syntax: LIST_SYSTEM [KEYWORD]

Options: keyword -- One of the keywords ELEMENTS, SPECIES, PHASES or CONSTITUENT (as described in the LIST_DATABASE command) must be used to indicate what to list.

NEW_DIRECTORY_FILE

Description: This command opens a new database initiation file (or called database directory file) generated by a local database manager or user, for accessing additional databases that are not predefined in the original database initiation file. It is available since the TCCM.

The original database initiation file is automatically copied by the TCC installation script to the main database area, on a local computer for an independent installation locates or a connected server for a server installation. That is the TC_INITD (or TC_INITD.TDB) file in the \DATA\ area that is under the directory defined by the TCPATH parameter on Windows environments, or the initd.tdb file in /data/ area that is under the directory defined by the TC_DATA parameter on Linux. However, if there are too many databases at one installation or if there are some user-specified databases for some particular purposes, a local database manager can generate some additional database initiation files, or each user could have his 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 several additional database initiation files that define accessing paths to various database groups (various 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_DICTORY_FILE <ANOTHER DATABASE-INITIATION-FILE NAME>

Prompt: File with database directory /TC_INITD/: <database-initiation-file name>

Options: another database-initiation-file name -- the name of the next database initiation file (either additional or original) that is to be switched on in the following sessions.

Notes: 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, an "Open" window is popped up on screen for specifying the new database initiation file.

If the Thermo-Calc Console Mode is run on a Linux platform, this new database initiation file must be located in the current working directory.

REJECT

Description: This command rejects 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). Phases/species/constituents that are not included on the list cannot be entered without first being restored. 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, will be 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 1: REJECT [KEYWORD]

(keyword = ELEMENTS or SPECIES or PHASES)

Prompt: If keyword: <name1, name2, ...>

Syntax 2: REJECT [KEYWORD]

(if keyword = CONSTITUENT)

Prompt: PHASE: <phase name>

SUBLATTICE NUMBER: <sublattice number in the phase>

CONSTITUENT: <constituent(s) in the sublattice of the phase>

Syntax 3: REJECT [KEYWORD]

(if keyword = SYSTEM)

Options: keyword -- ELEMENTS/SPECIES/PHASES/CONSTITUENT/SYSTEM

name& -- names of the pre-defined elements/species/phases/constituents

Notes: One of the keywords ELEMENTS, SPECIES, PHASES, CONSTITUENT or SYSTEM must be used to indicate what is rejected:

ELEMENTS: The given elements will be rejected.

SPECIES: The given species will be rejected, making it impossible to form them from the defined elements.

PHASES: The given phases will be rejected, making it impossible to form them from the defined elements or species.

CONSTITUENT: The given constituent in one phase will be rejected.

SYSTEM: The DATA module is reset to its initial state; GIBBS is reinitiated and data already entered to GES5 will be lost.

In case of rejecting constituents, further details should be specified in the subsequent prompts:

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

Description: This command does the opposite of the command REJECT, *i.e.,* restoring already explicitly rejected elements, species, phases or constituents; but it cannot 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).
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, will be restored in the defined system.

Syntax 1: RESTORE [KEYWORD]

(if keyword = ELEMENTS or SPECIES or PHASES)

Prompt: keyword: <name1, name2, ...>

Syntax 2: RESTORE [KEYWORD]

(if keyword = CONSTITUENT)

Prompt: PHASE: <phase name>

SUBLATTICE NUMBER: <sublattice number in the phase>

CONSTITUENT: <constituent(s) in the sublattice of the phase>

Options: keyword -- ELEMENTS/SPECIES/PHASES/CONSTITUENT

name& -- names of the pre-defined elements/species/phases/constituents

Notes: One of the keywords ELEMENTS, SPECIES, PHASES, or CONSTITUENT must be used to indicate what is restored:

ELEMENTS: The given elements will be restored.

SPECIES: The given species will be restored and thus possible to form from the defined elements.

PHASES: The given phases will be restored and thus possible to form from the defined elements or species.

CONSTITUENT: The given constituent in one phase will be restored.

In case of restoring constituents, further details should be specified in the sequent prompts:

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 of the constituent to be restored.

SET_AUTO_APPEND_DATABASE

Description: This command enforces an automatic action that will append thermodynamic data from a chosen database regarding your system to the data you already have about the system from your current database (that is, the default database or the database that was last chosen with SWITCH_DATABASE or APPEND_DATABASE). This is useful for simultaneously retrieving both thermodynamic and mobility data for a defined system when performing a DICTRA simulation.

This command should be used before defining a system (by the command DEFINE_SYSTEM, DEFINE_ELEMENT, or DEFINE_SPECIES) and retrieving the data (executed by the GET_DATA command) from the primarily switched 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 primarily switched database will be appended. There is no possibility to manually list, reject and/or restore any phase that are available in the appending database. When retrieving data by executing the GET_DATA command afterwards, all the phases that exist in the appending database but not in the primarily switched database will be automatically rejected. The prompt will be keep as for the primarily switched database, TDB_XYZ: (where XYZ stands for the name of the primarily switched database) until the execution of the GET_DATA command.

Therefore, if one wants to selectively append more phases from a secondary database to a system that has been defined and retrieved from a primarily switched database, then the APPEND_DATABASE command and those sequential commands (such as DATABASE_INFORMATION, DEFINE_SYSTEM, DEFINE_ELEMENT, DEFINE_SPECIES, LIST_SYSTEM, REJECT, RESTORE and GET_DATA) should be used instead, prior to this SET_AUTO_APPEND_DATABASE command.

Syntax 1: SET_AUTO_APPEND_DATABASE <ADDITIONAL DATABASE NAME>

Prompt: DATABASE NAME /XYZ/: <additional database name>

Options: additional database name -- the name of an existing database or a USER database that corresponds to the automatically-appending database.

Notes: Although it is impossible to manually list, reject and restore any phase from the appending database, the DATA module will automatically ignore all the phases that do not exist in both the primarily switched database and the appending database, and just append the data for the phases (that are also available in the primarily switched database) from the appending database, as informed on screen when executing the GET_DATA command afterwards.

SWITCH_DATABASE

Description: This command switches (or changes) from the current database to a new one, and reinitializes 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 (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 the NEW_DIRECTORY_FILE command, are listed by pressing the <RETURN> key without giving any argument. The user can also supply his own database by first giving the argument USER and then the database name and, if it is not located in the current working directory, its correct path. The second part of the prompt TDB_XYZ indicates the present database XYZ.

Syntax: SWITCH_DATABASE <NEW DATABASE NAME>

Prompt: Use one of these databases

```
...
PURE    = SGTE pure element database
...
SSUB    = SGTE substance database 1997
...
USER     = user defined database

DATABASE NAME /XYZ/: <new database name>
```

Options: `new database name` -- the name of an existing database or a `USER` database definition file (`***setup.TDB`).

Notes: `DATABASE NAME`: The new database is specified by typing the abbreviation given in the front of one of the available predefined databases. For your convenience when switching/appending your own databases or lately purchase databases from TCS or its agents, you can simply add them to the predefined database list in the database initiation file `TC_INITD.TDB` or `initd.tdb` of your installed Thermo-Calc/DICTRA packages.

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.

In Linux, the filename of a USER database, or one of the predefined names, can be used under the ensuing prompt:

FILENAME: 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.

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 will be popped up 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 very 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 will be used in the calculations. It is not advisable to use this method for large databases since they load very 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.

4 EDIT_EXPERIMENTS commands

Some special commands are designed commands only for the ED_EXP module and can also be used in *.POP or *.DOP files. See Appendix – Commands in the Experimental Data files (*.POP, *.DOP) for more information.

Some important notes for running the ED_EXP module:

- The first necessary command after entering the ED_EXP module for the first time (through the EDIT_EXPERIMENTS command in the PARROT module) should always be the READ (READ_BLOCK) command, in order to load the experimental data block from the current work *PAR-file for editing. Furthermore, the READ command must also be used prior to any other ED_EXP command if the module has been reinitiated, or if no experimental data block has not been previously read from a work file compiled with a proper experimental data **.POP/.DOP** file, or if the user wants to change to another data block for editing.

- Do not forget to give a SAVE-WORKSPACE command before going BACK to the PARROT module, if any change has been made in the ED_EXP module.
- Note that although the TABLE_HEAD, COMMENT and FLUSH_BUFFER commands are visible in the ED_EXP module, they cannot be used in the ED_EXP module; they are only functional in an *.POP or *.DOP file.

COMPUTE_ALL_EQUILIBRIA

Description: In ED_EXP and PARROT modules, each experiment is treated as an individual equilibrium with some measured values. These are created with the CREATE-NEW-EQUILIBRIUM command, and stored in an experimental data *.POP file and then compiled and saved in the POLY3 workspace of a PARROT work *.PAR file.

With the COMPUTE-ALL-EQUILIBRIA command, 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 will be skipped.

Note that "current" or "present" experimental point means the lately selected or calculated one. If one has already used this command once, the current point will turn to be the last point in the data block; in such a case the user must first use the SELECT_EQUILIBRIUM command so that the current point will be switched to a desired one.

Syntax: COMPUTE_ALL_EQUILIBRIA

Note: During the COMPUTE-ALL-EQUILIBRIA command, there will always be a list output on screen, which consists of 6 columns for all experimental points available in the current data block. The first column is the *equilibrium identifier* (a number) assigned by the CREATE_NEW_EQUILIBRIUM command, the second the *data label* assigned by the LABEL-DATA command, 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 command 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 has been used for some experimental points, the listing will be slightly different for such points. The 3-4 columns will be displayed with "* alt *", instead. If an alternate calculation is failed at one experimental point, the point will automatically be assigned with a zero weight, and a warning message is shown above the data line (with all six columns).

EXPERIMENT

Description: This command is mainly 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: the first one as a quantity relation, and the second one as the uncertainty of the value for the quantity, with a colon "`:`" separated them. The quantity relation part 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%).

Note that an "experiments" that use an inequality will give 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 will increase with the value.

Syntax: EXPERIMENT <EXPERIMENT>

The `experiment` must be typed after the EXPERIMENT command. Several experiments can be given on the same line.

The Syntax of EXPERIMENT is similar to that of a `condition`. Usually, a state variable set equal to a value with a given uncertainty is specified after the colon "`:`". For instance, `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%
```

Important Note: A typical "experiment" that is added interactively is to specify 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 one may by this command add an experimental information to enforce that. As an example, one may have to suppress the HCP phase in an existing experimental point:

```
CHANGE-STATUS HCP=DORMANT
EXPERIMENT DGM(HCP)<-.001:.0001
```

Note that `DGM` is the POLY variable for the driving force that is already divided by `RT`, and it is thus dimensionless. Only `DORMANT` phases can have driving forces larger than zero, as this indicate that they would be stable if they were allowed to form. The "experiment" in the above case will try 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. Note that the reference state of the component Pb must have been set with a SET-REFERENCE-STATE command. 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

Description: This command makes it possible to transfer a calculated value from an equilibrium to an optimising 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), and this option should be used with care and understanding.

Syntax: EXPORT <FUNCTION NAME>#<N>

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 "#".

Important note: The function name is UPPER/lowercase sensitive, and it should normally be given in UPPER case. For example,

```
ENTER_FUNCTION STRNGY=GM(FCC).X(CU);
EXPORT STRNGY#6
```

This will transfer 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

Description: This command is the inverse of the `EXPORT` command. This makes it possible to transfer the value of one of the optimizing variables to a constant. Normally, it is directly used inside an experimental data (`*.POP` or `*.DOP`) file.

One case when this is useful is that if several experiments have been 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 of course 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 will try to find the activity that gives the best fit. Note that if an approximate value of the activity is known, one can of course supply that as an experiment.

Another case when this command is useful is given in the `*.POP` file of the *Example 36* as given in the `/TCEX/` subdirectory of a Thermo-Calc installation and described in *Thermo-Calc Console Examples*.

Syntax: IMPORT <CONSTANT NAME>#<N>

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 `#`. Note that the constant name is UPPER/lowercase sensitive, and it should normally be given in UPPER case. For example:

```
ENTER_CONSTANT ACU=0.1
IMPORT ACU#2
```

This will transfer the value of the optimizing variable 2 (*i.e.,* "`V2`") to the constant `ACU`.

LABEL_DATA

Description: This command adds 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 `SET-WEIGHT` command to assign the same weight to all equilibria with the same label.

Syntax: LABEL_DATA A<BCD>

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

Description: This command is used to 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. The user can always use the SELECT_EQUILIBRIUM command to select a specific point as the current equilibrium point for such a list.

Syntax: LIST_ALL_EQUILIBRIA

MAKE_POP_FILE

Description: 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 `*.POP/*.DOP` file.

This output is not always written *without* errors, and it should be checked carefully. 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. Note that if an existing *.POP file name is specified, it will be overwritten. In DICTRA, an *.DOP file is written or overwritten.

Syntax: MAKE_POP_FILE <POP/DOP-FILE NAME>

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 (READ_WORKSPACE = READ_BLOCK)

Description: This command (previously named as READ-BLOCK while now as READ-WORKSPACE) is equivalent to the READ-WORKSPACES command in the POLY module, but it will only read the `poly3` workspace from the work file set by the SET-STORE-FILE command in the PARROT module.

Important Note: This command must be given each time the ED_EXP module is entered, unless one has already given it once in ED_EXP and has not used any LIST-RESULT or OPTIMIZE command 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 `FLUSH_BUFFER` command in the original POP or DOP file, then there is only one data block with the number `1`.

RESTORE_ALL_WEIGHTS

Description: This command is used to restore a specific **weight-set** (that has previously saved by the STORE_ALL_WEIGHTS command) and assign all the different experimental points in the current data block with their previously-set weights in the sequential assessments.

Syntax: RESTORE_ALL_WEIGHTS <WEIGHT-SET NUMBER>

Prompt: `Set number (0 for list) /0/: <weight-set number>`

Specify a previously-stored weight-set for all the different experimental points. The default value `0` is for a list on screen.

SET_ALTERNATE_CONDITION

Description: This is a very special command mainly used in the experimental data `*.POP/*.DOP` file but also possible in the ED_EXP module. It is used only when one has specified that the *alternate mode* is set on in the PARROT module. The meaning of the alternate mode is described in the User's Guide of the PARROT module.

The command syntax for this command is the almost the same as for the POLY command `SET-CONDITION`, while the uncertainty should also be specified.

Syntax: SET_ALTERNATE_CONDITION <alternate condition(s)>

Each alternate condition (including the normally POLY-module condition plus uncertainly; see details below) must be given explicitly, but can be given on the same line or on separate lines with each one started with the command.

Prompt: State variable expression: <state variable name or linear expression>

This question is rather cryptic but the user is expected to give either just a state variable or a linear expression of 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)

There are many more state variables can be used in conditions. For more information, give an INFO STATE_VARIABLES command in the POLY module. A condition is normally just a value of a single state variable with its value. For instance,

```
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
```

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. Note that after the equal sign only a numeric value is allowed.

Factor: <a factor for the state variable, or a continuation>

This question is even more cryptic, and it means that the user 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 will be prompted if a single state variable name has not given in either the prompt “State variable expression” or “Factor”, or a state variable expression has been 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. The user needs to specify here a state variable or a complete state variable expression, or complete the unfinished state variable expression. If a numeric factor has been given before this prompt, only one state variable can be specified; otherwise, the program will only take 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%).

SET_WEIGHT

Description: Each experimental value has an uncertainty associated with it, specified by the value after the colon ":" in the EXPERIMENT command. During an optimization the absolute difference between the experimental and calculated values gives a contribution to the sum of error.

By the SET_WEIGHT command, one may 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 they should have higher weight. The special value zero means that the set of equilibria should be ignored in editing in ED_EXP (obviously as the COMPUTE_ALL_EQUILIBRIA command is given) and in optimization in PARROT.

Note that the weight may be needed to obtain a balance between different kinds of experiments. For example, if there are only 5 experimental values of the composition of a phase diagram but 500 experimental values of activities or enthalpies, then the 5 composition points may have to be given higher weight than unity, otherwise they will not be well described by the optimizing procedure.

Syntax: SET_WEIGHT <WEIGHT VALUE> <SELECTION>

Prompt: `Value /1/: <weight value>`

Specify a weight of the "experiments" in the specified equilibria (which will be asked in the next prompt). The contribution to the "sum of errors" of these experiments will be multiplied by this weight. Note that the weight is squared, thus one should 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>`

The equilibria, which are available in the read data block from the current work file and will be given the above-defined weight, should be specified here.

The selection may be given as a range but the equilibrium numbers must then be separated by a minus sign but nothing else. Note that the starting and ending equilibrium numbers must be available in the read data block read from the current work file. For instance, `5-101`, means that all equilibria starting from the identifier `5` until the identifier `101`. Some of numbers between these two identifiers may be not existing in the data block.

The selection may also be given as one or more labels (which should be separated by space but nothing else). All equilibria with the specified label(s) in the read data block will be given the above-defined weight.

By pressing `<RETURN>` to accept the default value `/PRESENT/`, the above-defined weight will only be assigned to the "current" or "present" experimental point (which is selected by the POLY command SELECT_EQUILIBRIUM). Note that if a COMPUTE_ALL_EQUILIBRIUM command has been used but no specific equilibrium point has been selected afterwards, the last point in the read data block remains the `PRESENT` point.

STORE_ALL_WEIGHTS

Description: This command is used to 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 very useful when a user wishes 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 by the `RESTORE_ALL_WEIGHTS` command sequentially.

Syntax: STORE_ALL_WEIGHTS <WEIGHT-SET NAME>

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

Description: One of the tedious problems in the ED_EXP module is 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 of the same kind equilibrium and each must have its start value set. When one has managed to calculate equilibrium one of such experimental point, this command is useful to transfer the site fractions from a previous successfully-calculated equilibria to the present experimental point.

The <equilibrium number> should be the numeric identifier for the equilibrium from which the start values should be copied to the present experimental point, or "PREVIOUS" by pressing <RETURN>.

Syntax: TRANSFER_START_VALUE <EQUILIBRIUM NUMBER>

Prompt: `From equilibrium /PREVIOUS/: <equilibrium number>`

Specify the equilibrium number from which the start values should be copied to the present experimental point. The PREVIOUS is the default selection; if not, then the numeric identifier must be specified here.

5 GIBBS commands

ADD_COMMENT

Description: This command allows users to jot down notes about parameters or something else that might be difficult to figure out in the future.

Syntax: ADD_COMMENT

AMEND_ELEMENT_DATA

Description: 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, simply because of that the element data available in the database have been set by the database-developer for the purpose of internally consistency. Except for the mass, the other values have no influence on the calculations.

Syntax: AMEND_ELEMENT_DATA

Prompt: `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 <RETURN> to accept the default SER or specify a new SER for the element.

Important Note: The default name of SER (the Stable Element Reference State) for elements should not be changed, if for which data have been 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, one can specify a new SER.

```
NEW ATOMIC MASS /xx.xxxx/: <yyyyyy>
```

Press <RETURN> 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 <RETURN> 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 <RETURN> 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>
```

Specify a proper index for the default listing parameters (symbol), or press <RETURN> to accept the pre-set index.

The index is for defining the symbol printed in listings of parameters. The symbol can be G, H298 or H0:

- `G` means that the data are referred to Gibbs energy at a variable temperature (also called *Lattice Stability*).
- `H298` means that the data are referred to the enthalpy of the element at 298.15 K and 1 bar.
- `H0` means the same as `H298` but at the temperature 0 K.

Note that the index only changes the symbol, not any value! Normally the index is set correctly by the database. Only when entering data manually, one must 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”.

AMEND_PARAMETER

Description: The user can interactively change the temperature-pressure function of a parameter with this command. This is useful in order to correct typing errors because the old function will be made available for interactive editing on the terminal. For details on how a parameter is defined.

Syntax: AMEND_PARAMETER

Prompt: PARAMETER: <parameter name>

Specify a correct parameter name. If a parameter name is not acceptable or the user only presses <RETURN>, the following warning message appears:

```
*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY
```

The program will further prompt 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 the following types of legal identifiers (G or L, or TC, or BM, or V0 or VA or VB or VC or VK). Same as in the ENTER_PARAMETER command.

```
Phase name (/ABCD/): <phase name>
```

Specify the phase name. Same as in the ENTER_PARAMETER command.

```
Constituent (in SUBLATTICE # /abc/): <species
name>
```

Specify the constituent name. Same as in the ENTER_PARAMETER command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE #
/xyz/): <species name>
```

Specify the interacting constituent name: If there is no interacting constituent, just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE #
/xyz/): <species name>
```

If there are more than one interacting constituents, specify them; otherwise just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
Degree /#/: <degree>
```

Specify a numerical number as the degree of the phase parameter. Same as in the ENTER_PARAMETER command.

After the parameter name is specified correctly, the program will lists out on screen its current definition (either present in database or previously 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 the program prompt for changing the parameter definition, as shown below:

```
DO YOU WANT TO CHANGE THE NUMBER OF RANGES
/NO/: <Y or N>
```

If the user wants to change the number of ranges for the chosen function, or change some of the temperature limits in the definition, by typing Y (Yes), he has to retype both the low/high temperature limits and functions (see all the remaining details in the ENTER_PARAMETER command. Sorry for that there is no other possibility to do so.

If the user does not want to change the number of ranges but wishes to change the function(s) in one or more ranges, by pressing <RETURN> to accept the default answer N (No), the whole definition of the chosen parameter in all ranges (if any) will be listed out 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 following prompt will then be presented:

```
DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y or N>
```

If there are more than one range, this question will be prompted. But, sorry again for that this option (as intended by answering Y) has not been implemented yet. So you just press <RETURN> here!

```
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 one can press <RETURN> 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 the AMEND_SYMBOL command. This routine prompts the user in the following way:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons, "::".

The following commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurences> 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.

Note that when the string is typed the character at the current position has been replaced by an underscore "_". This help can also be obtained on-line by typing "?".

To finish the editing of the current function, one has to type E at the prompt.

```
RANGE NUMBER (0 TO EXIT) /0/: <range number>
```

Give a range number to edit that function, or press <RETURN> or type 0 to exit this command.

AMEND_PHASE_DESCRIPTION

Description: This command can be used to specify/amend phase descriptions, in terms of that if a phase has a miscibility gap, uses a special excess energy model, has a specific additional contribution to its Gibbs energy, among other aspects. They are summarized in the following:

- EXCESS_MODEL
- MAGNETIC_ORDERING
- DEBYE_HUCKEL
- STATUS_BITS
- NEW_CONSTITUENT
- RENAME_PHASE
- COMPOSITION_SETS
- GLASS_TRANSITION
- DISORDERED_PART
- MAJOR_CONSTITUENT
- ZRO2_TRANSITION
- REMOVE_ADDITION
- QUASICHEM_IONIC
- QUASICHEM_FACT00
- QUASICHEM_IRSID
- TERNARY_EXTRAPOLAT
- HKF_ELECTROSTATIC
- DEFAULT_STABLE
- SITE_RATIOS
- FRACTION_LIMITS

For various amending options, their corresponding Prompts and answers are different:

```
For RENAME_PHASE,
  NEW PHASE NAME /ABCD/: <phase name>
For SITE_RATIOS,
  Sites in first sublattice /xx/: <yy>         Sites
in ...
For NEW_CONSTITUENT,
  SUBLATTICE /#/: <sublattice number>
  SPECIES: <species name>
```

```
For COMPOSITION_SETS,
  NEW HIGHEST SET NUMBER /#/: <set number n>
         Major constituent(s) for sublattice 1:
/AB/: <constituent(s)>
For MAJOR_CONSTITUENT,
  Composition set /1/: <composition set number>
  Major constituent(s) for sublattice 1: /AB/:
<constituent(s)>
For FRACTION_LIMITS,
  Element: <el>
  Low fraction limit: /0/:  <appropriate low limit>
  High fraction limit: /1/: <appropriate high limit>
For MAGNETIC_ORDERING,
  THE ANTIFERROMAGNETIC FACTOR /xx/: <yy>
  SHORT RANGE ORDER FRACTION OF THE ENTHALPY /xx/:
<yy>
For DISORDERED_PART,
  Disordered phase name: <phase name>
For EXCESS_MODEL,
  Model name /ABCDEFG/: <model name>
For TERNARY_EXTRAPOLAT,
  Extrapolation method: /TOOP-KOHLER/: <ternary
model> For DEBYE_HUCKEL,       no Prompt.
For HKF_ELECTROSTATIC,  no Prompt.
For QUASICHEM_IONIC,
  Number of bonds/atom /2/: <number of bonds per
atom> For QUASICHEM_FACT00,   no Prompt.
For QUASICHEM_RISID,    no Prompt.
For GLASS_TRANSITION,   no Prompt.
For ZRO2_TRANSITION,    no Prompt.
For REMOVE_ADDITION,    no Prompt.
For STATUS_BITS,
  New status word (in hexadecimal) /04204000/: <new
status bits>
For DEFAULT_STABLE,     no Prompt.
```

Notes:

It is illegal to use the NEW_CONSTITUENT option to add new constituents to the ionic liquid phase. The ZRO2_TRANSITION option has not been completely implemented yet.

There are many more solution models that have been implemented in the GIBBS module (try to type INFO MODELS to see more details on various thermodynamic models). However, they are switched on and handled in different ways (via various GIBBS routines/commands):

- The *first* way is to model the non-ideality as EXCESS parts, *i.e.,* by amending the phase's EXCESS_MODEL and/or TERNARY_EXTRAPOLAT models.
- The *second* way is to model 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* way is to implement 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.

A user may occasionally prefer to change the names of some specific phases for some reasons (for instance, a phase named FE3O4_S may be better to be renamed as “Magnetite” for convenience and presentation preference during the sequential calculations and postprocessing). The option RENAME_PHASE can then be used for this purpose.

If the user for some reasons wants to change the numbers of sites (i.e., the stoichiometric coefficients of various sublattices) in a sublattice phase, he can use the SITE_RATIOS option.

If a user wants to add a new constituent to a phase this can be done by the NEW_CONSTITUENT option. But remember it is illegal to add new constituents to an ionic two-sublattice liquid phase.

For the COMPOSITION_SET term, it is only needed for solution phases that may have miscibility gap(s); however, nowadays this becomes less important and often unnecessary to define additional composition set(s), thanks to that 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).

The MAJOR_CONSTITUENT option can be used for setting major constituent(s) on each sublattice in each composition set for a solution phase. This is useful in order to make calculation 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.

If the user must limit the application range (in terms of mole-fractions of all the involved elements) of a particular solution phase, he can use the option FRACTION_LIMITS. This is very useful and essential for the purposes of avoiding 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.

Phases which can have an order/disorder transformation will have parameters split on two phases; therefore, it is referred as the “two phase” description in the GIBBS module. One of these has sublattices for ordering, the other one represents the disordered state. For the DISORDERED_PART term, it 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). 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 so-called Two Substitutional-Sublattice Ordering Model or Four Substitutional-Sublattice Ordering Model, respectively.

Using the MAGNETIC_ORDERING option, the user can change the magnetic ordering parameter for a certain phase with magnetic ordering contribution, in terms of its anti-ferromagnetic factor [which by default is –1 for BCC phase and –3 for all other phases (FCC, HCP, etc.)] and fraction value of the total enthalpy that is due to short-range ordering above the magnetic transition temperature (which by default is 0.40 for BCC phase, and 0.28 for all other phases (FCC, HCP, etc.)].

If the user wants to change the Excess Model (for interaction energies) from a default model to another one, he can set the option EXCESS_MODEL, in which the model for handling the excess interaction energies in a solution phase can be chosen among:

```
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 slag solution phase (C-A-M)
```

If the user wants to change the extrapolation method from the default Redlich-Kister_Muggianu Extrapolation Model to another extrapolation model, he can set the option TERNARY_EXTRAPOLAT, in which the method for extrapolating from binary to ternary (and higher-order) excess interaction parameters in a solution phase can be chosen among:

```
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
```

If the user wants to use the DHLL (Debye-Hückel Limiting Law) Model and SIT (Specific Ionic Interaction Theory) Model for a dilute AQUEOUS solution phase, he can switch the **DEBYE_HUCKEL** part on, and it will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS phase.

If the user wants to use the hypothetical electrostatic contribution for the chosen phase (It must be an aqueous solution phase using the Complete Revised_HKF Model!), he can switch the ELECTROSTATIC_HKF part on, and it will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS solution phase.

If the user wants to use the Quasichemical-Ionic Model for the chosen liquid phase, he 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 will create a completely new liquid solution phase [namely the QUAS_IONIC phase which will use the Quasichemical Model for the entropy, according to Prof. Mats Hillert (2001: J of Alloys and Compounds, 320, 161-176)], while the original IONIC_LIQ phase will remain in the system and will not be changed by this option. If the user wants to use the Quasichemical Model developed by Kongoli et al. in the F*A*C*T group in Montreal, he must have already 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 will remove the previously-set (if any) ADDITIONAL part on the non-ideality for the chosen liquid phase. Two examples concerning this Quasichemical Model can be found in the Thermo-Calc Console Examples (Examples 49 and 50).

If the user wants to use the Kapoor-Frohberg-Gaye Quasichemical Cell Model [i.e., the Quasichemical Model developed by ISRID, France) for a liquid SLAG solution phase, he/she can use a normal liquid model with specified associates or species and turn on the option QUASICHEM_ISRID.

If the user wants to use the special model for glass transition of a liquid phase, he can switch the GLASS_TRANSITION part on, and it will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen liquid phase.

If the user wants to remove all the selected ADDITIONAL part from the Gibbs energy description for the chosen phase, he can set the option REMOVE_ADDITION. If the user prefers to set a specific phase-status bits for a phase, he can be the PHASE_BITS option to do so, provided that he must completely know the restrict meaning of each part of a phase- status bits and ONLY WHEN HE IS ENTIRELY SURE OF WHAT HE IS ACTUALLY DOING.

If the user wants to have a better guess of which phases that should be stable at the first calculation, he can use the option DEFAULT_STABLE to set default phases that should be stable.

Syntax: AMEND_PHASE_DESCRIPTION

Prompt: `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.

By typing a question mark at the above prompt, a full list of all possibly amended subjects for a phase can be obtained.

In the following, it will be demonstrated that how to amend these different subjects for a specific phase by this AMEND_PHASE_DESCRIPTION command (and its sequent prompts and related inputs):

RENAME_PHASE

If a user wants to change the name of a specific phase for some good reasons (for instance, a phase named FE3O4_S may be better to be renamed as Magnetite for convenience and presentation preference during the sequential calculations and postprocessing), this can be easily done by this option. This is also a way to “delete” a phase by hiding it under a “new” name.

`NEW PHASE NAME /ABCD/: <phase name>`

Give a new phase name for the chosen phase, or press <RETURN> if the user does not want to change it as default shown.

SITE_RATIOS

If the user for some reasons wants to change the numbers of sites (i.e., the stoichiometric coefficients of various sublattices) in a sublattice phase, he can use the option SITE_RATIOS.

`Sites in first sublattice /xx/ : <yy> Sites in second sublattice /xx/ : <yy>`

Specify the desired site numbers for each of the prompted sublattices for the current phase. Pressing <RETURN> to accept the previous definitions.

NEW_CONSTITUENT If a user wants to add a new constituent to a phase this can be done by this option. It is illegal to add new constituents to the ionic liquid phase.

`SUBLATTICE /#/: <sublattice number>`

Specify the sublattice where the new constituents will locate.

`SPECIES: <species name>`

Give a valid species name.

COMPOSITION_SETS If the user wants to change the number of composition sets, he will be prompted:

`NEW HIGHEST SET NUMBER /#/: <set number n>`

The default value (#) will usually be 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 will be deleted. Note that one cannot take away the first composition set.

Give for COMPOSITION SET n the following:

`Major constituent(s) for sublattice #: /AB/: <major constituent(s)>`

Specify the new major constituent(s) for the sublattice #, or press <RETURN> to accept the default one which was automatically set according to the specified composition set of the phase.

This prompt will be 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 If a user wants to set major constituent(s) on each sublattice in each composition set for a phase, he will be prompted:

`Composition set /1/: <composition set number>`

Give the composition set (digit number) for the chosen phase, or press <RETURN> if the user wishes 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 <RETURN> to accept the default one which was automatically set according to the specified composition set of the phase.

This prompt will be 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 This option makes it possible to 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 will automatically ignore 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 very important for solution phases for which the thermodynamic models and assessed data for the phases are for specific composition ranges, for instance, 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.

```
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 >
```

Specify the desired 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). This has to be done for all the possible elements (defined in the phase) in a single AMEND_PHASE_DESCRIPTION FRACTION_LIMITS command-sequence, for instance:

```
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  !
```

MAGNETIC_ORDERING

If the user wants to change the magnetic ordering parameter for the phase, he will be prompted:

```
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>
```

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.).

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 here).

```
Disordered phase name: <phase name>
```

Give the disordered phase name for which there is no ordering sublattice.

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 will also include the Gibbs energy for the disordered phase.

Phases which can have an order/disorder transformation will have parameters split on two phases; therefore, it is referred 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.

This facility was available already in previous versions, but it has now become used more regularly for multicomponent ordering. The “two phase” description means that the “ordered” phase has only parameters that describe the ordering. The “disordered” phase has all parameters for the reference state and those for describing the disordered phase.

The reason for having a “two phase” description is that the ordered phase may occur only is some systems whereas the disordered phase may occur very 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.

But in order to avoid this, the “two phase” description was developed. In this case the parameters describing the disordered phase are not changed but the Gibbs Energy system is simply informed that the Gibbs Energy for the two phases should be added. This is done with the AMEND_PHASE_DESCRIPTION command using the option DISORDERD_PART.

More information on this is given elsewhere in this part. But it may be useful to give the actual method to calculate the Gibbs Energy of a “two phase” model with two or four substitutional sublattices for ordering.

$$G_m = G_m^{ord}(y_i', y_i'') + G_m^{dis}(x_i) - G_m^{ord}(y_i' = x_i, y_i'' = x_i)$$

Note that the mole fractions xi are calculated from the site fractions yi. 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 will have no contribution to the disordered state.

EXCESS_MODEL

If the user wants to change the excess energy model for the phase, he will be prompted:

```
Model name /ABCDEFG/: <model name>
```

The default model is the pre-set model for the solution phase, which is normally the REDLICH-KISTER_MUGGIANU model. One can choose one of the following models 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

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. Note that the REDLICHKISTER_KOHLER model is implemented only for ternary systems.

The MIXED-EXCESS-MODELS option, that works only for a substitutional phase with no sublattice (such as the metallic LIQUID phase), 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)]. Please note that binary interaction parameters for such a pair of constituents must have been entered prior to turning on the non-default Legendre or Polynom models. The standard Example 52 in the Thermo-Calc Console Mode Examples Book is specifically demonstrating the use of such mixed excess binary models. This MIXED-EXCESS-MODELS option will further prompt the following 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 will repeatedly be prompted with the first sub-option (i.e., "First (the independent) constituent:" 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, simply type <RETURN> at such a prompt (implying that there will be no more changes of binary excess model for all other possibly-remaining binary pairs that shall still use the default REDLICH-KISTER model), so you can finish this MIXED-EXCESS-MODELS option.

```
Excess model type: /LEGENDRE/: <desired binary
excess model type>
```

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 shall be used for a certain binary pair, you can either specify the model name (or in short as R-K) or simply type <RETURN>. 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 typing <RETURN>) of the MIXED-EXCESS-MODELS option.

The HKF and PITZER models are legal only for the AQUEOUS solution phases.

The FLORY-HUGGINS POLYMER MODEL is for polymers.

The CENTRAL_ATOM_MODEL is for liquid slag (not completely implemented yet).

TERNARY_EXTRAPOLAT

If the user wants to change the extrapolation method from the default Redlich-Kister_Muggianu Extrapolation Model to another extrapolation model, he can set the option TERNARY_EXTRAPOLAT, in which the method for extrapolating from binary to ternary (and higher-order) excess interaction parameters in a solution phase can be chosen.

```
Extrapolation method: /TOOP-KOHLER/: <ternary
model>
```

Choose an appropriate ternary extrapolation model for the current solution phase, from the following models:

- TOOP-KOHLER for Toop-Kohler Extrapolation Model
- KOHLER-ALL for Kohler Extrapolation Model

- MUGGIANU_RESTOR for Muggianu-Restor Extrapolation Model

Note that 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 a R-K-K Model will be 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, the user shall 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 one has 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 utilized 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). As also demonstrated by the standard Example 52 in the Thermo-Calc Console Examples, this TOOP-KOHLER option will further prompt the following sub-options:
	`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.

DEBYE_HUCKEL	If the user wants to use the DHLL (Debye-Hückel Limiting Law) Model for an dilute AQUEOUS solution phase, he/she can switch this ADDITIONAL part on. Note that this will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS solution phase.
HKF_ELECTROSTATIC	If the user wants to use the hypothetical electrostatic contribution for the chosen phase (IT MUST BE AN AQUEOUS SOLUTION PHASE USING THE COMPLETE REVISED_HKF MODEL!), he/she can switch this ADDITIONAL part on. Note that this will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen AQUEOUS solution phase.
GLASS_TRANSITION	If the user wants to use the special model for glass transition of a liquid phase, he can switch this ADDITIONAL part on. Note that this will remove previously set (if any) ADDITIONAL part on the non-ideality for the chosen liquid phase.
ZRO2_TRANSITION	It is not implemented yet.

QUASICHEM_IONIC — If the user wants to use the Quasichemical Model for the chosen liquid phase, he must first enter it as an ionic liquid phase, and then use this amending option. It will create a completely new phase, the original IONIC_LIQUID phase will remain in the system and will not be changed by this option. The QUAS_IONIC phase will use the Quasichemical Model for the entropy, according to Prof. Mats Hillert (2001: J of Alloys and Compounds, 320, 161-176).

```
Number of bonds/atom /2/: <number of bonds per
atom>
```

Specify the Z value (number of bonds per atom) that is used in the Quasichemical Ionic Model according to Prof. Mats Hillert (2001).

QUASICHEM_FACT00 — If the user wants to use the quasichemical model developed by Kongoli et al. in the F*A*C*T group in Montreal, he can use a normal liquid model with specified associates or speicies and turn on the option QUASICHEM_FACT00. Two examples concerning this quasichemical model can be found in the *Thermo-Calc Console Examples* (Examples 49 and 50).

QUASICHEM_ISRID — If the user wants to use the Quasichemical Model developed by ISRID, France, for a liquid SLAG solution phase, he/she can use a normal liquid model with specified associates or speicies and turn on the option QUASICHEM_ISRID.

REMOVE_ADDITION — If the user wants to remove all the selected ADDITIONAL part from the Gibbs energy description for the chosen phase, he can set this option.

STATUS_BITS — If the user wants to correct the phase status bits (ONLY WHEN THE USER IS ENTIRELY SURE OF WHAT HE IS ACTUALLY DOING!), he will be prompted:

```
New status word (in hexadecimal) /04204000/:
<new status bits>
```

See the command LIST_STATUS for information on the status bits. Normally, these should never be changed in this way, because such phase status bits will be automatically set on when all the phase descriptions (predefined in databases and amended in the GIBBS module) have been defined properly! The only two exceptions are the following: 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.

DEFAULT_STABLE — If the user wants to have a better guess of which phases that should be stable at the first calculation, he can use this option to set such phases as default-stable.

AMEND_SYMBOL

Description: This command can be used to interactively calculate the current values of a function or table (predefined by the switched database or previously entered by the ENTER_SYMBOL command) at the current temperature and pressure condition, and also possible to modify the definitions of variables or functions.

Note that the current values of the temperature and pressure used in calculating the functions or tables are the ones listed in the LIST_STATUS command, and that there is no way to change the current temperature and pressure values interactively.

For modifying 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, their values can be changed.
- For functions, their low and high temperature limits in various ranges (if any), as well their expressions, can be changed.
- For parameters, their low and high temperature limits in various ranges (if any), as well their expressions, can be changed.

Syntax: AMEND_SYMBOL

Prompt: NAME: <symbol name>

Specify the name of a previously entered symbol.

For tables and functions (or parameters which are treated as functions after they were 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 will automatically calculate their values under the current temperature and pressure conditions, and lists out on screen their current values, such as:

```
FUNCTION VALUE  2.52500000E+01
TABLE VALUE  1.56000000E+02
```

For variable and functions (or parameters), there will be some further sequent prompts, which are different depending upon the symbol type to be amended, as shown below:

For a variable:

Value /current value/: <new value>

For a variable, its current value is displayed, and the user can change it to a 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>

If the user wants to change the number of ranges for the chosen function, or change some of the temperature limits in the definition, by typing Y(YES), he/she HAS TO RETYPE both the low/high temperature limits and functions (see all the remaining details in the ENTER_SYMBOL command. Sorry for that there is no other possibility to do so.

If the user does not want to change it, by pressing <RETURN> to accept the default answer N(NO), the whole definition of the chosen function in all ranges (if any) will be listed out on screen, such as:

```
DIFFERENT FUNCTIONS IN THESE RANGES
298.15<T<2000.00
2000.00<T<4500.00
4500.00<T<6000.00
```

and the followings will be prompted:

```
DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y or N>
```

If there are more than one range, this question will be prompted. But, sorry again for that this option (as intended by answering Y) has not been implemented yet. So you just press <RETURN> here!

```
RANGE NUMBER (0 TO EXIT) /0/: <range number>
```

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 one can press <RETURN> 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. This routine prompts the user in the following way:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons (“::”).

The following commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurences> 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.

Note that when the string is typed the character at the current position has been replaced by an underscore "_". This help can also be obtained on-line by typing "?". To finish the editing of the current function, one has to type E at the prompt.

`RANGE NUMBER (0 TO EXIT) /0/: <range number>`

Give a range number to edit that function, or press <RETURN> or type 0 to exit this command.

CHANGE_STATUS

Description: 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 will show that which elements/species/phases have been suspended or restored (entered). Consequently, the status bits for the specified elements or species or phases will be changed, as listed by the LIST_STATUS command.

Syntax: `CHANGE_STATUS <KEYWORD> <NEW STATUS> <NAME1; NAME2; ...>`

Prompt: `FOR ELEMENT, SPECIES OR PHASE /SPECIES/: <keyword>`

Specify the keyword (element or 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, they should be separated by a space and terminated with a ";" 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 ";". Example: 5 1 7-12 FE;

DELETE

Description: Not implemented yet.

Syntax: `DELETE`

ENTER_ELEMENT

Description: The user may specify his system interactively with this command. The program will search the currently switched or pre-set database for data for the given elements. The data for the elements in the database are:

- the mass in g/mol
- the name of the Selected Element Reference State (SER) which normally is the stable phase for the element at 298.15 K
- the enthalpy difference for the element in the SER state at 298.15 K and zero K
- the 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 the CHANGE_STATUS command.

If an aqueous solution phase is involved, one should in the GIBBS module 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 <ELEMENT NAME>

Prompt: ELEMENT NAME: <element name>

The user may 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 will be found in the database for them.

Note that 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 REINITIATE command. Note that the REINITIATE command removes all data, and thus should be executed before any other command.

ENTER_PARAMETER

Description: 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 rather 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 referred to as `TP-Functions`, which is described in more detail in the subject "`FUNCTIONS`" and *Section 11.2.5 – Functions for Temperature and Pressure*. 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. This will be further described in *Section 11.5* about the *Data Structure*. 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.

The user can enter `TP-Function(s)` for a specific parameter for a phase interactively by 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 could be changed later by the command `AMEND_PARAMETER`.

As described in Section 11.4 (Thermodynamic Parameters), a completely valid parameter should have the general form of:

<identifier>(<phase>, <constituent array>; <digit>) <xxx> <expression> <yyy> <keyword Y or N> <zzz> !

where

- <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 sign "!" is used to indicate that the current parameter definition is ended.

The GES parameter name has a general form of:

<identifier>(<phase>,<constituent array>;<digit>)

Examples of `GES parameter name`:

- `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 `GES parameter name` consists of several parts. The first is a type-identifier. The following type-identifiers are legal:

- `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 $\omega_{Pr,Tr}$ for aqueous solute species).

One may also use G for interaction parameters; and on output list (performed by the GES 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 $\omega_{Pr,Tr}$ of aqueous solute species since TCCP/TCW2.

The following types of identifiers are available since TCCN/TCW1 and are further expanded since TCCR/TCW4:

- VO: Molar volume at 298.15 K and 1 bar (a numeric value only);
- VA Integrated thermal expansivity;

$$\int_{298.15}^{T} \alpha(T)dT$$

- VB Bulk modulus at 1 bar;
- VC Isothermal compressibility;
- VK High-pressure fitting parameter.

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.

Specifying the phase name always in uppercase is recommended; however, if one would prefer to write it as a mixture of uppercase and lowercase, it will then automatically convert 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 the ENTER_PARAMETER is executed.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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 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))`

If you press `<RETURN>` when you are asked for a parameter name or if you have improperly input the entire parameter name, you will be asked for each of these items in the name.

Syntax: `ENTER_PARAMETER`

Prompt: `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 the user only presses `<RETURN>`, the following warning message appears:

```
  *** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY
```

and the program will further prompt 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 the following types of legal identifiers:

- `G` Standard energy parameter (Gibbs energy of formation);

- L Excess energy parameter (Gibbs energy of interaction);
- TC Curie temperature parameter for magnetic ordering;
- BM Bohr magneton number parameter (BMAGN) for magnetic ordering; (or Born function $\omega_{Pr,Tr}$ for aqueous solute species)
- V0 Molar volume at 298.15 K and 1 bar (a numeric value only);
- VA Integrated thermal expansivity;
- VB Bulk modulus at 1 bar;
- VC Isothermal compressibility;
- VK High-pressure fitting parameter.

Note that quantities as H (enthalpy), S (entropy), V (Volume), F (Helmholtz energy), *etc.,* can be calculated from the Gibbs energy when necessary.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> for the same type identifier, or specify a new type.

```
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.

Specifying the phase name always in uppercase is recommended; however, if one would prefer to write it as a mixture of uppercase and lowercase, it will then automatically convert all lowercases to uppercase, as the GIBBS module only recognizes uppercase phase names. It is important that if a phase bears a legal phasetype (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 the ENTER_PARAMETER is executed.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> for the same phase, or specify a new phase name.

```
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. Note that it is the species name, not the stoichiometric formula, that is needed here.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> if the constituent is the same, or specify a new species name.

For phases with several sublattices, the program will ask for one constituent in each sublattice.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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))

```
INTERACTING CONSTITUENT (IN SUBLATTICE  #
/xyz/): <species name>
```

An interaction parameter, which is used to describe the excess term of a quantity, must have two or more constituents that "interact each other" on a specified sublattice site of the given phase. It is arbitrary which of these constituents that is given as the first constituent and which are given as the interacting constituents. But the program will always sort the constituents (in each sublattice) in alphabetical order when the parameter name is written as prompt (for entering its parameter value) and when the parameter is listed out (using the GIBBS command LIST_PARAMETER or LIST_PHASE_DATA). This is very important for all asymmetric interaction parameters where the sign of the interaction parameter must depend on the appearance order.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> if the constituent is the same, or specify a new species name.

Note! To cancel the default value of the interacting constituent you must type NONE or the name of another constituent.

This question will be repeated till all the interested interacting constituent(s) on a specific sublattice in the phase have been specified, and finally a simple <RETURN> has been enforced.

```
Degree /#/: <degree>
```

Specify an integer number (a value from 0 through 9) as the degree of compositiondependent interaction contribution for the phase parameter. This prompt is only valid 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).

The meaning of the degree is model-dependent. 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.

This question may not be prompted if this command is used for entering standard G parameter for a pure component (end-member), as its degree should be always 0.

After this prompt, the program will echo on the screen the full TP-Function of the desired 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. *Same as for entering functions.*

```
Function: <definition for a function>
```

A TP-Function consists of terms in T and P. *Same as for entering functions.*

```
& <continuation of the definition for the
current function>
```

Continuation of a TP-Function definition. *Same as for entering functions.*

```
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* has been entered as a negative number) for the current TPFunction. *Same as for entering functions.*

```
Any more ranges /N/: <Y or N>
```

Specify a Y (YES) for more function(s) or N (NO) for ending this command. Same as for entering functions.

ENTER_PHASE

Description: Normally, data for a system are retrieved from the preset database after the elements and species have been specified. This is automatically done through the `GET_DATA` command inside DATA module. In some cases, the user wants to specify a phase interactively by this command.

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 thus 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) that is used to describe the binary, ternary and/or higher-order interactions between constituents (see more details in the section of “Regular/Subregular Solution Model”);
- 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.

By the current ENTER_PHASE command, the phase name, phase-type, sublattice number, and constituents or constituent array(s) for the phase can be 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 and AMEND_PHASE_DESCRIPTION should be used sequentially.

Syntax: ENTER_PHASE

Prompt: 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 phases. One can press <RETURN> if the phase is ordinary.

The following phase-type codes can be defined:

- G: for a gaseous mixture phase. Note that there may only exist one gas phase in a system.
- A: for an aqueous solution phase.
- L: for a liquid solution phase [but not A (aqueous) or Y (ionic liquid)].
- Y: for an ionic liquid solution phase (that is specially treated by the Ionic TwoSublattice Liquid Model).
- I: for a phase with charged species [but not G (gaseous), A (aqueous) or Y (ionic liquid)].
- F: for an ordered FCC or HCP solution phase with 4 substitutional sublattices (additionally, such a phase can also have an interstitial sublattice).
- B: for an ordered BCC solution phase with 4 substitutional sublattices (additionally, such a phase can also have an interstitial sublattice).

Note that 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) modeled as a substitutional phase without sublattice, too.

Since TCCS/TCW5, the phase-type F has been extended, and the new phase-type B has been implemented. These two 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 4 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 neighbors. 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, they 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.

Previously (within versions prior to TCCS/TCW4), no excess interaction parameters were allowed with the phase-type option `F`, but this has been modified since TCCS/TCW5. However, there are restrictions on the excess parameters allowed with the phase-type option `F`. One can only have excess parameters of the following types:

```
L(phase,A,B:*:*:*;0…9)
```

```
L(phase,A,B:C,D:*:*;0…9)
```

The asterisk "`*`" means that the interaction parameter is independent of the constituents on its corresponding sublattice. No ternary interaction parameters (*i.e.,* with 3 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 SubstitutionalSublattice 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 4 substitutional sublattices for ordering).

For ordered BCC phases, the phase-type option B means the same thing but the situation is a bit more complicated, as the 4-sublattice ordering phase represents an irregular tetrahedron with 2 pairs of sites that are next nearest neighbors. 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 4 nearest neighbor bonds between A and B atoms, whereas the end-member described by the G(phase,A:B:A:B) term has 2 nearest neighbor bonds between A and B atoms and 2 next nearest neighbor bonds (for detailed reference one may visit the Crystal Lattice Structure web page http://cst-www.nrl.navy.mil/lattice/index.html).

The first end-member [described by the G(phase,A:A:B:B) term] represents B2ordering and the second [described by the G(phase,A:B:A:B) term] stands for B32-ordering. There are 2 permutations of the G(phase,A:A:B:B) term and 4 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 additional advanced feature with the phase-type options `F` and `B` is that a composition set that represents the solution phase will have 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

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 recommended to use integer numbers as sites if possible. For substitutional phases with no sublattices, this question will not shown up.

NAME OF CONSTITUENT: <constituent name(s)>

For each sublattice the user must 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 will be repeated till pressing only <RETURN> or giving a semi-comma ";", meaning that the list of constituents is terminated by ";" or an empty line.

Will you add constituents later /NO/: <Y or N>

If Y (Yes) is given as the answer, the user may later add other constituents to the phase by specifying new components in *e.g.* an ENTER_PARAMETER command. The default answer is N (No) (or pressing <RETURN>).

If illegal constituents are used, the program will later on give an error message, *e.g.,* when parameters are entered.

Do you want a list of possible parameters /NO/: <Y or N>

If Y (Yes) is given, the program will generate a list of all possible parameters up to the fifth interaction order for the entered phase. By pressing <RETURN> or typing N (No), no list will show up.

ENTER_SPECIES

Description: The user can interactively specify species made up from the already entered elements. For each species, its name and chemical formula must be given. Note that all elements will automatically be entered as species simultaneously. 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 <SPECIES NAME> <CHEMICAL FORMULA>

Prompt: `SPECIES NAME: <species name>`

The user must supply a unique name of a species. This is usually its chemical formula but it may be any string which starts with a letter and contains only letters or digits, parenthesis "(" and ")", and the underscore character "_". The special combination /– or /+ used to denote charge is also legal. For instance,

```
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 stoichoimetry of the species is the chemical formula of the species. The entered name of the species is the default answer to this question.

The elements in the chemical formula are normally separated by stoichiometric numbers. Neither parenthesis "(", ")" nor the underscore character "_" is allowed in the chemical formula, while the special combination /- or /+ can be used.

In upper case mode (see the REINITIATE command), 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.

Note that all elements in a species must have been entered prior to entering the species.

ENTER_SYMBOL

Description: Normally, a symbol can be used to represent a numeric quantity, a function or a table. The entered symbol (with a symbol name of maximum 8 characters) can be used later when parameters are entered, and this represents a flexible way to manipulate thermodynamic functions. Symbols are especially useful if the same function or table is used in several thermochemical parameters.

A symbol can be entered by this command in the GIBBS module. The symbols entered in the GIBBS module can also be listed and used in the PARROT module (for data optimization) where one may use the GIBBS-entered symbols (variables, functions or parameters) to define parameters that shall be optimized. But symbols entered in the GIBBS module are not the same symbols as defined in the POLY and POST modules.

If a parameter is specified, this command is equivalent to the ENTER_PARAMETER command. If the user wants to define a constant quantity, he can enter it as a simple function with a constant value in all (temperature) ranges.

Syntax: ENTER_SYMBOL

Prompt: VARIABLE, FUNCTION, TABLE OR PARAMETER? /FUNCTION/: <keyword>

The keyword can be variable, function, table or parameter.

- Functions are expressions of state variables or other functions.
- Variables are similar to functions because they can also be expressions of state variables. It is possible to enter a variable with a new expression anytime.
- Tables are used for listing results. A table consists of a list of any number of state variables, functions or variables.
- Parameter symbols are specially used for assigning the values of phase parameters (which are defined by both the commands ENTER_PARAMETER) to specific characterized symbols by this option. Note that if the phase parameter has not been defined yet, the parameter symbol will be assigned with zero or with a symbol which is valued as zero. In this way the user is easy to refer such 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.

Depending upon the chosen type of symbol (variable, function, table or parameter), the different questions will be proceeded, as shown below:

If the symbol is a variable, its value should be assigned:

VALUE: <numeric value of a variable>

For a *variable*, assign its value here. Note that only a constant numeric value, not an expression, is acceptable.

If the symbol is a function, you will be asked to 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>
```

For a *function*, 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 will be 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 will also be taken as pressure limit values (in Pascal).

The temperature/pressure limits for the functions are checked during calculations. An indicator will be 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 will be conducted using the TP-Function 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. Not more than 78 characters can be written on a line. If this is not sufficient space on one line, press <RETURN> and continue on the next line. The program will prompt with an ampersand "&" that a continuation is expected.

Note that 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 10 different temperature ranges (and thus have up to 10 different TP-Functions).

This question will be repeated if the current function has more than one expression (PTfunction) in various temperature ranges [or pressure ranges; only if the *lowest limit* has been 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 one may continue to write the function on the new line if one line is not enough for the current TP-Function. If one has finished the current TP-Function, just press <RETURN> 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 always 6000 K.

This question will be repeated if the function has more than one expression (TPFunction) in various temperature ranges. Note that all the temperature limits for the presently-be-entered TP-Function MUST be in an increasing order; otherwise the whole function will be given a single value of zero.

If a negative number has been given as the lowest-temperature limit, it is interpreted as the low-pressure limit (in Pascal), and the "high-temperature limit" will also be taken as the high-pressure limit (in Pascal) for the current TP-Function.

The high-temperature/pressure limit for the current applicable expression (TPFunction) will be 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 will be 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 will be conducted using the TP-Function valid in the nearest temperature/pressure range.

```
Any more ranges /N/: <Y or N>
```

Answer N (No) to finish the definition of the current function. If the user answers Y (Yes) to this question, the program will ask 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: The user must himself make sure that the function and its first derivatives are continuous across break points.

If the symbol is a table, one should 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 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.

If the symbol is a parameter, one should 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. This is the same as in the ENTER_PARAMETER command.

```
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 the user only presses <RETURN>, the following warning message appears:

```
 *** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY
```

The program will further prompt 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 the following types of legal identifiers (G or L, or TC, or BM, or V0 or VA or VB or VC or VK). Same as in the ENTER_PARAMETER command.

```
Phase name (/ABCD/): <phase name>
```

Specify the phase name. Same as in the ENTER_PARAMETER command.

```
Constituent (in SUBLATTICE  # /abc/): <species
name>
```

Specify the constituent name on the specified sublattice site of the given phase. Same as in the ENTER_PARAMETER command.

```
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, just press <RETURN>. Same as in the ENTER_PARAMETER command.

INTERACTING CONSTITUENT (IN SUBLATTICE # /xyz/): <species name>

If there are more than one interacting constituents on the specified sublattice site of the given phase, specify them; otherwise just press <RETURN>. Same as in the ENTER_PARAMETER command.

Degree /#/: <degree>

Specify an integer number (a value from 0 through 9) as the degree for the phase parameter. *Same as in the ENTER_PARAMETER command.*

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. Same as for entering a function (see above).

Function: <definition for a function>

Specify the desired TP-Function for the current temperature range (or current pressure range; only if the *lowest limit* has been entered as a negative number). *Same as for entering a function (see above).*

& <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 has been entered as a negative number). Same as for entering a function (see above).

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 has been entered as a negative number) for the current TP Function. Same as for entering a function (see above).

Any more ranges /N/: <Y or N>

Answer as Y (Yes) or N (No). Same as for entering a function (see above).

LIST_CONSTITUENT

Description: This command lists on the screen all the constitution (as site-fractions) for all the phases in the defined system. This is mainly for the benefit of the software managers. Note that there are no commands in the GIBBS monitor to set the constitution.

Syntax: LIST_CONSTITUENT

LIST_DATA

Description: All data for the current system that have been defined (retrieved from database in the DATA module, or directly entered in the GIBBS module) within the current GIBBS workspace are written in a readable manner on the output file. If no output file is specified, the data are shown up 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.

Note that for encrypted commercial databases one may not be able to use this command to list any retrieved data. However this command can be used for listing 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

Prompt: 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 <RETURN>).

OPTIONS?: <option(s)>

Choose one or several of the following 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

Description: This command lists the TP-Function(s) of a specific parameter for a phase on screen.

You must supply the name of the phase parameter; for more details, see the command ENTER_PARAMETER.

The parameter name:
<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 name of a parameter consists of several parts. The first is a type-identifier. The following type-identifiers are legal:

- `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 $\omega_{Pr,Tr}$ for aqueous solute species).
- `V0` Molar volume at 298.15 K and 1 bar (a numeric value only);$_T$
- `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.

One may 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 $\omega_{Pr,Tr}$ 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.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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))`

If you press `<RETURN>` when you are asked for a parameter name or if you have improperly input the entire parameter name, you will be asked for each of these items in the name.

Note that for encrypted commercial databases, one may not be able to use this command to list any retrieved parameter.

Syntax: LIST_PARAMETER <IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)

Prompt: `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 the user only presses `<RETURN>`, the following warning message appears:

```
*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY
```

and the program will further prompt 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 the following types of legal identifiers (G or L, or TC, or BM, or V0 or VA or VB or VC or VK). Same as in the ENTER_PARAMETER command.

```
Phase name (/ABCD/): <phase name>
```

Specify the phase name. Same as in the ENTER_PARAMETER command.

```
Constituent (in SUBLATTICE # /abc/): <species name>
```

Specify the constituent name on the specified sublattice site of the given phase. Same as in the ENTER_PARAMETER command.

```
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, just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE # /xyz/): <species name>
```

If there are more than one interacting constituents on the specified sublattice site of the given phase, specify them; otherwise just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
Degree /#/: <degree>
```

Specify a numerical number as the degree for the phase parameter. Same as in the ENTER_PARAMETER command.

LIST_PHASE_DATA

Description: 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.

Note that for encrypted commercial one 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 <PHASE NAME>

Prompt:

```
PHASE NAME: <phase name>
```

Specify a phase name (if abbreviated, it should be unique).

LIST_STATUS

Description: This command lists the entered elements, phases and species with their status word. The command is included mainly for the system managers' benefit.

Syntax: LIST_STATUS

Important Note: Each element, species, phase and symbol has a set of status bits. The values of these bits are listed by the `LIST_STATUS` command. They are normally of no interest to a user, but the meaning is explained here.

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 (*Bit. Meaning):*

1. set if element cannot be deleted (only vacancy and electron)
2. set if suspended (inclusive OR of bit `3` and `4`)
3. set if explicitly suspended
4. set if implicitly suspended (cannot occur for an element)

Species status word (*Bit. Meaning)*

1. set if species record for an element (each element has a species record)
2. set if suspended (inclusive OR of bit 3 and 4)
3. set if explicitly suspended
4. set if implicitly suspended (*e.g.,* if a species element is suspended)
5. set if charged (inclusive OR of bit 6 and 7)
6. set if negative charge
7. set if positive charge
8. set if vacancy
9. set if component (by default the elements are the components)

Phase status word (*Bit. Meaning):*

1. set if ideal (no sublattices and no excess parameters)
2. set if suspended (inclusive OR of bit 3 and 4)
3. set if explicitly suspended
4. set if implicitly suspended (*e.g.,* if all constituents suspended)
5. set if gas phase
6. set if liquid phase
7. set if solution phase
8. set if only one constituent (in each sublattice)
9. set if ions (inclusive OR of bit 10 and 12)

10. set if new ionic liquid model (charge balance handelled internally)
11. set if constituents cannot be added
12. set if external charge balance needed
13. set if aqueous model
14. set if charged species (ions)
15. set if dilute entropy
16. set if last calculation exceeded temperature range for any parameter
17. set if Kapoor-Frohberg-Gaye cell model
18. set if turbo calculation used
19. set if turbo calculation impossible
20. set if turbo calculation illegal
21. set if phase is not ideal
22. set if current site fractions saved
23. set if this phase is the ordered part
24. set if this phase is the disordered part (bit 2 & 3 also set)
25. set if shadow phase with diffusion data
26. set if error in ionic model
27. set sometimes
28. set if CVM-SRO (short-range ordering) entropy expression
29. set if CVM initialization has been made
30. used to test quasi-chemical ordering model
31. set if major constituents check even if no miscibility gap (ordering)
32. set if Hoch-Arpshofen model

Symbol status word *Bit Meaning*

1. set if constant
2. set if variable
3. set if function
4. set if table
5. set if value must not be amended
6. set if undefined

LIST_SYMBOL

Description: This command lists on the screen 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.

Note that for encrypted commercial databases one may not be able to use this command to list any retrieved symbol.

Syntax: LIST_SYMBOL

Prompt: 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). Note that only those symbols that match this name will be listed.

Or press <RETURN> 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 simple textual file can later on be opened and edited by any text editor). The default value is the screen (by pressing <RETURN>).

READ_GES_WORKSPACE

Description: The data area previously-saved onto a *.GES5 file by a SAVE_GES_WORKSPACE command is read back into the GIBBS workspace. Note that these 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.

Syntax: READ_GES_WORKSPACE <FILE NAME>

The user shall give a name in the “File name” box, for which the user can further specify the working directory when he wish to save the file in the “Look in” box. Normally the default file-type in the “Files of type” box is the proper one for the GIBBS workspace format (i.e., *.GES5 file).

REINITIATE

Description: All data in the stored GIBBS workspace are erased, and all variables are initiated to their default values.

Syntax: REINITATE

Prompt: 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 <RETURN>), or the elements with two letter names have the first letter in Upper and the second in lower case (by typing N). Note that in upper case mode all input in lower case will be converted to upper case automatically.

LOWER TEMPERATURE LIMIT /298.15/: <lowest T in K>

This value will be used as the lower temperature limit when data are entered interactively.

UPPER TEMPERATURE LIMIT /6000/: <highest T in K>

This value will be 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 <RETURN> to accept the preset index (1).

The *index* is for defining the symbol printed in listings of parameters. The symbol can be G, H298 or H0:

- G means that the data are referred to Gibbs energy at a variable temperature (also called *Lattice Stability*).
- H298 means that the data are referred to the enthalpy of the element at 298.15 K and 1 bar.
- H0 means the same as H298 but at the temperature 0 K.

Note that the index only changes the symbol, not any value! Normally the index is set correctly by the database. Only when entering data manually, one must 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

Description: 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 command. Note that the output is unformatted and cannot be printed.

Syntax: SAVE_GES_WORKSPACE <FILE NAME>

The user shall give a name in the "File name" box, for which the user can further specify the working directory when he wish to save the file in the "Save in" box. Normally the default file-type in the "Files of type" box is the proper one for the GIBBS workspace format (i.e., *.GES5 file).

SET_R_AND_P_NORM

Description: 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

Prompt:

```
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 the user changes this value, it will only affect the interpretation of energy data given interactively from the terminal, *e.g.,* if the value is set to `1.98717` the user 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 will be divided by this value and then multiplied with the value of one bar expressed in pascal before it is used inside the GIBBS module.

6 PARROT commands

AMEND_PARAMETER

Description: The user can 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 will be made available for interactive editing on the terminal.

This command is identical to the `AMEND_PARAMETER` command in the GIBBS module. Please consult the description of the `ENTER_PARAMETER` command for details on how a parameter is defined or the `AMEND_PARAMETER` command in the GIBBS module, or use this command inside the GIBBS module if you are not sure about the questions.

Syntax: `AMEND_PARAMETER`

Prompt:

```
PARAMETER: <parameter name>
```

Specify a correct parameter name. If a parameter name is not acceptable or the user only presses `<RETURN>`, the following warning message appears: `*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY`

and the program will further prompt 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 the following types of legal identifiers (`G` or `L`, or `TC`, or `BM`, or `V0` or `VA` or `VB` or `VC` or `VK`). Same as in the `ENTER_PARAMETER` command.

```
Phase name (/ABCD/): <phase name>
```

Specify the phase name. Same as in the `ENTER_PARAMETER` command.

```
Constituent (in SUBLATTICE # /abc/): <species
name>
```

Specify the constituent name. Same as in the `ENTER_PARAMETER` command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE #
/xyz/): <species name>
```

Specify the interacting constituent name; if there is no interacting constituent, just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE #
/xyz/): <species name>
```

If there are more than one interacting constituents, specify them; otherwise just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
Degree /#/: <degree>
```

Specify a numerical number as the degree of the phase parameter. Same as in the ENTER_PARAMETER command.

After the parameter name is specified correctly, the program will lists out on screen its current definition (either preset in database or previously 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 the program prompt for changing the parameter definition, as shown below:

```
DO YOU WANT TO CHANGE THE NUMBER OF RANGES
/NO/: <Y or N>
```

If the user wants to change the number of ranges for the chosen function, or change some of the temperature limits in the definition, by typing Y (Yes), he has to retype both the low/high temperature limits and functions (see all the remaining details in the ENTER_PARAMETER command. Sorry for that there is no other possibility to do so.

If the user does not want to change the number of ranges but wishes to change the function(s) in one or more ranges, by pressing <RETURN> to accept the default answer N (No), the whole definition of the chosen parameter in all ranges (if any) will be listed out 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 followings will be prompted:

```
DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y or
N>
```

If there are more than one range, this question will be prompted. But, sorry again for that this option (as intended by answering Y) has not been implemented yet. So you just press <RETURN> here!

```
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 one can press <RETURN> 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 the AMEND_SYMBOL command. This routine prompts the user in the following way:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons, "**:** **:**".

The following commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurences> 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.

Note that when the string is typed the character at the current position has been replaced by an underscore "_". This help can also be obtained on-line by typing "?".

To finish the editing of the current function, one has to type E at the prompt.

```
RANGE NUMBER (0 TO EXIT) /0/: <range number>
```

Give a range number to edit that function, or press <RETURN> or type 0 to exit this command.

AMEND_SYMBOL

Description: This command can be used to interactively calculate the current values of a function or table (predefined by the switched database or previously entered by the ENTER_SYMBOL command) at the current temperature and pressure condition, and also possible to modify the definitions of variables or functions. This command is identical to the AMEND_SYMBOL command in the GIBBS module.

Note that the current values of the temperature and pressure used in calculating the functions or tables are the ones listed in the LIST_STATUS command, and that there is no way to change the current temperature and pressure values interactively.

For modifying 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, their values can be changed.
- For functions, their low and high temperature limits in various ranges (if any), as well their expressions, can be changed.
- For parameters, their low and high temperature limits in various ranges (if any), as well their expressions, can be changed.

Syntax: AMEND_SYMBOL

Prompt: NAME: <symbol name>

Specify the name of a previously entered symbol.

For tables and functions (or parameters which are treated as functions after they were 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 will automatically calculate their values under the current temperature and pressure conditions, and lists out on screen their current values, such as:

```
FUNCTION VALUE  2.52500000E+01
TABLE VALUE  1.56000000E+02
```

For variable and functions (or parameters), there will be some further sequent prompts, which are different depending upon the symbol type to be amended, as shown below:

For a variable:

```
Value /current value/: <new value>
```

For a variable, its current value is displayed, and the user can change it to a new value.

For a function (or an parameter entered as a symbol):

```
DO YOU WANT TO CHANGE THE NUMBER OF RANGES
/NO/: <Y or N>
```

If the user wants to change the number of ranges for the chosen function, or change some of the temperature limits in the definition, by typing Y(YES), he/she HAS TO RETYPE both the low/high temperature limits and functions (see all the remaining details in the ENTER_SYMBOL command. Sorry for that there is no other possibility to do so.

If the user does not want to change it, by pressing <RETURN> to accept the default answer N(NO), the whole definition of the chosen function in all ranges (if any) will be listed out on screen, such as:

```
DIFFERENT FUNCTIONS IN THESE RANGES
298.15<T<2000.00
2000.00<T<4500.00
4500.00<T<6000.00
```

and the followings will be prompted:

```
DO YOU WANT TO CHANGE RANGE LIMITS /NO/: <Y or N>
```

If there are more than one range, this question will be prompted. But, sorry again for that this option (as intended by answering Y) has not been implemented yet. So you just press <RETURN> here!

```
RANGE NUMBER (0 TO EXIT) /0/: <range number>
```

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 one can press <RETURN> 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. This routine prompts the user in the following way:

```
1:+:>
```

The prompt consists of the current position in the string and the character at that position between colons, "**:** **:**".

The following commands can be given:

- Help: ?
- Move CP to last or first character: <+/-> A
- Delete characters from CP: <+-#characters> D
- Exit: E
- Find: <#occurences> 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.

Note that when the string is typed the character at the current position has been replaced by an underscore "_". This help can also be obtained on-line by typing "?". To finish the editing of the current function, one has to type E at the prompt.

RANGE NUMBER (0 TO EXIT) /0/: <range number>

Give a range number to edit that function, or press <RETURN> or type 0 to exit this command.

COMPILE_EXPERIMENTS

Description: 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 has been created (by the CREATE-NEW-STORE-FILE command) or set (by the SET_STORE_FILE command) or opened (by the READ_PARROT_WORKSPACE command) prior to this command.

This command will also list the compilation details on screen or onto a listing file (*.TXT). Note that if a syntax error in the commands is detected, the compilation will be terminated, then the user needs to correct that error in the *.POP file (using a textual editor) and recompile it with the *.PAR work (store) file.

After a successful compilation, the program will transplate all the experiemental data points in the *.POP file into a graphical experiemental data file (*.EXP) in the DATAPLOT syntax, which can be further edited and used as an appendable graphical experiemental data file (*.EXP) for the purposes of imposing onto calculated/plotted phase diagrams and/or property diagrams for comparisons during the assessment process.

Syntax: COMPILE_EXPERIMENTS <*.POP FILE > <SCREEN OR *.TXT FILE> <Y OR N> <*.EXP FILE>

An “Open” window will pop up for opening the original experiemental data file (*.POP). The user shall give an *.POP file name in the “File name” box, for which the user can further specify the working directory when he/shees wish to open the *.POP file in the “Look in” box. Normally the default file-type in the “Files of type” box is the proper one for the experimental data format (i.e., *.POP file). Click the “Open” button to proceed with opening the *.POP file, or the “Cancel” button to adandon this operation.

Prompt: OUTPUT TO SCREEN OR FILE /SCREEN/: <*.POP file name>

A list of the source code in the *.POP file and error messages during compilation will be written on screen (by pressing <RETURN>) or on a list *.TXT file under a specific file name which can later on be opened and edited by any simple 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 the user specifies that the store file should be initiated before compilation (Y), only the compiled block will be stored on the current work file (consequently, the *.PAR file will be updated). If a negative answer (N) is chosen here, the current work file will not be changed.

Then, an "Open" will pop up for generating/storing the graphical experiemental data file (*.EXP) in the DATAPLOT syntax. The user shall give a desired *.EXP file name in the "File name" box, for which the user can further specify the working directory when he/shees wish to open the *.EXP file in the "Look in" box. Normally the default filetype in the "Files of type" box is the proper one for the graphical experimental data format (i.e., *.EXP file). Click the Open button to proceed with generating/storing the *.EXP file, or the Cancel button to adandon this operation.

CONTINUE_OPTIMIZATION

Description: This command continues the optimization using the same Hessian matrix.

Note that in some cases, it is illegal 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 have been changed. The program will give a warning message if the user tries to continue when he may have made such changes.

Syntax: CONTINUE_OPTIMIZATION

Prompt:

```
Are you sure? /N/: <N or Y>
```

Prior this question, the following message will appear on screen to alert the user on if he is sure of a safe continuation (Y) or not (N):

```
It is safe to CONTINUE only after TOO MANY ITERATIONS
and no change in variables and experiments ...
Now anything can happen ...
```

One can always accept the default answer (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 previously specified iteration number is shown as the default number. The PARROT program will exactly try this 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

Description: The user must create a binary file to be used as a "work" file (or called store file) before any optimization can be done. The workspace used by the GIBBS, POLY and PARROT modules will be 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. Important note that this file is hardware dependent, and cannot be read by any text editor.

A work file that has been created at a previous run can be used in the PARROT module by the other command SET-STORE-FILE.

Syntax: CREATE_NEW_STORE_FILE <*.PAR FILE NAME>

A "Save" window as shown in Figure 12-4 will pop up. The user shall give an *.PAR file name in the "File name" box, for which the user can further specify the working directory when he wish to save the *.PAR file in the "Save in" box. Normally the default file-type in the "Files of type" box is the proper one for the PARROT workspace format (i.e., PAR file). Click the Save button to proceed with creating the new work *.PAR file, or the Cancel button to adandon this operation.

EDIT_EXPERIMENTS

Description: By this command, the sub-module for editing experimental equilibria, *i.e.,* the ED_EXP module is initialized and becomes available to the user. 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, the user may calculate each equilibrium separately or all together, provide start values for equilibria which failed to converge, set weights, modify the values of experiments or conditions.

Syntax: EDIT_EXPERIMENTS

ENTER_PARAMETER

Description: This command is identical to the ENTER_PARAMETER command in the GIBBS module. 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 rather 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 referred to as 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. This will be further described in *Section 11.5* about the *Data Structure*. 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.

The user can enter `TP-Function(s)` for a specific parameter for a phase interactively by 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 could be changed later by the command `AMEND_PARAMETER`.

A complete/valid parameter should have the general form of: <identifier>(<phase>, <constituent array>; <digit>) <xxx> <expression> <yyy> <keyword Y or N> <zzz> !

where

- `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; and
- sign "`!`" is used to indicate that the current parameter definition is ended.

The GES parameter name has a general form of:
<identifier>(<phase>,<constituent array>;<digit>)

Examples of GES parameter name:

- `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 `GES parameter name` consists of several parts. The first is a type-identifier. The following type-identifiers are legal:

- `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 $\square_{Pr,Tr}$ 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.

One may 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. The type-identifier `BM` is also used for Born functions $\omega_{Pr,Tr}$ 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.

Specifying the phase name always in uppercase is recommended; however, if one would prefer to write it as a mixture of uppercase and lowercase, it will then automatically convert all lowercase to uppercase, as the GIBBS module only recognizes 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 the `ENTER_PARAMETER` is executed.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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))`

If you press <RETURN> when you are asked for a parameter name or if you have improperly input the entire parameter name, you will be asked for each of these items in the name.

Syntax: ENTER_PARAMETER

Prompt: 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 the user only presses <RETURN>, the following warning message appears:

*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY

and the program will further prompt 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 the following types of legal identifiers:

- G Standard energy parameter (Gibbs energy of formation);
- L Excess energy parameter (Gibbs energy of interaction);
- TC Curie temperature parameter for magnetic ordering;
- BM Bohr magneton number parameter (BMAGN) for magnetic ordering (or Born function $\omega_{Pr,Tr}$ 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.

Note that quantities as H (enthalpy), S (entropy), V (Volume), F (Helmholtz energy), *etc.,* can be calculated from the Gibbs energy when necessary.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> for the same type identifier, or specify a new type.

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.

Specifying the phase name always in uppercase is recommended; however, if one would prefer to write it as a mixture of uppercase and lowercase, it will then automatically convert all lowercases to uppercase, as the GIBBS module only recognizes uppercase phase names. It is important that if a phase bears a legal phasetype (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 the ENTER_PARAMETER is executed.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> for the same phase, or specify a new phase name.

`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. Note that it is the species name, not the stoichiometric formula, that is needed here.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing <RETURN> if the constituent is the same, or specify a new species name.

For phases with several sublattices, the program will ask for one constituent in each sublattice.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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 [$1-\sum y$(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))

`INTERACTING CONSTITUENT (IN SUBLATTICE # /xyz/): <species name>`

An interaction parameter, which is used to describe the excess term of a quantity, must have two or more constituents that "interact each other" on a specified sublattice site of the given phase. It is arbitrary which of these constituents that is given as the first constituent and which are given as the interacting constituents. But the program will always sort the constituents (in each sublattice) in alphabetical order when the parameter name is written as prompt (for entering its parameter value) and when the parameter is listed out (using the GIBBS command `LIST_PARAMETER` or `LIST_PHASE_DATA`). This is very important for all asymmetric interaction parameters where the sign of the interaction parameter must depend on the appearance order.

If this command has been used once or more, the previous value on this prompt will be set as default. Once can accept it by pressing `<RETURN>` if the constituent is the same, or specify a new species name.

Note! To cancel the default value of the interacting constituent you must type `NONE` or the name of another constituent.

This question will be repeated till all the interested interacting constituent(s) on a specific sublattice in the phase have been specified, and finally a simple `<RETURN>` has been enforced.

`Degree /#/: <degree>`

Specify an integer number (a value from `0` through `9`) as the degree of compositiondependent interaction contribution for the phase parameter. This prompt is only valid 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`).

The meaning of the degree is model-dependent. 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.

This question may not be prompted if this command is used for entering standard `G` parameter for a pure component (end-member), as its degree should be always `0`.

After this prompt, the program will echo on the screen the full *TP-Function* of the desired 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. Same as for entering functions.

```
Function: <definition for a function>
```

A TP-Function consists of terms in T and P. Same as for entering functions.

```
& <continuation of the definition for the
current function>
```

Continuation of a TP-Function definition. Same as for entering functions.

```
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 has been entered as a negative number) for the current TPFunction. Same as for entering functions.

```
Any more ranges /N/: <Y or N>
```

Specify a Y (YES) for more function(s) or N (NO) for ending this command. Same as for entering functions.

LIST_ALL_VARIABLES

Description: This command will produce, on screen or on a text file, a list of the values and status of all variables in the PARROT workspace. After an optimization some statistical information are also written. The correlation matrix will be written if that option has been chosen, see the SET_OUTPUT_LEVELS command.

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 will be written on screen (by pressing <RETURN>) or on the list file under a specific file name which can later on be opened and edited by simple 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
 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

Description: By this command the present values of the output and optimizing conditions will be listed on screen or a simple text file which can be opened and edited by a text editor later on.

Syntax: LIST_CONDITIONS

Prompt: `OUTPUT TO SCREEN OR FILE /SCREEN/: <file name>`

A list of the current values of optimization conditions, as well the current status of listing, in the `PARROT` workspace will be written on screen (by pressing `<RETURN>`) or on the list file under a specific file name which can later on be opened and edited by simple text editor.

LIST_PARAMETER

Description: This command lists the `TP-function(s)` for a specific parameter for a phase on screen. You must supply the name of the phase parameter; for more details, see the command `ENTER_PARAMETER`. This command is identical to the `LIST_PARAMETER` command in the GIBBS module.

The parameter name is:
<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 name of a parameter consists of several parts. The first is a type-identifier. The following type-identifiers are legal:

- `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 $\omega_{Pr,Tr}$ 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.

One may 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. The type-identifier `BM` is also used for Born functions $\omega_{Pr,Tr}$ 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.

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 in the following prompt). Note that solution phases with sublattices may have interacting constituents in each sublattice.

One may 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))

If you press <RETURN> when you are asked for a parameter name or if you have improperly input the entire parameter name, you will be asked for each of these items in the name.

Note that for encrypted commercial databases one may not be able to use this command to list any retrieved parameter.

Syntax: LIST_PARAMETER <IDENTIFIER>(<PHASE>,<CONSTITUENT ARRAY>;<DIGIT>)

Prompt: 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 the user only presses <RETURN>, the following warning message appears:

*** ERROR, PLEASE RE-ENTER EACH PART SEPARATELY

The program will further prompt 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 the following types of legal identifiers (G or L, or TC, or BM, or V0 or VA or VB or VC or VK). Same as in the ENTER_PARAMETER command.

Phase name (/ABCD/): <phase name>

Specify the phase name. Same as in the ENTER_PARAMETER command.

Constituent (in SUBLATTICE # /abc/): <species name>

Specify the constituent name on the specified sublattice site of the given phase. Same as in the ENTER_PARAMETER command.

```
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, just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
INTERACTING CONSTITUENT (IN SUBLATTICE  #
/xyz/): <species name>
```

If there are more than one interacting constituents on the specified sublattice site of the given phase, specify them; otherwise just press <RETURN>. Same as in the ENTER_PARAMETER command.

```
Degree /#/: <degree>
```

Specify a numerical number as the degree for the phase parameter. Same as in the

ENTER_PARAMETER command.

LIST_PHASE_DATA (from GIBBS)

Description: All data for a specific phase are written in a readable manner 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

Description: By this command, the result of the current optimization run will be listed on screen or a specified file. How much details will be listed by this command can be prescribed by the SET_OUTPUT_LEVELS command.

The LIST_RESULT command has a 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 the user 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.

The output normally consists of the following parts:

- A title showing the data of the action;
- A message on successful optimization with the iteration number in the last optimization;
- 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_LEVEL 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) predefined for each phase in the system (optionally shown, pre-set by the SET_OUTPUT_LEVEL 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 has been 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 have been 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).

Syntax: LIST_RESULT

Prompt:
```
FULL, CONDENSED, DETAILED OR GRAPHIC FORMAT:
/C/: <C or D or F or G>
```

Specify a desired format (C for Condensed, or D for Detailed, or F for Full, or G for Graphical).

- The C (condensed) format is the default one.
- The D (detailed) format lists all experiments, including those with fulfilled inequalities which are suppressed with the default option C.
- The F (full) format is obsolete but the question has been retained for backward compatibility.
- 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 the user can scale it giving commands in the POST module.

```
OUTPUT TO SCREEN OR FILE /SCREEN/: <file name>
```

A list of the current optimization results will be written on screen (by pressing <RETURN>) or on the list file under a specific file name which can later on be opened and edited by simple text editor.

Example output: The following is the listed result in the example TCEX36, 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.  DATE 2000. 8.17   17:10: 3
==================================================
 *** 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
  1     R         80000000  8.3145100E+00
  2     RTLNP     20000000 +R*T*LN(1E-05*P)
  3     V1        48000000  2.0374946E+04
  4     V2        48000000 -2.9428637E+01
   13   V11       48000000 -2.1739567E+04
   14   V12       48000000  1.5210718E+01
   17   V15       48000000  2.4210656E+04
   18   V16       48000000 -8.3872397E+00
   19   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)
  1     Alternate equilibrium calculation                  0.4183
  2     Alternate equilibrium calculation                  0.1932
  3     Alternate equilibrium calculation                  0.1016
  4     Alternate equilibrium calculation                 1.4354E-
 03
  5     Alternate equilibrium calculation                 2.5063E-
 02
  6     Alternate equilibrium calculation                 8.3929E-
 03
  10    Alternate equilibrium calculation                   141.4
  11    Alternate equilibrium calculation                   141.4
  12    Alternate equilibrium calculation                   141.4
  13    Alternate equilibrium calculation                   141.4
  20    Alternate equilibrium calculation                 1.2532E-
 03
  21    Alternate equilibrium calculation                 5.4781E-
 04
  22    Alternate equilibrium calculation                 1.5404E-
 03
  23    Alternate equilibrium calculation                 1.2702E-
 03
   100 ACR(B)=9.4E-1               0.9397      2.85E-02 -2.7745E-
   04 -9.7472E-03
   101 ACR(B)=8.4E-1               0.8395      2.82E-02 -4.9038E-
   04 -1.7396E-02
   102 ACR(B)=7.4E-1               0.7407      2.81E-02  7.3804E-
   04  2.6305E-02
   103 ACR(B)=6.4E-1               0.6424      2.81E-02  2.3935E-
   03  8.5272E-02
   104 ACR(B)=5.4E-1               0.5434      2.82E-02  3.4449E-
   03  0.1220
   105 ACR(B)=4.4E-1               0.4428      2.85E-02  2.8265E-
   03  9.9024E-02
   106 ACR(B)=3.4E-1               0.3394      2.90E-02 -5.8174E-
   04 -2.0040E-02
   107 ACR(B)=2.3E-1               0.2320      2.97E-02  2.0260E-
   03  6.8208E-02
   108 ACR(B)=1.2E-1               0.1194      3.06E-02 -6.4192E-
   04 -2.0981E-02
   110 HMR(LIQUID)=-1964            -1957.      5.00E+02   7.439
   1.4878E-02
   111 HMR(LIQUID)=-3500            -3478.      5.00E+02   21.67
   4.3338E-02
   112 HMR(LIQUID)=-4588            -4565.      5.00E+02   22.69
   4.5382E-02
   113 HMR(LIQUID)=-5239            -5217.      5.00E+02   21.50
   4.3008E-02
   114 HMR(LIQUID)=-5454            -5435.      5.00E+02   19.11
   3.8216E-02
   115 HMR(LIQUID)=-5233            -5217.      5.00E+02   15.50
   3.1008E-02
   116 HMR(LIQUID)=-4575            -4565.      5.00E+02   9.691
   1.9382E-02
```

```
117 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

Description: This command writes on screen the name of the store file and its full path.

Syntax: LIST_STORE_FILE

LIST_SYMBOL_IN_GES

Description: This command 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 provides a way to find out how the functions depend on the optimizing variables. It is nearly identical to the `LIST_SYMBOL` command in the GIBBS module, but in PARROT module the list will only be shown on screen, not written to any file.

Syntax: LIST_SYMBOL_IN_GES

Prompt: `NAME: <symbol name>`

Specify a symbol name. Note that only those symbols that matches this name will be listed. Or press `<RETURN>` for a list of all available symbols entered in the system.

OPTIMIZE_VARIABLES

Description: This command will perform an optimization of variables. All system-definition data needed for the optimization will be read from the current work file (`*.PAR`). The result of the optimization is automatically stored onto 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 previously specified iteration number is shown as the default number. The PARROT program will try this number of different sets of values of the optimizing variables unless it has converged earlier, or has given up earlier. Note that it will initially take a small step in each variable to find the steepest slope. One may give zero iteration just to calculate the error in all selected experiments, and then use the `LIST_RESULT` command to check how good (bad) are the current fit.

READ_PARROT_WORKSPACES

Description: If the user is not happy with the late changes made through various PARROT/GIBBS/POLY commands, he/she can use this command to read the previous PARROT/GIBBS/POLY3 workspaces back to replace the current PARROT/GIBBS/POLY3 workspaces. The previous PARROT/GIBBS/POLY3 workspaces are always associated with the latest action to either open a work file by the SET_STORE_FILE command, or to created a work file by the CREATE_NEW_STORE_FILE command (either interactively in the PARROT module or through a MACRO_FILE_OPEN of an *SETUP.TCM file), or to update the work file by the SAVE_PARROT_WORKSPACE command.

Unlike the READ commands in other modules (e.g., GIBBS or PLOY), this command in the PARROT module will not ask for the file name where to read a previously opened/created/updated PARROT/GIBBS/POLY3 workspaces, for the reason described above. One cannot use this command if there is no work file opened or created previously.

Syntax: READ_PARROT_WORKSPACE

RECOVER_VARIABLES

Description: By this command the values of all variables will be set back to their start values.

Syntax: RECOVER_VARIABLES

REINITIATE

Description: By this command the workspace used by the PARROT program will be reinitiated. All output and optimizing conditions will be given their default values. All variables are set fixed with their value equal to zero. Note that this command should never be used, unless you really want to destroy the current `PARROT` workspace. However, this PARROT command will not reinitiate the GIBBS/POLY3 workspaces.

Syntax: REINITIATE

RESCALE_VARIABLES

Description: By this command the current values of all the parameters will be copied to their start values and the scaling factors for further optimization. Thus it should be done now and again, in particular if the user thinks the optimization results have been improved after the previous run(s), or if any variable has changed more than a factor of `10`.

Syntax: RESCALE_VARIABLES

SAVE_PARROT_WORKSPACES

Description: If the user is happy with the late changes made through various PARROT/GIBBS/POLY commands, he/she can 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 the SET_STORE_FILE command or been created by the CREATE_NEW_STORE_FILE command (either interactively in the PARROT module or through a MACRO_FILE_OPEN of an *SETUP.TCM file). Note that the PARROT/GIBBS/POLY3 workspaces are updated after each PARROT/GIBBS/POLY command, and by this command the current PARROT/GIBBS/POLY3 workspaces will be 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 will not ask for the file name where to save the current PARROT/GIBBS/POLY3 workspaces, for the reason described above. One cannot use this command if there is no work *.PAR file opened or created previously.

Syntax: `SAVE_PARROT_WORKSPACE`

SET_ALTERNATE_MODE

Description: This command can turn on or off the alternate mode. The alternate mode is described in detail elsewhere. 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 answer `Y` will turn on the alternate mode. Anything else will turn it off.

SET_EXTERNAL_PARAMETER

Description: With this command, the user can additionally 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.

Syntax: SET_EXTERNAL_PARAMETER

Prompt: `External name: <parameter name in the external model>`

Correctly specify the parameter name defined within the user-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
```

Notes: In order to be able to build the user-specified external model and to use SET_EXTERNAL_PARAMETER command, it requires the user to first write his/her own 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 will be 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 has been bilaterally agreed and signed in advance between the user and Thermo-Calc Software Company.

SET_FIX_VARIABLE

Description: Using this command, the user can prescribe a fixed value to a variable. The variable will be considered as a constant at the optimization. Normally, one can do so after some 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 the `SET_OPTIMIZING_VARIABLE` command.

Syntax: SET_FIX_VARIABLE

Prompt: `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.,* `2-8`. The parameters within the limits will be 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 has been specified at the previous prompt). The current value for the selected variable will be shown as default.

If a range of variable numbers has been specified at the previous prompt, this question will not be prompted, and their current values will be used as fixed values.

SET_OPTIMIZING_CONDITION

Description: By this command, the user can specify the conditions for the optimization. The default values have been chosen with some cares automatically by the PARROT program, and they should not be changed unless the user perfectly knows what he is doing.

Syntax: SET_OPTIMIZING_CONDITION

Prompt: `RELATIVE STANDARD DEVIATION FOR EXPERIMENTS? /N/: <Y or N>`

The user can choose whether the standard deviation of the experimental determinations are absolute values (N) or if they are used as relative weighting factors (Y). The estimated standard deviations of the optimized variables might be different for the two cases.

`MIN SAVE ON FILE? /Y/: <Y or N>`

The user can specify whether the program should minimize the transfer of data to and from the current work file during the optimization:

If yes (Y), it will speed up the optimization. After optimization one can use the SAVE_PARROT_WORKSPACE command to update the progress onto the current work file.

If minimum saving on file is not specified (N), the workspaces will be 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. Just press <RETURN> here.

`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.

The user can 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.

`DMAX (VA05AD): /100/: <n>`

An estimate of the maximum distance between the start and the final values of the variables. A smaller value will make 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. Note that the optimization stops when the sum of errors decreased by this value, even though it might be possible to increase it further with a new `OPTIMIZE` command.

SET_OPTIMIZING_VARIABLE

Description: By this command, the user can specify which variable value should be estimated at the optimization. Such a variable may have a value of zero or another start value, or may have been fixed by the `SET_FIX_VARIABLE` command prior to the previous optimization run.

For a good reference prior to proceeding this command, a list of all variables (which have been used in defining various parameters for phases in the optimizing system, as pre-entered in the `*SETUP.TCM` file or interactively entered in the GIBBS module) can be obtained by using the `LIST_ALL_VARIABLE` command.

Syntax: SET_OPTIMIZING_VARIABLE

Prompt: `VARIABLE NUMBER: <variable number(s)>`

Specify the number(s) for the variable(s) to be set. Any non-zero parameter within the range will be allowed to be optimized. A parameter with a current value equal to zero must be specified explicitly here in order to be optimized.

One may 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 will be asked.

`START VALUE /xxxx.xxxxx/: <a guess value>`

Specify a start guess of the optimum value (only when one variable number has been specified at the previous prompt). The current numerical value for the selected variable will be shown as default.

Note that this guess is very critical as the initial guess of all parameters must make it possible to calculate the selected equilibria.

If a range of variable numbers has been specified at the previous prompt, this question will not be prompted, and their current values (as start values) will not be changed.

SET_OUTPUT_LEVELS

Description: By this command, the user can choose what type of information the PARROT module should give during the optimization procedure and when the result is listed.

Syntax: SET_OUTPUT_LEVELS

Prompt: `LIST INCREMENT /1/: <increment in iteration number>`

Specify the increment in iteration number for which information will be listed out on screen during the optimization. By giving a larger number it will shorten the list.

`LIST SUM OF SQUARES: /Y/: <Y or N>`

The user can choose whether the sum of squares will be listed on screen during the optimization procedure (`Y`) or not (`N`).

`LIST SCALED VARIABLES: /Y/: <Y or N>`

The user can choose whether the scaled variable values will be listed on the terminal during the optimization procedure (`Y`) or not (`N`).

`LIST WEIGHTENED RESIDUALS: /N/: <N or Y>`

The user can choose whether the wheightened residuals will be listed on the terminal during the optimization procedure (`Y`) or not (`N`).

`LIST ALL PARAMETERS: /N/: <N or Y>`

The user can choose whether all parameters in the models will be listed at the `LIST_RESULT` command (`Y`) or not (`N`).

`LIST CORRELATION MATRIX: /N/: <N or Y>`

The user can choose whether the correlation matrix of the variables will be listed at the `LIST_RESULT` and `LIST_ALL_VARIABLES` commands (`Y`) or not (`N`).

SET_SCALED_VARIABLE

Description: This command is similar to the `SET_OPTIMIZING_VARIABLE` command by specifying start values for optimizing variables, but it is functional for only one optimizing variable at one time. Furthermore, it also prescribes a minimum and maximum value for the variable. During the following optimization runs, the variable value will be limited within this min-max range. Improvements have been made since TCCS, so that SET_SCALED_VARIABLE command now works correctly for specifying lower and upper limits for optimizing variables.

Such a variable may have a value of zero or another start value, or may have been fixed by the `SET_FIX_VARIABLE` command prior to the previous optimization run. For a good reference prior to proceeding this command, a list of all variables (which have been used in defining various parameters for phases in the optimizing system, as preentered in the `*SETUP.TCM` file or interactively entered in the GIBBS module) can be obtained by using the `LIST_ALL_VARIABLE` command.

Syntax: SET_SCALED_VARIABLE

Prompt: `VARIABLE NUMBER: <variable number(s)>`

Specify the number for an optimizing variable to be set. Any non-zero parameter will be 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 will be shown as default.

Note that this guess is very 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

Description: The user can use this command to specify a store file (work file) to be used for compilation and optimization. The `*.PAR` work file must have been created (by the `CREATE-NEWSTORE-FILE` command) 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 <*.PAR FILE NAME>

The user shall give an *.PAR file name in the "File name" box, for which the user can further specify the working directory when he/she wishes to open the *.PAR work file in the "Look in" box. Normally the default file-type in the "Files of type" box is the proper one for the PARROT workspace format (i.e., *.PAR file). Click the "Open" button to proceed with setting the *.PAR work file, or the "Cancel" button to adandon this operation.

7 POLY commands

ADD_INITIAL_EQUILIBRIUM

Description: This command is used to add initial equilibrium points from which a phase diagram is calculated (through the MAP command). Normally, this command is not needed for calculating a property diagram (through the STEP NORMAL command-sequence). In many cases, the ADD_INITIAL_EQUILBRIUM command is not required for MAP commands either, as the mapping procedure will then start from the current equilibrium state that has already been calculated. In order to calculate a simple phase diagram, the user may just 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 will start 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 will handle all the initial equilibrium points in a very robust and automatic way. Therefore, a user does not need to have any "good" guess of starting point and to add any initial equilibrium point prior to the calculations of various types of phase diagrams. However, if the user prefers to use this command to add any initial equilibrium points in certain directions, the POLY module will use the user-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 re-written MAP routines will not be 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.

Syntax: ADD_INITIAL_EQUILBRIUM <DIRECTION CODE>

Prompt: Direction /default/: <direction code>

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 will search for a line in the diagram (*i.e.,* a line where the amount of a phase is zero, imply that if just starts to become stable on one side of the line and disappear on the other side) in the given direction.

Options: `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.

Notes: Since TCCM, the default direction is treated in a new way. The `ADD_INITIAL_EQUILBRIUM` command with the default direction will scan along the axis variables and generate start points each time the scanning procedure crosses a phase boundary. In addition, it will generate 4 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 will be 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 will subsequently stop 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 will generate 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 will ensure successful finding of 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 by issuing the `LIST_INITIAL_EQUILBRIUM` command. A certain initial equilibrium point (including its conditions and equilibrium results) can be loaded into the current equilibrium, if needed, by the `LOAD_INITIAL_EQUILBRIUM` command. Furthermore, any specific or all of the initial equilibrium points can be deleted from current POLY workspace, if desired, by the `DELETE_INITIAL_EQUILBRIUM` command.

ADVANCED_OPTIONS

Description: At present moment, the following 14 advanced options can be set:

```
 EQUILIBRIUM_CALCUL       OUTPUT_FILE_FOR_SHOW
STABILITY_CHECK
 GLOBAL_MINIMIZATION      PARAEQUILIBRIUM
STEP_AND_MAP
 LIST_PHASE_ADDITION      PHASE_ADDITION           T-
ZERO TEMPERATURE
 MAJOR_CONSTITUENTS       PRESENT_PHASE
TOGGLE_ALTERNATE_MODE  NEW_COMPOSITION_SET
SHOW_FOR_T=
```

Note that four options previously under SPECIAL_OPTIONS are now renamed under ADVANCED_OPTIONS (mainly for the purpose of removing ambiguousness when specifying the first few letters and thus tying only fewer letters is faster and easier):

- "SET_MISCIBILITY_GAP" is replaced by "NEW_COMPOSITION_SET";
- "SET_MAJOR_CONSTITUENTS" is replaced by "MAJOR_CONSTITUENTS";
- "SET_PRESENT_PHASE" is replaced by "PRESENT_PHASE"; and
- "SET_PHASE_ADDITION" is replaced by "PHASE_ADDITION".

Furthermore, the following five options that were previously available under the SPECIAL_OPTIONS command have been removed, because they are no longer necessary or relevant:

- "MISC_GAP_TEST_INTERVAL", "SET_BREAK_CONDITION",
- "OUTPUT_AT_MAP_AND_STEP",
- "NEVER_ADJUST_MINIMUM_Y" "SHOW_OUTPUT_FILE"

Syntax: ADVANCED_OPTIONS

Prompt: Which option? /STEP_AND_MAP/: <Option>

Options: Choose one of the 14 advanced options or NONE (for not any advanced option). For some options, some questions will be further asked.

EQUILIBRIUM_CALCUL

The SET_MINIMIZATION_OPTIONS command that was only available in TCCR/TCW4 has been split and replaced by the advanced options EQUILIBRIUM_CALCUL and GLOBAL_MINIMIZATION.

This advanced option EQUILIBRIUM_CALCUL is designed to decide how to perform the so-called Ordinary POLY Minimization (*i.e.,* the traditional Gibbs Energy Minimization) in the POLY module. By default, the Global Minimization Technique for various singlepoint equilibrium calculations and for stepping or mapping calculations is used. The user can use the advanced option GLOBAL_MINIMIZATION 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, one 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.

This advanced option EQUILIBRIUM_CALCUL will prompt the following syntaxes (and default options) which will affect the ways on 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>
```

The user can choose to answer Y (Yes) or N (No), to determine the way on 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 then the local maxima. Furthermore, if an eigenvalue (for a phase) of the Hessian matrix is near 0.0, the step-size is very large and the /Y/ option will set a lower limit to this eigenvalue, and thus will reduce the step-size and improve the convergence. The POLY command-sequence SHOW_VALUE QF(phase) shows the size of the lowest eigenvalue of a phase in an equilibrium state.

By answering Y (Yes), it might help the Ordinary POLY Minimization to converge, if there is such problems, especially if QF(phase)=0 for phases.

By answering N (No), the Ordinary POLY Minimization routine will be 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:

$$G_m(X) = G_m(T, P; y_1, y_2, \ldots, y_n)$$

We may compute a Hessian matrix as which describes the curvature of the Gibbs energy curve of this phase at the defined composition X:

$$\partial^2 G_m / \partial y_i \partial y_j$$

Diagonalize this matrix and call the Eigenvalues as $e_1, e_2, \ldots, e_n$. The QF(phase) quantity for this phase is the lowest of these eigenvalues divided by the same values for a corresponding ideal (stoichiometric) phase:

$$QF(ph) = \min(\{e_1, e_2, \ldots, e_n\}) / \min(\{se_1, se_2, \ldots, se_n\})$$

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>`

The user can choose to answer Y (Yes) or N (No), to determine whether (by answering Y) or not (by answering N) to introduce a control of step-size while reaching the minimum of an equilibrium state in a normal POLY optimization procedure. By answering Y (Yes), it might help the POLY optimization to converge, especially if when some site fractions are less than 1E-4.

GLOBAL_MINIMIZATION

The SET_MINIMIZATION_OPTIONS command that was only available in TCCR/TCW4 has been split and replaced by the advanced options EQUILIBRIUM_CALCUL and GLOBAL_MINIMIZATION.

The advanced option GLOBAL_MINIMIZATION is 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. The user can use this advanced option 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, one 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.

This advanced option GLOBAL_MINIMIZATION will prompt the following syntaxes (and default options) which will affect the manners on 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>`

The user may choose to answer Y (Yes) or N (No), 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.

By answering Y (Yes), the Global Minimization calculation is done when possible (depending on the condition settings), and Global Minimization test is always performed when an equilibrium is reached. This costs more computer time but the calculations are more accurate.

By answering N (No), the Global Minimization calculation is not used to reach the equilibrium state. Please note that the following question "Use global minimization for test only?" will determine on if the Global Minimization test will be made against the calculated equilibrium states obtained by the Ordinary POLY Minimization calculation.

```
Use global minimization for test only? /N/: <Y or N>
```

By answering Y (Yes), 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 will be an error message.

By answering N (No), 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 will be permanently turned off, and that consequently only the Ordinary POLY Minimization routine will be used.

```
Maximum number of gridpoints per phase /2000/: <integer number>
```

Specify the maximum number of grid points that will be 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.

STEP_AND_MAP

The advanced option STEP_AND_MAP is used for determining on how to perform Global Minimization test and how to handle initial equilibrium points.

This advanced option STEP_AND_MAP will prompt the following syntaxes (and default options):

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 user-supplied 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 10th 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 will perform Global Minimization test at every n-th step during STEP and MAP calculations. Of course, the Global Minimization test will always be carried out at a phase change or a node 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 the ADD_INITIAL_EQUILIBRIUM command 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 instance, if set to 2 (two), initial equilibrium points will be added at the beginning, center, and end of each axis-variable forming 2 intervals.

`Use inside meshing points /N/: <Y or N>`

If answering N (No), the INITIAL_EQUILIBRIUM mesh consists of initial equilibrium points added only along the edge/border of the diagram defined by the axis-variables. If answering Y (Yes), the initial equilibrium points added are also added inside edge.

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. The command is a subcommand of the ADVANCED_OPTIONS command. Normally the database creates as many composition sets as is necessary but the user can 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 becomes unnecessary if the Global Minimization Technique is in use, thanks to that 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 the user has his own data file, this advanced option must be used to indicate that a solution phase may 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 for amending composition-set) or been manipulated through the GIBBS module (using the AMEND_PHASE_DESCRIPTION COMPOSITION_SET command-sequence). This advanced option will ask 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.

The advanced option NEW_COMPOSITION_SET asks the following three questions for the user to specify the additional composition set(s) for handling 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 will usually be one higher than the current value. Each phase has initially one composition set. If a lower value is given, composition sets will be deleted. One cannot take away the first composition set.

The following message appears to show that the user is requested to specify the composition for the composition set # (2,3,...): “Give for composition set #”.

```
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 should have different major constituents for each composition set.

This question will be 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

This gives an option 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 option.

The advanced option MAJOR_CONSTITUENT asks the following three questions for the user 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 (#) will usually be /1/ as the other composition sets will be 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 will be repeated for each sublattice in the phase.

PHASE_ADDITION

In some cases it may be interesting to add a constant contribution to the Gibbs energy of a phase (stoichiometric or solution). This can, of course always, be done also 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. Two advanced options related to this aspect had been added to the ADVANCED_OPTIONS command, *i.e.,* PHASE_ADDITION and LIST_PHASE_ADDITION.

The user may 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.

The advanced option PHASE_ADDITION gives the following two prompts for the user 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 will be added to the Gibbs energy of the (stoichiometric or solution) phase. It can represent a nucleation barrier, surface tension, elastic energy or whatsoever. Note that it is not composition-, temperature- or pressure-dependent.

LIST_PHASE_ADDITION

The values set as additional contributions (given by the advanced option 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.

PRESENT_PHASE

The phase specified with this option must be stable at all equilibria calculated during a MAP command. It is a convenient 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 will be mapped.

The advanced option PRESENT_PHASE just asks the user to specify the name of the

```
PRESENT phase:
```

```
Phase name: <name of a phase>
```

Specify the name of the phase that should be present at all calculated equilibria.

T-ZERO TEMPERATURE

This advanced option calculates the temperature when two specific phases have the same Gibbs energy, *i.e.,* the so-called T_0 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.

```
Name of first phase:  <phase A>
```

```
Name of second phase: <phase B>
```

At these two prompts, the names of phases A and B must be given, for which the T_0 temperature (where their Gibbs energies are equal) is to be calculated.

If the T_0 temperature between the two specified phases has been successfully calculated, the following two messages will appear on screen, *e.g.,*

The T0 temperature is 840.82 K

Note: LIST-EQUILIBRIUM is not relevant

The first message shows the calculated T_0 temperature between the two specified phases.

The second one indicates that after this advanced-option calculation the LIST_EQUILIBRIUM command is irrelevant and does not list the equilibrium for the system at the T_0-temperature.

PARAEQUILIBRIUM This advanced option 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 by the SET_AXIS_VARIABLE command.

To ensure a successful point calculation of paraequilibrium state between two specific phases in a defined alloy system, it is very important that you must 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 instance 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.

It is recommended to keep the following three important issues in mind:

- 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 SPEICAL using the following POLY command:

```
CHANGE_STATUS COMPONENT
<interstitial component> = SPECIAL
```

By doing this, you will get a clear picture on u-fractions of various substitutional and interstitial components, which are of course different from the overall composition in the system. The SPECIAL status means that specified component(s) will not be included in summations for mole or mass fractions. Therefore, all the composition variables plotted from paraequilibrium calculations are u-fraction related quantities.

```
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 typed at the above two prompts subsequently or on the same (first) line at once (separated by an empty space, *e.g.,* “FCC#1 BCC” or “FCC#2 M23C6”).

Please note that you must have completely understood what you are dealing with here in terms of calculating a paraequilibrium state between the two specified phases. Specifically, there are four distinguished cases which need to pay a great attention: (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 simply impossible to calculating 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 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.

Please pay attentions on the following aspects:

Such specified interstitial component(s) must be appropriately defined according to the phase constitution definitions of the two selected phases: it/they 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 <RETURN> 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 just pressing <RETURN> 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 has been successfully calculated, the following messages will appear on screen, *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 they 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.

However, if the single-point calculation of the paraequilibrium state between the two specified phases has failed, the following messages will appear on screen:

```
*** 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 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 so-called “spinodal curve” or simply “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 so-called “spinodal decomposition” (i.e., decomposing to a mixture of regions with the two stable compositions, one on each side of the miscibility gap).

Since TCCP, this advanced 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 thus find out if any phase is subject to spinodal decomposition during the subsequent calculations. If there is an unstable phase locates inside a miscibility gap in a calculation, it gives a warning so that the user shall suspend the unstable phase, or use FORCED automatic start values for phase constituents (*i.e.,* SET_ALL_START_VALUE FORCE command-sequence), or create other composition sets; the user could also simply ignore the warning message if he/she knows that the unstable phase will not be formed anyhow in the current calculations.

```
Stability check on? /Y/: <Y or N>
```

The user may answer Y (Yes) or N (No). 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 will be no stability check in various calculations.

```
Check also for unstable phases? /Y/: <Y or N>
```

If the automatic stability-check option has been switched on (in the last prompt), the user can choose here on if he/she wishes to also check the stability for unstable phases. The default is Y (Yes); consequently, 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 will inform the user to selectively make some necessary 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 (No), the stability check will be enforced only to stable phases in the system.

TOGGLE_ALTERNATE_MODE

With this advanced option, the toggle of ALTERNATE mode for experimental equilibrium calculation can be set to DEFAULT, ALWAYS and NEVER, during data-assessments using the PARROT optimization. The DEFAULT means that the experiment points will be calculated according to the so-called Alternate Technique depending on the SET_ALTERNATE command. If the toggle is set to ALWAYS the experiment points will always be calculated according to the Alternate Technique even if the Alternate Mode is switched off in the PARROT module. If the toggle is set to NEVER the experiment points will be calculated as normal equilibria even if one has SET_ALTERNATE in the PARROT module.

Note that single-phase experiments will never be calculated with the ALTERNATE mode.

SHOW_FOR_T= This advanced option makes it possible to show (on screen) 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 feature is particularly useful for knowing *e.g.* 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 has been changed.

Note that this advanced option must be used with care! You must have successfully calculated a real equilibrium state under one temperature condition (normally the temperature for the last heat treatment). This advanced option 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.

```
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 various thermodynamic properties of the entire system, of components, or of phases) in the currently-defined system (being in a "frozen" state) will be 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 instance, one 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).

NONE

This means to take away any previously-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, the following messages will appear 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 means that smallest site fraction in the current POLY3 workspace has been 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 to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

OUTPUT FILE FOR SHOW

With this advanced option, the name of a text file will be asked for and all the results output from the command SHOW_VALUE will be written to this file.

AMEND_STORED_EQUILIBRIA

Description: This command gives information about the calculated blocks (and phase regions included in blocks) after the `STEP` or `MAP` calculation(s). It allows the user to list all or parts of the calculation results, to suspend all or parts of the calculation results that are redundant or where metastable equilibria have been 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.

One may use one the following options:

- `L` for listing the calculated equilibria (all or a specified block)
- `S` for suspending everything (all blocks and their regions)
- `Q` for suspending each set of equilibria individually (specified blocks and/or regions)
- `R` for restoring everything (all blocks and their regions)

Syntax: AMEND_STORED_EQUILIBRIA

Prompt: `Name: <name of a defined table>`

Options: L(ist) S(uspend) Q(uery suspend) R(estore) /L/: <option>

One may just 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`. Alternatively, one can include all the blocks in the amending option, by accepting the wildcard "`*`" (pressing `<RETURN>`).

In case of the option 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 have been calculated during stepping or mapping will be listed out on screen or in a textual file (see later).

In case of the option Q, each ranger in a block (if a block number is specified) or in all blocks (if the wildcard "*" is accepted/used) will be shown up on screen and the user will be asked if suspending or keeping it (see later):

```
S(uspend) K(eep) /K/: <S or K>
```

In case of the option S or R, if a block is specified, the user will be asked on which region in the block shall be suspended or retored; however, if the wildcard "*" is accepted/used, the user will be asked the following question (and needs to answer Y or N):

```
Really suspend all /N/: <Y or N>
```

```
Really restore all /N/: <Y or N>
```

If the answer to the above question is N, the program will sequentially ask 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>
```

Confirm your choice: Y for suspending everything, or N for not suspending everything

```
Really restore all /N/: <Y or N>
```

Confirm your choice: Y for restoring everything, or N for not restoring everything

```
Range: <range(s) of region>
```

The user may specify one or more ranges to be suspended (the S option) or restored (the R option). By typing the wildcard "*" to suspend or restore 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) will ask 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>
```

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 <RETURN>. A list of stored equilibria as various blocks (with all their ranges) will be shown out on screen or the file.

CHANGE_STATUS

Description: This command 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 usage of this command 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 means that the component(s) or species are included in the calculation. This is the default status.
- SUSPENDED means that the component(s) or species are not considered in the calculation.
- SPECIAL means that specified component(s) will not be included in summations for mole or mass fractions. It only works for component(s).

Note that only component(s) may have the status SPECIAL, which implies that they will not be included in summations for mole or mass fractions. For instance, for the so-called “ufractions” or other normalized fractions, when one or more of the components are excluded from the summation, one 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.

For example, to obtain the metallic fraction in a system with carbon as an interstitial component, one can set the component status for carbon as SPECIAL:

```
Change_status comp C=special
```

Please also note that the SUSPENDED status for components and species will not always work, as there are bugs in the program!

For a phase, it may have one of four statues:

- ENTERED means that the phase(s) are included in the equilibrium calculations and they will be stable if that minimizes the total Gibbs energy in the defined system. This is the default status for all phases alreadyretrieved 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 means that the phase(s) are not considered in the equilibrium calculations.
- DORMANT means that the phase(s) are not considered in the equilibrium calculations but their driving forces for precipitation are calculated.
- FIXED means that 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.

Syntax: CHANGE_STATUS <KEYWORD> <NAME(S)>=<STATUS> <VALUE, OPTIONAL>

Prompt: `For phases, species or components? /PHASES/: <keyword>`

keyword = phase or species or components

`Phase name(s): <name(s) of the phase(s)>`

In case of "phase" as the keyword, the names of the phases that shall have their status changes must be given (all on one line). A comma or space must be used as separator. The status to be assigned to the phases can also be given on the same line if preceded with an equal sign "=". Note that an asterisk, "*", can be used to denote all phases. The special notations "*S", *i.e.,* a widecard * directly followed by a "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)>`

In case of "species" or "component" as the keyword, the names of the species or components that shall 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 "=". Note that an asterisk, "*", can be used to denote all species or components. The special notations "*S", *i.e.,* a widecard * directly followed by a "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 will be excluded from sums for mole fractions and mass fractions, which is useful when calculating the so-called "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 will be calculated. FIXED means that it is a condition that the phase is stable at a certain amount.

For instance, for the so-called "u" fractions, when one or more of the components are excluded from the summation, one 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 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, one can 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, one should instead use a properly-entered symbol (function or variable), for instance: "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 instance, 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 modeled by certain models, and their NA values must be evaluated in different ways, as described below:

  LIQUID (C,Cr,$CrO_{3/2}$,Fe,FeO,$FEO_{3/2}$,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 has been modeled by the Two-Sublattice Ionic Liquid Model, i.e:

  IONIC_LIQ (Cr^{+3},Fe^{+2},Ni^{+2})$_p$(VA,C,N,O^{-2},$FEO_{3/2}$)$_q$,7

then the evaluation of its NA value becomes even more complicated:

$$NA = p + q^{*}y_{C2} + q^{*}y_{N2} + q^{*}y_{O2\square 2} + q^{*}y_{FeO2\,3/2}$$

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 multiplesublattice model with ionic constituents, such as `SPINEL` and `HALITE` phases in some databases.

- There will be no strange thing when using a zero value [*i.e.,* `0`] in a `FIXED` phase-status, since it simply 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 just 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 has been 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, one 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 very 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

$$N/[p+q^{*}y_{C}^{2}+q^{*}y_{N}^{2}+q^{*}y_{O}^{2}{}_{\square 2}+q^{*}y_{FeO}{}^{2}{}_{3/2}].$$

COMPUTE_EQUILIBRIUM

Description: 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 subsequential single-point calculations or stepping/mapping calculations within the current TCC run) if having decisively switched it off by the user (or possibly in some special modules) through changing the minimization option using the ADVANCED_OPTIONS MINIMIZATION_OPTION command-sequence.

Syntax: COMPUTE_EQUILIBRIUM

Notes: The "C_E –" command-combination 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

Technique has been 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 (we call it "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 (we call it "Indirect Global Minimization").

If there is any problem with convergence, you may try the powerful "C_E *" command-combination. The character * will enforce the command to use an advanced technique to obtain a complex equilibrium. However, it is worth notice that after a successful "C_E *" calculation, you may repeat the "C_E" command and can check the status of phases, species or components (by the LIST_STATUS CPS command-sequence) and equilibrium conditions (by the LIST_CONDITION command) and list out the calculation results (by the LIST_EQUILIBRIUM command), because such actions may tell you how to further modify various settings for your current calculation. This command-combination is not that useful any more, 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. One may force the program to correctly calculate the driving forces of metastable phases, by simply giving repeated C_E commands until the number of iterations (that is shown on screen after this command) is reduced to 2.

See also the POLY command SET_NUMERICAL_LIMIT which can set the "Approximate driving force for metastable phases" option *on* (as Y) or *off* (as N) in all the subsequent POLY calculations within the current TCC run.

If an equilibrium state for the defined system has not been found, an error message is given on screen. In such a case, you may try to repeat this command a few times, or to 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

Description: This command is actually a combination of the CHANGE_STATUS, SET_CONDITION and COMPUTE_EQUILIBRIUM commands, so 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 has already been calculated successfully; otherwise, the user will be 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 will call the `CHANGE_STATUS` command to temporarily change the phase status of a user-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 the user). The program will calculate a new equilibrium in which that specific phase is stable but its equilibrium amount in the system is zero. The released condition will then be assigned with a calculated value that ensures the calculated equilibrium. Afterwards, the program will automatically change the phase status of that specific phase back to `ENTERED`, and reset the temporarily released condition as one of the conditions and assign it with the value that has been calculated to ensure the zero-amount formation of that specific phase.

This command is particularly convenient to find melting temperature, boiling temperature, or solubility limits, and generally useful when the user wants to set the most optimal conditions for calculating an equilibrium where a specific phase will become stable. It can also be used when the user wishes to know exactly how far is one of the defined conditions away 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, the user may issue a `COMPUTE_EQUILIBIRUM` calculation to assure the calculated transitional equilibrium is a really stable one, and can also use the `LIST_EQUILIBRIUM` command to see the details of transitional equilibrium state.

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 one wishes, and thus this new feature is very useful for finding out all possible phase transformations along a certain released condition.

Syntax: COMPUTE_TRANSITION

Prompt: Phase to form: <phase name>

A new phase name, *e.g.,* BCC that is expected to form, must be specified here. This will change the status of this new phase to be FIXED as 0 amount, and the program will show the information such as below:

```
You must remove one of the these conditions
P=100000, T=800, N=1, X(FE)=.5    DEGREE OF FREEDOM 0
```

It is also possible to find out any new phase to be formed in a certain direction of the released condition if the key word ANY is given as the phase name when it is prompted for "Phase to form:". In such a case, the above informative message will also be shown on screen.

```
Give the state variable to be removed /T/: <one
condition>
```

One condition as listed in the above message must be removed, in order to be able 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 following message may come up (after a successful calculation) if having typed, *e.g.,* X(Fe):

```
To form BCC the condition is set to
X(FE)=.48605791769
```

This calculated value will then be assign as the parameter of that removed condition, in this case, the X(FE) variable. So the following message will be shown on the screen, if the LIST_CONDITIONS command is typed:

```
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 following line will be further 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 it is only estimated value, so any value within its reasonable scale would be enough). For instance, if a combination of -0.02 is input, the following message may come up (after a successful calculation):

```
To form BCC_A2#1 the condition is set to
X(FE)=.493708756187
```

This calculated value will then be assign as the parameter of that removed condition, in this case, the `X(FE)` variable. So the following message will be shown on the screen, 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

Description: During data-assessments using the PARROT/ED_EXP modules, one may, 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. If the user wishes to keep the set of conditions and phase for this equilibrium, he may 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`

Prompt:

```
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 by the `SELECT_EQUILIBRIUM` command later on.

```
Initiation code /2/:
```

When an equilibrium is created, one may 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. Note that the entered components and phases can lately be changed with the `CHANGE_STATUS` command.

DEFINE_COMPONENTS

Description: The set of components can be changed by this command. By default, the elements are used as components. The set of components may be important because some conditions can only be 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`.

Note that this command implies a `REINITIATE` command and thus it should be given as the first command in the POLY module.

Syntax: DEFINE_COMPONENTS <ALL NEW COMPONENTS>

Prompt: `Give all new components /existing components/: <new components>`

The `new components` must be specified all on one line. These will replace the `existing component` definitions. Note that the number of components cannot be changed with this command. To change the number of components, use the `CHANGE_STATUS` command.

If you want to keep some existing components, then 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.

Note that the new components need not already be present as species.

DEFINE_DIAGRAM

Description: This command is an extension of the `DEFINE_MATERIAL` command and has been available since TCCM. It allows automatic calculation and plotting of a diagram with a single command. It is the exactly same as the `DEFINE_MATERIAL` command up to the point when the first equilibrium has been calculated. Therefore, the "`alloy OPTION`" feature is also available in this command for specifying alloying compositions for a special alloy predefined by the `OPTION` keyword in a selected database (*e.g.,* the TCNI Ni-based Superalloys Database).

One can use this command to calculate all types of phase diagrams after specifying all composition value and an initial temperature (if temperature will be used as an axis). However, for binary and ternary diagrams, one may prefer the special BIN and TERN modules.

This command will then list all the independent variables for the defined system (*i.e.,* temperature and the components) and asks for a variable as the X-axis. The user 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 will then calculate and plot a *Phase Diagram,* as there are two independent quantities on the axes.

Alternatively, the user may 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 will calculate and plot how this quantity depend on the condition on the X-axis. This is called a *Property Diagram*.

This command will end up within the POST module monitor. One can refine the calculated phase diagram or property diagram as desired.

Moreover, many more property diagrams with axes *other than compositions* can also be plotted (after the calculation), using the normal `SET_AXIS_VARIABLE` command in the sequent POST monitor.

Syntax: `DEFINE_DIAGRAM`

Prompt: The following prompts will be presented:

```
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 °C>
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 will be calculated here, as by the DEFINE_MATERIAL command. 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 will be asked only when the user has already defined some axis variables, or has previously used the command DEFINE_DIAGRAM. 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>
```

Select one of the independent conditions by giving its index on the condition list as the Xaxis 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 <RETURN> 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 <RETURN> 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 simply giving its corresponding index given on the condition lists. Note that it must be different from the X-axis variable that has already been selected.

If selecting one of dependent quantities (by simply 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. In case of the “composition of a phase” is selected, the phase name will be 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 will be 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 will be stored (saved as a desired `*.POLY3` file) should be specified; the default file name is `RESULT.POLY3` (under MS-Windows) or `RESULT.poly3` (under Linux).

DEFINE_MATERIAL

Description: This command makes it possible to 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 will read the system from the specified database, set the composition and temperature (and pressure equal to 1 bar) and calculate the equilibrium state before the user is prompted for a new command. The user may then list the results by the `LIST_EQUILIBRIUM` command or set a new composition or set axis for a `STEP` or `MAP` command.

Previously, data were always retrieved from the DATA module and this is still the recommended way. But the growing number of "occasional" Thermo-Calc (Console Mode) users makes it necessary to provide simpler ways through the program, in addition to the various Windows interface developed in parallel. Note that one cannot "append" data from different databases in this way. One may also use this command with data from a `USER` database.

Syntax: DEFINE_MATERIAL

Prompt: `Same elements as before /Y/?  <Y or N>`

This question will be asked only as the user has already read some data from the database, or has previously used the command `DEFINE_MATERIAL` or `DEFINE_DIAGRAM`. It then offers a convenient way to change the composition and temperature with one command.

Note that the `DEFINE_MATERIAL` command only works properly in case that the composition of the material system has already been defined as in the mole-percent or mass-percent unit.

`Mole percent of <element> /##/: <value>` *or*

`Mass percent of <element> /##/: <value>`

If the user has already decided to use the same materials system (available in the current POLY3 workspace) by accepting the default answer (Y) to the previous prompt "Same elements as before /Y/?", one of the two alternative prompts given above will appear for each of the components in the defined system, depending on how the composition has previously been defined (either in mole-percent, or in mass-percent).

Such prompts will be repeated until all the defined components have been gone through. Then, the program will prompt for specifying the temperature condition.

Database /ABCDE/: <database name>

The database with the description for the material must be given, or just press <RETURN> if using the current database. It is possible to give a USER database.

Major element or alloy: <element name>

The material must have a "major" element, usually the element which is present in the largest amount. The fraction of this element will not be set but be "the rest".

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 the user stays within these limits the calculation should give reasonable results.

Composition in mass (weight) percent? /Y/: <Y *or* N>

The default is that input will be taken as mass percent, but it is possible to change to mole percent by answering N (No) to this question.Note that composition should be given in PERCENT not FRACTION, as it is required for the W and X state variables in the SET_CONDITION command.

1st alloying element: <element name>

The first alloying element must be given.

All alloying elements will be asked for in a sequence. They can be given in any order. The user must know if they are present as assessed systems in the database. There is no error or warning messages if data are missing. Please check in the documentation of the database selected.

If an alloy is selected, a list of legal alloying elements and their maximum percent will be 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 one cannot use the normal flexibility of Thermo-Calc for conditions, but all must be given in mass percent. However, one may afterwards change the conditions using the SET_CONDITION command.

```
2nd alloying element: <element name>
```

The second alloying element must be given. If only one, just press <RETURN>. If an element name is given then the program will ask 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>
```

The user may go on giving elements and mass (weight) fractions until all elements specified. When all alloying elements and their compositions (as in the above prompt) have been specified, just press <RETURN> as answer to this question to finish the materials definition.

```
Temperature (C) /1000/: <Temperature of
interest in °C>
```

POLY will make the first calculation after retrieving the data for this temperature. By pressing <RETURN> to accept the default temperature. The value should be given in Celsius (°C).

Note that in this command the pressure will be set to 1 bar.

```
Reject phase(s) /NONE/: <list of phase(s) to be
rejected>
```

This is a question generated by the database allowing the user to select the phases.Normally, all phases should be included and the user just presses <RETURN>. 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 just those that should be included. The question will be repeated until the user press <RETURN> after rejected all undesired phases or an asterisk "*".

```
Restore phase(s) /NONE/: <list of phase(s) to
be restored>
```

The user may restore phases that were 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 will be repeated until the user presses <RETURN> after restoring all desired phases.

```
OK? /Y/: <Y or N>
```

All phases to be selected from the database are listed and the user must confirm the selection. If he has made some errors or wants to amend the selection, he can answer N (No) and will then be back at the question about rejecting phase(s).

If the selection is confirmed by answering Y (Yes), the software will retrieve 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 the user should specify this here. It will cost some computation time and may make subsequent MAP or STEP more difficult to converge.

If the user does not want to have any such phase with miscibility gap in the calculation, he shall just press <RETURN>. Then, the DEFINE_MATERIAL command will start calculating the equilibrium, and be terminated.

If the user wants to set such a phase with miscibility gap in the calculation, he shall answer Y (Yes). Then the software will ask some questions about the phase names and their constitutions, like below:

```
Phase with miscibility gap: <phase name>
```

The user must supply the phase name, which will have a miscibility gap under the specified system and conditions

```
Major constituent(s) for sublattice #: /AA/:
<constituent(s)>
```

The software will show up a default constituent in the sublattice # (1,2,3,...), according to the existing phase definition in the chosen database. The user may specify one or more major constituents for the sublattice # in the phase.

This question will be repeated until all sublattices have been specified.

```
Phase with miscibility gap: <phase name>
```

The user may 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 `<RETURN>`, the `DEFINE_MATERIAL` command will start calculating the equilibrium, and then be terminated.

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 have been 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 an "`?`" mark on the prompt "`Major element or alloy:`". When a specific predefined alloy (instead of a major element) is selected, the major element will be staked from the alloy definition and shown on the screen (with a message like "`Alloy found with major element NI`").

The user is only allowed to specify alloying elements and their compositions (weight percent or mole percent). Typing an "`?`" mark at any of the prompts for the alloying element names, *e.g.,* "`1st alloying element:`", "`2nd alloying element:`", will list all the alloying elements and their composition limits in the alloy. If the composition of an alloying element is outside of its limit, there will be a warning message (like "Amount above limit: 30.0000") and a prompt "Override limit ? /N/:". If the user decide on enforcing the overriding by answering Y (Yes) on this prompt (i.e., accepting the over-limit alloying composition), another warning message (like "Amount of major element below limit: 70.0000") and prompt "Override limit ? /N/:". Then the user can further decide on if enforcing the overriding: if Y (Yes) then accepting the major element's composition below the limit; if N (No) then using the predefined major element composition limit.

The intention of the "alloy OPTION" feature is to provide a safety net for inexperienced users. At present, there is no check inside the Thermo-Calc software that the selected set of composition conditions is within the applicable composition limits of a database. Experienced users can still use the old command without this safety feature.

DELETE_INITIAL_EQUILIBRIUM

Description: This command deletes 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; see also the ADD_INITIAL_EQUILIBRIUM command.

Syntax: DELETE_INITIAL_EQUILIBRIUM <NUMBER OF AN 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 the LIST_INITIAL_EQUILIBRIA command to figure out the numbers for all the existing initial equilibrium points that have already been added (*manually* through the ADD_INITIAL_EQUILIBRIUM command, 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

Description: Symbols (*i.e.,* constants, variables, functions or tables, which are entered by the command ENTER_SYMBOL) can be deleted by this command.

Syntax: DELETE_SYMBOL <NAME OF A 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

Description: Symbols are a very useful feature of the POLY module to define quantities that are convenient for the user. Symbols can be defined with values of constants, variables, functions or tables.

Within the POLY module, symbols are normally defined prior to an equilibrium calculation (enforced by a C_E command), stepping calculation (enforced by a STEP_WITH_OPTIONS command) or mapping calculation (enforced by a MAP command). They can also be entered after an equilibrium calculation; however, for defined functions, variable or tables, it requires using the EVALUATE_FUNCTIONS command before showing their corresponding values in the calculated equilibrium state. Functions or tables (with defined functions as values) can even be entered in the POST module after a stepping or mapping calculation, for purposes of plotting such entered functions or tables as axis variables; for details, see *Chapter 9 – PostProcessor Module (POST).*

Syntax: ENTER_SYMBOL <KEYWORD> <NAME>=<VALUE, EXPRESSION OR VARIABLES>

Prompt:

```
Constant, variable, function or table?
/FUNCTION/: <keyword>
```

The `keyword` can be constant, variable, function or table.

Constants can only be entered once and is just 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 will only be evaluated when they are entered or if they are explicitly named in an EVALUATE command. It is possible to enter a variable with a new expression anytime. This expression will be 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.Note that there is a special connection between tables and variables. If a variable is used in a table, it will be 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 "\", period ".", are illegal for symbol names.

If one wishes to enter the symbol name and the value or function on the same line, they must be separated with an equal sign "=", for instance, `TC=T-273.15` or `T_C=T273.15` which stand for a definition of temperature in Celsius. Otherwise, the following questions will be proceeded; note that for different types of symbol (constant, function, variable or table) which is under definition here, the questions will be prompted differently.

```
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 simply pressing <RETURN> at the next prompt).

Examples of function epressions:

```
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
```

The temperature in Celsius

```
T-273.15
<continuation of the definition for the symbol>
```

The ampersand "&" means that one may continue to write the function on the new line if one line is not enough for the function. If one has finished the function, then just press <RETRUN> 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 will be 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 function will be 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 calculation is through a table. Example:

```
ENTER TABLE
K=T,X(LIQ,C),X(LIQ,CR),ACR(C)
```

Which means that the table called K will contain four columns, *i.e.,* the temperature, the mole fractions of C and Cr in the LIQUID phase, and the activity of C.

If one wishes to have the temperature in Celsius in a table, one must first 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 one may continue to write the table on the new line if one line is not enough for the table. If one has finished the table just press <RETURN> again.

EVALUATE_FUNCTIONS

Description: The value of one or more or all entered functions or variables are evaluated and listed. Note that variables are evaluated only if they are explicitly named.

Syntax: EVALUATE_FUNCTIONS <NAME(S) OF DEFINED FUNCTION(S)>

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.

LIST_AXIS_VARIABLE

Description: All the axis variables for a stepping or mapping calculation that have already been set by the SET_AXIS_VARIABLE command are listed on screen by this command.

Syntax: LIST_AXIS_VARIABLE

All the axis variables that have been set in the current POLY3 workspace will be listed on the screen.

LIST_CONDITIONS

Description: 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 on the screen. 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, one may perform a COMPUTE_EQUILIBRIUM command. If it is larger than zero, some more conditions are required, and the user must further set additional ones, using the SET_CONDITION or CHANGE_STATUS command. If it is negative, the 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 output: Here is an example of the kind of output you will receive.

```
P=100000, T=800, N(NI)=1E-1, N=1
FIXED PHASES
FCC_A1=1 LIQUID=0   DEGREE OF FREEDOM 0
```

LIST_EQUILIBRIUM

Description: The result *(always in SI units)* from the last calculated equilibrium is listed on screen or in a textual file. Note that one may also execute this command if no calculation has been made or if the calculation failed. It is the user's responsibility to interpret the result accordingly.

Syntax: LIST_EQUILIBRIUM <RETURN OR FILE NAME> <OPTION(S)>

Prompt: `OUTPUT TO SCREEN OR FILE /SCREEN/: <file name>`

The name of the textual file where the list of the calculation results shall be written. Default is `SCREEN` (only shown on the terminal).

`Options /VWCS/: <option(s)>`

The user may select the output units and formats by optionally specifying a combination of the following 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 just simply 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, one can simply type "L_E,,," or the following line:

```
LIST_EQUILIBRIUM ,,,
```

LIST_INITIAL_EQUILIBRIA

Description: This command lists all the equilibria added with the `ADD_INITIAL_EQUILIBRIUM` command. All the initial equilibria are used for the `MAP` (and `STEP`) calculations.

Syntax: LIST_INITIAL_EQUILIBRIA

LIST_STATUS

Description: The status of components, species or phases can be listed with this command. The user may select all or some of these.

Syntax: LIST_STATUS <KEYWORD(S)>

Prompt: Specify what you want 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.

Default is `CPS`. By pressing `<RETURN>`, a complete list with status for components, phases and species is obtained. By just giving the option `P`, a list of just the phase statuses is obtained. If you are also interested in component status, then type `C`. You may also simply input `CS` so that a list of statuses for both components and species can be listed out.

The statuses of components, phases and species can be changed with the `CHANGE_STATUS` command (see above).

Results: Depending upon the key word specified in the CHANGE_STATUS options, a table with the current statuses of phases or species or components, or their combinations, is shown up:

- For components, their statuses and reference states are listed.
- For `ENTERED` and `FIXED` phases, their statuses, driving forces and equilibrium amounts are listed. Note that the metastable phases are listed in descending order of stability. To avoid long outputs only 10 metastable phases (in `ENTERED` status) will be 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.
- For species, only the statuses are listed out.

LIST_SYMBOLS

Description: For all constants, functions, variables and tables, their definitions can be listed by this command. The defined variables will be listed up together with the defined functions, but variable names will be followed by a percentage sign “%”.

In order to find the value of a function or variable, use the `SHOW` or `EVALUATE` commands. A table is tabulated with the `TABULATE` command.

Syntax: LIST_SYMBOLS

LOAD_INITIAL_EQUILIBRIUM

Description: This command 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 <NUMBER OF AN 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 the LIST_INITIAL_EQUILIBRIA command.

MAP

Description: This `MAP` 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. Note that a phase diagram consists of mapped phaseboundary lines/curves; on one side of each such phase-boundary line/curve, the amount of one specific stable phase is zero (*i.e.,* the so-called "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 `MAP` command.

Normally, before it can be evoked, one must have already calculated at least one initial equilibrium point and have also defined at least two indenpendt varying variables (*i.e.,* the controlling conditions in the system) that have been set by the `SET_AXIS_VARIABLE` command. As a unique feature of the Thermo-Calc software, one can also have three, or four or maximum five independent varying variables that have been defined by the `SET_AXIS_VARIABLE` command).

The Thermo-Calc Console Mode will list the current values of each of the indenpendt axis variables for each of the calculated equilibrium points along each of the mapped phaseboundaris, and also list the corresponding information when the set of stable phases changes.

The Global Minimization Technique is by default enforced in this command ("`MAP`"), while it can be disabled if having decisively switched it off by the user (or possibly in some special modules) through changing the minimization option using the `ADVANCED_OPTIONS MINIMIZATION_OPTION` command-sequence. The user can additionally choose how often to do a Global Minimization test (via the "`Global Test Interval`" option) in the `ADVANCED_OPTIONS STEP_AND_MAP` command-sequence.

Syntax: MAP

Notes:

During a MAP calculation, the values of mapped axis-variables for presenting each phaseboundary (lines/points) of the calculated phase-regions will be listed, and the corresponding stable-phase sets will be shown up. Since TCCN, the lengthy output of calculated equilibria had be removed by default; moreover, the special option "SHOW_OUTPUT_FILE" or "OUTPUT_AT_MAP_AND_STEP" that was enforced by the previously-available SPECIAL_OPTION command for setting back a long listing has been removed since TCCS.

The user may terminate the mapping of a line by pressing a single CTRL-A (on MS-Windows) or CTRL-C (on Linux). This can be useful in order to stop a longish calculation without loosing 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 the MAP command, the following messages will appear 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 has been 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 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 have now been a special MAP procedure which checks for the best phase to use as axis variables in order to ensure reasonable increments between the tie-lines. This procedure produces smoother curves and also gives a better stability in finding adjacent phase-regions.

Complex miscibility gaps of various solution phases will be automatically detected during a mapping calculation, and two or more composition sets for each of such solution phases will be 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); in earlier versions up to TCCR, that was done by the `MISC_GAP_TEST_INTERVAL` option of the `SPECIAL_OPTIONS` command. *As a particular case,* such an automatic `Global Minimization Test` ensures that one 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; previously, such calculations had to be made from low temperatures. In other words, 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 the `ADD_INITIAL_EQUILIBRIUM` command) inside various phase-region boundaries under certain direction(s).

POST

Description: This command switches to the post-processor, the POST module, which has its own command repertoire.

Syntax: POST

Comments: After a STEP or MAP calculation, through the POLY module (as well as via TAB module, and any special module *e.g.* BIN, TERN, POT, SCHEIL and POURBAIX) in the Thermo-Calc (and DICTRA) software system, the calculation/simulation results should be handled in the POST module, to generate various *graphical* presentations [on screen, and into desired files with various types of graphical formats (*e.g.,* PS, EMF, PNG, BMP, PDF, JPG, TIF, among many others) or data forms (*e.g.,* EXP, TAB and WRL)] of phase diagrams, property diagrams, diffusion profiles and many other types of plots a user would expect, as well as to create some *tabulated* forms (on screen, or into simple textual files (TXT) or spreadsheet).

Within the POST module, one can choose any state variable, any derived variable, or any entered symbol (functions or variable) as the X/Y-axis. When a diagram has been plotted, varied appearance parameters for defining a high-standard graph can be further specified, *e.g.,* curvelabeling 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 labeling on phase boundaries and phase regions, graphic formats, text fonts, colors, raster plot, *etc*. A user can easily append experimental data onto a plotted diagram. One 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, one can translate a plotted property diagram (after a steeping calculation) into a tabulated form, such as a simple list on screen or a textual file, and a spreadsheet (*e.g.,* an MS Excel file with an extension of .XLS).

READ_WORKSPACES

Description: The POLY3 and GIBBS workspaces and the calculated results from the MAP and STEP commands can be READ from a file where they must have been saved previously with a SAVE_WORKSPACES command. Such an *.POLY3 file is not printable.

Syntax: READ_WORKSPACES <FILE NAME>

Options: file name -- The name of a previously saved POLY3-file where the POLY3 and GIBBS workspaces shall be read from must be specified. The user does not need to type the extension if it is the default "*.POLY3" (under MS-Windows) or "*.poly3" (under Linux), otherwise the user must type the whole POLY-file name.

An "Open" window will pop up on the screen, so that the path (in the Look in box) and file name (in the File name box) can be appropriately specified, as shown in Figure 8-6. The file type (i.e., POLY3, as in the "Files of type" box) cannot be changed. By pressing the "Open" button, the program proceeds with opening the POLY3 and GIBBS workspaces from the specified, previously-saved *.POLY3 file. The user may also "Cancel" such an "Open" window session, and thus no previously-saved POLY3 and GIBBS workspaces will be opened.

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 previously been read from an existing POLY3 file under the current work area, while some additional changes in the settings may have been made but do not need to be kept in further steps in the current ThermoClac (Console Mode) run or any diagram has been plotted in the POST module, one can simply type "`READ,,`" or the following line:

```
READ_WORKSPACE ,,
```

RECOVER_START_VALUES

Description: This command can recover the start values in equilibrium calculations.

Syntax: RECOVER_START_VALUES

REINITIATE_MODULE

Description: The `REINITIATE_MODULE` command makes sure of that the whole POLY module (and thus the entire POLY workspace) is reinitiated to the state as it was when first entered. 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.

Syntax: REINITIATE_MODULE

RESET_CONDITION

Description: This command does not exist (!), but a condition is reset by using the `SET_CONDITION` command again with a new value. A set condition is removed with the value `NONE`.

Syntax: SET_CONDITION <CONDITION>=<NEW VALUE>

Examples: The following will reset the condition `T` (temperature) as 1673.15 K:

```
SET_CONDITION T=1673.15
```

This will remove the condition on `T` (temperature):

```
SET_COND T=NONE
```

Notes: You can remove all conditions by giving the command as below:

```
SET_CONDITION *=none
```

SAVE_WORKSPACES

Description: Thermo-Calc has a unique feature that 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. A user can do so for later uses or when he has to terminate the current Thermo-Calc (Console Mode) run for any reason.

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, the user can always come back to exactly the state he had when he issued the SAVE_WORKSPACE command by simply giving a READ_WORKSPACE command.

After saving the POLY3 and GIBBS workspaces on a file, one may leave the program and at a later time READ the file and continue from the saved state. Note that a STEP or MAP command automatically saves on the work file with the most lately specified name. Do not SAVE after a MAP or STEP!

The results from the MAP or STEP commands are destroyed by the SAVE_WORKSPACE command. You may append several results obtained by sequential MAP or STEP calculations without destroying the previous results, whilst SAVE_WORKSPACE will erase them all. Keeping this in mind is very important and useful particularly for calculating various isothermal (or isoplethal) sections and plotting them on the same diagram in a single ThermoCalc run.

You may append several results by MAP or STEP without destroying the previous results but the SAVE command will erase them all. To suspend some of the MAP or STEP results, use the AMEND_STORED_EQUILIBRIA command.

Syntax: SAVE_WORKSPACES <FILE NAME> <Y OR N>

Options: file name -- A user-desired file name must be specified. The default extension of the POLY workspace file is "*.POLY3" (under MS-Windows) or "*.poly3" (under Linux), while the user can have any other extension as wished.

A "Save" window will pop up on the screen, so that the path (in the "Save in" box) and file name (in the "File name" box) can be appropriately specified. The file type (*i.e.,* POLY3, as in the "Files of type" box) cannot be changed. By pressing the "Save" button, the program proceeds with saving the POLY3 and GIBBS workspace in the specified *.POLY3 file.

Notes: When saving a POLY3 workspace under a name that already exists under the current work area, which has ben saved by default (after running a special module, *e.g.,* BIN, TERN, POT, SCHEIL and POURABIX) or in an earlier stage of the current Thermo-Calc (Console Mode) run or in a previous run (which has been READ into the current POLY3 workspace), one can simply type "SAVE,,y" or the following line:

```
SAVE_WORKSPACE ,,y
```

However, this must also be avoided if some results from previous MAP or STEP calculations shall not be destroyed.

SELECT_EQUILIBRIUM

Description: If the user has created more than one initial equilibrium (during data-assessments using the PARROT/ED_EXP modules), he can switch between them using this command.

Syntax: SELECT_EQUILIBRIUM <CHOICE ON EQUILIBRIUM>

Prompt:

```
Number /NEXT/: <choice on equilibrium>
```

One may answer FIRST, LAST, NEXT, PREVIOUS or PRESENT.

Most commands affect only the PRESENT equilibrium. However, the REINITIATE and DEFINE_COMPONENT commands will remove all the stored equilibria.

SET_ALL_START_VALUES

Description: This command is unnecessary unless one wants to supply 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 will be 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

Prompt:

```
T /XXXX/: <temperature in K>
```

If the temperature is not a condition, the user must supply a guess of its final value (in K).

```
P /100000/: <pressure in pa>
```

If the pressure is not a condition the user must supply a guess of its final value (in pa).

```
Automatic start values for phase constitutions?
/N/: <Y, N or F>
```

The user may answer N (No), Y (Yes) or F (for FORCE). The default is N (No).

The reason for the F option is that in some cases the calculation has failed because the user has set impossible conditions, *e.g.,* W(C)=1.5 [W(C) is mass fraction and must thus be less than unity]. The program tries to fulfill this condition by putting maximum amount of carbon in all phases, but will fail anyway. When the user detects the error and set W(C) to 0.015, the calculation may still fail because it could start from the previous values. To get back to "fresh" start values, one gives the answer F for FORCE.

If the user answers Y (Yes), this command will immediately terminate, and the program will automatically set the start values for phase constitutions in all possible phases.

If the user accepts N (No) by pressing <RETRUN>, he will be asked to supply an initial amount of each entered phase and their major constituents or site fraction of each constituent.

```
Should <phase> be stable /N/: <Y/1 or N/2>
```

A guess on if this phase should be stable or not is requested. All entered phases will be prompted for this question and next two. Note that 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 phase name may have a hash sign "#" followed by a digit, *e.g.,* BCC_A2#2. For phases with miscibility gaps, there should be two phases with the same name but with different numbers after the hash sign.

```
Major constituent(s): <name of major
constituent(s) in the phase>
```

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 will be asked for individual fractions in the phase.

```
Y(<phase>,<constituent>) /.XXXXXXXXXX/:
<.YYYYY>
```

The current value (.XXXXXXXXXX) is default. The user may accept the default by pressing <RETURN> 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 will also be suffixed with a hash sign "#" followed by a digit.

SET_AXIS_VARIABLE

Description: In order to calculate a diagram, one must 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 will, 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 very powerful feature of the Thermo-Calc software, up to 5 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.

One may also give the `SET_AXIS_VARIABLE` command without having previously set a condition on the axis variable. Under such a circumstance, the relevant condition will be automatically created and the value set between the minimum and maximum axis limits; however, be aware of that, as a side effect, the POLY module will create two conditions, `P=1e5` and `N=1` (in they have not been defined as condition yet), in case that the user sets an axis variable which is not already a condition.

One may 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. One specifies the logarithmic axis by giving a `*` after the increment value. Note that the increment in this case is treated as a factor. For example,

```
S-A-V 1 P 1E5 1E25 5*
```

will make axis 1 a logarithmic axis where the difference between two calculated values will max be a factor 5. Note that the factor must be larger than 1.0.

Note that in some cases, such as when the DEFINE_DIAGRAM command has been used or a special advanced module (*e.g.,* BIN, TERN, POT, SCHEIL or POURBAIX) has been called, some axis variables would have been automatically set by the program, not necessarily by this command.

Syntax: SET_AXIS_VARIABLE <AXIS NUMBER> <CONDITION> <MIN> <MAX> <LENGTH>

Prompt: `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.

`Condition /NONE/: <one condition>`

Here the condition that should be varied along the axis must be given. The condition is specified as in the SET_CONDITION command, 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.

SET_CONDITION

Description: This is the main command to 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 a very-much mixed types of conditions; this brings the extreme flexibility and powerfulness to the complex calculations with the Thermo-Calc software system.One 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 (through the CHANGE_STATUS command), it is equivalent to one condition (implying that particular phase is stable in the defined system).

Syntax: SET_CONDITION <CONDITION(S)>

Each condition must be given explicitly, but can be given on the same line (separated by a comma sign " , " 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 user sets the temperature 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.

This question is rather cryptic but the user is expected to give either just a state variable.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)

There are many more state variables can be used in conditions. For more information, give an `INFO STATE_VARIABLES` command. A condition is normally just a value of a single state variable with its value. For instance,

```
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. Note that 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 is even more cryptic, and it means that the user 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 will be prompted if a single state variable name has not given in either the prompt "`State variable expression`" or "`Factor`", or a state variable expression has been 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. The user needs to specify here a state variable or a complete state variable expression, or complete the unfinished state variable expression. If a numeric factor has been given before this prompt, only one state variable can be specified; otherwise, the program will only take 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 instance T=NONE will take away the temperature condition.

SET_INPUT_AMOUNTS

Description: This command can be used to specify how a system is made up from mixing of various substances. It is most useful with a substance database. In the C-H-O-N system, one may, for example, give:

```
S-I-A N(H2)=10, N(H2O)=25,
N(C1O2)=5, N(N2)=100
```

The POLY module will automatically and appropriately convert 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 with

```
SET-CONDITION N(H)=70, N(O)=35,
N(C)=5, N(N)=200
```

Note that one may also give negative amounts in the SET-INPUT-AMOUNTS command!

Syntax: SET_INPUT_AMOUNTS <N(<SPECIE>) OR B(<SPECIE>)>=<VALUE>

Prompt: `Quantity: <N(<specie>) or B(<specie>)>`

Here you should give `N(<specie>)` or `B(<specie>)`. You may give the amount also preceded with an equal sign "=" [*e.g.,* `N(H2)=10` or `B(H2O)=1000`], or just press <RETURN> for the next prompt on the amount of the quantity.

`Amount: <value of the quantity>`

The numerical `value of the quantity` must be specified.

SET_NUMERICAL_LIMITS

Description: This command makes it possible to change the criteria for convergence. This is very helpful to speed up a calculation in a complex system.

Syntax: SET_NUMERICAL_LIMITS

Prompt: `Maximum number of iterations /200/: <xxx>`

By default, the program will try 500 iterations before it gives up. As some models will give computation times of more than 1 CPU second/iteration, this number is also used to check the CPU time and the calculation will terminate if 500 CPU seconds/iterations have been 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 will normally mean fewer iterations but less accurate solution. 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 very unstable. It is normally only in the gas phase one may 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 the user has used the `SET_NUMERICAL_LIMITS` command to re-set an even-lower value (*e.g.* `1E-45`, that will naturally be 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 they may not have reached their most favorable composition and thus their driving forces may be only approximate [corresponding to the `Y` (Yes) answer for this question]. The user may also force the calculation to converge for the metastable phases by answering this question by `N` (No), if it is very 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.

Notes: If there is any convergence problem in finding a stable solution at any stage during a calculation procedure enforced by the `COMPUTE_EQUILIBRIUM`, or `STEP_WITH_OPTION`, or `MAP` or `ADVANCED_OPTIONS` command), the following messages will appear 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 has been 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 `SET_NUMERICAL_LIMITS` command to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits.

SET_REFERENCE_STATE

Description: 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 one 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 the user himself may select the reference state for a component if the reference state in the database does not suit.

One may also 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 will not be prompted.

In order to specify conditions in the user specified reference state, one may 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 <COMPONENT> <PHASE> <TEMPERATURE> <PRESSURE>

Prompt: `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 will result 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 will have the lowest energy and consequently become the reference state.

Note that 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 will remain the reference state even at calculations performed at lower temperatures where O2 has a lower energy than O1.

`Temperature /*/: <temperature for the reference state>`

One may select the temperature (in K) for the reference state. The widecard value * means the current temperature is used for the calculation.

`Pressure /1E5/: <pressure for the reference state>`

One may select the pressure (in pa) for the reference state. The widecard 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

Description: This command is similar to the SET-ALL-START-VALUES command, but is used for an individual phase that may have a wrong constitution.

Syntax: SET_START_CONSTITUTION

Prompt: `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 a "*", the user selects 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

Description: A start value of an individual variable can be set by this command. It is rarely used any more as the automatic start values can be handled for most problems.

Syntax: SET_START_VALUE

Prompt: 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

Description: This command is used to show on the screen the current value of any state variable, function or variable.

Syntax: SHOW_VALUE <NAME(S) OF STATE VARIABLE(S) OR SYMBOL(S)>

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. Note that the wildcard "*" can be used to denote all phases or all components. One may also use the dollar-sign "$" to denote all stable phases. Thus, the command SHOW W(*,*) will list the mass fraction of all phases, whereas SHOW W($,*) will list the mass fraction of all stable phases.

Note that if you SHOW a function, all functions will be evaluated using the current values of the state variables. But if you SHOW a variable, it will retain its value from the time it was ENTERED or last EVALUATED. Thus if one wishes to save a value from one calculation to another, one may ENTER it to a variable. This is frequently done in the PARROT module to transfer values between equilibria.

STEP_WITH_OPTIONS

Description: This STEP_WITH_OPTIONS command initiates the stepping procedure for making 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). Note that 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 evoked, one must have already calculated an initial equilibrium point and have also set one indenpendt varying variable *(i.e.,* the controlling condition in the system) that has been set by the SET_AXIS_VARIABLE command.

The Thermo-Calc Console Mode will list the current values of the axis variable for each calculated equilibrium, and also list 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 the user (or possibly in some special modules) through changing the minimization option using the ADVANCED_OPTIONS MINIMIZATION_OPTION command-sequence. The user can additionally 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:

- INITIAL_EQUILIBRIA: This means that an initial equilibrium will be stored at each calculated equilibria. (However, please note that this option has not been completely implemented yet). It has been 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 instance, one may first use the temperature condition as the stepping axis-variable, and give a STEP INITIAL_EQUILIBRIA commandsequence; this will calculate a number of initial equilibrium points and automatically ADD 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), one may alternatively chose a compositional condition as the new stepping axis-variable; then the STEP NORMAL command-sequence will use those created starting equilibrium points (at different temperatures) and step in such a new compositional variable. As a result, these two sequential STEP commands together will give a “matrix” of values. Actually, one may carry this option even deeper.
- SEPARATE_PHASES: This means that each of all the entered phase will be calculated separately and in parallell at each step, i.e., all the phases are calculated for the same composition (if the phases exist for that composition). In this way, one may calculate how the Gibbs energy for a number of phases varies for varying compositions; and this option is particularily 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: This means that the stepping calculation will be 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 particularily 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 will at least start the stepping in ecah of the phases in their default most-stable compositions (the major constititutions). Therefore, this option is always recommended for STEPPING heterogeneous systems with both ordered phases and their disordered pairs.
- EVALUATE: This means that the variables given will be 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: This means that a stepping calculation that is specially designed for the Extended Scheil-Gulliver Approach (partial-equilibrium transformations) for simulating alloy solidifications with back-diffusion of one or more fast-diffusing interstitial components (such as C, N, O, S, etc.) in solid phases will be performed. 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.
- T-ZERO: This means that T0 (T-zero) lines in a diffusionless transformation [where two specific partially-equilibrated phases have the same Gibbs energy] will be calculated along a composition variable (set as the stepping variable).
- PARAEQUILIBRIUM: This means that 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.] will be calculated along varied temperature or along a composition variable (of the matrix or one substitutional component).

Syntax: STEP_WITH_OPTIONS <A CHOSEN OPTION, N OR I OR S OR O OR E OR M OR T OR P>

Prompt: `Option? /NORMAL/: <a chosen option, N or I or S or O or E or M or T or P>`

Options: One has to choose one of the following eight options:

NORMAL This option allows a stepping calculation with the chosen independently-varying equilibrium condition (axis-variable). By repeating the `STEP NORMAL` command-sequence, a user 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 has been specified by the user before the first `STEP NORMAL` command-sequence). Please note that 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.

The user can choose to alter one or more controlling conditions for calculating starting equilibriumpoint(s) and can also optionally change the independently-varying condition (stepping-variable) and its settings for the stepping. All such alternations/changes will be 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 the user to easily/efficiently generate some special types of property diagrams and even to plot special phaseboundary lines on phase diagrams, as described below:

- Calculate/plot missing part(s) of a specific property [that arecalculated under another controllingcondition 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 controllingconditions 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 phasetransformations 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 `MAP` calculation) for the same system.

INITIAL_EQUILIBRIA An initial equilibrium stored at every step. *This option has not been completely implemented yet.* It has been intended to make a stepping 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 instance, one may first use the temperature condition as the stepping axis-variable, and give a `STEP INITIAL_EQUILIBRIA` command-sequence; this will calculate a number of initial equilibrium points and automatically `ADD` 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), one may alternatively chose a compositional condition as the new stepping axis-variable; then the `STEP NORMAL` command-sequence will use those created starting equilibrium points (at different temperatures) and step in such a new compositional variable. As a result, these two sequential `STEP` commands together will give a "matrix" of values. *Actually, one may carry this option even deeper.*

SEPARATE_PHASES Each phase calculated separately. This option performs a stepping calculation in which all the entered phases in the system will be 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, one can calculate how the Gibbs energy for a number of phases varies for varying compositions. This option is very useful when the user wants 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. With this new option, which has been available since TCCS, 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.

Please note that the "ONE_PHASE_AT_TIME" option is almost identical to the older option "SEPARATE_PHASES", while the newer one might be (sometimes) more powerful for stepping calculations in which some ordering/disordering phase-transformations occur; this is because of that, the STEP ONE_PHASE_AT_TIME command-sequence will 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 will at least start the stepping in ecah of the phases in their default most-stable compositions (the major constititutions). Therefore, this option is always recommended for STEPPING 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 one may change the conditions during the stepping. After a successful `STEP EVALUATE` calculation, one may go to the POST module to manually define and plot various desired property diagrams.

With this option, the following prompt needs to be specified:

```
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 have been simplified by the general improvement of convergence but a number of additional fixes have been 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`) if a phase has two or more composition sets.

The miscibility gap test means that one 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 will first become metastable with a composition different from the metallic FCC phase and later also stable. Previously such `STEP EVALUATE` calculations had to be made from low temperature.

One good case of using this advanced option is described in the `INFORMATION` command with the subject as `SOLIDIFICATION` (Solidification Path Calculations). With this option, one 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 document of Thermo-Calc Console Mode Examples Book. 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

Extended Scheil-Gulliver type solidification with back diffusion.

This option, which has been available since TCCQ, 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. Also see more information in *Section 8.4.3, Solidification path simulations*.

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 very important to pay special attentions 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 have been already defined in terms of mole-fraction or mass-fraction, *e.g.,* `X(C)` and `W(N)`. Otherwise, a stepping calculation with this option will fail.
- An initial equilibrium in which the LIQUID mixture phase is the only stable must have been 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 have been 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, the following 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;
- 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, just type NONE to ignore back diffusion entirely, meaning a normal Scheil-Gulliver simulation will be performed.

Please note that 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.*): It/they shall be located on the interstitial/vacancy sublattices in such alloy solution phases; otherwise, such a ScheilGulliver simulation may not make sense at all.

```
Allow BCC -> FCC ? /Y/: <Y or N>
```

Type Y (for Yes, or by pressing <RETURN> to accept the default Y) or N (for No) to allow or not allow the BCC-->FCC phase transformation (which is usually a typical phenomena 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
       1733.1500          0.3294          0.6707
-8032.6240
       1733.0250          0.3237          0.6763
-8095.1490    . . . . . .
```

After a successful STEP MIXED_SCHEIL calculation, one may go to the POST module to manually define and plot various property diagrams for the alloy solidification process according to the chosen Scheil-Guillver 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 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 thee 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 massfraction), 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, 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.

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 by the SET_AXIS_VARIABLE command. Note that the temperature condition cannot set as the stepping variable if one wants to make a STEP T-ZERO calculation.

To ensure a successful calculation of T_0 line between two specific phases after an initial equilibrium calculation in the current system, it is recommended to make a single T_0 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.

With this option, the following prompts need to be specified:

```
Name of first phase:  <phase A>
Name of second phase: <phase B>
```

The names of the two target phases, for which the Gibbs energies equal to each other at each point on the T_0 line, must be typed at the above two prompts subsequently.

During the STEP T-ZERO calculation procedure, the calculated T_0 values are shown after the corresponding scanned conditions (of the stepping composition variable), *e.g.,*

```
 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
  Phase Region from   1.000000E-01 for:
      BCC_A2
      FCC_A1
    1.000000E-01          940.24
    1.075000E-01          939.62
    . . . . . .
    2.950000E-01         1084.87
      3.000000E-01         1080.99
```

After a successful STEP T-ZERO calculation, one may go to the POST module to plot the T_0 line against the stepping composition variable or another varying axis, or to impose the calculated T_0 line onto a normal phase diagram [normally plotted as an T-X isopleth]. See *Examples 23* and *41* in the document of *Thermo-Calc Console Mode Examples Book*.

PARAEQUILIBRIUM — Paraequilibrium state between two specific phases calculated

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.

Therefore, the STEP PARAEQUILIBRIUM command-sequence will ensure that the two partiallyequilibrated phases will 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 by the SET_AXIS_VARIABLE command.

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.

Please also note that the composition condition for the interstitial components cannot be set as the stepping variable if one wants 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 very 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 instance 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.

It is also recommended to keep the following four important issues in mind:

- Always check if the chosen `phaseA` and `phaseB` 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 `SPEICAL` using the command:

  ```
  CHANGE_STATUS COMPONENT
  <interstitial component> = SPECIAL
  ```

  By doing this, you will get a clear picture on u-fractions of various substitutional and interstitial components, which are of course different from the overall composition in the system. The `SPECIAL` status means that specified component(s) will not be 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, the following 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`").

Please note that you must have completely understood what you are dealing with here is the calculation of a paraequilibrium state between the two specified phases. Particularily, there are four distinguished issues which need to pay a great attention: (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 simply impossible to calculating 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 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, just press `<RETURN>` key to start the paraequilibrium line stepping-calculation.

Please pay attentions on the followings aspects:

- Such specified fast-diffussing interstitial component(s) must be appropriately defined according to the phase constitution definitions of the two selected phases: It/they must be located on the interstitial/vacancy sublattices in both chosen phases.

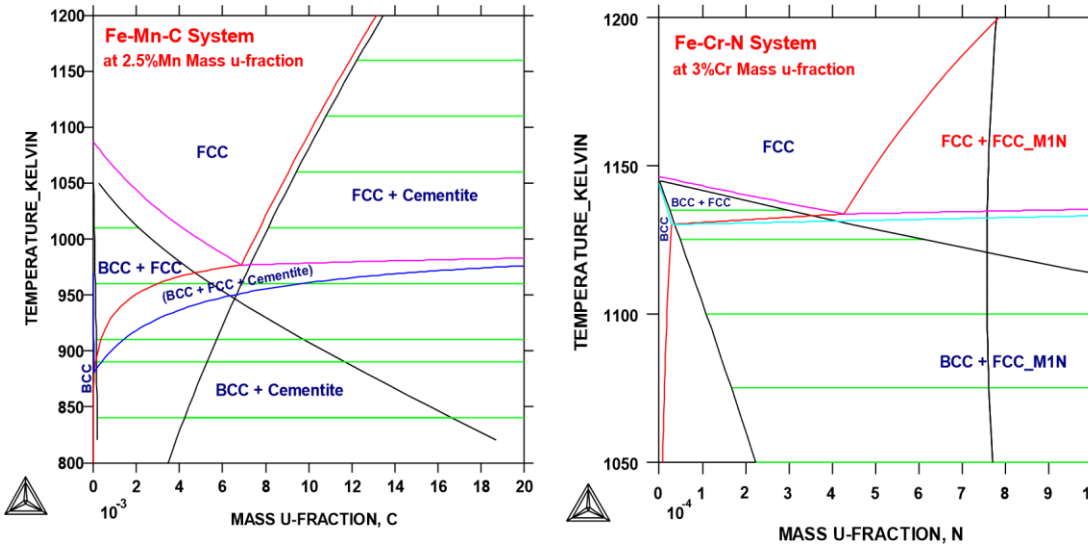
- If there is only one fast-diffusing component which is carbon, press <RETURN> to accept the default value (C) at the first prompt; if the single fast-diffusing interstitial component is another element (*e.g.,* N), appropriately type interstitial component the name at the first prompt;
- If there are 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);
- the amounts of the phase A in mole number *[i.e.,* NP(phaseA)];
- the amounts of the phase B in mole number [*i.e.,* NP(phaseB)];
- the contents of the interstitial component(s) in the phase A expressed in u-fractions [*i.e.,* U-F(phaseA,component(s))];
- the contents of the interstitial component(s) in the phase B expressed in u-fractions [*i.e.,* U-F(phaseB,component(s))];
- the LNACR(component) value(s) for the interstitial(s).

After a successful STEP PARAEQUILIBRIUM calculation, one may 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; as illustrated in *Figure 8-7* as an example]. See Examples 23 and 42 in the document of Thermo-Calc Console Mode Examples Book.

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 later 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 Febased alloys, with equilibrium results appended. Note that 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.

Notes:

During a `STEP` calculation, the value of the stepping axis variable for each calculated equilibrium will be listed and also the set of stable phases

The user may terminate the mapping of a line by pressing a single `CTRL-A` (under MSWindows) 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, the following messages will appear 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 has been 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 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 will be automatically detected during a stepping calculation, and two or more composition sets for each of such solution phases will be 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); in earlier versions up to TCCR, that was done by the MISC_GAP_TEST_INTERVAL option of the SPECIAL_OPTIONS command. *As a particular case,* such an automatic Global Minimization Test ensures that one 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; previously, such calculations had to be made from low temperatures. In other words, the advanced stepping 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.

Occasionally, it may be necessary to have a user-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

Description: This command will give the values of an already-defined table (with various properties varied with the stepping axis variable) from equilibria calculated with a STEP command. One can tabulate a table even the table is entered after the STEP calculation.

Syntax: TABULATE <NAME OF A DEFINED TABLE> <RETURN OR A FILE NAME>

Prompt: `Name: <name of a defined table>`

The name of the table must be given. The table must have been ENTERED, of course.

`Output to Screen or file /SCREEN/: <file name>`

The table can be listed on the screen (by pressing <RETRUN>), on a file (the `file name` must be specified here).

8 POST commands

ADD_LABEL_TEXT

Description: With this command the user may add a text to an area in a phase diagram or a property diagram. The label will be written starting from the specified coordinates. Optionally, one 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. Please note that such an optional calculation for automatic phase region labeling may fail in some cases. It is not possible to automatically add labels for certain sets of axes, like entropy, enthalpy, pH, Eh, *etc*.

The labels are volatile and will disappear if any axis is changed afterwards.

The labeling texts and their coordinates, either added by this command or modified by the MODIFY_LABEL_TEXT command or manually edited on the “Thermo-Calc Console Results” window, are stored in the workspace, and they may later be saved onto an *.EXP file (through the MAKE_EXPERIEMENTAL_DATAFILE command).

Syntax: ADD_LABEL_TEXT <X COORDINATE> <Y COORDINATE> <Y/N> <LABEL TEXT> <TEXT SIZE>

Prompt: `Give X coordinate in axis units: <value of the X coordinate>`

Specify the value of the X coordinate where the label will start.

`Give Y coordinate in axis units: <value of the X coordinate>`

Specify the `value of the Y coordinate` where the label will start.

`Automatic phase labels? /Y/: <Y or N>`

If the answer to this question is `Y`, the program will automatically calculate the equilibrium at the given coordinates and generate a label with the names of the stable phases. However, the automatic calculation procedure works only for phase diagrams that have been mapped with two axes in the POLY module, and may fail sometimes, especially for complex heterogeneous interaction systems; in such cases, a warning message will appear on screen.

`Text: <text for the label>`

If the answer to the previous question was N, the user may specify a text himself. Provided that they are supported by the font you are using, you can include the following types of characters in a label:

- Uppercase and lowercase greek letters: Enter the letters as an argument to the "^GR" code and end the string with a dollar sign ("$"). For example, to enter "α B γ δ ε Φ γ η ι φ κ" into a label, enter "^GRa B c d e F g h i j k$".
- Numbers, parentheses, "+", "-", "i" and "n" in sub- or superscript: Enter these characters as an argument to the "^DO" code (subscript) or the "^UP" code (superscript) and end the string with a dollar sign ("$"). For example, to enter "10^{2+i+n}" into a label, enter "10^UP(2+i+n)$".
- Right- or left-facing arrows can be added by using the character combinations "!A" and "!B".
- Any unicode character can be added by typing "^UC" followed by the code(s) for the character(s) that you want to include in the label. For example, to enter "ѡ ђ ѣ Ѫ ѫ" into a label, enter "^UC461 452 463 46a 46b$".

You can rotate the label by beginning the label text string with the command "^RO", immediately followed by the number of degrees you want the label to be rotated. For example, to rotate the label "10^{2+i+n}" forty-five degrees, enter "^RO45 10^UP(2+i+n)$".

`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. Our recommendation is *e.g.* 0.25 for an even better looking diagram.

APPEND_EXPERIMENTAL_DATA

Description: This command is typically used for adding experimental data and text onto a calculated diagram. This is achieved 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. This is described in the document *DATAPLOT User's Guide & Examples*.

The picture generated from the data is superimposed on the ordinary graphical output from the postprocessor. Such an experimental datafile (`*.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` (see below), which will dump 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.

Syntax: `APPEND_EXPERIMENTAL_DATA`

Prompt: `USE EXPERIMENTAL DATA (Y or N) /N/: <Y or N>`

Specify whether the data from an experimental datafile should be included in the next plot. If the answer is `N`, no experimental data will be plotted.

An "Open" window will then pop up on the screen for specifying the name of the file with the experimental data.

`PROLOGUE NUMBER: /0/: <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. Note that by answering -1, a list of all prologues on the file is given; and by 0, no experimental data will be 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. Note that by answering -1, a list of all datasets on the file is given; and by 0, no experimental data will be appended.

DUMP_DIAGRAM

Description: This is command for quick printing. Similar to the PRINT_DIAGRAM command, after a Thermo-Calc graph has been created (and basically has been shown up on screen by the PLOT_DIAGRAM command), one can use this command to save (dump) the diagram to a file.

Syntax: DUMP_DIAGRAM <FORMAT> <FILENAME> <Y/N>

Prompt: OUTPUT FORMAT /: <format or RETURN>

Specify an appropriate graphic format for dumping the graph or pressing <RETURN> to accept the default format.

A "Save" window will be opened if you have not specified a filename. Select a directory path and enter a name in the "File name" field. If a file with the specified name already exists, then you will be 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").

Important Note: Similar to the case for the PRINT_DIAGRAM command, it does not matter what graphic device has been set up for the graph, this command will always be able to convert the graph shown on screen to the selected graphic format.

After the printing process is finished, the temporally set graphic device for conversion and printing will be switched back to the one previously set by the command SET_PLOT_FORMAT, so that the user can save the graph on desired file(s) using the PLOT_DIAGRAM command.

ENTER_SYMBOL

Description: Symbols are a very useful feature of POLY and POST to define quantities that are convenient for the user. Symbols can be constants, variables, functions or tables.

Please note that all the symbols which are entered in the POST module are not saved in the currently-loaded POLY3 workspaces. Therefore, if you wish to apply such symbols also in other similar calculations for the same defined system, you shall use the ENTER_SYMBOL command prior to the STEPPING or MAPPING calculation in the POLY module, instead.

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 just 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 will only be evaluated when they are entered or if they are explicitly named in an EVALUATE command. It is possible to enter a variable with a new expression anytime. This expression will be 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.

Note: There is a special connection between tables and variables. If a variable is used in a table, it will be 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. If one wishes to enter the symbol name and the value or function on the same line, they must be separated with an equal sign "=", *e.g.,* TC=T-273.15. Otherwise, the following questions will be proceeded; note that for different types of symbol (constant, function, variable or table) which is under definition here, the questions will be prompted differently.

```
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 simply pressing <RETURN> at the next prompt). Examples of functions:

- 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 one may 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 <RETRUN> 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 will be 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 function will be 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 will contain four columns, i.e., the temperature, the mole fractions of C and Cr in the LIQUID phase, and the activity of C.

If one wishes to have the temperature in Celsius in a table, one must first 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 one may continue to write the table on the new line if one line is not enough for the table. If one has finished the table just press RETRUN again.

FIND_LINE

Description: This ommand identify 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

Prompt: `X coordinate /.484623611/: <value of the X coordinate>`

Specify the `value of the X coordinate` where you wish 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 wish to find the details of property curve (or phase region).

Then the POST module will show (on screen) 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

Description: This command (available since TCCR) 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 have been plotted on the *Thermo-Calc Console Results window*), but only after a STEP calculation. The listing output can be either on screen or into a spreadsheet file in the MS Excel format (under a user-specified file name with the default file extension of XLS), which can be further edited/manipulated or used for further graphical handling by opening the file (using the MS-Excel program) outside of the Thermo-Calc software system; as illustrated in *Figure 9-14*.

Please note that it works only for a property diagram after a stepping calculation (but not for any phase diagram after a mapping calculation).

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) will be saved, or press <RETURN> if want to see a simple textual table (for the already-defined properties that change alone with an independent variable) on screen.

LIST_PLOT_SETTINGS

Description: The present values of most parameters specifying the type of diagram to be plotted will be listed on the terminal.

Syntax: LIST_PLOT_SETTINGS

LIST_SYMBOLS

Description: For all constants, functions, variables and tables, their definitions can be listed by this command. The defined variables will be listed up together with the defined functions, but variable names will be followed by a percentage sign "%".

Syntax: LIST_SYMBOLS

MAKE_EXPERIMENTAL_DATAFILE

Description: This command makes it possible to write the graphical information onto screen or an formatted textual file (always having an extension of EXP) using the DATAPLOT graphical format. In order to merge two or more diagrams from separate calculations, one may write them out on files with this command, and then add/edit them together with a normal textual editor.

Syntax: MAKE_EXPERIMENTAL_DATAFILE <NAME OF A FILE OR RETURN>

Prompt: OUTPUT TO SCREEN OR FILE /SCREEN/: <name of a file or RETURN>

Specify the name of the desired file. One can accept /SCREEN/ (by pressing <RETURN>) for a listing. Otherwise, a file name where the graphical information will be written must be given here. The default file extension is "EXP" (under MSWindows) or "exp" (under Linux).

MODIFY_LABEL_TEXT

Description: This command make it possible to move a label created with the ADD_LABEL_TEXT command to another position, or replace its text with another one.

The labeling texts and their coordinates, either added by ADD_LABEL_TEXT command or modified by this command or manually edited on the "Thermo-Calc Console Results" window, are stored in the workspace, and they may later be saved onto an *.EXP file (through the MAKE_EXPERIEMENTAL_DATAFILE command).

Syntax: MODIFY_LABEL_TEXT

Prompt: `Which label to modify? /#/: <number of the label>`

Before this prompt, all the labels created by the ADD_LABEL_TEXT command will be listed with an identifying number. Please 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 shown up and can be accepted by pressing RETRUN.

`New Y coordinate: /yyy/: <new Y position>`

Specify the `new Y position.` The previous Y coordinate is shown up and can be accepted by pressing RETRUN.

`New text /ABCDEFGH.../: <new labeling text>`

Specify the `new labeling text.` The previous text is shown up and can be accepted by pressing <RETRUN>. Note that the new labeling text must not be longer than the previous text.

PLOT_DIAGRAM

Description: The graphic information will be plotted on a specific graphic device using the plot format set by the SET_PLOT_FORMAT command. Note that the diagram axes must be set first!

The functionality of this command has been split up into two: one remained in the PLOT_DIAGRAM command for normal screen display and file saving; *and* the other implemented in the PRINT_DIAGRAM command (under MS-Windows environment) for direct hard-copy printing.

Therefore, with this command, a calculated/plotted diagram will be shown on screen, if the default graphic device has already been defined, or be saved onto a desired file.

Syntax: PLOT_DIAGRAM <NAME OF A FILE OR RETURN FOR SCREEN>

Prompt: `OUTPUT TO SCREEN OR FILE /SCREEN/: <name of a file or RETURN>`

Specify the name of the desired file (Note that under Linux, the prompt is "PLOTFILE /SCREEN/:" instead). Consequently, a file with a proper extension for the selected graphic format (previously set by the SET_PLOT_FORMAT command) will be saved under the current working directory. If no extension is given in the file name, the default extension (as shown in the list given in *Section 9.3.5*, *e.g.*, "ps" for Postscript portrait/landscape modes, "P7" or "p7" for HPGL landscape, "P8" or "p8" for HPGL portrait, *etc.*) will be automatically given to the file name.

If the default device has been set (by the command SET_PLOT_FORMAT), one can either accept /SCREEN/ to plot the graph in the Thermo-Calc Graph window on screen by pressing <RETURN>, or save it on a file by giving a file name. If no extension is given in the file name, the default.

PRINT_DIAGRAM

Description: This is a very useful command for immediate printing with high quality. After a Thermo-Calc graph has been created (and basically has been shown up on screen by the PLOT_DIAGRAM command), one can directly print the diagram on a connected printer using this command.

Note that this command has the same function as that of the "Print Plot" option directly on the Thermo-Calc Console Results windows, for printing out a plotted diagram. Therefore, the use can print the plotted diagram to a connected printer (also possible to a fax machine, as a word-editing document or graphical-formated file, depending upon how you have set your local printer settings).

Syntax: PRINT_DIAGRAM

QUICK_EXPERIMENTAL_PLOT

Description: This command is similar to the APPEND_EXPERIMENTAL_DATA command 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) datafile.

Syntax: QUICK_EXPERIMENTAL_PLOT <NAME OF AN EXPERIMENTAL DATA FILE>

An "Open" window (which is the same as in the APPEND_EXPERIMENTAL_DATA command) will pop up on the screen instead.

Prompt: PROLOGUE NUMBER: /0/: <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. Note that by answering `-1`, a list of all prologues on the file is given; and by `0`, no experimental data will be 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. Note that by answering `-1`, a list of all datasets on the file is given; and by `0`, no experimental data will be appended.

REINITIATE_PLOT_SETTINGS

Description: With this command, all parameters describing the diagram will be given default values. Therefore, all plotting setting made in the POST module will be lost, and everything will return to their initial settings when the POST module is entered.

Syntax: REINITIATE_PLOT_SETTINGS

RESTORE_PHASE_IN_PLOT

Description: This command restores a phase that has been previously suspended from the plot with the `SUSPEND_PHASE_IN_PLOT` command. Naturally it works only after at least one phase has been suspended by the `SUSPEND_PHASE_IN_PLOT` command.

Syntax: RESTORE_PHASE_IN_PLOT <NAME OF A PHASE>

Prompt: `PHASE NAME: <name of a phase>`

Give the name of the phase to be restored on the diagram.

SELECT_PLOT

Description: This command can be used to select a plot within the current Results tab in the Console Results window, or to create a new plot tab.

Syntax: SELECT_PLOT <N/NEW>

Where “N” is a positive integer.

Prompt: `PLOT /NEW/:`

Enter the number of the plot that will be selected in the Console Results window (in the currently selected Results tab). If a plot with that number does not exist, it will be created.

The default parameter value is “NEW”. If this is entered, then a new plot is created. A new plot can also be created by pressing Ctrl+Shift+T.

SET_AXIS_LENGTH (not supported in Thermo-Calc 3.1)

Description: This command can be used to 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`. This command has been added due to some idiosyncrasies in the graphical package used by POST. 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

Prompt: `AXIS (X, Y or Z): <name of an axis>`

The user must specify which axis to set the length.

`AXIS LENGTH /11.5/: <new relative axis length>`

The user can specify the new real axis length in inches. The relative length `1` corresponds to `10` tic-marks on the axis.

SET_AXIS_PLOT_STATUS (not supported in Thermo-Calc 3.1)

Description: With this command, the user can specify whether the axis title texts and axis labeling texts are to be plotted or not on a diagram. It does not work for the axis lines and tic-marks. If one wishes to plot a diagram without tic-marks, the `SET_TIC_TYPE` command should be used 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 <Y OR N>

Prompt: `Axis plot (Y or N) /Y/: (Y or N)`

The user can specify to plot axis texts (`Y`) or not (`N`).

SET_AXIS_TEXT_STATUS

Description: This command can be used to change the axis text from the automatic text given by the axis specification to a text given by the user.

Syntax: SET_AXIS_TEXT_STATUS

Prompt: `AXIS (X, Y or Z): <name of an axis>`

The user must specify for which axis's text status is to be changed.

`AUTOMATIC AXIS TEXT (Y or N) /N/: <Y or N>`

The user must specify if automatic axis text is to be used (`Y`) or not (`N`). If automatic axis text has not been chosen, the user will be further prompted for his own axis text:

`AXIS TEXT: <desired axis text>`

The user shall enter his own desired axis text.

SET_AXIS_TYPE

Description: This command is to change the axis type among linear, logarithmic and inverse axis.

Syntax: SET_AXIS_TYPE

Prompt: `AXIS (X, Y or Z): <name of an axis>`

The user must specify for which axis you want 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 (not supported in Thermo-Calc 3.1)

Description: Some devices that support colors or line thickness. With this command, one may select different Colors or LineTypes on some types of information on a diagram. Note that this command will fail if a chosen device does not support colors or line thickness.

This command will arise four sequent prompts (*i.e.,* `Text and axis`, `Invariant equilibria`, `Tie-lines`, and `All other lines`), with the `Keyword` and `default option` (both listed as shown below). Depending on if the switched-on graphic device (by the `SET_PLOT_FORMAT` command) supports with colors or line types, the POST will *automatically* switch 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

Notes*:* 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` command); the POST module will *automatically* switch on the appropriate `Keyword`.

The 22 legal options for `Color` are:

- BACKGROUND
- FOREGROUND (default)
- RED (default for invariant equilibria)
- GREEN (default for tie lines)
- BLUE
- YELLOW
- MAGENTA
- CYAN
- PURPLE
- GOLD4

- TURQUOISE4
- PINK
- GRAY
- ORANGERED3
- MAROON
- PLUM
- SEAGREEN
- OLIVEDRAB
- SIENNA
- ORANGE1
- CORAL1 USERDEF

The 8 legal options for `LineType` are:

- INVISIBLE
- NORMAL (default)
- VERY_THICK (default for invariant equilibria)
- THIN (default for tie lines)
- THICK
- VERY_THIN
- DASHED
- DOTTED

SET_CORNER_TEXT

Description: This command writes texts in the corners on a plotted diagram. Normally, one can write such texts as a subtitle.

Syntax: `SET_CONER_TEXT`

Prompt: `CORNER /LOWER_LEFT/: <RETURN or new option>`

Use one of following five options:

- 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.

Notes: 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` has been switched off by the command `SET_PLOT_OPTION`, the calculated conditions will be always write on plotted diagrams; and only when the plotting option `PLOT HEADER` has been switched off by the command `SET_PLOT_OPTION`, the used database will also always appear. Under such circumstance, one should avoid writing texts at the `UPPER_LEFT` corners.

SET_DIAGRAM_AXIS

Description: The user can 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.

Note that the axis variables in the plot can be different from the number of axis used for the mapping.

Please also note a catch that even for frequent users it may misunderstand. 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 will only give one side of the two-phase region. The reason for this is complicated to explain. 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`".

Syntax: SET_DIAGRAM_AXIS <AXIS NAME> <VARIABLE NAME & SPECIFICATION>

Prompt: `Axis (X, Y or Z): <axis name>`

The user must specify for which axis (X or Y or Z) to set with a variable.

`Variable type: <variable name>`

Specify a variable for the chosen axis here.

The valid variables are listed below:

- TEMPERATURE-CELSIUS as temperature in °C
- 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 will further ask for a column number.
- Any entered function or variable.
- Any entered table.

If you specify an entered table as the variable, then you will be presented with the following prompts:

`For component: <component name>`

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

Note: The axis must have exactly the same number of columns, or one axis must have just one column. In the first case the columns will be plotted matching them one by one; in the latter all columns on one axis will be plotted against the single column. For example, one may 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 command. It simplifies the use of Thermo-Calc and DICTRA, but this option has not been made visually available in Thermo-Calc, because users may believe that they cannot set any other axis than used in the mapping calculation. The flexibility to set any set of state variables as diagram axes after a MAP calculation is one of the most powerful features of Thermo-Calc.

However, it has been implemented into the Thermo-Calc software system since TCCM, while the user does not need to specify this option after a `MAP` calculation. In this way, the Thermo-Calc beginners may also avoid the tricky problem to understand the difference among `x(comp)`, `x(*,comp)` and "`mole-fraction of comp`".

If the state variable `x(comp)` has been used in a `MAP` calculation, then mole-fraction of the component will be used as diagram axis; if a potential or some other state variables has been used, then that will be used as a diagram axis.

Note that after a `STEP` calculation, automatic diagram axis will not be set as there is only one axis used in the calculation.

SET_DIAGRAM_TYPE

Description: With this command, the user can choose the diagram type as perpendicular plot or triangular plot (Gibbs triangle, especially for ternary systems). The default diagram type is with 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 will be removed.

Note that 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

Prompt: `TRIANGULAR DIAGRAM (Y or N) /N/: <Y or N>`

The user can specify triangular plot by answering `Y` (Yes). Otherwise (by pressing `<RETURN>` to accept `N`), perpendicular axis will be set and this command is terminated. By default, such perpendicular diagrams will 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), the user should answer Y here 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 has been selected, the user can specify if a 3rd axis (by answering Y here), connecting the end points of the X- and Y-axes, is plotted.

`CLIP ALONG 3:RD AXIS (Y or N) /N/: <Y or N>`

If a triangular plot has been selected, all lines outside the region limited by a line joining the end points of the X- and Y-axis will be removed by answering `Y` (Yes) at this prompt.

SET_FONT (not supported in Thermo-Calc 3.1)

Description: The user can select the font to be used for labels and numbers when plotting the diagram under the currently selected graphic device (by the `SET_PLOT_FORMAT` command). For some devices there may be other fonts available and these are selected by the `SET_PLOT_FORMAT` command.

Syntax: `SET_FONT`

Prompt: `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 `<RETURN>`. By typing a question mark "`?`" here, the program will list 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

Description: This command 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.

Syntax: `SET_INTERACTIVE_MODE`

SET_LABEL_CURVE_OPTION

Description: This command makes it possible to 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.

Note that 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. Two more options (for colorful labeling), `E` and `F`, have been made available since TCCP.

Syntax: SET_LABEL_CURVE_OPTION <CURVE OPTION>

Prompt: `CURVE LABEL OPTION (A, B, C, D, E, F OR N) /A/: <option>`

The options means:

- 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. Note that 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) will disable all curve labels and lists.

If, for example, one has used `T` (temperature in K) as the variable axis in a `STEP` command and then plots the amount of stable phases (by the command `SET_DIAGRAM_AXIS 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

Description: With this command the user can 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 `MS-Windows` and `9 for XWindow` for Linux). This default graphic device value may also be altered with the `SET_PLOT_ENVIRONMENT` command 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 "`?`", as shown below:

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

Description: With this command, one will be prompted for and may toggle on (`Y`) or off (`N`) the plotting of some options on all subsequently generated diagrams.

For the "`Always initiate POST on re-entering`" question, the default answer is `Y` (Yes) which implies that the POST module will automatically reinitiate whenever the POLY3 workspaces has been changed upon re-entering. If answered `N` (No), the same diagram axis variables will be kept even if the workspaces in the POLY module has been read from or written to an `*.POLY3` file. The `N` (No) answer is particularly useful when the user has several sets of `*.POLY3` files with the same calculation but for different set of model parameters.

For the "`Always solid line`" question, the default answer is `N` (No) which means that the POST module will 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 the alternative answer `Y` (Yes) is chosen, all the lines will always be plotted in solid lines.

Syntax: `SET_PLOT_OPTIONS`

Prompt: `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>`

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>
```

As a new feature since TCCP, 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>
```

The user can choose to accept the default answer Y (Yes) to let the POST module automatically reinitiate whenever the POLY3 workspaces has been changed upon reentering. If answered N (No), the same diagram axis variables will be kept even if the workspaces in the POLY module has been read from or written to an *.POLY3 file. The N (No) answer is particularly useful when the user has several sets of *.POLY3 files with the same calculation but for different set of model parameters.

```
Always solid line: /N/: <Y or N>
```

The user can opt for default answer N (No) to allow 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 the alternative answer Y (Yes) is chosen, all the lines will always be plotted in solid lines.

SET_PLOT_SIZE (not supported in Thermo-Calc 3.1)

Description: With this command, the user can 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 the user has chosen by the `SET_PLOT_FORMAT` command. The default plot size (corresponding to the default global plot size `1`) is adjusted to the chosen device.

Syntax: SET_PLOT_SIZE <RELATIVE SCALING FACTOR>

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 `<RETURN>`, the default relative scaling factor (`1`) is accepted.

SET_PREFIX_SCALING

Description: This command set the prefix scaling of remote exponents with certain powers (default as `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>`

The user must specify for which axis is to have a prefix scaling.

`USE PREFIX SCALING /Y/: (Y or N or #)`

The user must answer either Y or N or an integer number as described above.

SET_RASTER_STATUS

Description: With this command, it is possible to have a raster (*i.e.,* with gridlines on both axis directions) plotted in the diagram. The default status is no raster plotted.

Syntax: SET_RASTER_STATUS <Y OR N>

Prompt: `RASTER PLOT (Y or N) /N/: <Y or N>`

Enables (`Y`) or disables (`N`) the raster plot.

SET_REFERENCE_STATE

Description: The reference state for a component is important when calculating activities, chemical potentials and enthalpies. After a `STEP` or `MAP` calculation, 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, the user 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

Prompt: `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 or DORMANT or SUSPENDED, must be 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, one would like to have the most stable species as reference state of oxygen, i.e., O2 in this case. Therefore, the program will calculate the Gibbs energy of all possible states with the phase with the pure component at the current temperature and select the most stable one.

`Temperature /*/: <reference temperature>`

One may select the temperature (in K) for the reference state. The value * means the current temperature is used for the calculation.

`Pressure /1E5/: <reference pressure>`

One may select the pressure (in pa) for the reference state. The value * means the current pressure is used for the calculation.

SET_SCALING_STATUS

Description: When an axis variable is selected by the `SET_DIAGRAM_AXIS` command, the scaling status for the axis is always set to automatic scaling. By this command, the user can choose between manual or automatic scaling on a specified axis. If manual scaling is chosen, the user must specify a minimum and a maximum value. Manual scaling can be used to magnify interesting parts of a diagram.

Syntax: SET_SCALING_STATUS <NAME OF AN AXIS> <Y OR N> <MIN> <MAX>

Prompt: `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>
```

You can choose between automatic (`Y`) and manual scaling (`N`). If manual scaling is chosen, you must further specify the following parameters:

```
MIN VALUE: <minimum value>
```

The `minimum value` at the start point of the specified axis.

```
MAX VALUE: <maximum value>
```

The `maximum value` at the end point of the specified axis.

SET_TIC_TYPE

Description: This command makes it possible to 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 they are adjusted based on the selected graphic device and the defined relative scaling factor (global plot size).

Syntax: SET_TIC_TYPE <1 OR -1 OR 0>

Prompt:

```
TIC TYPE /1/: <1 or -1 or 0>
```

The tic type `1` is default, *i.e.,* tic are drawn outside the diagram. `-1` means that the tics are inside the diagram and `0` means no tics.

SET_TIELINE_STATUS

Description: If the tie-lines are in the plane of the calculation, one may select to plot the tie-lines in twophase fields with this command.

Syntax: SET_TIELINE_STATUS

Prompt:

```
PLOTTING EVERY TIE-LINE NO /0/: <number of tie-lines>
```

The tie-lines plotted will not be equally spaced by the graphics software. Instead, the user may 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 will be plotted.

SET_TITLE

Description: The user can specify a title that will appear on all listed table and plotted diagrams from the POST module in the same run of Thermo-Calc Console Mode.

Syntax: SET_TITLE <TITLE>

Prompt:

```
TITLE: <title>
```

Enter the `title` you want to appear on all tabulation or graphic outputs. Note that there is a maximum length of about 60 characters for the title.

SET_TRUE_MANUAL_SCALING (not supported in Thermo-Calc 3.1)

Description: 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 behavior 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 behavior, just 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 for which axis (X or Y or Z) you want to toggle between automatic adjustment or avoid the adjustment of the given maximum and minimum values.

SUSPEND_PHASE_IN_PLOT

Description: This command allows one to specify that lines originating from the presence of a certain phase shall not be plotted on a diagram. If one wants to bring the suspended phase back to the diagram, just use the `RESTORE_PHASE_IN_PLOT` command.

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

Description: For any entered table, this command will give 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.

Not like the `TABULATE` command in the TAB module, it is impossible to plot columns from the tabulated tables. For such plotting one can use normal POST commands.

Syntax: TABULATE

Prompt: `Name: <name of a table entered in either POLY or POST>`

Specify a table name that has been 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 `<RETURN>` if want to see the table values on screen.

9 REACTOR commands

AMEND_RECORD

Description: This command is to amend some of the already-created record, but has not been implemented yet.

Syntax: AMEND_RECORD

CREATE_DIVIDERS

Description: This command defines 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

Prompt: `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 instance, if the stage box number has been defined as `4`, the divider number should be defined as `5` here.

`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

Description: This command defines 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

Prompt: `From record: <a record number>`

Specify a record number. If there are 4 segments in the reactor, the following records will be 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 repeated for all inputs and/or outputs for mass and/or heat in current record.

CREATE_PROBE

Description: This command is to create probe, but has not been implemented yet.

Syntax: CREATE_PROBE

CREATE_STAGE_BOXES

Description: This command creates several reaction stages as boxes (segments), where 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, must be specified.

Syntax: CREATE_STAGE_BOXES

Prompt: NUMBER OF STAGE BOXES /4/: <the number of stage boxes>

Specify the number of stage boxes in the reactor. Then the following message appears:

YOU MUST FIRST DEFINE FEED FROM SURROUNDINGS!

Now the feed of mass and/or energy (from surroundings to) the rector can be defined in subsequent prompts.

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 ensuring prompt).

Input temperature /298.15/: <input temperature>

GIVE FEED TO SYSTEM: <feed>

Specify other feeds (mass or energy) to the reactor. Press <RETURN> to finish the feed inputs, and then the following message appears:

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>

At present moment, only the EQUILIBRIUM type is allowed; by pressing <RETURN> to accept it.

Is the stage heat controlled? /Y/: <Yes or No>

Y (Yes) means the current segment is heat controlled, while N (No) means it is not.

```
Initial guess of temperature? /1000/: <initial temperature>
```

Give a guess of initial temperature (in K) for the current segment; by pressing <RETURN> to accept the default suggestion.

```
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 ensuring prompt).

```
Input temperature /298.15/: <input temperature>
Give initial amount: <initial amount of mass or heat>
```

Further specify initial amount (mass or energy) in the current segment. Press <RETURN> to finish the initial amount inputs, and then the following message appears:

```
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 <RETURN> 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 has been defined.

EXECUTE_POLY3_COMMAND

Description: This command provides the possibility of directly utilizing all kinds of POLY commands inside the REACTOR module.

Syntax: EXECUTE_POLY3_COMMAND <A VALID POLY COMMAND>

Any the legal POLY command can be used here.

LIST_DESIGN

Description: This command is to list the design details of the creator, but has not been implemented yet.

Syntax: LIST_DESIGN

LIST_RECORDS

Description: This command lists on the screen all the already-created records for the reactor (including stage boxes, dividers, pipes, *etc.*)

Syntax: LIST_RECORDS

READ_WORKSPACE

Description: The REACTOR workspace (along with the related POLY3 and GIBBS workspaces) and the simulated results from the SIMULATE command can be read by this READ_WORKSPACE command from a binary file where they must have been saved previously with a SAVE_WORKSPACE command. This *.RCT file is not printable.

Syntax: READ_WORKSPACE <FILE NAME>

Options: file name -- The name of a previously saved *.RCT file where the REACTOR and POLY3/GIBBS workspaces shall be read from must be specified. The user does not need to type the extension if it is the default ".RCT", otherwise the user must type the whole *.RCT file name.

An "Open" window will pop up on the screen if a file name is not given after the command or its path is incorrect, so that the path (in the "Look in" box) and file name (in the "File name" box) can be appropriately specified. The file type (i.e., RCT, as in the "Files of type" box) cannot be changed. By pressing the "Open" button, the program proceeds with opening the REACTOR (and POLY3/GIBBS) workspaces from the specified previously-saved *.RCT file.

SAVE_WORKSPACE

Description: 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, the user can always come back to exactly the state he had when he issued the SAVE command by simply giving a READ command.

After saving the REACTOR (and POLY3/GIBBS) workspaces on a file, one may leave the program and at a later time READ the file and continue from the saved state. Note that 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 has been enforced, very similar to that one should not SAVE after a STEP or MAP command! You may append several results by START_SIMULATE without destroying the previous results but SAVE will erase them all. To suspend some of the SIMULATE results, use the AMEND_RECORD command.

Syntax: SAVE_WORKSPACE <FILE NAME>

Options: `file name` -- A user-desired file name must be specified. The default extension of the `RCT` file is "`.RCT`", while the user can have any other extension as wished.

A "Save" window will pop up on the screen if a file name is not given after the command, so that the path (in the "Save in" box) and file name (in the "File name" box) can be appropriately specified, as shown in Figure 10-20. The file type (i.e., RCT, as in the "Files of type" box) cannot be changed. By pressing the "Save" button, the program proceeds with saving the REACTOR (and POLY3/GIBBS) workspaces in the specified *.RCT file. The user may also "Cancel" such a "Save" window session, and thus the current REACTOR (and POLY3/GIBBS) workspaces will not be saved.

However, if there is already a file with the same file name under the directory (as indicated by the Save in box), a warming message will pop up on the screen. If clicking on the "OK" button, the program will accordingly overwrite the current REACTOR (and POLY3/GIBBS) workspaces onto the existing *.RCT file. If clicking on the "Cancel" button, the program returns to the REACTOR module; nothing will be saved, and the previous content will not be overwritten. Later on, the user may choose save the workspaces onto a file in another "Save in" path or as a different "File name"; the user may also use the SAVE command with an unspecified name (i.e., through SAVE ,,, command-sequence) to save the REACTOR (and POLY3/GIBBS) workspaces.

SHOW_PROBE

Description: This command is to show the details of the created probe, but has not been implemented yet.

Syntax: SHOW_PROBE

START_SIMULATION

Description: This command starts the reactor simulation, and lists the results for all iterations. The output for each iteration will consist of the conditions set in each segment, and the user may also select some state variables to be listed out. After each loop, the temperatures in all segments are listed.

Syntax: START_SIMULATION

Prompt: `Max number of loops /10/: <the max number of loops>`

Give the max number of loops. The REACTOR module will repeat the reactor simulation till 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 <RETURN>) or saving them on an experimental file (with the default extension .EXP). Note the conditions in each segment will not be saved in the file, only on the screen.

`Output variables /T BP($)/: <variable name(s)>`

Specify the variable name(s) that will also be listed on the screen together with the conditions in each segment, or in the output experimental file.

Example output: For a 4-stage reactor, the output at a certain iteration may look like below:

```
 ====================================
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
```

10 SYSTEM_UTILITIES commands

ABOUT

Description: This command gives some basic information about the development history and ownship of the Thermo-Calc and DICTRA software.

Syntax: ABOUT

CLOSE_FILE

Description: A previously opened text file is closed by this command.

Syntax: CLOSE_FILE

Prompt: UNIT NUMBER: <unit number>

A unit number previously given by the program in an OPEN_FILE command must be specified.

DISPLAY_LICENSE_INFO

Description: This command uses a license diagostics program implemented in the SYS module to detect the license information for the Thermo-Calc installations on the current client computer (and on the connected license server), and lists out all the details of such license information, including the current status of available license file and all its included licensefeatures for the current installation(s), as well as the Safenet Environment Variables and detailed Local (Client) Information, on screen or onto a textual file (*.TXT).

This feature is useful for having a comprehensive understanding of your license details, and for obtaining basic technical supporting services when there would eventually be an installation/license problem (by simply sending such an *.TXT file back to our support@thermocalc.se account).

Syntax: DISPLAY_LICENSE_INFO

Prompt: OUTPUT TO SCREEN OR FILE /SCREEN/: <MyLicenseInfo.TXT>

A list of detailed license information will be written on screen (by pressing <RETURN>) or onto a listing *.TXT file under a specific file name which can later on be opened and edited by any simple textual editor.

If you are running a network client installation of Thermo-Calc, you can see how many of the client instance licenses that are currently “checked out” and how many of them that are left. You can also see which clients computers it is that have checked these client instance licenses out.

HP_CALCULATOR

Description: This is the QBA simple interactive calculator using reverse polish notations.

Syntax: HP_CALCULATOR

Notes: All available OPCODEs (HPC codes) are listed below:

One must use the BACK command to quit the HP_CALCULATOR, in order to go back to the SYS module.

Input are numbers, + - * / and ^ and OPCODEs. Use HELP to list 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

Description: A text file can be opened by this command for uses in other commands where a unit number is necessary. The program will automatically assign a unit number.

Syntax: OPEN_FILE

Prompt: FILE NAME: <file name>

A legal file name must be specified.

SET_COMMAND_UNITS

Description: 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 by the OPEN_FILE command.

Note that the first two lines of an input file will be skipped before any input is read from an input file.

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 will be 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

Description: This command is specially designed for uses within MACRO files of the Thermo-Calc Console Mode (and DICTRA) software. By simply adding this command at the every beginning of a MACRO `*.TCM` file [or of the primary MACRO file on the top level if any sub-level(s) of MACRO files are used], it is very convenient and extremely useful to automatically show up on screen the complete details of various commands in all the sequential operations in the Thermo-Calc Console Mode (and DICTRA) software, that are enforced according to the MACRO file(s). When MACRO files are not used, it is irrelevant to apply this command.

Syntax: SET_ECHO

SET_INTERACTIVE_MODE

Description: This command resets the input and output units to their initial values, i.e., keyboard and screen. Remember to add this command as the last command into your MACRO files, so that you will be able interact with the software after executing the MACRO commands.

Syntax: SET_INTERACTIVE_MODE

SET_LOG_FILE

Description: This command makes it possible to save, on a simple text file, everything the user types 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 is wrong.

This command will also make 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 for a new user to follow.

The saved log file can then be edited as a MACRO file by using a simple textual editor. This is very useful to run the MACRO file(s) for similar calculations (the same system but slightly varied temperature-pressure-composition conditions), or run the Thermo-Calc Examples MACRO files.

Syntax: SET_LOG_FILE

A “Save” window will be popped up on the screen for specifying a file name for the *.LOG file, so that the path (in the “Save in” box) and file name (in the “File name” box) can be properly specified. If an *.LOG file with the same name exists in the current working directory, it will be overwritten. The file type (i.e., LOG, as in the “Files of type” box) cannot be changed. Press the “Save” button to proceed with saving all the subsequent commands in the specified *.LOG file, or the “Cancel” button to abandon this action.

SET_PLOT_ENVIRONMENT

Description: This command allows a user, preferably in his initiation file `tc.ini`, to set the plot devices he normally has access to as this may vary at different installations of Thermo-Calc.

Note that this command must be terminated with an empty line or two commas.

Syntax: SET_PLOT_ENVIRONMENT

Prompt: `DEFAULT PLOTDEVICE NUMBER /default number/: <device number>`

The number given here 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 "?" will always give 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 the user is 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

Description: This command lets the user pre-define the "default" values of three general Thermo-Calc options for performing all the sequential single-points, stepping and mapping calculations.

Syntax: SET_TC_OPTIONS

Prompt: `USE_GLOBAL_MINIMIZATION /YES/: <Y or N>`

The user may choose to answer `Y` (Yes) or `N` (No), to decide if using Global Minimization Technique from start of calculation or not. By answering `Y` (Yes), the Global Minimization will be done when possible (depending on the conditions setting), and Global Minimization test is always performed when an equilibrium is reached. This costs more computer time but the calculations are more robust. By answering `N` (No), the Global Minimization is not used to reach the equilibrium.

`GLOBAL_TEST_INTERVAL /0/: <an integer value>`

The user-supplied 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 will perform Global Minimization test at every *n*th step during STEP and MAP calculations. Of course, the Global Minimization test will always be 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 the ADD_INITIAL_EQUILIBRIUM command 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 indepedent axis-variables have already been defined by the POLY command

SET_AXIS_VARIABLE. With the Y (Yes) option, the POLY module will always enforce the automatic procedures for adding initial equilibrium points during mapping; with the N (No) option, no initial equilibrium points will thus be added during mapping.

SET_TERMINAL

Description: This command is now obsolete. It was previously used to define the characteristics of the terminal. Nowadays, users who are connected through a packet-switching network will normally have the echo provided from their PAD, and are thus not interested by a second echo from the host also.

Syntax: SET_TERMINAL

Prompt: Number of lines shown before stop on full screen /0/: <number>

Specify a number of the lines you wish to show before stop on full screen.

XON/XOFF enabled? /Y/: <YES or NO>

Give your answer (Y or N).

Use EMACS like line editor? /N/: <YES or NO>

Please note that such an editor is ONLY implemented on VMS machine. So you should simply press <RETURN> to accept the default answer NO.

Are you using packet-switched networks "X.25"? /N/: <Y or N>

Give your answer (Y or N).

STOP_ON_ERROR

Description: This command is useful in batch jobs in order to prevent that an erroneous sequence of commands to the program will cause waste of computer activity. If the argument ON is given, the program will terminate after an illegal or ambiguous command. It is possible to reset the effect of this command by giving the argument OFF. Default value is ON.

Syntax: STOP_ON_ERROR <ARGUMENT>

Give the preferred argument (ON or OFF) after the STOP command.

11 TABULATION_REACTION commands

ENTER_FUNCTION

Description: This command can be used to define a new thermodynamic function to be included in the sixth column of tabular output. The last entered function will be 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.

A limited number of state variables, G, H, S, T, P, V and H298 can be used in the function definition. For instance, -(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.

Syntax: ENTER_FUNCTION

Prompt:
```
Name: <name>
Function: <definition>
```

Options: name -- name of the function (maximum 8 characters)
definition -- 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 <RETURN> in the prompt "Output file /SCREEN/:", and the figure by typing 6 in the prompt "Plot column /2/:":

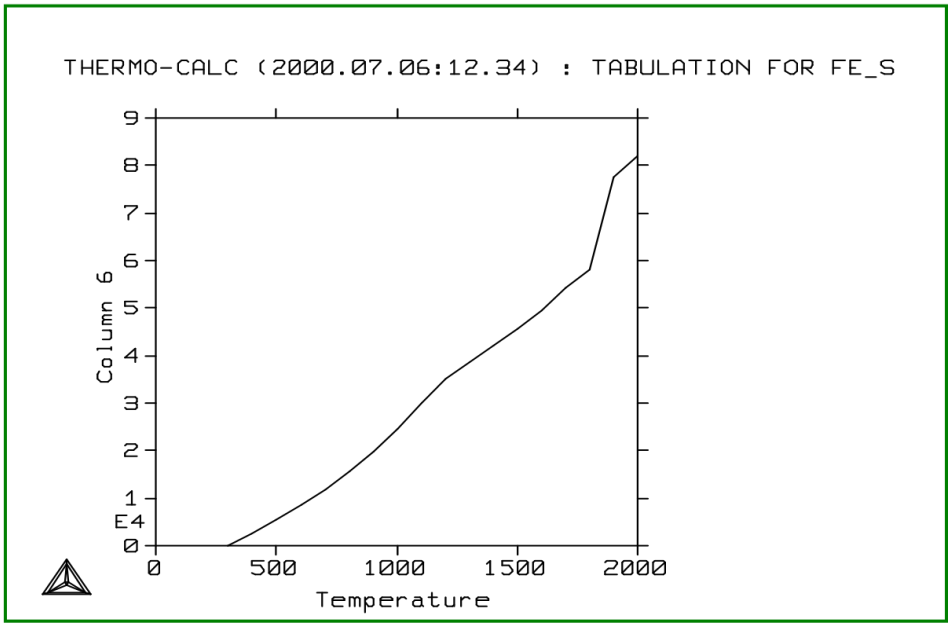
```
TAB: ENTER-FUNCTION
Name: InEnergy
Function: G+T*S-P*V;
TAB: TABULATE_SUBSTANCE
Substance (phase): Fe
Pressure /100000/: <RETURN>
Low temperature limit /298.15/: <RETURN>
High temperature limit /2000/: <RETURN>
Step in temperature /100/: <RETURN>
Output file /try1/: try2
Graphical output? /Y/: <RETURN>
```

```
Plot column? /2/: 6
```

This will give the following output:

```
                O U T P U T  F R O M  T H E R M O - C A L C
                2012.10.22                              12.31. 3
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
```

The following graph resulted from the TAB module, showing the relation between the entered function `G+T*S-P*V` (*U*, *i.e.,* column 6 on the resulted table) and temperature (K) for the substance Fe (data retrieved from the SSUB database). The graph can be refined in the POST Module:



ENTER_REACTION

Description: This command is identical to TABULATE_REACTION 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 the SWITCH_DATABASE command (thus there is no need to use the DATA module before this command). The user may select any other database by using the SWITCH_DATABASE command in the TAB module. The rules for specifying a reaction are the same as in the command TABULATE_REACTION.

Syntax: ENTER_REACTION

Prompt:

```
Same reaction? /Y/: <Y or N>
```

This prompt appears only if there is at least one reaction that has already been defined either by the command ENTER_REACTION or TABULATE_REACTION. By answering Y (Yes), the next prompt for defining reaction will not appear.

```
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 typing <RETURN>, 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 will be both shown on screen and saved as an `*.EXP` or an `*.TAB` file, and the program will further prompt the following questions:

```
Graphical output /Y/: <Y or N>
```

If answering `N` (No), the program will create the table output on the screen, and also save the same table as a simple textual file with the default extension `.TAB` under the current working directory. In this case, the graph cannot be plotted.

If answering `Y` (Yes), the program will both create a table with all thermodynamic functions as normal (which will be shown on screen) and generate a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which will be plotted on screen and saved as an `*.EXP` file), and will further ask for which column 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 (*verse* 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 will be 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 will appear on the screen as usual. The plot will then appear 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 will always take you back to the TAB module.

LIST_SUBSTANCES

Description: This command makes it possible to list all species in the current database with a certain set of elements. This is useful for specifying species as pure substances in the `TABULATE_SUBSTANCE` command.

Syntax: LIST_SUBSTANCES

Prompt:

```
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 (no phases!) in the current database will be searched. If the answer to this question is Y (the default), no other elements will be allowed in searching, except for those specified. If N is given as the answer, all species containing at least one of the specified elements will be listed.

Example output: For the SSOL5 database, various options gives different lists, as given below:

```
TAB: l-sub
With elements /*/: <RETURN>
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
Exclusivly with those elements? /Y/: <RETURN>
C                        CR                       FE
N                        NI                       O
C1                       C2                       C3
C4                       C5                       C6
C7                       N2                       O2
TAB:l-sub fe cr ni c n o
Exclusivly 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

Description: This command sets 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 <UNIT>

Prompt: `Energy unit (C=Cal, J=Joule) /J/: <unit>`

Options: `unit` -- C (calories) or J (joule)

SET_PLOT_FORMAT

Description: This command sets the plotting format while the result is plotted on screen or the file is saved as `EXP` file [using DATAPLOT graphical format; please refer to *Chapters 12, 13* and *15* in this *Thermo-Calc (Concole Mode) User's Guide*] and `TAB` file (as a simple text file and as shown on screen). Note that this command is different from the `SET_PLOT_FORMAT` command in the POST module, but is identical to the command `SET_PLOT_ENVIRONMENT` command in the SYS (system utility) monitor.

With this command, the user can 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 your `TC.INI` file.

Syntax: SET_PLOT_FORMAT <UNIT>

Options: `unit` -- the default set unit is `1` (MS-Windows) or `9` (Linux) for graphical presentation on screen, or all other units for outputs as `*.EXP` and `*.TAB` files.

SWITCH_DATABASE

Description: By default, the TAB module will always automatically retrieve 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, the user may select thermodynamic data for complex non-ideal solution phases from another database. This is possible by using the `SWITCH_DATABASE` 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 <RETURN> key without giving any argument. The user can also supply his own database by first 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` for substances, and `ENTER_REACTION` or `TABULATE_REACTION` for reactions.

When a predefined database or an appropriate USER database has been switched on, the DATA command GET_DATA (*Section 5.4.11*) is automatically executed; therefore, the TAB module will immediately show on screen a message like below (in case the PSUB database is 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 <NEW DATABASE NAME>

Prompt: `Database /XYZ/: <new database name>`

Specify an appropriate database name. XYZ stands for the default SSUB database or the current database which has previously switched on. If an appropriate database name is not given and the USER option is not used, the following list will shown up:

```
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>
```

Options: `new database name` -- the appropriate 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 your convenience when switching/appending your own databases or lately purchase databases from TCS or its agents, you can simply add them to the predefined database list in the database initiation file `TC_INITD.TDB` or `initd.tdb` of your installed Thermo-Calc/DICTRA packages.

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.

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 will be popped up for specifying the filename of the database setup file name (`*setup.TDB`) of the to-be-switched USER database.

Please note that unlike the situation in the DATA module, if a USER database is used, the Gibbs energy system will also be reinitialized and only data from the USER databases will be retrieved for tabulations.

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.

TABULATE_DERIVATIVES

Description: This command is mainly for system debugging purposes and for experienced users. It will automatically calculate all partial derivatives for a pure phase or solution phase at a given composition. Note that this is not the same as the chemical potential!

Before this command, the user must have already defined the system and gotten thermodynamic data from an appropriate solution database in the DATA module before using this 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, 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 will then optionally go through a specific sublattice or all the possible sublattices for the specified phase, and will go through 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), when it prompts for inputs of site fractions for various species possibly occupying such sublattices. Therefore, one 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).

Syntax: TABULATE_DERIVATIVES

Prompt: `Phase Name /XXXX/: <name of the phase>`

Specify the phase name of a pure phase or a solution phase, *e.g.,* FCC, CEMENTITE, LIQUID, SLAG, AQUEOUS, GAS, Fe-S, Wustite, Fe2O3-Hematite, *etc*.

XXXX is the last phase the TAB module accounted for. It will automatically prompt for other options and necessary inputs for the definition of the composition of the specified phase.

For a pure phase (such as `Fe-S`, `Wustite`, and `Fe2O3-Hematite`), no further composition definition is needed. For a solution phase 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 will automatically be assigned as the rest. Note that the sum of input site fractions must not exceed unity. For instance, 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 will be 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 solution phase with two or more sublattices (such as `FCC` alloy solution, and `ION_LIQ` ionic liquid solution), it will first ask 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 will prompt 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 will automatically be assigned as the rest. For instance, 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` will be 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/: <RETURN>
 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/: <RETURN>
Pressure /100000/: <RETURN>
 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
 First 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

Description: With this command, thermodynamic property changes for a chemical reaction can be tabulated 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). The user may select any other database by using the `SWITCH_DATABASE` command in the TAB module.

It is very important to keep the rules for specifying a reaction in mind:

1. The reaction is specified, by giving the reactants and products, such as `Ga+S=GaS`, `5Al+2Fe=Al5Fe2`.
2. 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`.
3. Elements with a single letter symbol must be followed by a stoichiometry factor, even if it is 1; thus, the symbol `CO` will be taken as cobalt, carbon monoxide must be given as `C1O1` (or `O1C1`).

4. All the reactants and products must have been 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 instance, if a database does not have a species defined with a stoichiometry of `Al5Fe2` {*e.g.,* the `Al` and `Fe` (but not `Al5Fe2`) may have been defined as species, and the `Al5Fe2` phase may have been 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.
5. One should make sure that the Gibbs free energy expression for a stoichiometric phase is really corresponding to the species-stoichiometry. For instance, if a database has a definition of `Function GAL5FE2` for `1/7` of the `AL5FE2` formula, then the relation `G(Al5Fe2)=7*GAL5FE2` must have been defined; otherwise, the tabulated thermodynamic properties for the reaction `5Al+2Fe=Al5Fe2` will be wrong.

Syntax: TABULATE_REACTION

Prompt: `Same reaction? /Y/: <Y or N>`

This prompt appears if there is at least one reaction already defined either by the command ENTER_REACTION or TABULATE_REACTION. If you answer Y, then the next prompt for defining reaction will not appear.

```
 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> 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 typing `<RETURN>`, 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 will be both shown on screen and saved as an `*.EXP` or a `*.TAB` file, and the program will further prompt the following questions:

```
Graphical output /Y/: <Y or N>
```

If answering `N` (No), the program will create the table output on the screen, and also save the same table as a simple textual file with the default extension `.TAB` under the current working directory. In this case, the graph cannot be plotted.

- If answering `Y` (Yes), the program will both create a table with all thermodynamic functions as normal (which will be shown on screen) and generate a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which will be plotted on screen and saved as an `*.EXP` file), and will further ask for which column to be plotted on the resulting graph:

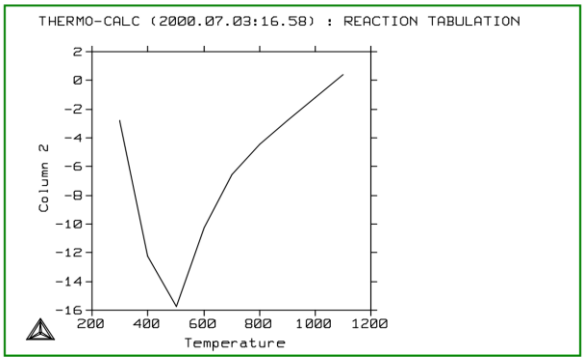
```
Plot column ? /2/: <1 or 2 or 3 or 4 or 5 or 6>
```

Specify which property (as column number) to be plotted as the Y-axis (*verse* 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 will be 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 will appear on the screen as usual. The plot will then appear 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 will always take you back to the TAB module.

Example output: For the reaction `Ga+S=GaS` (using the SSUB5 database), the following table is obtained by typing `<RETURN>` 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
               2012.10.22                          16.54. 0
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
 Temperature range exceeded for GA1S1
```



The graph resulted from the TAB module, showing the relation between the heat capacity change (ΔC_p, *i.e.*, column 2 on the resulted table) and temperature (K) for the reaction `Ga+S=GaS` (data retrieved from the SSUB database). The graph can be refined in the POST Module.

TABULATE_SUBSTANCE

Description: This command can be used to tabulate various 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 TCFE7), or of a solution phase with a fixed composition (getting data from a specific solution database such as SSOL5 and TCFE7), under a constant pressure and various temperatures.

In case of a substance, the default database is set as the SSUB database. The user can use the SWITCH_DATABASE command in either the DATA or TAB module to set an appropriate database as the current database for subsequent tabulations. The notation for a chemical formula of a substance treats upper and lower case as the same, thus elements with a single letter symbol must be followed by a stoichiometric factor, even if it is 1. The symbol CO will be taken as cobalt, carbon monoxide must be written as C1O1 (or O1C1). It does not matter how you arrange the order of the elements when you type the chemical formula (*e.g.,* calcite as CaC1O3 or CaO3C1 or C1O3Ca or C1CaO3 or O3C1Ca), since the TAB module will search for the available species with the defined elements and their stoichiomtric coefficients through all the species in the current database. If the specified substance is not valid species in any of available phases in the current database, a warning message appears on screen (when retrieving data) about what is incomplete or inappropriate, and thus the tabulation will fail. For this reason, the command does not work for charged species, *e.g.,* H1+1, H1-1, H2+1 *etc*. in GAS phase, Fe1/+2, Cr1/+3, O1/2, *etc*. in LIQUID phase, H+1, OH-1, Fe+2, *etc*. in AQUEOUS phase. Note that the module will automatically find out in what form (phase) the specified substance is the most stable under a certain temperature range at the given pressure, and thus the tabulated thermodynamic data are always for the most stable substances under various temperatures.

In case of a pure stoichiometric phase or a solution phase with a fixed composition, the user must have already defined the system and gotten 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 will then optionally go through a specific sublattice or all the possible sublattices for the specified phase, and will go through 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), when it prompts for inputs of site fractions for various species possibly occupying such sublattices. Therefore, one 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 for specifying an appropriate substance or solution database (*e.g.,* typing `SSUB2`, `SSUB3`, `SSUB4`, `SSUB5`; `SSOL2`, `SSOL4`, `SSOL5`; `TCFE3`, `TCFE4`, `TCFE5`, `TCFE6`, `TCFE7`, *etc.*). Prior to this command, the user can also use the `SWITCH_DATABASE` 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.

Note that before other Prompts, a list of the used database, defined elements, getting-data sequence, references, *etc*. is shown on screen. From such information, the user would know about 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 very important that the user must 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 will automatically prompt 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 will automatically be assigned as the rest. Note that the sum of input site fractions must not exceed unity. For instance, 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 will be 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 inter-metallic 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 inter-metallic solution phase), it will first ask 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 will prompt 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 will automatically be assigned as the rest. For instance, the following 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 will be 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
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 typing <RETURN>, 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 will be both shown on screen and saved as an *.EXP or a *.TAB file, and the program will further prompt the following questions:

```
Graphical output /Y/: <Y or N>
```

If answering N (No), the program will create the table output on the screen, and also save the same table as a simple textual file with the default extension .TAB under the current working directory. In this case, the graph cannot be plotted.

If answering `Y` (Yes), the program will both create a table with all thermodynamic functions as normal (which will be shown on screen) and generate a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which will be plotted on screen and saved as an `*.EXP` file), and will further ask for which column 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 (*verse* 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 will be 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 will appear on the screen as usual. The plot will then appear 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 will always take you back to the TAB module.

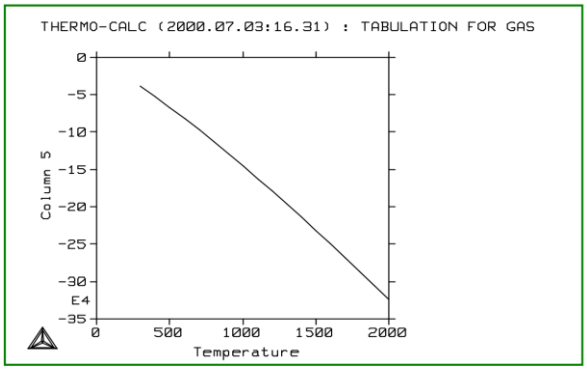
Example Output 1: For the `FCC` alloy phase as a non-ideal solution with a fixed composition

```
[Fe0.80,Cr0.10,Ni0.0995,O0.005][C0.05,N0.05,VA0.90]
  (using the SSOL2 database), the following table is
obtained by typing <RETURN> 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
  2012.10.23                              11.55.46
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
```

Note that for a phase, as either an end-member or real solution, the listed species name is irrelevant!

Example Output 2: For the H2 species as a pure substance (using the SSUB5 database), the following table is obtained by typing <RETURN> 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
                2012.10.23                              14. 5.26
 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
 TAB:
```



The graph resulted from the TAB module, showing the relation between Gibbs energy (*i.e.,* column 5 on the resulted table) and temperature (K) for the pure substance H2 (data retrieved from the SSUB database). The graph can be refined in the POST Module.

12 Appendix – 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 one edits 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 simple text file and can be opened and edited by any text editor. For this reason, an *.POP or *.DOP file is not hardware dependent, and can thus be used by Thermo-Calc or DICTRA in any CPU platform.

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 a ***syntax checker*** when the PARROT command COMPILE_EXPERIMENT 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 appendix can only be used in the *.POP or *.DOP files.

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
- CHANGE_STATUS
- SET_CONDITION
- SET_REFERENCE_STATE
- ENTER_SYMBOL
- SAVE_WORKSPACE

Note that: as the last command in an *.POP or *.DOP file, one must always have the SAVE command.

Other legal commands from the POLY module that are used less often are:

- DEFINE-COMPONENTS
- EVALUATE-FUNCTION
- SET-ALL-START-VALUES
- SET-NUMERICAL-LIMITS
- SET-START-VALUE
- ADVANCED-OPTIONS

Note that the DEFINE-COMPONENTS command must be always used as the first command in an *.POP or *.DOP file, as it will automatically reinitiate the whole workspaces.

Most of the special commands for the ED_EXP module are also often used in an *.POP or *.DOP file. For example:

- EXPERIMENT
- EXPORT
- IMPORT
- LABEL-DATA
- SET-ALTERNATE-CONDITION
- SET-WEIGHT

However, 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 the Thermo-Calc/DICTRA software family, *e.g.,* in *.TDB, *.DAT, *.TCM, *.POP/*.DOP and *.EXP files. Such a sign and all information afterwards in the same comment line will be 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.

There are only a few other commands that can be used only in the POP or DOP file. These are described below.

TABLE_HEAD, TABLE_VALUES and TABLE_END

Description: 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.

Syntax: TABLE_HEAD <N>

TABLE_VALUES

TABLE_END

Comments: 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. Note that 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.

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 will be used to generate unique numeric identifiers for all the equilibria in the table. The numeric identifier will be incremented by one for each experimental (equilibrium) point in the table. An example of the use of a table for enthalpy measurements in the liquid is given here. Much more elaborate tables can be used.

```
$ Enthalpies of mixing, Topor and
Kleppa, Met Trans 1984  TABLE-HEAD 1

CREATE_NEW @@ 1

CHANGE-STATUS PHASE LIQ=FIX 1
```

```
SET-REFERENCE-STATE AU LIQ * 1E5
SET-REFERENCE-STATE CU LIQ * 1E5
SET-CONDITION P=1E5 T=1379
X(LIQ,AU)=@1
LABEL ALH
EXPER HMR=@2:5%
TABLE_VALUES
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 the following details:

`@@` This is used for the purpose of automatically generating a series of identifying numbers, instead of just creating only one identifying number, at the `CREATE-NEW-EQUILIBRIUM` command. This means that the program will automatically generate the identifying numbers starting after the number given after the `TABLE-HEAD` command (in this case it is `1`).

`@1` Values that are in the table are specified by a `@` followed by a column number. In the above case, the mole fraction of Au in the liquid is thus in column 1 (see the line for `SET-CONDITION`). The columns may have other information than values, and one may 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"

A peculiarity of the *syntax checker* is also that 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`, `@2`, *etc.* One may have columns that are not used but there must be exactly the same number of columns of each line, otherwise the syntax checker will give an error message in the PARROT module.

COMMENT

Description: 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 <COMMENT>

A comment can have maximum about 60 characters, written on the same line.

FLUSH_BUFFER

Description: This command is only used in the *.POP/*.DOP files. It is needed if the number of experiments require more space than they can be fitted into one buffer. When the PARROT program compiles experiments, it will inform if any FLUSH commands are needed. The FLUSH command will terminate the current block, save it to the work file and start a new block. With the READ command in the EDIT-EXP module, one can select the block to edit.

After a FLUSH command, the workspace is reinitiated and all functions or constants must be entered again in the *.POP file. One may take advantage of this; for instance, one can use the FLUSH command to have blocks with different components in one *.POP file.

Syntax: FLUSH_BUFFER

GRAPHICS_PLOT

Description: This command create one experimental data file directly from a POP file during compiling in the PARROT module. Several datasets and different symbols are also allowed according to user's specifications. It is automatically executed when the POP file has been compiled in the PARROT module.

Syntax: GRAPHICS_PLOT <DATASET#> <X VALUE> <Y VALUE> <SYMBOL INDEX>

Comments: Please note that this command is now totally irrelevant and unable to use in the ED-EXP module, even the command is still visible on the command list.