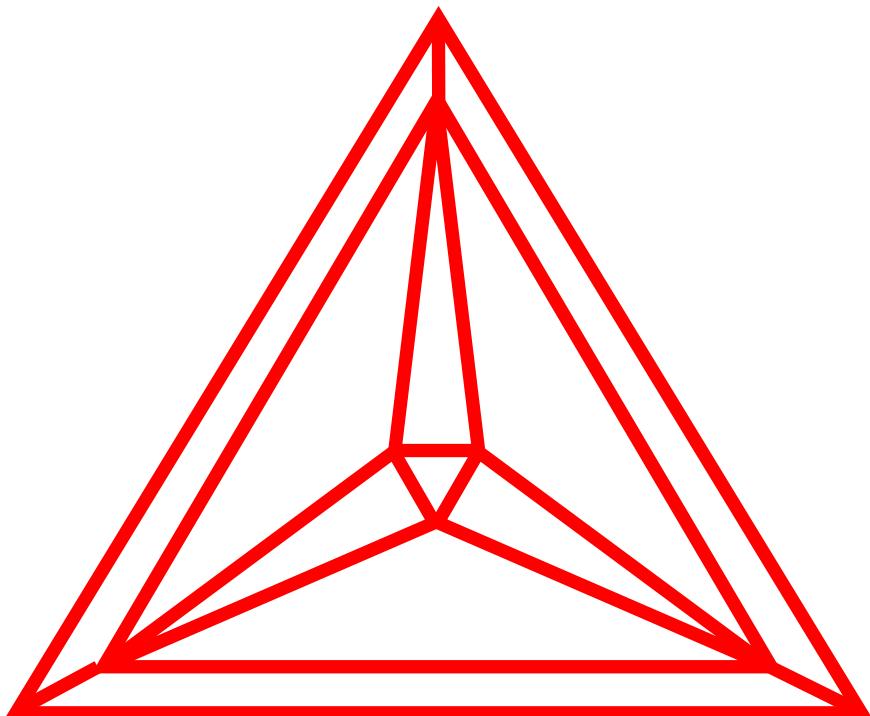


TCCTM

Thermo-Calc® Software

User's Guide

Version S



**Thermo-Calc Software AB
Stockholm Technology Park
Björnnäsvägen 21
SE-113 47 Stockholm, Sweden**

Copyright © 1995-2008 Foundation of Computational Thermodynamics
Stockholm, Sweden

Copyright:

The Thermo-Calc® (TCC™ and TCW™) and DICTRA® software are the exclusive copyright properties of the STT (Foundation of Computational Thermodynamics, Stockholm, Sweden) and TCS (Thermo-Calc Software AB, Stockholm, Sweden). All rights are reserved worldwide!

Thermo-Calc Software AB has the exclusive rights for further developing and marketing all kinds of versions of Thermo-Calc® and DICTRA® software/database/interface packages, worldwide.

This **TCCS (Thermo-Calc Classic version S) User's Guide**, as well as all other related documentation, is the copyright property of Thermo-Calc Software AB.

It is absolutely forbidden to make any illegal copies of the TCS-provided software, databases, programming interfaces, and their manuals (User's Guide and Examples Book) and other technical publications (Reference Book and Technical Information). Any unauthorized duplication of such copyrighted products, is a violation of international copyright law. Individuals or organizations (companies, research companies, governmental institutes, and universities) that make or permit to make unauthorized copies may be subject to prosecution.

The utilization of the Thermo-Calc® (TCC™ and TCW™) and DICTRA® software and associated database and programming interfaces, as well as their manuals and other technical information, are extensively and permanently governed by the **Thermo-Calc Software END USER LICENSE AGREEMENT (TCS-EULA)**, which is connected with the software.

Disclaimers:

Thermo-Calc Software AB and STT (Foundation of Computational Thermodynamics, Stockholm, Sweden) reserve the rights to further developments of the Thermo-Calc® (TCC™ and TCW™) and DICTRA® software and associated database and programming interface products, and to revisions of their manuals and other publications, with no obligation to notify any individual or organization of such developments and revisions. In no event shall Thermo-Calc Software AB and the STT Foundation be liable to any loss of profit or any other commercial damage, including but not limited to special, consequential or other damage.

There may be some minor differences in contents between this **TCCS User's Guide** and the actual appearance of the program as seen on the screen when running the **TCCS (Thermo-Calc Classic, version S)**. This is because that some of the contents may need to be updated in the program's on-line help and in the future release of the program. Please visit the Thermo-Calc Software web site (www.thermocalc.com) for any patch (with modifications and/or improvements that have been incorporated into the program and its on-line help), or any amendment that have made to the content of the User's Guides and Examples Books, or to the FAQ lists and other technical information publications.

Acknowledgement of Copyright and Trademark Names:

Various names that are protected by copyright and/or trademarks are mentioned for descriptive purposes, within this **TCCS User's Guide** and other documents of the Thermo-Calc® (TCC™ and TCW™) and DICTRA® software/database/programming-interface packages. Due acknowledgement is herein made of all such protections.

Availability of This Document:

For the purpose of environment-friendlessness, this **TCCS User's Guide** and all other operational manuals (User's Guides and Examples Books), as well as Reference Lists and other technical documentations, for the TCS-provided software, databases and programming interfaces are provided along the delivered TCS Standard Products CDs and installed on each of designated installation, which can be reviewed and accessed easily and conveniently. If desired and preferred, a user can locally print such a manual BUT it is only for the purpose of the user's internal use. For a hard copy of such a manual physically printed and delivered by TCS, a certain fee shall be applied.

To make manual updating more prompt and efficient, the later manual revisions or additions will be made available on the Internet. Our users may therefore download such revised documents from our company's web site www.thermocalc.com.

Editors of This Document:

Dr. Pingfang Shi
Thermo-Calc Software AB (TCSAB)
Stockholm Technology Park
SE-113 47 Stockholm, SWEDEN
E-Mail: pingfang@thermocalc.se

Prof. Bo Sundman
Dept. of Materials Science & Engineering
Royal Institute of Technology (KTH)
SE-100 44 Stockholm, SWEDEN
E-Mail: bosse@mse.kth.se

Contents

CONTENTS	I
CHAPTER 1 GENERAL INTRODUCTION	1-1
1.1 COMPUTATIONAL THERMODYNAMICS AND KINETICS	1-1
1.2 THERMO-CALC SOFTWARE/DATABASE/INTERFACE PACKAGE	1-1
1.3 ACKNOWLEDGEMENTS	1-6
1.4 REVISION HISTORY	1-6
1.5 GENERAL STRUCTURE OF THE THERMO-CALC PACKAGE	1-21
1.6 AVAILABILITY OF THE THERMO-CALC PACKAGE ON VARIOUS HARDWARE.....	1-23
1.7 BENEFITS OF USING THE THERMO-CALC PACKAGE	1-25
CHAPTER 2 HOW TO BECOME A THERMO-CALC EXPERT	2-1
2.1 HOW TO USE THIS USER'S GUIDE	2-1
2.2 HOW TO INSTALL AND MAINTAIN THE THERMO-CALC PACKAGE	2-5
2.2.1 <i>License requirements</i>	2-5
2.2.2 <i>Installation procedures</i>	2-6
2.2.2.1 <i>Installing the entire TCC software/database package</i>	2-6
2.2.2.2 <i>Adding database(s) to an already installed TCC package</i>	2-7
2.2.3 <i>Maintaining current and previous versions</i>	2-8
2.2.4 <i>Making the TCC performance convenient for you</i>	2-9
2.3 HOW TO BECOME A THERMO-CALC EXPERT	2-12
2.3.1 <i>Get prompt technical supports from TCS and its agents worldwide</i>	2-12
2.3.2 <i>Make daily use of various Thermo-Calc functionality</i>	2-13
2.3.3 <i>Present your results in professional and high quality standards</i>	2-13
2.3.4 <i>Exchange your experience with others through varied channels</i>	2-14
CHAPTER 3 THERMO-CALC SOFTWARE SYSTEM	3-1
<i>[PLEASE GO TO THE DOCUMENT “THERMO-CALC SOFTWARE SYSTEM”]</i>	
CHAPTER 4 THERMO-CALC DATABASE DESCRIPTION	4-1
<i>[PLEASE GO TO THE DOCUMENT “THERMO-CALC DATABASE GUIDE”]</i>	
CHAPTER 5 DATABASE MODULE (TDB).....	5-1
5.1 INTRODUCTION	5-1
5.2 USER INTERFACE OF THE TDB MODULE	5-2
5.3 GETTING STARTED	5-3
5.3.1 <i>SWITCH_DATABASE</i>	5-3
5.3.2 <i>LIST_DATABASE ELEMENT</i>	5-3
5.3.3 <i>DEFINE_ELEMENTS</i>	5-3
5.3.4 <i>LIST_SYSTEM CONSTITUENT</i>	5-3
5.3.5 <i>REJECT PHASE</i>	5-3
5.3.6 <i>RESTORE PHASE</i>	5-3
5.3.7 <i>GET_DATA</i>	5-3
5.4 DESCRIPTION OF ALL AVAILABLE TDB MONITOR COMMANDS	5-4
5.4.1 <i>AMEND_SELECTION</i>	5-4
5.4.2 <i>APPEND_DATABASE</i>	5-4

5.4.3	<i>BACK</i>	5-5
5.4.4	<i>DATABASE_INFORMATION</i>	5-5
5.4.5	<i>DEFINE_ELEMENTS</i>	5-6
5.4.6	<i>DEFINE_SPECIES</i>	5-6
5.4.7	<i>DEFINE_SYSTEM</i>	5-6
5.4.8	<i>EXCLUDE_UNUSED_SPECIES</i>	5-7
5.4.9	<i>EXIT</i>	5-7
5.4.10	<i>GET_DATA</i>	5-7
5.4.11	<i>GOTO_MODULE</i>	5-8
5.4.12	<i>HELP</i>	5-8
5.4.13	<i>INFORMATION</i>	5-8
5.4.14	<i>LIST_DATABASE</i>	5-10
5.4.15	<i>LIST_SYSTEM</i>	5-10
5.4.16	<i>MERGE_WITH_DATABASE</i>	5-10
5.4.17	<i>NEW_DIRECTORY_FILE</i>	5-10
5.4.18	<i>REJECT</i>	5-11
5.4.19	<i>RESTORE</i>	5-12
5.4.20	<i>SET_AUTO_APPEND_DATABASE</i>	5-13
5.4.21	<i>SWITCH_DATABASE</i>	5-13
5.5	EXTENDED COMMANDS.....	5-15

CHAPTER 6 DATABASE MANAGER'S GUIDE..... 6-1*[PLEASE GO TO THE DOCUMENT “THERMO-CALC DATABASE GUIDE”]***CHAPTER 7 TABULATION MODULE (TAB)..... 7-1**

7.1	INTRODUCTION.....	7-1
7.2	GENERAL COMMANDS.....	7-2
7.2.1	<i>HELP</i>	7-2
7.2.2	<i>GOTO_MODULE</i>	7-2
7.2.3	<i>BACK</i>	7-2
7.2.4	<i>EXIT</i>	7-3
7.2.5	<i>PATCH</i>	7-3
7.3	IMPORTANT COMMANDS	7-4
7.3.1	<i>TABULATE_SUBSTANCE</i>	7-4
7.3.2	<i>TABULATEREACTION</i>	7-8
7.3.3	<i>ENTERREACTION</i>	7-10
7.3.4	<i>SWITCH_DATABASE</i>	7-11
7.3.5	<i>ENTER_FUNCTION</i>	7-12
7.3.6	<i>TABULATE_DERIVATIVES</i>	7-14
7.3.7	<i>LIST_SUBSTANCES</i>	7-17
7.4	OTHER COMMANDS.....	7-18
7.4.1	<i>SET_ENERGY_UNIT</i>	7-18
7.4.2	<i>SET_PLOT_FORMAT</i>	7-18
7.4.3	<i>MACRO_FILE_OPEN</i>	7-18
7.4.4	<i>SET_INTERACTIVE</i>	7-21
7.5	PLOTTING OF TABLES	7-22

CHAPTER 8 EQUILIBRIUM CALCULATION MODULE (POLY)..... 8-1

8.1	INTRODUCTION.....	8-1
8.2	GETTING STARTED	8-3
8.3	BASIC THERMODYNAMICS.....	8-4
8.3.1	<i>System and phases</i>	8-4
8.3.2	<i>Species and constituents</i>	8-4

8.3.3	<i>Standard state variables and partial derivatives</i>	8-5
8.3.3.1	<i>Intensive properties of the system, or of a component</i>	8-7
8.3.3.2	<i>Intensive properties of a species relative to a specific phase</i>	8-7
8.3.3.3	<i>Normalizing of extensive properties</i>	8-9
8.3.3.4	<i>Energetic extensive properties of the system or of a phase</i>	8-10
8.3.3.5	<i>Total size of the system</i>	8-11
8.3.3.6	<i>Amount of compounds in the system or in a phase</i>	8-11
8.3.3.7	<i>Amount of a phase in the system</i>	8-13
8.3.3.8	<i>Special quantities of a phase</i>	8-13
8.3.3.9	<i>Constitution of a phase</i>	8-14
8.3.3.10	<i>Partial derivatives (derived variables)</i>	8-14
8.3.4	<i>Components</i>	8-18
8.3.5	<i>Conditions</i>	8-19
8.4	DIFFERENT TYPES OF CALCULATIONS	8-20
8.4.1	<i>Calculating a single equilibrium</i>	8-20
8.4.2	<i>Stepping calculations of property diagrams</i>	8-21
8.4.3	<i>Solidification path simulations</i>	8-21
8.4.4	<i>Paraequilibrium and T_0 temperature simulations</i>	8-22
8.4.5	<i>Mapping calculations of phase diagrams</i>	8-24
8.4.6	<i>Potential diagram calculations</i>	8-25
8.4.7	<i>Pourbaix diagram calculations</i>	8-25
8.4.8	<i>Plotting of diagrams</i>	8-25
8.4.9	<i>From Gibbs Energy Minimization to Global Minimization</i>	8-25
8.5	DIAGRAMS.....	8-28
8.5.1	<i>Binary phase diagrams</i>	8-28
8.5.2	<i>Ternary phase diagrams</i>	8-28
8.5.3	<i>Quasi-binary and quasi-ternary phase diagrams</i>	8-31
8.5.4	<i>Higher order phase diagrams</i>	8-31
8.5.5	<i>Property diagrams</i>	8-32
8.6	GENERAL COMMANDS.....	8-33
8.6.1	<i>HELP</i>	8-33
8.6.2	<i>INFORMATION</i>	8-33
8.6.3	<i>GOTO_MODULE</i>	8-34
8.6.4	<i>BACK</i>	8-34
8.6.5	<i>SET_INTERACTIVE</i>	8-34
8.6.6	<i>EXIT</i>	8-34
8.7	BASIC COMMANDS	8-35
8.7.1	<i>SET_CONDITION</i>	8-35
8.7.2	<i>RESET_CONDITION</i>	8-36
8.7.3	<i>LIST_CONDITIONS</i>	8-37
8.7.4	<i>COMPUTE_EQUILIBRIUM</i>	8-37
8.7.5	<i>LIST_EQUILIBRIUM</i>	8-38
8.7.6	<i>DEFINE_MATERIAL</i>	8-39
8.7.7	<i>DEFINE_DIAGRAM</i>	8-42
8.8	COMMANDS TO SAVE AND READ THE POLY DATA STRUCTURE	8-44
8.8.1	<i>SAVE_WORKSPACES</i>	8-44
8.8.2	<i>READ_WORKSPACES</i>	8-46
8.9	COMMANDS TO CALCULATE AND PLOT DIAGRAMS	8-47
8.9.1	<i>SET_AXIS_VARIABLE</i>	8-47
8.9.2	<i>LIST_AXIS_VARIABLE</i>	8-48
8.9.3	<i>MAP</i>	8-48
8.9.4	<i>STEP_WITH_OPTIONS</i>	8-49
8.9.5	<i>ADD_INITIAL_EQUILIBRIUM</i>	8-56
8.9.6	<i>POST</i>	8-57
8.10	OTHER USEFUL AND ADVANCED COMMANDS	8-58
8.10.1	<i>DEFINE_COMPONENTS</i>	8-58

8.10.2	<i>SET_REFERENCE_STATE</i>	8-58
8.10.3	<i>CHANGE_STATUS</i>	8-59
8.10.4	<i>LIST_STATUS</i>	8-64
8.10.5	<i>COMPUTE_TRANSITION</i>	8-65
8.10.6	<i>SET_INPUT_AMOUNTS</i>	8-66
8.10.7	<i>SET_NUMERICAL_LIMITS</i>	8-67
8.10.8	<i>SET_ALL_START_VALUES</i>	8-68
8.10.9	<i>SET_START_VALUE</i>	8-69
8.10.10	<i>SET_START_CONSTITUTION</i>	8-69
8.10.11	<i>RECOVER_START_VALUES</i>	8-69
8.10.12	<i>ENTER_SYMBOL</i>	8-70
8.10.13	<i>LIST_SYMBOLS</i>	8-71
8.10.14	<i>DELETE_SYMBOL</i>	8-71
8.10.15	<i>SHOW_VALUE</i>	8-72
8.10.16	<i>EVALUATE_FUNCTIONS</i>	8-72
8.10.17	<i>TABULATE</i>	8-72
8.10.18	<i>LIST_INITIAL_EQUILIBRIA</i>	8-73
8.10.19	<i>LOAD_INITIAL_EQUIlibrium</i>	8-73
8.10.20	<i>DELETE_INITIAL_EQUILIBRIUM</i>	8-73
8.10.21	<i>AMEND_STORED_EQUILIBRIA</i>	8-73
8.10.22	<i>ADVANCED_OPTIONS</i>	8-75
8.10.24	<i>CREATE_NEW_EQUILIBRIUM</i>	8-85
8.10.25	<i>SELECT_EQUILIBRIUM</i>	8-85
8.10.26	<i>MACRO_FILE_OPEN</i>	8-86
8.10.27	<i>REINITIATE_MODULE</i>	8-88
8.10.28	<i>PATCH</i>	8-88
8.11	AQUEOUS SOLUTIONS.....	8-89
8.11.1	<i>Phase definitions and reference states for the aqueous phase</i>	8-89
8.11.2	<i>Connected models, databases and modules for the aqueous phase</i>	8-89
8.11.3	<i>Additional derived variables for the aqueous phase</i>	8-90
8.11.4	<i>Modification of output from LIST_EQUILIBRIUM</i>	8-90
8.11.5	<i>Calculations with aqueous solution involving systems</i>	8-93
8.11.5.1	<i>Command-line calculations</i>	8-94
8.11.5.2	<i>Advanced-module calculations</i>	8-94
8.12	TROUBLE SHOOTING.....	8-95
8.12.1	<i>First step</i>	8-95
8.12.1.1	<i>Check conditions</i>	8-95
8.12.1.2	<i>Simplify the system</i>	8-95
8.12.1.3	<i>Phases</i>	8-95
8.12.1.4	<i>Zero fractions</i>	8-95
8.12.2	<i>Second step</i>	8-95
8.12.3	<i>Third step</i>	8-96
8.13	FREQUENTLY ASKED QUESTIONS.....	8-97
8.13.1	<i>Why do I only get half of the lines in my diagram?</i>	8-97
8.13.2	<i>Why can not I plot the diagram after I have SAVED it?</i>	8-97
8.13.3	<i>Why is G.T not always the same as -S?</i>	8-97
8.13.4	<i>How do I get the partial enthalpy for a component?</i>	8-98
8.13.5	<i>Why is H(LIQUID) zero but not HM(LIQUID)?</i>	8-98
8.13.6	<i>Why is my carbon activity less than unity even if graphite is stable?</i>	8-98
8.13.7	<i>How can I obtain the value of the excess Gibbs energy?</i>	8-99
8.13.8	<i>What is wrong when I get crossing tie-lines but no miscibility gap?</i>	8-99
8.13.9	<i>How can I calculate the maximum of a miscibility gap directly?</i>	8-99
CHAPTER 9 POST-PROCESSOR MODULE (POST)		9-1
9.1	INTRODUCTION	9-1
9.2	GENERAL COMMANDS.....	9-3

9.2.1	<i>HELP</i>	9-3
9.2.2	<i>BACK</i>	9-3
9.2.3	<i>EXIT</i>	9-3
9.3	IMPORTANT COMMANDS	9-4
9.3.1	<i>SET_DIAGRAM_AXIS</i>	9-4
9.3.2	<i>SET_DIAGRAM_TYPE</i>	9-5
9.3.3	<i>SET_LABEL_CURVE_OPTION</i>	9-6
9.3.4	<i>ADD_LABEL_TEXT</i>	9-6
9.3.5	<i>MODIFY_LABEL_TEXT</i>	9-7
9.3.6	<i>SET_PLOT_FORMAT</i>	9-8
9.3.7	<i>PLOT_DIAGRAM</i>	9-10
9.3.8	<i>Thermo-Calc Graph Window</i>	9-11
9.3.9	<i>PRINT_DIAGRAM</i>	9-18
9.3.10	<i>DUMP_DIAGRAM</i>	9-19
9.3.11	<i>SET_SCALING_STATUS</i>	9-20
9.3.12	<i>SET_TITLE</i>	9-20
9.3.13	<i>LIST_PLOT_SETTINGS</i>	9-20
9.4	EXPERIMENTAL DATA FILE PLOT COMMANDS	9-21
9.4.1	<i>APPEND_EXPERIMENTAL_DATA</i>	9-21
9.4.2	<i>MAKE_EXPERIMENTAL_DATAFILE</i>	9-22
9.4.3	<i>QUICK_EXPERIMENTAL_PLOT</i>	9-22
9.5	OTHER COMMANDS	9-23
9.5.1	<i>ENTER_SYMBOL</i>	9-23
9.5.2	<i>LIST_SYMBOLS</i>	9-24
9.5.3	<i>SET_AXIS_LENGTH</i>	9-24
9.5.4	<i>SET_AXIS_TEXT_STATUS</i>	9-25
9.5.5	<i>SET_AXIS_TYPE</i>	9-25
9.5.6	<i>SET_COLOR</i>	9-25
9.5.7	<i>SET_CORNER_TEXT</i>	9-26
9.5.8	<i>SET_FONT</i>	9-26
9.5.9	<i>SET_INTERACTIVE_MODE</i>	9-27
9.5.10	<i>SET_PLOT_OPTIONS</i>	9-28
9.5.11	<i>SET_PREFIX_SCALING</i>	9-29
9.5.12	<i>SET_REFERENCE_STATE</i>	9-29
9.5.13	<i>SET_TIELINE_STATUS</i>	9-29
9.5.14	<i>SET_TRUE_MANUAL_SCALING</i>	9-30
9.5.15	<i>FIND_LINE</i>	9-30
9.5.16	<i>TABULATE</i>	9-30
9.5.17	<i>LIST_DATA_TABLE</i>	9-31
9.6	ODD COMMANDS	9-32
9.6.1	<i>PATCH_WORKSPACE</i>	9-32
9.6.2	<i>RESTORE_PHASE_IN_PLOT</i>	9-32
9.6.3	<i>REINITIATE_PLOT_SETTINGS</i>	9-32
9.6.4	<i>SET_AXIS_PLOT_STATUS</i>	9-32
9.6.5	<i>SET_PLOT_SIZE</i>	9-32
9.6.6	<i>SET_RASTER_STATUS</i>	9-33
9.6.7	<i>SET_TIC_TYPE</i>	9-33
9.6.8	<i>SUSPEND_PHASE_IN_PLOT</i>	9-33
9.7	3D-GRAPHICAL PRESENTATIONS: COMMAND AND DEMONSTRATION	9-34
9.7.1	<i>CREATE_3D_PLOTFILE</i>	9-36
9.7.2	<i>View 3D Diagrams in Cortona VRML Client Viewer and SIM VRMLView</i>	9-38

CHAPTER 10 SOME SPECIAL MODULES	10-1
10.1 INTRODUCTION	10-1
10.2 FILES GENERATED OR USED BY SPECIAL MODULES	10-2
10.2.1 <i>POLY3 Files</i>	10-2
10.2.2 <i>RCT Files</i>	10-2
10.2.3 <i>GES5 Files</i>	10-2
10.2.4 <i>MACRO Files</i>	10-3
10.3 INTERACTIONS WITH SPECIAL MODULES	10-3
10.4 BIN MODULE	10-4
10.4.1 <i>Descriptions of the BIN module</i>	10-4
10.4.2 <i>Structures of special BIN-module databases</i>	10-9
10.4.3 <i>Demonstrative examples of the BIN-module calculations</i>	10-11
10.5 TERN MODULE.....	10-12
10.5.1 <i>Descriptions of the TERN module</i>	10-12
10.5.2 <i>Structures of special TERN-module databases</i>	10-18
10.5.3 <i>Demonstrative examples of the TERN-module calculations</i>	10-20
10.6 POTENTIAL MODULE	10-21
10.6.1 <i>Descriptions of the POTENTIAL module</i>	10-21
10.6.2 <i>Structures of the databases used by the POTENTIAL-module</i>	10-25
10.6.3 <i>Demonstrative examples of the POTENTIAL-module calculations</i>	10-25
10.7 POURBAIX MODULE.....	10-26
10.7.1 <i>Pourbaix diagram</i>	10-26
10.7.2 <i>Descriptions of the POURBAIX module</i>	10-30
10.7.2.1 <i>Development of the POURBAIX module</i>	10-30
10.7.2.2 <i>Basic requirements for calculating Pourbaix diagrams</i>	10-32
10.7.2.3 <i>Main features of the POURBAIX module</i>	10-33
10.7.2.4 <i>Descriptions of various properties for the aqueous solution phase</i>	10-47
10.7.2.5 <i>Thermodynamic models and databases for Pourbaix-module calculations</i>	10-48
10.7.3 <i>Applications of the POURBAIX module</i>	10-50
10.7.4 <i>Demonstrative examples of the POURBAIX-module calculations</i>	10-54
10.7.5 <i>References on the POURBAIX-module calculations</i>	10-54
10.8 SCHEIL MODULE	10-55
10.8.1 <i>Solidification processes</i>	10-55
10.8.2 <i>Scheil-Gulliver Solidification Model (under thermodynamic approximation)</i>	10-56
10.8.3 <i>Moving Boundary Model (under kinetic approximation)</i>	10-56
10.8.4 <i>Descriptions of the SCHEIL Module</i>	10-57
10.8.5 <i>Demonstrative examples of the SCHEIL-module simulations</i>	10-71
10.8.6 <i>References on the Scheil-Gulliver Simulations</i>	10-71
10.9 REACTOR MODULE	10-72
10.9.1 <i>Descriptions of the REACTOR module</i>	10-72
10.9.2 <i>Some concepts in multi-staged steady-state reactor simulations</i>	10-73
10.9.3 <i>Commands in the REACTOR module</i>	10-74
10.9.3.1 <i>AMEND_RECORD</i>	10-74
10.9.3.2 <i>BACK</i>	10-74
10.9.3.3 <i>CREATE_DIVIDERS</i>	10-74
10.9.3.4 <i>CREATE_PIPES</i>	10-74
10.9.3.5 <i>CREATE_PROBE</i>	10-74
10.9.3.6 <i>CREATE_STAGE_BOXES</i>	10-75
10.9.3.7 <i>EXECUTE_POLY3_COMMAND</i>	10-76
10.9.3.8 <i>EXIT</i>	10-76
10.9.3.9 <i>GOTO_MODULE</i>	10-76
10.9.3.10 <i>HELP</i>	10-76
10.9.3.11 <i>INFORMATION</i>	10-77
10.9.3.12 <i>LIST_DESIGN</i>	10-77
10.9.3.13 <i>LIST_RECORDS</i>	10-77
10.9.3.14 <i>PATCH_WORKSPACE</i>	10-77

10.9.3.15	<i>READ_WORKSPACE</i>	10-77
10.9.3.16	<i>SAVE_WORKSPACE</i>	10-78
10.9.3.17	<i>SHOW_PROBE</i>	10-80
10.9.3.18	<i>START_SIMULATION</i>	10-80
10.9.4	<i>Demonstrative examples of the REACTOR-module simulations</i>	10-81
10.9.5	<i>References on the REACTOR-module simulations</i>	10-81
CHAPTER 11 GIBBS ENERGY SYSTEM MODULE (GES)		11-1
11.1	<i>INTRODUCTION</i>	11-1
11.2	<i>THERMOCHEMISTRY</i>	11-2
11.2.1	<i>Definitions of some terms</i>	11-2
11.2.2	<i>Elements and species</i>	11-2
11.2.3	<i>Upper and lower case mode</i>	11-2
11.2.4	<i>Phases</i>	11-3
11.2.5	<i>Functions of temperature and pressure</i>	11-4
11.2.6	<i>Symbols</i>	11-5
11.2.7	<i>Miscibility gaps</i>	11-6
11.3	<i>THERMODYNAMIC MODELS</i>	11-7
11.3.1	<i>Standardized Gibbs energy</i>	11-8
11.3.2	<i>Ideal Substitutional Model</i>	11-8
11.3.3	<i>Regular/Subregular Solution Model</i>	11-8
11.3.4	<i>Use of constituents other than elements</i>	11-10
11.3.5	<i>Sublattice Model -- Compound-Energy-Formulism</i>	11-10
11.3.6	<i>Ionic Two-Sublattice Liquid Model, for liquids with a tendency for ordering</i>	11-11
11.3.7	<i>Associate Model</i>	11-11
11.3.8	<i>Quasichemical Models</i>	11-11
11.3.9	<i>Non-chemical contributions to Gibbs energy (e.g., ferromagnetism)</i>	11-12
11.3.10	<i>Phases with order-disorder transformations</i>	11-12
11.3.11	<i>CVM Approach: for chemical ordering/disordering phenomenon</i>	11-13
11.3.12	<i>Birch-Murnaghan Model: for high-PT contributions</i>	11-13
11.3.13	<i>Generalized PVT Model: for high-PT contributions</i>	11-13
11.3.14	<i>Ideal gas model vs. non-ideal gas/gaseous mixture models</i>	11-15
11.3.15	<i>DHLL and SIT Models: for dilute aqueous solutions</i>	11-15
11.3.16	<i>HKF and PITZ Models: for concentrated aqueous solutions</i>	11-15
11.3.17	<i>Generalized Two-State Model</i>	11-16
11.3.18	<i>Flory-Huggins Model: for polymers</i>	11-17
11.4	<i>THERMODYNAMIC PARAMETERS</i>	11-18
11.5	<i>DATA STRUCTURES</i>	11-21
11.5.1	<i>Constitution</i>	11-21
11.5.2	<i>Surface of reference for the Gibbs energy</i>	11-21
11.5.3	<i>Excess Gibbs energy</i>	11-21
11.5.4	<i>Stored private files</i>	11-21
11.5.5	<i>Encrypted and unencrypted databases</i>	11-22
11.6	<i>APPLICATION PROGRAMS FOR THE GES MODULE</i>	11-23
11.7	<i>USER INTERFACE</i>	11-24
11.7.1	<i>Modularity and Interactivity</i>	11-24
11.7.2	<i>Use of Control Characters</i>	11-24
11.8	<i>COMMANDS FOR HELP AND INFORMATION</i>	11-25
11.8.1	<i>HELP</i>	11-25
11.8.2	<i>INFORMATION</i>	11-25
11.9	<i>COMMANDS TO CHANGE MODULE AND TO TERMINATE THE PROGRAM</i>	11-26
11.9.1	<i>GOTO_MODULE</i>	11-26
11.9.2	<i>BACK</i>	11-26
11.9.3	<i>EXIT</i>	11-26

11.10	COMMANDS TO ENTER DATA	11-27
11.10.1	ENTER_ELEMENT	11-27
11.10.2	ENTER_SPECIES.....	11-27
11.10.3	ENTER_PHASE.....	11-28
11.10.4	ENTER_SYMBOL.....	11-31
11.10.5	ENTER_PARAMETER	11-35
11.11	COMMANDS TO LIST DATA	11-40
11.11.1	LIST_DATA	11-40
11.11.2	LIST_PHASE_DATA	11-40
11.11.3	LIST_PARAMETER.....	11-41
11.11.4	LIST_SYMBOL	11-43
11.11.5	LIST_CONSTITUENT	11-43
11.11.6	LIST_STATUS.....	11-43
11.12	COMMANDS TO AMEND DATA.....	11-45
11.12.1	AMEND_ELEMENT_DATA	11-45
11.12.2	AMEND_PHASE_DESCRIPTION	11-46
11.12.3	AMEND_SYMBOL	11-56
11.12.4	AMEND_PARAMETER	11-57
11.12.5	CHANGE_STATUS.....	11-59
11.12.6	PATCH_WORKSPACES	11-59
11.12.7	SET_R_AND_P_NORM	11-59
11.13	COMMANDS TO DELETE DATA	11-60
11.13.1	REINITIATE	11-60
11.13.2	DELETE.....	11-60
11.14	COMMANDS TO STORE OR READ DATA	11-61
11.14.1	SAVE_GES_WORKSPACE	11-61
11.14.2	READ_GES_WORKSPACE	11-62
11.15	OTHER COMMANDS	11-63
11.15.1	ADD_COMMENT.....	11-63
11.15.2	SET_INTERACTIVE	11-63

CHAPTER 12 OPTIMIZATION MODULE (PARROT) 12-1

12.1	INTRODUCTION	12-1
12.1.1	Extended Calphad Approach.....	12-3
12.1.2	Thermodynamic and Kinetic Models	12-5
12.1.3	Experimental Information.....	12-7
12.1.4	Thermodynamic Databases	12-8
12.1.5	Optimization Method	12-8
12.1.6	Requirements for Using the Least Square Method	12-9
12.1.7	Other Optimization Software	12-9
12.2	GET STARTED	12-10
12.2.1	Experimental data file: the POP file	12-10
12.2.1.1	Single-phase equilibrium	12-10
12.2.1.2	Two-phase equilibrium	12-11
12.2.1.3	Invariant equilibria experiments	12-11
12.2.1.4	Ternary and higher order experiments	12-12
12.2.1.5	Simultaneous use of binary and ternary experiments	12-12
12.2.2	Graphical experimental file: the EXP file	12-13
12.2.3	System definition file: the SETUP file.....	12-13
12.2.3.1	Models for phases.....	12-13
12.2.3.2	Model parameters	12-13
12.2.4	Work file or store file: the PAR file	12-14
12.2.5	Various file names and their relations	12-14
12.2.6	Interactive running of the PARROT module	12-15
12.2.6.1	Compilation of experiments	12-15
12.2.6.2	Setting the Alternate Mode	12-15

12.2.6.3	<i>Plotting intermediate results</i>	12-15
12.2.6.4	<i>Selection of experimental data</i>	12-16
12.2.6.5	<i>Critical sets of experimental data and of parameters</i>	12-16
12.2.6.6	<i>Optimize and continue optimize</i>	12-17
12.2.6.7	<i>Some hints</i>	12-17
12.2.6.8	<i>Analysis of results</i>	12-17
12.2.7	<i>Rounding off parameters</i>	12-18
12.2.8	<i>Changes that are interactively done require recompilation</i>	12-18
12.3	ALTERNATE MODE	12-19
12.3.1	<i>Common tangent construction</i>	12-19
12.3.2	<i>Preparation of the POP file for Alternate Mode</i>	12-19
12.3.3	<i>More examples of the SET_ALTERNATE_CONDITION command</i>	12-20
12.4	TRICKS AND TREATS	12-21
12.4.1	<i>Conflicting data</i>	12-21
12.4.2	<i>Lack of data or bad data</i>	12-22
12.4.3	<i>Time estimate</i>	12-22
12.4.4	<i>Number of parameters</i>	12-22
12.5	COMMAND STRUCTURE	12-23
12.5.1	<i>Definitions of some terms</i>	12-23
12.5.2	<i>Command connections with other modules</i>	12-23
12.5.3	<i>User interface</i>	12-23
12.6	GENERAL COMMANDS	12-24
12.6.1	<i>GOTO_MODULE</i>	12-24
12.6.2	<i>HELP</i>	12-24
12.6.3	<i>INFORMATION</i>	12-25
12.6.4	<i>BACK</i>	12-25
12.6.5	<i>SET_INTERACTIVE</i>	12-25
12.6.6	<i>EXIT</i>	12-25
12.7	MOST FREQUENTLY USED COMMANDS	12-26
12.7.1	<i>CREATE_NEW_STORE_FILE</i>	12-26
12.7.2	<i>SET_STORE_FILE</i>	12-27
12.7.3	<i>COMPILE_EXPERIMENT</i>	12-27
12.7.4	<i>SET_ALTERNATE_MODE</i>	12-28
12.7.5	<i>SET_OPTIMIZING_VARIABLE</i>	12-28
12.7.6	<i>SET_FIX_VARIABLE</i>	12-29
12.7.7	<i>EDIT_EXPERIMENTS</i>	12-29
12.7.8	<i>OPTIMIZE_VARIABLES</i>	12-30
12.7.9	<i>CONTINUE_OPTIMIZATION</i>	12-30
12.7.10	<i>LIST_RESULT</i>	12-30
12.7.11	<i>LIST_ALL_VARIABLES</i>	12-33
12.7.12	<i>RESCALE_VARIABLES</i>	12-33
12.8	OTHER COMMANDS	12-34
12.8.1	<i>SET_OUTPUT_LEVELS</i>	12-34
12.8.2	<i>SAVE_PARROT_WORKSPACES</i>	12-34
12.8.3	<i>READ_PARROT_WORKSPACES</i>	12-34
12.8.4	<i>MACRO_FILE_OPEN</i>	12-35
12.8.5	<i>REINITIATE</i>	12-38
12.8.6	<i>SET_OPTIMIZING_CONDITION</i>	12-38
12.8.7	<i>LIST_CONDITIONS</i>	12-39
12.8.8	<i>LIST_STORE_FILE</i>	12-39
12.8.9	<i>SET_SCALED_VARIABLE</i>	12-39
12.8.10	<i>RECOVER_VARIABLES</i>	12-40
12.8.11	<i>SET_EXTERNAL_PARAMETER</i>	12-40
12.9	COMMANDS FROM THE GIBBS ENERGY SYSTEM (GES)	12-41
12.9.1	<i>LIST_PHASE_DATA</i>	12-41

12.9.2	<i>ENTER_PARAMETER</i>	12-41
12.9.3	<i>AMEND_PARAMETER</i>	12-45
12.9.4	<i>LIST_PARAMETER</i>	12-47
12.9.5	<i>LIST_SYMBOL_IN_GES</i>	12-49
12.10	CONNECTIONS WITH OTHER MODULES AND INTERFACES	12-50
12.10.1	<i>Connections with the ED_EXP modules</i>	12-50
12.10.2	<i>Connections with the GES and POLY modules</i>	12-50
12.10.3	<i>Connections with the DICTRA software</i>	12-50
12.10.4	<i>Availability in the TCW software</i>	12-50

CHAPTER 13 EDIT-EXPERIMENT MODULE (ED-EXP)..... 13-1

13.1	INTRODUCTION	13-1
13.2	PERFORMANCE OF POLY COMMANDS IN THE ED_EXP MODULE	13-2
13.3	SPECIAL COMMANDS ONLY AVAILABLE IN THE ED_EXP MODULE	13-4
13.3.1	<i>COMPUTE_ALL_EQUILIBRIA</i>	13-4
13.3.2	<i>EXPERIMENT</i>	13-6
13.3.3	<i>EXPORT</i>	13-7
13.3.4	<i>IMPORT</i>	13-7
13.3.5	<i>LABEL_DATA</i>	13-8
13.3.6	<i>MAKE_POP_FILE</i>	13-8
13.3.7	<i>READ (READ_WORKSPACE = READ_BLOCK)</i>	13-8
13.3.8	<i>SET_ALTERNATE_CONDITION</i>	13-9
13.3.9	<i>SET_WEIGHT</i>	13-10
13.3.10	<i>TRANSFER_START_VALUE</i>	13-11
13.3.11	<i>STORE_ALL_WEIGHTS</i>	13-12
13.3.12	<i>RESTORE_ALL_WEIGHTS</i>	13-12
13.3.13	<i>LIST_ALL_EQUILIBRIA</i>	13-12
13.4	OTHER COMMANDS IN THE EXPERIMENTAL DATA (POP OR DOP) FILES	13-13
13.4.1	<i>TABLE_HEAD, TABLE_VALUES and TABLE_END</i>	13-14
13.4.2	<i>COMMENT</i>	13-15
13.4.3	<i>FLUSH_BUFFER</i>	13-16
13.4.4	<i>GRAPHICS_PLOT</i>	13-16

CHAPTER 14 SYSTEM UTILITY MODULE (SYS)..... 14-1

14.1	INTRODUCTION	14-1
14.2	GENERAL COMMANDS	14-2
14.2.1	<i>HELP</i>	14-2
14.2.2	<i>INFORMATION</i>	14-2
14.2.3	<i>GOTO_MODULE</i>	14-3
14.2.4	<i>BACK</i>	14-3
14.2.5	<i>EXIT</i>	14-3
14.2.6	<i>SET_ECHO</i>	14-4
14.2.7	<i>SET_LOG_FILE</i>	14-4
14.2.8	<i>MACRO_FILE_OPEN</i>	14-5
14.2.9	<i>SET_TC_OPTIONS</i>	14-7
14.2.10	<i>SET_PLOT_ENVIRONMENT</i>	14-8
14.3	ODD COMMANDS	14-9
14.3.1	<i>SET_INTERACTIVE_MODE</i>	14-9
14.3.2	<i>SET_COMMAND_UNITS</i>	14-9
14.3.3	<i>STOP_ON_ERROR</i>	14-9
14.3.4	<i>OPEN_FILE</i>	14-9
14.3.5	<i>CLOSE_FILE</i>	14-9
14.3.6	<i>SET_TERMINAL</i>	14-10
14.3.7	<i>HP_CALCULATOR</i>	14-10

CHAPTER 15 DATAPLOT GRAPHICAL LANGUAGE (DATAPLOT)	15-1
<i>[PLEASE GO TO THE DOCUMENT “DATAPLOT USER’S GUIDE”]</i>	
CHAPTER 16 REFERENCE LISTS ON THERMO-CALC PACKAGE AND APPLICATIONS	16-1
<i>[PLEASE GO TO THE DOCUMENT “THERMO-CALC REFERENCE LIST”]</i>	
CHAPTER 17 APPENDICES AND INDICES	17-1
APPENDIX A. THERMO-CALC CLASSIC FILE TYPES AND RELATIONS	17-2
APPENDIX B. THERMO-CALC CLASSIC QUICK REFERENCE CARD.....	17-3
APPENDIX C. UNITS OF STATE VARIABLES AND DERIVED VARIABLES	17-4
APPENDIX C. UNITS OF STATE VARIABLES AND DERIVED VARIABLES	17-5
<i>General Introduction</i>	<i>17-5</i>
<i>Basic Units: For Variables that are Globally Used</i>	<i>17-5</i>
<i>System Units: For Variables of a Defined System.....</i>	<i>17-6</i>
<i>Component Units: For Variables of a Specific Component in the System</i>	<i>17-7</i>
<i>Phase Units: For Variables of a Specific Phase in the System</i>	<i>17-8</i>
<i>Phase-Component Units: For Variables of a Specific Component in a Phase</i>	<i>17-10</i>
<i>Phase-Species Units: For Variables of a Specific Species in a Phase</i>	<i>17-11</i>
<i>User-Specified Units: For Functions that are Defined by a User</i>	<i>17-13</i>
<i>User-specified units for intensive variables of a defined system.....</i>	<i>17-14</i>
<i>User-specified units for extensive variables of a defined system.....</i>	<i>17-14</i>
<i>User-specified units for intensive variables of a system component.....</i>	<i>17-15</i>
<i>User-specified units for extensive variables of a system component.....</i>	<i>17-15</i>
<i>User-specified units for extensive variables of a phase</i>	<i>17-15</i>
<i>User-specified units for extensive variables of a component in a phase</i>	<i>17-16</i>
<i>User-specified units for intensive variables of a species in a phase.....</i>	<i>17-17</i>
<i>User-specified units for extensive variables of a species in a phase.....</i>	<i>17-17</i>
APPENDIX D. ON REFERENCE STATES AND STANDARD STATES	17-18
<i>Brief Introduction</i>	<i>17-18</i>
<i>How Does Thermo-Calc Handle Reference States</i>	<i>17-18</i>
<i>Some Important Issues about Chemical Potential and Activity.....</i>	<i>17-20</i>
<i>Some Specific Examples</i>	<i>17-21</i>
<i>Some Typical Questions about Reference States</i>	<i>17-26</i>
INDEX ON THERMO-CALC COMMANDS IN VARIOUS MODULES	17-28
INDEX ON THERMO-CALC COMMANDS IN VARIOUS TYPES OF FILES.....	17-39
LIST OF FIGURES.....	17-42
LIST OF TABLES	17-45

(This page is intended to be empty)

1 General Introduction

1.1 Computational Thermodynamics and Kinetics

The research and development linking materials science and engineering with computer calculations and simulations in the last decade have given birth to a revolutionary approach for quantitative conceptual design of various materials. A comprehensive combination of thermodynamic and kinetic models makes it possible to predict material compositions, structures and properties resulting from various materials processing.

The increased importance of mathematical modeling for product development and process control has proven the high need for thermodynamic calculations and kinetic simulations. Modern quantitative conceptual design of advanced materials has tremendously benefited from computational thermodynamics and kinetics.

The information about the equilibrium or partial-/local-equilibrium state of specific elements/species/phases in a heterogeneous interaction system and about kinetics of phase transformations (as well as chemical reactions, surface reactions, nucleation, particle growths/dissolutions, fluid flow, *etc.*) in a certain materials process, which can be provided by a software system, is essential for solving many real problems ranging from materials design and process development in chemical, metallurgical, automobile, aerospace and electronic industries, to resource exploitation, energy conversion and waste management in natural and environmental engineering. An important feature of a thermochemical/kinetic databank is that it also provides an easy way, compared to experimental work, to investigate how the equilibrium or partial-/local-equilibrium state and dynamic process can be affected by various external and internal factors. Moreover, a computerized thermochemical/kinetic databank has the great advantage, compared to handbooks, in providing the user with self-consistent, reliable and the most recent data.

A general thermochemical/kinetic databank must have high-quality and internally-consistent data for a number of fields that traditionally have been regarded as separate, *e.g.*, metallurgy, steels/alloys, ceramics, high temperature gas phase equilibria, aqueous chemistry, and geochemistry. In most applications, the number of components in a heterogeneous interaction system/process is so large that the equilibrium or partial-/local-equilibrium state and process kinetics can be efficiently and reliably calculated/simulated only by using computer software. The establishment of the Thermo-Calc®™ and DICTRA®™ databank systems is a successful attempt to provide such powerful and sophisticated software systems, which are easy to learn and use for all kinds of thermochemical calculations and for some types of kinetic simulations.

Thermochemical calculations by Thermo-Calc, and kinetic simulations by DICTRA, can dramatically enhance your capacity in designing/developing new materials, selecting temperatures for heat treatments, optimizing yields of manufacturing processes, supervising materials applications, protecting environments, *etc.* These comprehensive software/database/interface packages have been proved worldwide to be the most powerful and flexible engineering tools that help eliminate expensive and time-consuming experiments, improve quality performance, and control environmental impacts.

1.2 Thermo-Calc Software/Database/Interface Package

Thermo-Calc is a general and flexible software system (Sundman *et al.*, 1985; 1993; Jansson *et al.*, 1993; Sundman and Shi, 1997, Shi *et al.*, 2004), for all kinds of calculations of thermodynamic properties (as functions not only of temperature, pressure and composition, but also of magnetic contribution, chemical-/magnetic-ordering, crystallographic structures/defects, surface tension, amorphous glass-formation, plastic deformation, mechanic-induced deformation, electro-static state, electronic potential, *etc.*), equilibrium and partial/local-equilibrium quantities, chemical driving forces (thermodynamic factors), and for various types of stable/meta-stable phase diagrams and property diagrams of multicomponent systems for many types of materials. It can efficiently handle a very complex multicomponent and heterogeneous interaction system defined with up to 40 elements, 1000 species and many different solution or stoichiometric phases. There are also powerful facilities to calculate many other types of diagrams, such as CVD/PVD depositions, CVM calculations of ordering/disordering phenomena, Scheil-Gulliver solidification simulations (with or without considering back-diffusions of fast-diffusing components), liquidus surface, Pourbaix diagrams, Ellingham diagrams, partition coefficients, partial pressures in gases, and so forth. It is the only software that can calculate arbitrary phase diagram sections with up to five independent variables in a defined system (which, is very

useful, for instance, in finding the lowest melting temperature in a multicomponent alloy that has up to five independently-varying compositions), as well as the only software that can calculate chemical driving forces (thermodynamic factors, *i.e.*, the second derivatives of Gibbs free energy with respect to compositions) which are important fundamental information for kinetic simulations (such as in diffusion-controlled phase transformation, nucleation, particle growths/dissolutions, and so on).

Linked with various databases and interfaces, it stands for the **Thermo-Calc Software/Database/Interface Package**, or the *Thermo-Calc Databank*. The most important aim of the Thermo-Calc package is to make efficient and quick thermodynamic calculations available for applications in science and industry. The Thermo-Calc software system is based upon a powerful Gibbs Energy Minimizer, which has been further enhanced by the so-called Global Minimization Technique (that has been implemented in the current version). The Thermo-Calc software is especially designed for complex heterogeneous interaction systems with strongly non-ideal phases (which are specifically handled by a very wide range of sophisticated thermodynamic models), and can use many different thermodynamic databases, particularly those developed by the SGTE organization (*Scientific Group ThermoData Europe*, an international organization for collaborations on thermodynamic database developments) and CALPHAD community (*CALculations of PHase Diagrams*). Furthermore, there are several powerful application programming interfaces of the Thermo-Calc software engine, which can be utilized for user-written application programs or within third-party software packages for materials property calculations and materials process simulations.

Thermo-Calc and its sister software **DICTRA** (for Diffusion-Controlled phase TRAnsformation) have been developed originally at the *Department of Materials Science and Engineering of KTH (Royal Institute of Technology)*, Stockholm, Sweden, and since 1997 further by our company **Thermo-Calc Software (TCS)**. They are the results of more than 35 years and 150 man-years R&D and many national/international collaborations through various R&D projects.

The copyrights for the Thermo-Calc and DICTRA software and several related databases belong to a non-profit organization, *Foundation of Computational Thermodynamics (STT)*, Stockholm. Since 1997, the further developments, marketing/sales, technical supports and all other activities concerning the Thermo-Calc and DICTRA packages are managed by our company TCS that is also owned by STT.

There are two different user-interface types of the Thermo-Calc software, namely **TCC** (Thermo-Calc Classic) and **TCW** (Thermo-Calc Windows). The first version of TCC was released in 1981. Since 1999, TCW (as the second generation of the renowned Thermo-Calc software), with a completely GUI-driven (Graphic User Interface), has been made available to our users. There has been an update almost every year and the most recent ones, TCCR and TCW4, are released in May 2006.

Any modern PC (Microsoft Windows Vista/XP/2000/NT4, Linux) and UNIX workstation (SUN Sparc, Solaris, HP, IBM AIX, DEC Alpha OSF1, SGI) can be used for running the Thermo-Calc and DICTRA software/database/interface packages. However, since TCCP (Thermo-Calc Classic version P) and DICTRA22 (DICTRA version 22), some UNIX platforms (SUN Sparc, HP, IBM AIX and DEC Alpha OSF1) are not supported any more.

The classical version TCC (and DICTRA) has an interactive user interface, extensive documentation, and on-line help facilities. The general GUI-driven (Graphic User Interface) version TCW has been made available for Windows Vista/XP/2000/NT4 environments, while such a general GUI version for UNIX and Linux environments is still under development.

Thermo-Calc has gained a worldwide reputation as the best software for calculations of multicomponent phase diagrams. There are more than 1000 installations all over the world today, including academic institutions (universities and governmental institutes) and non-academic organizations (industrial companies and research companies), and is a reference in technical literature. Many application examples are given inside the **TCC Examples Book** and **TCW Examples Book**, and can also be found at our web site (www.thermocalc.com).

A thermodynamic software package is useless if without accurate and validated databases. Thermo-Calc allows you to utilize many critically-assessed and high-quality databases from various sources (such as SGTE, CAMPADA, CCT, ThermoTech, NPL, NIST, MIT, Theoretical Geochemistry Group, *etc.*). Such databases use different thermodynamic models for each phase in a certain heterogeneous interaction system. Currently-available Thermo-Calc databases cover a wide spectrum of materials, including steels, alloys, ceramics, melts, slag, slats, glasses, hard materials, semi-/super-conductors, solders, gas/fluids, aqueous solutions, organic substances, polymers, nuclear materials, earth materials, as well as geochemical and environmental systems, which can be applied to research and development in industrial engineering and natural systems: for instance,

the SSUB/SSOL databases for substances and solutions in inorganic and metallurgical systems, TCFE for steels and Fe-alloys, TCNI/TTNi for Ni-based superalloys, TTAI/TTMg/TTTi for Al-/Mg-/Ti-based alloys, SLAG for slag, ION for carbides/nitrides/oxides/silicates/sulfides (solids/liquids/gases), TCMP for materials processing and applications to environmental problems associated with metallurgical, chemical and waste-treatment processes, particularly in recycling, remelting, sintering, incineration and combustion), SMEC for semiconductors, NSLD/USLD for lead-free solders, SNOB for noble metals, NUMT/NUOX for nuclear substances and nuclear oxides, GCE for minerals, and TCAQ/AQS for aqueous solutions, *etc.* For more details on various available databases, please see the [**Thermo-Calc Database Guide**](#).

The Thermo-Calc and DICTRA groups at KTH-MSE have initiated and participated in many national and international projects in order to create general and validated databases. Thermo-Calc Software AB is now actively devoted to developments of more application-oriented databases of various industrial interests. There are also many users in various academic societies and industrial companies all over the world who have established their own databases or datasets under the assistance of the Thermo-Calc and DICTRA packages.

Both the TCC and DICTRA software also provide the users with a unique tool (the **PARROT** module) for critical assessment based upon varied experimental data such as PVT-EOS, thermochemical properties, phase equilibria, phase diagrams, diffusivity, and so on. By means of this module the user can efficiently expand some databases or reliably create various datasets or databases for some specific materials and applications. As a matter of fact, many of the existing databases and datasets for various materials are really the products of R&D activities using the PARROT module.

The Thermo-Calc software system has also three application-oriented programming interfaces, namely the **TQ**, **TCAPI** and the **TC-MATLAB[®] Toolbox**, which are designed for uses in various material property modeling and complex material process simulations. Powered by the Thermo-Calc software engine (plus some DICTRA extensions), such programming interfaces provide various kinds of calculations both for thermodynamic quantities and for local-/partial-equilibria and driving forces, which are needed by other user-written programs or third-party software packages in complex property modeling and process simulations. The most successful examples are the comprehensive DICTRA^{®TM} package, MISCRESSTM software and PrecipCalcTM software. Recently, there have also been many other developments on coupling the Thermo-Calc/DICTRA engines through such programming interface with third-party software packages such as FluentTM and PhoenixTM, as well as with FEM/PDE calculations and with Phase-Field simulations.

Concurrent calculation and experiment makes much more rapid and reliable progress for materials and process development than purely experimental trial-and-error methods. The full potential of thermodynamic calculation methods can now be better employed through the significant developments of the Thermo-Calc/DICTRA software, databases and interfaces. The package has been developed to such an extent that realistic calculations can be made in the field of metallurgy, alloy development, material science, semi-/super-conductors, chemistry, chemical engineering, geochemistry, energy conversion, power production, food industry, nuclear fuel waste repository, environmental controls, *etc.* Figure 1-1 shows a number of examples of calculated diagrams.

One of the main purposes of the Thermo-Calc/DICTRA packages is in its use for planning and reducing the need for new costly experiments. By calculation it is possible to predict the results of an experiment and this can limit the number of experiments that eventually have to be made. It may even be found that the calculations alone give results that are reliable enough to be used directly.

At present and in future, TCS, STT and KTH-MSE are highly devoted to the further development on the Thermo-Calc and DICTRA software, as well on various databases and interfaced programs for different materials and processes. The continuous and close cooperations with many international R&D partners and Thermo-Calc and DICTRA users all over the world ensure such developments.

The mission of Thermo-Calc Software is *to be extensively utilized as a powerful research and development tool for scientists and engineers in their daily work in laboratories or factories, and in teaching and training activities to bring new insight into realistic problems by linking fundamental models and critically assessed thermodynamic and kinetic data.*

The ultimate purpose of Thermo-Calc Software is *to assist you, to the highest degree, to save time and money in materials design, engineering and applications.*

The development strategy of Thermo-Calc Software is *to provide multiple applicability and increased efficiency to the materials industrials and R&D community.*

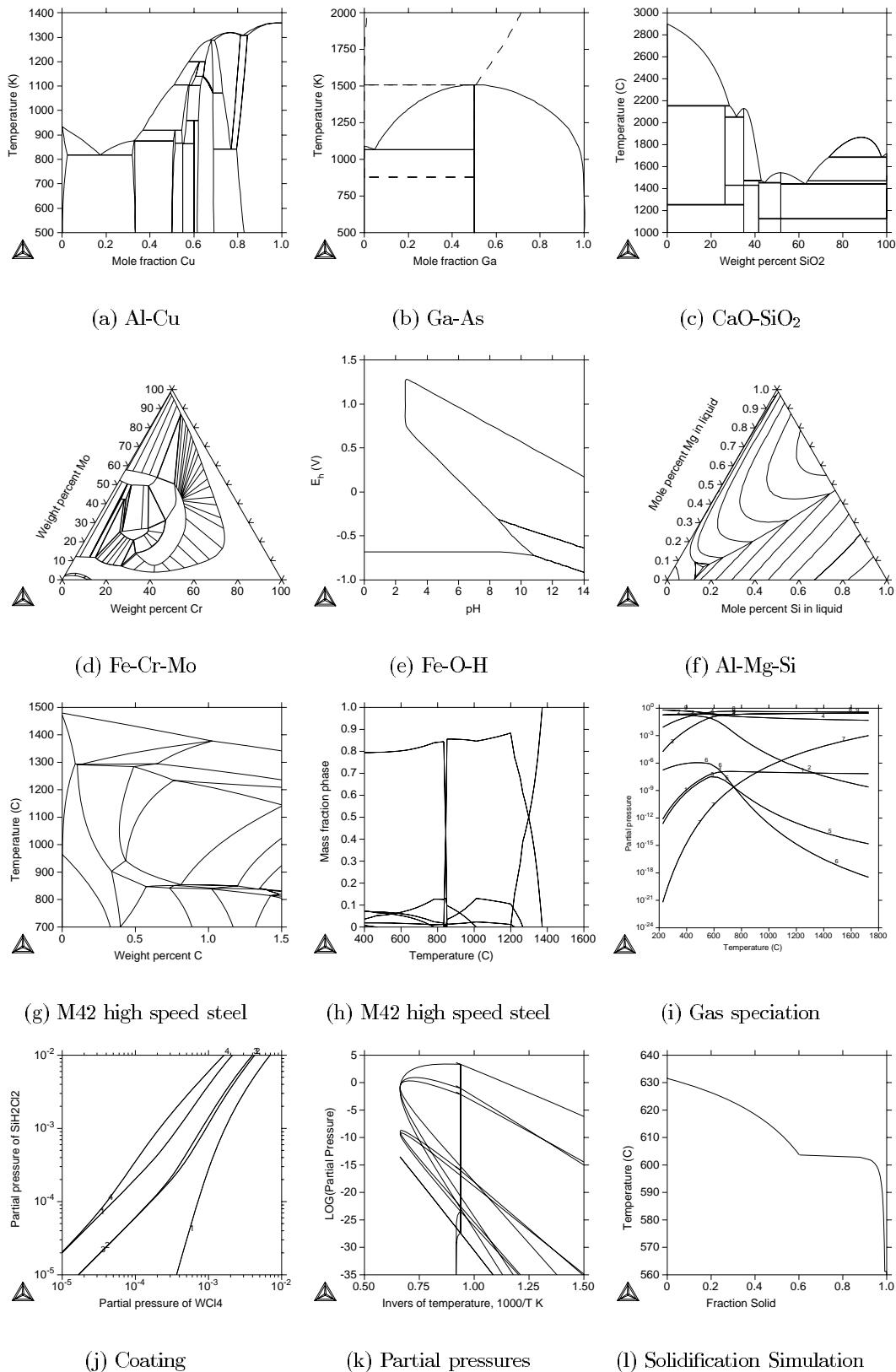


Figure 1-1. Examples of Calculations with Thermo-Calc

The diagrams illustrated in Figure 1-1 give some ideas of the capabilities of Thermo-Calc. Most of the figures are taken from the binder "Thermo-Calc Examples Book", which describes how to calculate and plot them. All diagrams have been calculated in Thermo-Calc after axes and other necessary selected conditions. Thermo-Calc can calculate just a single equilibrium, and indeed, the powerful set of commands makes it possible to directly calculate many important equilibria. There are many more diagrams which can be calculated in a few seconds by using one of the specialised modules of Thermo-Calc, e.g. BINary, TERnary, POTential, POURBAIX, SCHEIL modules.

The first two diagrams display the binary systems, Al-Cu and As-Ga. In diagram b), the solid lines represent a metastable system without the gas phase. The dashed lines show the equilibrium with the gas phase. Data for Al-Cu are from the COST-507 database and As-Ga derives from the Group III-V database. Diagram c) is a pseudo-binary CaO-SiO₂ and data are from the IONIC database. All of these databases are compatible with the SGTE databases.

The second row shows various diagrams for ternary systems. The first is an isothermal section at 1000 °C of a metallic system, Cr-Fe-Mo, plotted as a so-called Gibbs triangle. The second is a Pourbaix diagram for iron in water. The last is a liquidus surface of an Al-Mg-Si alloy showing isothermal and monovariant lines.

The third row pictures a phase diagram for a M42 steel with Co, Cr, Mn, Si, Mo, W, V and C. The alloy content has been kept fixed and only the carbon content and temperature vary. The lines separate regions with different sets of stable phases. The second diagram is for the same steel but here the carbon content is also kept fixed. The amount of the various phases is plotted as a function of temperature. Other quantities may be plotted like carbon activity, phase composition etc. from the same calculation and other state variable as the independent variable. The last diagram shows how the partial pressures of the major species in a C-H-O gas vary with temperature.

The bottom row shows some more "application" oriented diagrams. The first is a diagram used for CVD applications showing the various solid phases that are in equilibrium with a gas phase made up by mixing WCl₄ and SiH₂Cl₂ according to the axes and with an excess of argon. In the lower right region, only solid tungsten is deposited. The lines separate regions where one or two phases are deposited. The label identifies the phase that has zero amount along the line. The narrow region with just WSi₂ is the interesting one. The second diagram is from the same calculation as diagram b) in the first row but now the partial pressures of the gas species along the liquidus lines are plotted versus the inverse of temperature. The vertical line represents the eutectic temperature. The partial pressures are important to control the growth conditions from a gas phase. The final diagram shows a so-called Scheil-Gulliver solidification simulation. The system is Al-Mg-Si and the curve shows how the temperature varies with the fraction of solid phases, assuming no diffusion in the solid. The liquidus surface of this system is given in diagram f).

1.3 Acknowledgements

During the entire development history of the Thermo-Calc software/database/interface package over the past 35 years, more than 140 man-years at the Department of Materials Science and Engineering of KTH have been invested along the whole course of research and development related to this package. Various supports from NUTEK (Swedish National Council of Technical Development, previously STU), NFR (Swedish National Council of Natural Science), Sandvik Steels AB, SGTE (Scientific Group Thermodata Europe), NIF (Nordisk Industrifond), CAMPADA Project (Swedish National Strategic Consortium on Materials), CCT Project (Center of Computational Thermodynamics), *etc.* have been given during this time.

Different parts of the Thermo-Calc User's Guide were previously written by various authors at the Department of Materials Science and Engineering of KTH. Professor Bo Sundman had been responsible for continuous updates of the Thermo-Calc manual since 1981, and was the principal author of almost all parts of the manual, with co-authorship of Drs. Bo Jansson, Jan-Olof Andersson, Lars Höglund, Björn Jönsson, Malin Selleby, among others, and under various sponsorships from the organizations mentioned above. Valuable comments and suggestions from many colleagues at KTH and SGTE and from many international experts (especially Prof. Mats Hillert and Prof. L. Lukas) had also been incorporated. All those have formed the sound basis of this manual revision, and are acknowledged herein.

The current chief-editor for the TCCS/TCCR/TCCQ/TCCP/TCCN User's Guides (Dr. Pingfang Shi at Thermo-Calc Software; pingfang@thermocalc.se) is grateful for critical reviews on the manuscripts by Prof. Bo Sundman, Dr. Åke Jansson, Dr. Lars Höglund, Dr. Qing Chen, Dr. Anders Engström, Dr. Jan-Olof Andersson, Dr. Thomas Helander, and Dr. Nathalie Dupin.

We also appreciate the various comments and suggestions from our existing Thermo-Calc users all over the world. With such inputs, we can make further improvements on the Thermo-Calc software/database/interface package and its documentation.

In this revision of the Thermo-Calc manual set, *i.e.*, the **TCCS User's Guide**, **Thermo-Calc Software System**, **Thermo-Calc Database Guide**, **DATAPLOT User's Guide and Examples** and **TCCSR Examples Book**, all added new features, improved functionality, new or modified modules, new or updated databases, new application examples, and so forth, up to the TCC version S, are documented.

Please note that the Thermo-Calc manual set may have not always been updated simultaneously as the software revisions. However, whenever there is no manual update for any reason, some supplemental documents (such as **Thermo-Calc Technical Information**, **Thermo-Calc Special Database Applications**, **Thermo-Calc Reference Book**) or specially prepared materials documenting the software/database/interface revision will be released.

1.4 Revision History

The Thermo-Calc software has been updated nearly once a year since 1981. The most recent versions released in June 2008 are the TCCS (Thermo-Calc Classic, version S) and TCW5 (Thermo-Calc Windows, version 5), with which there are compatible versions of the other software DICTRA25 and of programming interfaces TQ7, TC API5 and TC-MATLAB Toolbox5. Each new release of the software (and associated databases and programming interfaces) comes as a result of many R&D activities at TCS (Thermo-Calc Software) and DCT (Division of Computational Thermodynamics, KTH-MSE).

Along with all the new software releases, some specific databases were updated or expanded in order to organize the software improvement, and some new ones were added due to the database development. Such updated or new databases are developed by TCS, DCT, our collaborating partners and experienced users.

The thermodynamic programming interface (TQ) was developed since the TCCL version, and has been updated simultaneously along with later TCC software revision. The full Windows version (TCW) and the application programming interfaces (TC API and TC-MATLAB Toolbox) are available since the TCCN version.

Table 1-1 shows the revision history of some later versions since 1996, with information on software's new feature and modifications, programming interface updates, and database additions.

All details of the major improvements in various versions of the Thermo-Calc software, databases and/or interfaces, can be found in the Thermo-Calc Software Newsletters (with the specific numbers as identified in the above table) or Thermo-Calc Technical Information. Please visit the TCS web site (www.thermocalc.com) for the latest and all previous releases of such documents.

Table 1-1. Revision History of the Thermo-Calc Software/Database/Interface Package

<i>TCC Version Number</i>	<i>TCC Releasing Date</i>	<i>Major Improvements: Software – new features and modifications; Interfaces – updates and new releases; Databases – additions and new releases.</i>	<i>Release-related Reference</i>
K	Apr. 1995	<ol style="list-style-type: none"> 1. Two new easy-to-use modules are added for automatic diagram calculations: the BIN module for binary phase diagrams, and the POTENTIAL module for potential diagrams. 2. Many functions in the POLY, GES, PARROT, TAB and POST modules are enhanced: for instance, <ul style="list-style-type: none"> • test on miscibility gap • mapping with “tie-line in the plane” • representation of additional Gibbs energy terms, such as stress or external influences • automatic diagram axis setting after mapping/stepping • direct plotting in the TAB module • calculation of U-fractions • better convergence in stepping • modifications in many POLY commands • improved MACRO-file opening function 3. The new version is available on OS/2, MS-DOS, and all UNIX platforms and VAX/VMS. 4. The ETTAN software interface is released for PC users to easily make stepping calculation of property diagrams. 5. The SSUB1 (SGTE Substances Database version 1) is released; the IRSID SLAG database is extended; the ThermoTech Al-/Ti-Ni-based alloy databases are introduced into Thermo-Calc. 	Thermo-Calc Newsletter No. 18
L	Nov. 1996	<ol style="list-style-type: none"> 1. Two more new easy-to-use modules are added for automatic diagram calculations: the SCHEIL module for simulating solidification processes; the POURBAIX module for calculating pH-Eh and many other property diagrams in aqueous-related systems. 2. Many functionality in the POLY, GES, PARROT, TAB and POST modules are enhanced: for instance, <ul style="list-style-type: none"> • new command <code>DEFINE_MATERIALS</code> is added • improved <code>STEP</code> and <code>MAP</code> commands • the limit of optimizing variable is increased from 100 to 1000 • the CVM model for modeling short-range ordering is improved 3. The TQ version 1 (compatible with TCCL) is released. 4. The academic version of the TCCL software, namely TC4A (version N) is released. 5. The FEDAT1 steels database and the AEAT nuclear materials databases are released. 	Thermo-Calc Newsletter No. 19
M	Aug. 1998	<ol style="list-style-type: none"> 1. The TERN module is added for automatic calculations of ternary diagrams; 2. Many functionality in the POLY, GES, PARROT, TAB and POST modules are enhanced: for instance, <ul style="list-style-type: none"> • new command <code>DEFINE_DIAGRAM</code> is added • new option to map monovariant lines • new <code>ADD_TABLE</code>, <code>MODIFY_TABLE</code>, <code>TABLE_DATA</code> and <code>COMMENT</code> are added • improved <code>ADD_INITIAL</code>, <code>CALCULATE_ALL</code> command • revised TAB module 3. The Alternate Mode is added to the PARROT module. 4. New composition-dependent parameters for volume. 5. The POURBAIX module is modified. 6. Multiple-database directory structure and simplified database management are added. 7. New graphic module for Windows NT/95/98. 8. The TQ version 2 (compatible with TCCM) is released. 9. The updated demo version for TCCM (TC4A version M) is released. 	Thermo-Calc Newsletter No. 22

N	Jun. 2000	<ol style="list-style-type: none"> 1. The FLEXLM Software Licensing System is implemented. 2. Under Windows, the Thermo-Calc Classic (TCC) and DICTRA software can be started by just simply clicking on a file. 3. The Flory-Huggins model for polymers is developed. 4. The Quasi-chemical model for ionic liquids is added. 5. The tetrahedron CVM model can be used together with regular solution excess (Redlich-Kister). 6. The complete Revised HKF model for aqueous solution has been implemented but needs more testing. 7. Many functionality in the POLY, GES, PARROT, TAB and POST modules are enhanced: for instance, <ul style="list-style-type: none"> • limit numbers in POLY increased to 40 components and 1000 species • the new POLY command COMPUTE_TRANSITION simplifies calculations of solubility limits • the phase code F saves much memory and computation time for 4-sublattice ordering • phase which never disorder can have a disordered excess • MACRO files can now be nested (5 levels) • the command SET_INTERACTIVE is added to the GES module • error messages from TDB module improved • output during LIST_STATUS shortened • output during MAP and STEP shortened • output during PARROT improved • some bugs fixed in TDB and PARROT module 8. The POURBAIX module is extensively revised, and the Option 4 for stepping calculations is added. 9. The test version of the fully-GU driven Windows version, TCW1 software (Thermo-Calc Windows version 1, <i>i.e.</i>, TCW-Basic, compatible with TCCN), is circulated among some groups. 10. The demo version of TCCN (TC4A version N) is released. 11. The TQ version 3 (compatible with TCCN) is released. 12. The TCAPI version 1 (compatible with TCCN) is released. 13. The TCFE steels/Fe-alloys database is updated to TCFE2 (TCFE2K). 14. Five new databases are released: <ul style="list-style-type: none"> • TCNI1 (for Ni-based super-alloys, v1) • TCER1 (for recycling/remelting processes, v1) • TCES1 (for sintering/incineration/combustion processes, v1) • AQS1 (for aqueous solutions using the HKF model, v1) • GCE1 (for minerals and sub-/super-critical fluids, v1) • COST2 (for light alloys, v2) 	TCS Newsletter Nos. 24 & 25
P	Nov. 2002	<ol style="list-style-type: none"> 1. Special option for automatic calculations of T_0 temperature, in both single equilibrium calculations and stepping calculations. 2. Special option for automatic calculations of para-equilibrium, in both single equilibrium calculations and stepping calculations. 3. Special option for internal stability check if any phase can be subject to spinodal decomposition. 4. The maximum possible number of constituents in a non-ideal solution phase has been increased from 80 to 200. 5. The convergence for the COMPUTE_EQUILIBRIUM * command sequence for a more robust calculation has been improved 6. The COMPUTE_TRANSITION command has been improved, so that is now possible to look for any new phase appearing when one condition is varied by using the keyword ANY instead of a phase name. 7. Automatic modification on the smallest fraction to ensure converged solutions during a single equilibrium calculation, special-options calculation, stepping or mapping; one can restore the user-desired smallest fraction afterwards, using the SET_NUM_LIMITS command. 8. The database and conditions used for a MAPPING or STEPPING calculation are automatically printed on the diagrams. The plotted conditions (especially there are too many) can also be taken away by using the SET_PLOT_OPTIONS command and switching off its WRITE_CONDITION option. 9. TCCP can now produce 3D graphs, by generating VRML (Virtual Reality Modelling Language) files from plotted triangular and tetrahedral diagrams; such plot files can be read, viewed and printed, using one VRML viewer from the many plug-ins freely available for web browsers or standalone programs. 	TCS Newsletter Nos. 26, 27 & 28

	<p>10. The Windows version of TCCP allows to convert any plotted diagram to graphical formats PNG, BMP, PDF, JPEG and TIFF, using the DUMP DIAGRAM command.</p> <p>11. In the TDB module, the new command SET_AUTO_APPEND makes it possible to conveniently append a database with the same system definition.</p> <p>12. A special routine has been implemented in the TDB module for easy converting database files from UNIX/Linux format to Windows format, or <i>vice versa</i>. The routine automatic gives a warning message if conversion is necessary.</p> <p>13. Each commercial database is provided with a specific database license key file, that allows its usage appropriate for the license category and installation type. The data-listing facility for encrypted commercial databases has been turned down in the GES module.</p> <p>14. In the sub-module ED_EXP of the PARROT module, it is now possible to store and restore a weight set that records all the weights given to different experimental points in an assessment, using the new STORE_ALL_WEIGHTS and RESTORE_ALL_WEIGHTS commands.</p> <p>15. In the sub-module ED_EXP of the PARROT module, it is also possible to list out all the calculated equilibrium points (using the LIST_ALL_EQUILIBRIA command) and to immediately calculate property diagrams with selective stepping options in which the calculations and the set stepping variable are based on a specific experimental point (using the GRAPHICS_PLOT command, which is identical to the POLY command STEP_WITH_OPTIONS).</p> <p>16. The complete Revised HKF model for aqueous solution has been fully implemented and extensively testing, along with the AQS2 aqueous solution database.</p> <p>17. The POURBAIX module is extensively revised, and now it allows either the TCAQ2 database (SIT model) or AQS2 (HKF model), to be used together with other appropriate databases for pure solid and solid solution phases in investigations of very complex heterogeneous interaction systems involving various aqueous solutions.</p> <p>18. Extensive and detailed on-line help can be obtained in various modules. The help files have been completely revised for all basic modules.</p> <p>19. The TCW full version 2 (compatible with TCCP) is released.</p> <p>20. The DICTRA version 22 (compatible with TCCP) is released.</p> <p>21. The updated demo versions for TCCP (TC4A vP) and for TCW2 (TC4U v2) are released.</p> <p>22. The TQ version 4 (compatible with TCCP) is released.</p> <p>23. The TCAPI version 2 (compatible with TCCP) is released.</p> <p>24. The TC-MATLAB® Toolbox 2 (compatible with TCCP) is released.</p> <p>25. A systematical naming strategy for database names are adopted (for detailed information, see <i>Part 4</i>).</p> <p>26. Encryption of commercial databases is applied; whilst, upon user's special request made to TCS and on the condition a TCS Database Agreement has been signed, a user can obtain the un-encrypted simple-textual version of a specific commercial database.</p> <p>27. Many new or updated databases (from TCS and its partners) are released in TCCP and TCW2:</p> <ul style="list-style-type: none"> • PURE4 (for pure elements; based on PURE3) • SSUB3 (for pure compounds/substances and gaseous species, v3) • SSOL4 (for alloy solutions, v4; based on SSOL2 and NSOL4) • TCFE3 (for steels/Fe-alloys, v3; based on TCFE2/TCFE2K) • TCMP2 (for materials processing, v2; based on TCER1) • SLAG2 (for Fe-containing and other types of slag, v2; based on SLAG1) • ION2 (for ionic solutions, v2) • TCAQ2 (for aqueous solutions using the SIT model, v2) • AQS2 (for aqueous solutions using the HKF model, v2;) • GCE2 (for geochemical and environmental systems, v2) • SEMC2 (for semi-conductors, v2) • NSOL4 (for alloy solutions, v4); but it is replaced by SSOL4 later on) • NSLD2 (for multicomponent solder alloys, v2) • NAL5 (for Al-based alloys, v5) • NOX2 (for oxide solution phases, v2) • SNOB1 (for noble-metal alloys, v1) • USLD1 (for multicomponent solder alloys, v1) 	
--	--	--

Q	Jun. 2004	<p>1. TDB Module:</p> <ul style="list-style-type: none"> • Since the TCCP version, all commercial databases are encrypted to avoid illegal copying. This has caused some problems in the simultaneous use of databases of different types. Efforts have been done to remove those inconveniences and it is now possible to combine encrypted databases with each other or with public databases. • A new option AFTER can be used for the keyword TYPE_DEFINITION in TDB files. It is similar to the GES option but executed after all parameters have been entered. It is specifically useful for changing binary excess models. (See TCCQ Example 50). • A new keyword ADD_REFERENCES can now be used in TDB files in order to have more reference lists. • Several ASSESSED_SYSTEMS keywords can now be used in the same TDB file in order to have more lists of assessed systems. <p>2. GES Module:</p> <ul style="list-style-type: none"> • Two unsymmetrical binary excess models, simple and legendre polynomials, have now been implemented in addition to the default symmetrical Redlich-Kister model. A switch to these models can be achieved through the new sub-option MIXED_EXCESS_MODELS to the option EXCESS_MODEL of the AMEND_PHASE_DESCRIPTION command. (See TCCQ Example 49). • The Toop-Kohler ternary extrapolation method has been added. There is now a new option TERNARY_EXTRAPOL for the AMEND_PHASE_DESCRIPTION command, under which one can select different extrapolation methods. • The quasi-chemical model developed by Kongoli <i>et al.</i> in the F*A*C*T group in Montreal has been implemented on the basis of the associate liquid model. This can be done through choosing the new option QUASICHEMI_KONGOLI which is available to the AMEND_PHASE_DESCRIPTION command. The working associates or species must have stoichiometries defined according to the model. (See TCCQ Example 49). • A new command ADD_COMMENT is now available, which gives users opportunity to jot down notes about parameters or something else that may be difficult to figure out in the future. ADD_COMMENT will first list whatever comment is stored, and then allow deleting, amending or adding to the comment text. • The AMEND_PHASE_DESCRIPTION command has now a new option DEFAULT_STABLE, which allows users to have a better guess of which phases that should be stable at the first calculation. • The AMEND_PHASE_DESCRIPTION command has now a new option SITE RATIOS, which can be used to change the number of sites in a single-sublattice phase to anything other than 1. <p>3. POLY Module:</p> <ul style="list-style-type: none"> • The SPECIAL_OPTIONS command has now a new option NEVER_ADJUST_MINIMUM_Y, which turns off the automatic adjustment of numerical precision. • The SPECIAL_OPTIONS command has now a new option TOGGLE_ALTERNATE, which is very useful in parameter optimization from experimental data (in PARROT/Ed-EXP module). • The SPECIAL_OPTIONS command has now a new option SHOW_OUTPUT_FILE, which asks for a name of a simple textual file, where the output from the command SHOW_VALUE shall be written. The output will still appear on the screen. • The STEP_WITH_OPTIONS command has now a new option MIXED_SCHEIL, which can lead to a Scheil-Gulliver solidification simulation with back diffusion of interstitials in solid phases if all the conditions are set up correctly. Users are recommended to go directly to the extended SCHEIL_SIMULATION module and make simulations automatically. • Generally speaking, the convergence of equilibrium calculation has been improved. For problems with calculating the first equilibrium, the COMPUTE-EQUILIBRIUM * is recommended. • Elements other than C can be also chosen as the fast moving interstitial components, and more than one such interstitials (<i>e.g.</i>, C, N, O, S, <i>etc.</i>) can now be considered during paraequilibrium calculations in the SPECIAL_OPTIONS and/or STEP_WITH_OPTIONS commands. (See TCCQ Example 42). 	TCS Newsletter No. 29
---	-----------	--	--------------------------

	<p>4. <u>POST</u> Module:</p> <ul style="list-style-type: none"> • The ADD_LABEL and MAKE_EXPERIMENTAL_DATAFILE commands have been improved. It is now possible to store the added labels and their locations in the workspace, and it may later be saved to an EXP file. • The GRAPH window has been further improved. It now allows freely moving and easily editing any plotted/added labels or texts on a diagram. <p>5. <u>PARROT</u> and <u>ED_EXP</u> Modules:</p> <ul style="list-style-type: none"> • Experimental data files can be created from POP files by using the new command GRAPHICS since TCCP. This has now been improved to allow several datasets and using different symbols according to user's specification. • The ALTERNATE mode has been improved. It is now possible to include functions to be evaluated together with an alternate calculation and to select ALTERNATE mode for each experimental equilibrium. • The POLY command SPECIAL_OPTIONS has now a new option TOGGLE_ALTERNATE, which can set the use of alternate technique in PARROT (ED-EXP) for each experimental equilibrium calculation to DEFAULT, ALWAYS or NEVER. • The LIST_RESULT command has a new 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. <p>6. <u>REACTOR</u> Module:</p> <ul style="list-style-type: none"> • The REACTOR module has been upgraded. It can simulate not only steady-state processes but also dynamic ones that change with time, for example decarburization of liquid steel. <p>7. <u>SCHEIL</u> Module:</p> <ul style="list-style-type: none"> • The SCHEIL_SIMULATION module has one new option now for users to define one or more fast diffusing components (usually interstitials, such as C, N, O, S, etc.) so that their back diffusions in solid phases can be automatically taken into account during the Scheil-Gulliver simulation. For steels with primary ferrite, an additional option (allowing BCC→FCC transition) has been given to consider its possible transformation to austenite. (See TCCQ Example 48). <p>8. Three more examples (TCCQ Examples 48 through 50) are added, and several existing examples are further revised.</p> <p>9. Extensive and detailed on-line help can be obtained in various modules. The help files have been completely revised for all basic modules.</p> <p>10. The TCW version 3 (compatible with TCCQ) is released.</p> <p>11. The DICTRA version 23 (compatible with TCCQ) is released.</p> <p>12. The updated demo versions for TCCQ (TC4A vQ) and for TCW3 (TC4U v3) are released.</p> <p>13. The TQ version 5 (compatible with TCCQ) is released.</p> <p>14. The TCAPI version 3 (compatible with TCCQ) is released.</p> <p>15. The TC-MATLAB® Toolbox v3 (compatible with TCCQ) is released.</p> <p>16. Many updated or new databases (from TCS and its partners) are released in TCCQ and TCW3:</p> <ul style="list-style-type: none"> • All ten public databases (<i>i.e.</i>, PURE4, PSUB, PBIN, PTER, PKP, PCHAT, PG35, PION, PAQ2 and PGEO) have been further modified, and their new sub-versions are freely included in TCCQ and TCW3. • Some of the commercial databases (<i>e.g.</i>, SSUB3, SSOL2 and SSOL4, TCMP2, SLAG2, ION2, TCAQ2, AQS2, GCE2, SMEC2, NSLD2, SNOB1, TTNI6, TTAl3, TTMs2) have been further revised (mostly as sub-versions), and such updated versions will be provided if the user has the valid database license for the same major version number (<i>e.g.</i>, from TCMP2.1 to TCMP2.2); otherwise, a upgraded version from an older major version of a specific database (<i>e.g.</i>, from TCMP1.1 to TCMP2.2) is only available on the condition that such a database-upgrade is purchased. • Recently, the entire copyrights and intellectual rights of the NUMT2, NUOX4, NUTO1 and NUTA1 databases that were previously owned by AEA Technology plc. (UK), UES Software Inc. (USA) and ESI Group (Switzerland) have been brought by and transferred to Thermo-Calc Software AB. 	
--	---	--

R	Aug. 2006	<p>1. <u>POLY Module:</u></p> <ul style="list-style-type: none"> • One of the major improvements in this new version of the Thermo-Calc software, TCCR and TCW4 (as well as the new version of various Thermo-Calc programming interfaces) is that the newly-implemented Global Minimization Technique is used to assure that the present minimum in an equilibrium calculation is the most stable minima for the specified conditions. The new technique, which is based on the <u>traditional GEM (Gibbs Energy Minimization) Technique</u> (<i>i.e.</i>, the ordinary POLY minimization routines used in previous versions), will ultimately prevent a calculation from reaching an undesired metastable or unstable (local) equilibrium in a defined system, and automatically detect possible miscibility gap(s) and create additional composition sets in a solution phase if needed for handling single or multiple miscibility gaps (<i>i.e.</i> there will no longer be necessary for the user to specify these in advance). • The Global Minimization Technique is normally used as default in a single-point equilibrium calculation when performing the COMPUTE_EQUILIBRIUM command, but can of course be turned <i>off</i> (<i>and on again</i> by repeating this command) by the user (see more below). • In the current versions TCCR and TCW4, fully-supported equilibrium conditions in the Global Minimization Technique are: T, P, N, N(component), X(component), B, B(component) and W(component). If other types of equilibrium conditions are used (in the current versions TCCR and TCW4), after the initial POLY optimization (using the traditional GEM Technique), a Global Minimization test and corrections are performed until the lowest minimum is found. • Concerning the single-point equilibrium calculations, there has been some necessary changes in how to use the COMPUTE_EQUILIBRIUM command: <ul style="list-style-type: none"> ➤ COMP_EQ Using the new Global Minimization Technique ➤ COMP_EQ - Using the traditional GEM Technique ➤ COMP_EQ * Since the Global Minimization Technique is even more robust, when the Global Minimization mode is on, it is not that useful anymore! It can be appropriately used ONLY after the Global Minimization mode has already been disabled. • For stepping and/or mapping calculations, the Global Minimization Technique is currently only performed during the initialization (for finding the starting points), but not during the actual stepping/mapping. This strategy has been selected in order to keep computational times short. A complete rewrite of the STEP and MAP commands is ongoing and will be made available in the next major-version release. • The new command SET_MINIMIZATION_OPTION can be used for deciding on how to perform the Gibbs energy minimizations in the POLY module (also in TCW4 but through choosing the option “Global Minimization...”). By default, the POLY module in TCCR/TCW4 uses the Global Minimization Technique for various single-point equilibrium calculations (and during the initialization stages of stepping or mapping calculations). The user can use this command to permanently (for all sequential calculations throughout the current TCC run) turn off the Global Minimization Technique (<i>Of course, the same command can be used again to switch on the Global Minimization mode afterwards</i>) and consequently use only the normal POLY optimization routine (which is similar to that in previous versions <i>e.g.</i> TCCQ/TCW3, TCCP/TCW2 and so on). With this command the user can additionally adjust the way 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 way on how to control the steps in reach Gibbs energy minima in an equilibrium state, in order to improve convergence. <p>2. <u>GES Module:</u></p> <ul style="list-style-type: none"> • The Generalized PVT Model (or General Thermal EOS Model, developed by Chen and Sundman, 2005) has been fully implemented into the Thermo-Calc software, which handles the calculations of volume contributions and pressure-dependence of Gibbs free energies in various 	<p>TCS Newsletter Nos. 31, 32 & 33</p>
---	-----------	---	--

	<p>types of materials [<i>e.g.</i>, metallic alloys/liquids and other varied forms of solids/minerals (carbides, nitrides, hydrides, oxides, sulfides, hydroxides, borides, phosphites, halites, silicates, carbonates, sulphates, nitrates, phosphates, <i>etc.</i>)]. This new model is based on fundamental physics; compared to the traditional Murnaghan Model, Birch-Murnaghan Model and Modified Birch-Murnaghan Model (all three had been implemented inside the Thermo-Calc software and used in some existing Thermo-Calc databases), it doesn't produce anomalies at high temperature and pressure, and allows the dependence of volume, thermal expansion, bulk modulus/compressibility and some other physical properties on temperature and pressure to be precisely described/calculated with less parameters. Thanks to its significant advantages over the traditional PVT models mentioned above, the new model has been applied in the critical assessments of volume data in various steels/Fe-alloys, and the resulted data set (based on the work by Lu and Sundman, 2005) has become a merged part of the newly-developed TCFE4 database (updated from TCFE3).</p> <ul style="list-style-type: none"> • The <code>AMEND_PHASE_DESCRIPTION</code> command has now a new option <code>FRACTION_LIMITS</code>, which allows users to set applicable compositional limits of a solution phase (in terms of mole-fractions of elements in the phase), for which a certain thermodynamic model has been used while the model itself and related critically-assessed data can only be used within a specified composition range. For instance, the <code>FE_liquid</code> solution phase in the SLAG database is only applicable to Fe-rich liquid in slag-bearing interaction system, and it is necessary (in the current versions of software TCCR/TCW4 and of Thermo-Calc programming interfaces, where the newly-implemented Global Minimization Technique is used) to set such applicable compositional limits for the concerned solution phase; otherwise, some additional composition sets [of possible miscibility gap(s)] may be automatically generated whilst such composition sets may be located outside the applicable range of the chosen model and data. Similar cases are the <code>AQUEOUS</code> solution phase in the TCAQ2/PAQ2 and AQS2 databases. One can also make use of this new option for some solution phases in any user-specified database and/or in any particular calculations. <p>3. <u>TDB Module:</u></p> <ul style="list-style-type: none"> • The new option <code>FRACTION_LIMITS</code> in the <code>GES</code> command <code>AMEND_PHASE_DESCRIPTION</code> (as described above) can also be used directly for the keyword <code>TYPE_DEFINITION</code> in TDB files. This is exactly that have been done in the SLAG2, TCAQ2/PAQ2 and AQS2 databases which are now distributed/used along with TCCR/TCW4; therefore, the users do not need to bother to amend the <code>FRACTION_LIMITS</code> in the phase descriptions of the SLAG and/or <code>AQUEOUS</code> solution phases when such databases are used. <p>4. <u>POST Module:</u></p> <ul style="list-style-type: none"> • Now it is possible to save a plotted diagram as an editable EMF (Enhanced Windows Metafile) graphical file under PC Windows XP/2000/NT4 environment. This can be done by choosing the new graphical format (using the <code>SET_PLOT_FORMAT</code> command and sequentially selecting the new device number 19) and plotting the diagram (using the <code>PLOT</code> command). Such a graphical file (with the default extension of EMF) can then be edited and manipulated using an appropriate graphical editor (such as in MS PowerPoint). • Using the new command <code>LIST_DATA_TABLE</code>, it is also possible to list on screen various properties (<i>i.e.</i>, the Y-axis variable on a property diagram after a stepping calculation) that change with an independent variable (<i>i.e.</i>, the X-axis variable on a property diagram), and/or export such property diagram data 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; • The GRAPH window has been further improved. It now allows: <ul style="list-style-type: none"> ➤ Automatically detect any plotted line/curve on a property diagram (the X-/Y-coordinates and the property name will be simultaneously displayed at the bottom part of the GRAPH window), by simply moving the mouse-cursor over or near the property lines. This feature is available only when a standard 	
--	---	--

	<p>variable (but not any user-specified or module-predefined symbol) has been chosen as one axis variable (and the independent variable as the other axis).</p> <ul style="list-style-type: none"> ➤ Automatically detect (or identify) stable phases inside regions of an isothermal or isoplethal phase diagram section (the X/Y-coordinates of the mouse-cursor and the stable phase-assemblage will be simultaneously shown up at the bottom part of the GRAPH window), by simply positioning the mouse-cursor inside the phase regions. ➤ Automatically add phase labels inside regions of an isothermal or isoplethal phase diagram section, by simply right-clicking the mouse (after positioning the mouse-cursor inside the phase regions) and then choosing the “Add Label” option on the action list displayed. ➤ Using the new option PLOT_SYMBOLS_AT_NODE_POINTS under the SET_PLOT_OPTION command, it is now possible to indicate that one would like to visualize the actual points calculated by plotting a symbol at each such node point, and the symbol size can be chosen by the user as well (using the new option SYMBOL_SIZE under the same command). ➤ By right-clicking the mouse on an already-plotted phase diagram or property diagram, the user can, from the action list displayed, choose to “Print” the diagram directly onto a connected printer, or “Save EMF to file” (in the fixed EMF form which can not be edited) or “Copy EMF to Clipboard”. On the same action list displayed, the user can also choose to add user-specified texts (through the “GD_ITEM/Add text item” option) or (as mentioned above) add automatically-detected/identified stable-phase-region labels on a plotted phase diagram (through the “Add Label” option, which of course does not work for a plotted property diagram). <p>5. <u>TERNARY</u> Module:</p> <ul style="list-style-type: none"> • The new option Liquid Surface has been added into the TERNARY module, which allows automatic calculations of not only monovariant lines involving liquid phase but also isothermal liquidus surface projections (at a defined temperature interval) of <i>already-assessed</i> ternary alloy systems. This option also automatically identifies and writes out the reaction types and reaction details of various invariant points, onto the plotted liquidus surface projection diagram on screen or saved graphical files, as well as in a simple textual file. • The further improved/modified TERNARY module now also accepts calculations of any ternary alloy system, even the switched-on database (especially some existing specific alloy solution databases) does not contain the specific ASSESSED_SYSTEM information for a selected ternary sub-system (in such a case, a warning message will be issued on screen). Of course, IT IS VERY IMPORTANT TO KEEP THE FOLLOWING IN MIND: <i>the user must well know that the chosen database must contain sufficient and appropriate data sets for the considered ternary sub-system as a whole which have already been critically-assessed in reality; otherwise, the obtained results from such a TERNARY-module calculation may be incomplete or totally wrong (in the worst cases)</i>. <p>6. <u>SCHEIL</u> Module:</p> <ul style="list-style-type: none"> • The further-revised/improved SCHEIL module now automatically superimposes a full-equilibrium solidification path [calculated through using the simple lever rules] onto the simulated freezing range simulated by the non-equilibrium method [<i>i.e.</i>, using the traditional Scheil-Gulliver model] or partial-equilibrium approach [<i>i.e.</i>, using the modified Scheil-Gulliver model which allows considering back diffusion of fast-diffusing components (usually interstitials) and possible BCC→FCC phase transformation]. However, such an overlapped graph is only available for the first diagram plotted immediately after a SCHEIL-module simulation [<i>i.e.</i>, on the default “Temperature vs Mole_Fraction_of_All_Solid_Phases” plot $T(^{\circ}C)$-NS]. <p>7. Several existing TCC examples are further revised.</p> <p>8. The structure of the <i>TCC User’s Guide</i> has been significantly changed: started from this TCCR version, five individual documents have been separately prepared, extracted from the relevant chapters of the TCC User’s</p>	
--	---	--

		<p>Guide. This is because that: <i>a</i>) the <i>User's Guide</i> has become bigger and bigger in later versions, and has thus turned to be more and more difficult to maintain and read; <i>b</i>) some chapters are (in some senses) independent upon software-versions, and are really common for other software (such as TCW and DICTRA) and application programming interfaces (such as TQ, TCAPI and TC-MATLAB Toolbox). The new document set includes the following five individual parts:</p> <ul style="list-style-type: none"> ➤ <i>TCCR User's Guide</i>; ➤ <i>Thermo-Calc Software System</i>; ➤ <i>Thermo-Calc Database Guide</i>; ➤ <i>DATAPLOT User's Guide and Examples</i>; ➤ <i>Thermo-Calc Reference List</i>. <p>Many chapters in these documents have been extensively revised and updated.</p> <ol style="list-style-type: none"> 9. The on-line help feature for various modules/commands has been accordingly revised/updated. 10. Other software and programming interfaces, which are compatible with TCCR, are released at the same time: <ul style="list-style-type: none"> ➤ TCW4 (TCW Software version 4); ➤ DICTRA24 (DICTRA Software version 24); ➤ Demo versions for TCCR (TC4A vR) and for TCW4 (TC4U v4); ➤ TQ6 (TQ Programming Interface version 6); ➤ TCAPI4 (TCAPI Programming Interface version 4); ➤ TC-MATLAB® Toolbox version 4. 11. The new public database, TCBIN (TC Binary Solutions Database, version 1) is free-of-charge included along with the TCCR/TCW4 software. However, unlike other public databases, <i>it is always and can only be distributed in the encrypted database form, and can only be used inside the BIN module (of TCCR) or Binary Phase Diagram module (of TCCW4)</i>. 12. Many new or updated commercial Thermo-Calc databases (from TCS and its partners) have been made available around the releasing time of the TCCR/TCW4 software: <ul style="list-style-type: none"> • TCFE4 is released. This database has been developed by TCS, based on the TCFE3 version, while critically-assessed data on molar volume (and its dependence on temperature, using the newly-implemented Generalized PVT Model) have been incorporated. • TTNi7 (updated from TTNi6), TTAI5 (updated from TTAI4/TTA13), TTMg3 (updated from TTMg2), TTTi3 (updated from TTTi2) and TTZr1 (new), developed by our partner ThermoTech, are released. • CCC1 (new, developed by our partner CCT Project) is released. • STBC1 (new, developed by our partners MPI-MF and SGTE) is released. • Some of the commercial databases (<i>e.g.</i>, SLAG2, ION2, SSOL4, TCMP2, TCAQ2, AQS2, NSLD2, NOX2) have been further revised (mostly as sub-versions), and such updated versions will be provided if the user has the valid database license for the same major version number (<i>e.g.</i>, from SLAG2.0/SLAG2.1 to SLAG2.2); otherwise, a upgraded version from an older major version of a specific database (<i>e.g.</i>, from SLAG1.1 to SLAG2.2) is only available on the condition that such a database-upgrade is purchased. • Some more new/updated Thermo-Calc databases are in the pipeline for further releases and can be used within the TCCR/TCW4 software. For details, refer to the further-coming Thermo-Calc Software Newsletters. 	
S	Jun. 2008	<ol style="list-style-type: none"> 1. <u>POLY Module</u>: <ul style="list-style-type: none"> • The Global Minimization Technique has been further improved on a greater scope, in the new version of the Thermo-Calc software, TCCS and TCW5 (as well as the new version of various Thermo-Calc programming interfaces TQ7, TCAPI5 & TC-MATLAB Toolbox5). <ul style="list-style-type: none"> ➤ It has been made as a standard routine for not only single-point calculations (as in the TCCR and TCW4) but also for stepping and mapping calculations (which have been rewritten in TCCS/TCW5). Therefore, it is now very powerful and efficient in automatically detecting possible miscibility gap(s) and creating additional composition sets in a solution phase if needed for handling single or multiple miscibility gaps (<i>i.e.</i> there will no longer be necessary for the user to specify these in advance) for all 	TCS Newsletter No. 34

	<p>types of calculations (of complex heterogeneous stable/meta-stable equilibrium states, phase diagrams and property diagrams of multicomponent system).</p> <ul style="list-style-type: none"> ➤ Through the POLY command ADVANCED_OPTIONS (with the “GLOBAL_MINIMIZATION” or “STEP_AND_MAP” option) or the SYS command SET_TC_OPTIONS, the Global Minimization Technique can be used on a full or limited scale, and, of course, it can also been turned off completely. If not turned off, a Global Minimization test of the equilibrium is done at the beginning and end of each phase region. Furthermore, along a phase region, equilibria are tested at an interval set by the user; by default such tests are done inside regions for STEP but not for MAP. ➤ The Global Minimization algorithm in TCCS/TCW5 has been improved under most conditions leading us to reduce the default number of grid points from 20000 in TCCR/TCW4 to 2000 in TCCS/TCW5. This reduction makes equilibrium calculations much faster while improvements in the algorithm keep and in many situations improve the robustness for the single-point, stepping and mapping calculations. • For single-point calculations (via the COMPUTE_EQUILIBRIUM command) of full stable equilibria or partial meta-stable equilibria, the improved Global Minimization Technique is more robust and faster, so that it can <i>usually</i> (if hesitating in saying <i>always</i>) find the global Gibbs energy minimum and properly handle all the possible phase-splitting for stable solution phases with complex miscibility-gaps (e.g., FCC_A1, BCC_A2 and HCP_A3 phases in C/N-bearing multicomponent alloy systems). • For stepping calculations (via the STEP command) of various types of property diagrams, the Global Minimization Technique can ensure the automatic detections of possible phase-splitting for stable solution phases with complex miscibility-gaps in the defined system, as well as enforce precise calculations of global Gibbs energy curves along the varied controlling conditions. The rewritten STEP NORMAL routine can now regularly use Global Minimization to test that the current equilibrium is the global and also at every phase change; and as a direct result, it in principle is not sensitive to any user-specified starting point in a defined multicomponent system. • For mapping calculations (via the MAP command) of various types of phase diagrams, the Global Minimization Technique can guarantee the automatic detections of possible phase-splitting for stable solution phases with complex miscibility-gaps in the defined system, as well as enforce precise calculations of global Gibbs energy surfaces over the changing conditions. Thanks to this great improvement, it is in principle unnecessary to set/add any user-specified starting point for calculating various types of phase diagrams in a defined multicomponent system. • The new POLY command ADVANCED_OPTIONS replaces and combines the SPECIAL_OPTIONS command (that had been available in earlier versions up to TCCR) and SET_MINIMIZATION_OPTIONS command (that was added in TCCR). Four options previously under SPECIAL_OPTIONS are now renamed under ADVANCED_OPTIONS (<i>mainly for the purpose of removing ambiguities when specifying the first few letters and thus tying only fewer letters is faster and easier</i>): the “SET_MISCIBILITY_GAP” option is replaced by “NEW_COMPOSITION_SET”, “SET_MAJOR_CONSTITUENTS” by “MAJOR_CONSTITUENTS”, “SET_PRESENT_PHASE” by “PRESENT_PHASE”, and “SET_PHASE_ADDITION” by “PHASE_ADDITION”. Furthermore, five options that were previously available under the SPECIAL_OPTIONS command, <i>i.e.</i>, “MISC_GAP_TEST_INTERVAL”, “SET_BREAK_CONDITION”, “OUTPUT_AT_MAP_AND_STEP”, “NEVER_ADJUST_MINIMUM_Y” and “SHOW_OUTPUT_FILE”, have been removed, because they are no longer necessary or relevant. • The new advanced options EQUILIBRIUM_CALCUL and GLOBAL_MINIMIZATION and STEP_AND_MAP (which are replacing the command SET_MINIMIZATION_OPTION within the TCCR/TCW4 version and are also extended in TCCS/TCW5) have been greatly modified/improved, and merged as integrated parts of the new command 	
--	---	--

	<p>ADVANCED_OPTIONS. A user can choose to enforce the Global Minimization Technique on either a full scale (<i>i.e.</i>, “Use global minimization as much as possible”) or on a limited scale (<i>i.e.</i>, “Use global minimization for test only”), or alternatively decide to completely turn it off (<i>i.e.</i>, if typing N to both of aforementioned questions). Additionally, a user can alter the default settings for the following calculation sub-options:</p> <ul style="list-style-type: none"> ➤ “Force positive definite phase Hessian” for normal POLY Minimizations (default as Y); ➤ “Control stepsize during minimization” for normal POLY Minimizations during stepping calculations (default as Y); ➤ “Use global minimization as much as possible” for Global Minimizations (the default input is normally Y, for which on start-up may be changed by the SYS command SET_TC_OPTIONS); ➤ “Use global minimization for test only?” for Global Minimizations (default as N); ➤ “Maximum number of grid-points per phase” for Global Minimizations (default as 2000); ➤ “Global Test Interval” for Global Minimization tests during stepping/mapping calculations (the default value is normally 0, for which on start-up may be changed by the SYS command SET_TC_OPTIONS); ➤ “Automatically add initial equilibria” during stepping/mapping calculations (the default input is normally Y, for which on start-up may be changed by the SYS command SET_TC_OPTIONS); ➤ “Number of mesh along an axis” during stepping/mapping calculations (default as 3); <i>and</i> ➤ “Use inside meshing points” during stepping/mapping calculations (default as Y). <p>The default values for three of the above-mentioned sub-options can be defined by the user on start-up in the SYS module, using the SET_TC_OPTIONS command, for its corresponding sub-options:</p> <pre style="margin-left: 40px;">“USE_GLOBAL_MINIMIZATION /YES/” “GLOBAL_TEST_INTERVAL /0/” “AUTOMATICALLY_ADD_INITIAL_EQUILIBRIA /YES/”</pre> <ul style="list-style-type: none"> • The new option SHOW_FOR_T has been added into the new command ADVANCED_OPTIONS, for the purposes of showing (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 <i>e.g.</i> 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. • The RECOVER_START_VALUES command has been deleted from the POLY-Module menu, as it has never been relevant in either POLY or PARROT calculations. • The IWSW dimension in the PLOY workspace has been increased from 40000 to 200000, and thus one can not read POLY3 files created with TCCS/TCW5 in earlier versions. <p>2. <u>GES Module:</u></p> <ul style="list-style-type: none"> • The <i>Four Substitutional-Sublattice Ordering Model</i> (which always requires that a solution phase must have 4 sublattices for substitutional ordering and can additionally have an interstitial sublattice) has been extended for chemically-ordered FCC (or HCP) solution phases (<i>i.e.</i>, with the phase-type F), and also implemented for chemically-ordered BCC solution phases (<i>i.e.</i>, with the new phase-type B). 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 	
--	--	--

	<p>can be: _L12 or _L10 for ordered FCC, or _A1 for disordered FCC; _B2, _B32, _D03 or _L21 for ordered BCC, or _A2 for disordered BCC; _D019 and _B19 for ordered HCP, or _A3 for disordered HCP.</p> <ul style="list-style-type: none"> The Quasichemical_Ionic Model has been extended that it allows specification of the Z factor (number of bonds per atom). For various encrypted databases, the GES commands LIST_PHASE_DATA or LIST_DATA can now also list definitions of phases/phase-constituents and references (but not thermodynamic parameters) for a defined system. The MIXED_EXCESS_MODELS option for amending EXCESS_MODEL of binary interactions (in a subsitutational solution phase (without sublattice), as well as the TOOP_KOHLER model for amending TERNARY_EXTRAPOLT method, have been improved. The new standard Example 52 in the TCCS Examples Book demonstrates the ways of using such improved models. <p>3. <u>TDB Module:</u></p> <ul style="list-style-type: none"> The CASE directive in TDB files is now working, that makes it possible to have additional TDB, GES or POLY commands executed depending on the user selection of elements, species or phases . The references for a defined system retrieved from the database are now stored in the GES5 file and can be correctly listed in the GES module any time (even after reading a POLY3 file). In order to handle chemically-ordered solution phases with partitioning (in 4-substitutional sublattices), the phase-type option F for ordered FCC/HCP phases has been improved and simplified, and the phase-type option B for ordered BCC solution phases has been implemented. A phase can now have several so-called ADDITIONAL parts of different types (that are enforced by certain TYPE_DEFINITIONS which call the GES command AMEND_PHASE_DESCRIPTION for describing various contributions to Gibbs energy); however, the TDB/GES modules will automatically delete any early-defined ADDITIONAL part(s) of the same type. Now the TDB module can now selectively retrieve functions which are necessary for a defined system from a database that has functions stored in its setup file or SEQ sequential function file, while all other functions irrelevant for the defined system are simply ignored and will not be saved in associated GES5 and POLY3 workspaces. Previously, this can only be done for large databases that have functions stored in RND1 random or FTP function files. <p>4. <u>POST Module:</u></p> <ul style="list-style-type: none"> Two new options, <i>i.e.</i>, “Always initiate POST on re-entering” and “Always solid line”, have been added in the SET-PLOT-OPTION command: <ul style="list-style-type: none"> 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 a 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. The IDGMAR dimension in the POST workspace has been increased from 500 to 2000, so that one can now plot extremely complex diagrams with many phase boundaries.
--	--

	<p>5. <u>PARROT and ED-EXP Modules:</u></p> <ul style="list-style-type: none"> The command <code>SET_SCALED_VARIABLE</code>, which allows specifying a lower and upper limit for an optimizing variable, now works correctly. The optimizer tries a little harder in cases when the initial set of optimizing variables is far away from the solution. In the POP file and in the ED-EXP module, one can use the new type of symbols in the <code>ENTER-SYMBOL</code> command for ordered phases (treated by the so-called <i>Four Substitutional Sublattice Ordering Model</i>) which have the phase-type of F or B. This is called <code>INTERNAL</code> and can be specified as L12, L10, D019, B19, D03, L21, B2 or B32; in addition, a phase and its composition-set must be specified. The symbol value will be zero if the phase is not ordered according to the specification; otherwise, it will have a positive value depending on the degree of order, with a maximum value of unity. This is useful when optimizing ordered phases (such as ordered FCC, HCP and BCC phases), as the order may change during the optimization. The <code>LIST-RESULT</code> command have a slightly-modified options <code>D</code> to list all experiments, including those with fulfilled inequalities which are suppressed with the default option <code>C</code>. The option <code>G</code> for the <code>LIST-RESULT</code> command has been improved, so that the graph is automatically plotted and the user can scale it giving commands in the POST module. The new <code>SET_EXTERNAL_PARAMETER</code> command has been added in the PARROT module, for the purpose of optimizations of parameters used in so-called external models (which may not be as fully-implemented/integrated parts inside the GES system). Normally, the Global Minimization Technique for calculating equilibria is turned off during optimization. However, it is possible to turn it on for (some) individual experiment points with the <code>POLY/ED_EXP</code> command-sequence <code>ADVANCED-OPTION TOGGLE-ALTERNATE</code>. <p>6. <u>TERNARY Module:</u></p> <ul style="list-style-type: none"> All the calculation routines for three TERNARY-Module options Phase Diagram (for calculating isothermal sections), Monovariants (for calculating monovariant lines involving liquid) and Liquid Surface (for calculating liquid surface projections) have been further modified and improved, so that the Global Minimization Technique can be used on a full scope. <p>7. <u>SCHEIL Module:</u></p> <ul style="list-style-type: none"> When a database (such as TCFE6/TCFE5/TCFE4) containing volume data (<i>i.e.</i>, molar volume, thermal expansivity, and compressibility) has been used for a multicomponent alloy systems, the further-revised/improved SCHEIL-Module can now plot diagrams using one of four additional properties varied during the solidification process [<i>i.e.</i>, CP/BCP – Apparent heat capacity per mole/gram, NV/NV (ph) – Molar volume of the system or a phase, DS/DS (ph) – Average density of the system or a phase, and BT – Apparent volumetric TEC of the system] as an axis variable. The further improved SCHEIL-Module can now also handle back diffusion of C in cases where graphite appears during solidification as in cast irons. For previous versions, the simulations always stop where graphite comes out; that was due to that the activity of C in graphite is always unity, then one cannot make the chemical potential of C the same in every phase. In the new treatment, the graphite is simply frozen at its amount, and does not take part in the redistribution of C among various phases; of course, the mass conservation of C is still kept. <p>8. Several existing TCC examples are further revised. Moreover, two new standard TCC examples have been added:</p> <ul style="list-style-type: none"> + TCEX52 for amending <code>MIXED_EXCESS_MODELS</code> for binary excess models and <code>TOOP_KOHLER</code> for ternary extrapolation method; + TCEX53 for calculating the so-called Pourbaix diagrams in aqueous-bearing heterogeneous interaction systems through the ordinary TDB-GES-POLY-POST module-routines (rather than via the special POURBAIX module as shown in TCEX40). 	
--	--	--

	<p>9. The <i>TCC User's Guide</i> has been further improved; and (similar to TCCR) the new document set for TCCS includes the following 5 individual parts:</p> <ul style="list-style-type: none"> ➤ <i>TCCS User's Guide</i>; (<i>this document</i>) ➤ <i>Thermo-Calc Software System</i>; ➤ <i>Thermo-Calc Database Guide</i>; ➤ <i>DATAPLOT User's Guide and Examples</i>; ➤ <i>Thermo-Calc Reference List</i>. <p>Many chapters/sections in these documents have been extensively revised and updated.</p> <p>10. The TCC on-line help features for various new and modified modules/commands have been accordingly revised and updated.</p> <p>11. Other software and programming interfaces, which are compatible with TCCS, are released at the same time:</p> <ul style="list-style-type: none"> ➤ TCW5 (TCW Software version 5); ➤ DICTRA25 (DICTRA Software version 25); ➤ Demo versions for TCCS (TC4A v5) and for TCW5 (TC4U v5); ➤ TQ7 (TQ Programming Interface version 7); ➤ TCAPI5 (TCAPI Programming Interface version 5); ➤ TC-MATLAB® Toolbox version 5. <p>12. Many new or updated commercial Thermo-Calc databases (from TCS and its partners) have been made available around the releasing time of the TCCS/TCW5 software:</p> <ul style="list-style-type: none"> • TCFE6 is formally released. This database has been developed by TCS, based on the TCFE5 and TCFE4 versions (that were released along with TCCR/TCW4 in 2006 and 2007, respectively). • TTAl6 and TTMy4 (developed by our partner ThermoTech; updated from TTAl5 and TTMy4, respectively), as well as SNUX6 (developed by SGTE/ThermoData and TCS, based on NUOX4 that is now owned by TCS), were released along with TCCR/TCW4 in 2007, and now become available for uses within the TCCS/TCW5 software. • Some commercial databases (<i>e.g.</i>, TCNI1, TCMP2, SNOB1, TCAQ2 and AQS2) have been further revised as sub-versions (<i>i.e.</i>, TCNI1.2, TCMP2.5, SNOB1.2, TCAQ2.5 and AQS2.5); such updated sub-versions are provided and can be used within the TCCS/TCW5 software if the user has the valid database license for the same major version number (<i>e.g.</i>, from TCMP2.3/TCMP2.4 to TCMP2.5 at no extra cost); otherwise, a upgraded version from an older major version of a specific database (<i>e.g.</i>, from TCMP1 to TCMP2.5) is only available under the condition that such a database-version-upgrade has been purchased. • For the commercial database SSOL4, a further-revised sub-version SSOL4.10 will be made available in the fall of 2008; such a updated sub-version will be provided with the TCCS/TCW5 software (to be distributed on the <i>TCS Standard Product CD</i>) if the user has the valid database license for the same major version number (<i>e.g.</i>, from SSOL4.8/SSOL4.9 to SLAG4.10 for free); otherwise, a upgraded version from an older major version (<i>e.g.</i>, from SSOL2 to SSOL4.10) is only available under the condition that such a database-version-upgrade has been purchased. <p>13. Public Thermo-Calc databases (within the TCCS/TCW5 software):</p> <ul style="list-style-type: none"> • PURE4, PBIN, PTERN, PG35 and PGEO have been slightly updated/modified, and their corresponding sub-versions are now PURE4.6, PBIN1.2, PTERN1.3, PG35_1.2 and PGEO1.2, respectively. • PAQ2 (as PAQ2.4) has been extensively revised in its file-structure, and PAQS2 (as PAQS2.4) is now distributed for free. For the included aqueous solution phase, they use the SIT Model and Complete Revised HKF Model, respectively. 	
--	--	--

1.5 General Structure of the Thermo-Calc Package

The general structure of the Thermo-Calc package is illustrated in *Figure 1-2*.

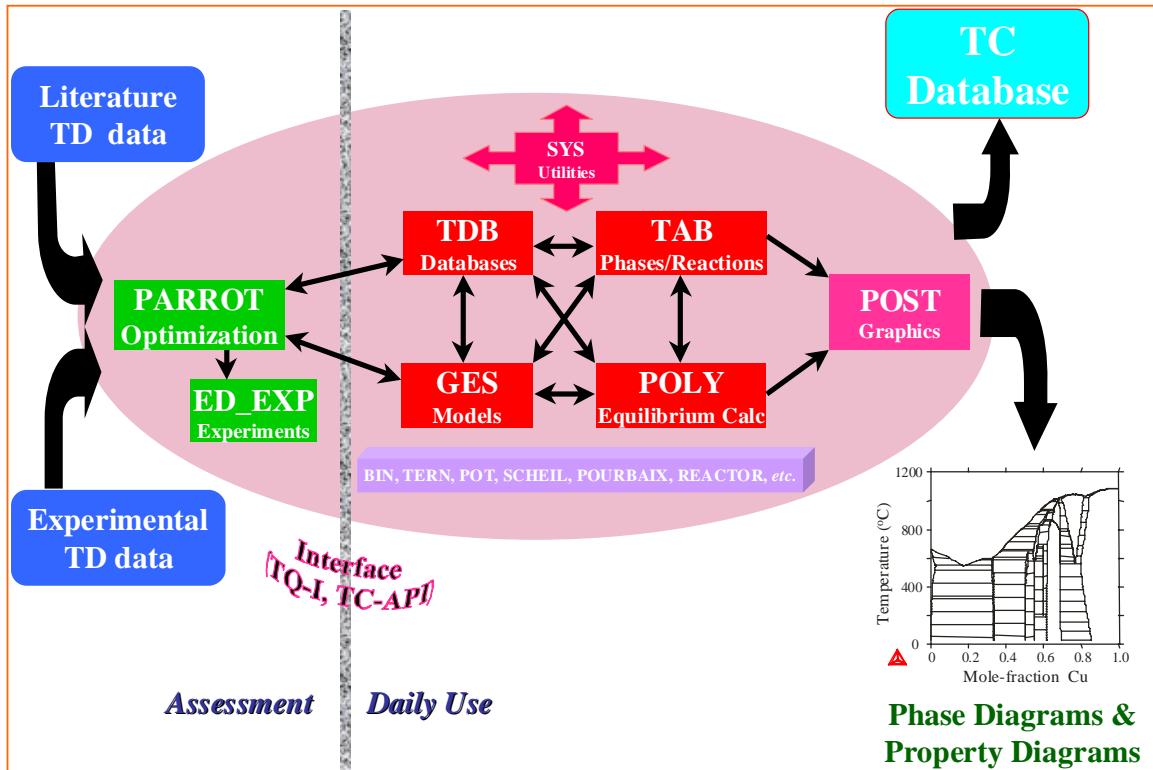


Figure 1-2. General Structure of the Thermo-Calc Package

The TCC (Thermo-Calc Classic) software consists of seven basic modules, *i.e.*,

- TDB for database retrieval and management;
- GES for thermodynamic model handling and data treatments for various phases;
- TAB for thermodynamic property tabulations of phases and reactions;
- POLY for multicomponent heterogeneous equilibrium and stepping/mapping calculations;
- POST for post-processing of various phase diagrams and property diagrams;
- PARROT for parameter optimizations in data assessments/ovulations; *and*
- ED_EXP for experimental points editing and equilibrium calculations.

All of these basic modules are generally necessary for making a thermochemical calculation or simulation. Further, these modules are internally connected to each other, while there are separate workspaces for some individual modules (*e.g.*, SYS, GES, POLY and PARROT). The POST module is often referred as a sub-module of the POLY module, and ED_EXP a sub-module of PARROT; the direct pathways to these two sub-modules are always started from the POLY and PARROT modules, respectively; and the workspaces are mostly (but not completely) shared between POLY and POST, and between PARROT and ED_EXP. There is also a bridge connecting the TAB and POST modules.

In the TCC software, there are also some special modules, or so-called easy-to-use modules (except for the last one REACTOR), for specific calculations and simulations, *i.e.*,

- BIN for binary phase diagram calculations;
- TERN for ternary phase diagram calculations;
- POT for potential diagram calculations;

- POURBAIX for Pourbaix diagram and property diagram calculations;
- SCHEIL for Scheil-Gulliver solidification simulations; and
- REACTOR for steady-state reaction simulations.

These special modules are designed in special ways that the users do not need to interact directly with the basic modules (except for the REACTOR module which has a similar user-interaction manner as for the basic modules); instead, just answers to some simple questions are needed. The software/database system automatically makes the calculations and simulations and then presents the results as graphs in a high professional standard. More of such modules for specific purposes will gradually be implemented into the Thermo-Calc package in the near future.

The TCC software always starts with the SYS module, which is designed for each computer system and its environment settings (*i.e.*, interactions with various operation systems). This module also usually acts as a kind of communication center with accessibility to all basic and special modules.

The TCW (Thermo-Calc Windows) software has a different user-interface, while its built-in module-structure is rather similar to the TCC software in many senses. For the details of TCW software structure, please refer to the TCW User's Guide and TCW Examples Book.

Beside the TCC and TCW software, two *programming interfaces*, namely the TQ and TCAPI, can be connected with the Thermo-Calc Engine (with DICTRA extensions), as well as with Thermo-Calc (and DICTRA) databases. These programming interfaces are designed for users who want to do application-oriented programming in their own user-written programs or third-party software packages, for other types of materials property calculations and materials process simulations. The powerful Thermo-Calc Engine provides accurate, reliable and fast thermodynamic calculations. The interfaces also service Thermo-Calc software/database-related and currently-existing toolboxes in some third-party software packages (such as TC-MATLAB Toolbox in the MATLAB® package) in order to apply thermochemical calculations and simulations to many different fields.

Through these software/databases and programming interfaces, all users may make daily use of thermochemical calculations and simulations, as well as perform their own assessment work. Users are highly encouraged to also build their own application programs, utilizing the Thermo-Calc Engine (with DICTRA extensions) through various programming interfaces in various R&D activities.

1.6 Availability of the Thermo-Calc Package on Various Hardware

Since May 2000, the Thermo-Calc software is identified as two separate versions as described below:

Thermo-Calc Classic	(TCC)	---	Command-line based Thermo-Calc software
Thermo-Calc Windows	(TCW)	---	Windows-GUI based Thermo-Calc software

The *Thermo-Calc Classic* was developed in the very beginning of Thermo-Calc history, and it will exist in all forth-coming versions. Several of its unique features will be continuously appreciated by many users, especially those who have already used such a version for some years. For instance, the MACRO-File-Open feature can make daily use more routine and flexible as one can use the same but modified MACRO file to make many similar calculations without bothering over all the calculation details. Moreover, by interactively running through various modules and commands, one can make maximum use of the software, and explore the possibility of dealing with very complicated systems, and in this way one can better understand how the software works.

The *Thermo-Calc Windows*, existing along with TCC started from TCCN, is fully GUI-driven. Its latest version, TCW5, is compatible with TCCS. It really simplifies thermodynamic calculations, especially for beginners and for teaching activities. A TCW user can perform calculations in a more straightforward way under a user-friendly Windows environment. However, the user may take a longer time to understand the inside picture, implying that the easily-obtained TCW calculation results may need to be appropriately understood and used for applications.

Important Note: *This edition of the Thermo-Calc manual, including not only the TCCS User's Guide and TCCS Examples Book, but also Thermo-Calc Software System and Thermo-Calc Database Guide (which are intentionally separated from the TCCS User's Guide started from this version; see Section 2.1) concerns the Classic version. While the Windows version has a separate manual, a TCW5 user is also recommended to refer to Thermo-Calc Software System, Thermo-Calc Database Guide, and some parts of this TCCS User's Guide.*

The current version TCCS (Thermo-Calc Classic, version S) is available for many types of hardware (see *Table 1-2*).

Table 1-2. Available Hardware Suitable for the Thermo-Calc Classic (version R)

Hardware		O/S	Version	Comments
PC		Windows	Vista/XP/2000/NT4	
		Linux	2.2 and above	
UNIX-Platform	SPARC	SOLARIS	2.5 and above	
	SGI	IRIX	5.3 and above	

Note: From TCCP on, several of the previously-available UNIX platforms [e.g., SPARC (SunOS), HP (HPUX), DEC AXP (DEC Unix) and IBM RS6000 (AIX)], as well as Macintosh (PowerMac and iMac), were abandoned from our support list.

Two special versions are currently available (under PC Windows Vista/XP/2000/NT4 environments), but only for the particular purposes of teaching and academic uses by university users and of testing and demonstrations by industrial users. They are:

Thermo-Calc For Academic	(TC4A)	---	Demo version of the TCC (Thermo-Calc Classic)
Thermo-Calc For University	(TC4U)	---	Demo version of the TCW (Thermo-Calc Windows)

Both TC4A and TC4U have been developed to handle thermodynamic and phase diagram calculations and assessment for systems of up to 3 elements, under PC Microsoft Windows Vista/XP/2000/NT4 environments. They are free of charge for university teaching and academic uses without external financial supports, and at no cost for industrial testing and demonstrations; and their associate licenses are only valid for a limited time.

However, it is absolutely not allowed to use them for any kind of commercial purpose. The bonded **TC4A/TC4U General Public License Agreement** governs their uses.

Installations of all software, databases and interfaces provided by Thermo-Calc Software (TCS) and its agents worldwide are handled by a very client-friendly license management system, **FLEXlm® Software Licensing Management System**. It allows user-preferred installations, updates and upgrades on individual computers and/or servers. For more information, please refer to *Section 2.2* and the related *Installation Guides*.

Table 1-3 gives a benchmark calculation of a specific system on various hardware, using the TCCS (Thermo-Calc Classic, version S). This information is useful when equipping the best-suited computer in your office.

Table 1-3. A Benchmark calculation of the Thermo-Calc Classic (version S)

Definition of the identical system in calculations	<i>Isopleth calculation of the Fe-Cr-C system using the SSOL-database</i>					
	<i>present phases:</i>					
	LIQUID FCC_A1 BCC_A2 GRAPHITE SIGMA CEMENTITE M23C6 M7C3 M3C2					
	<i>conditions:</i>					
	1 atm, 13 weight percent Cr					
	<i>mapped in:</i>					
	0 to 2 weight percent C, 0.05 weight percent steps 800 to 2000 degrees Kelvin, 30 degrees Kelvin steps					
	<i>start point at:</i>					
	1200 degrees Kelvin, 1 weight percent C					
Hardware	Model	O/S	Version	Time (s)	Compiler	
PC INTEL	PII 266MHz	NT	4.0	9	VF5.0	
PC INTEL	PII 400MHz	Linux	2.2.1	13	g77	
PC INTEL	Pentium 266MHz	NT	4.0	16	VF5.0	
SPARC	Ultra-1	SOLARIS	2.6	21	SC3.0	
PC INTEL	PPro 200MHz	Linux	2.2.1	26	g77	
SGI	Indy MIPS R4600	IRIX	5.3	29	g77	
SGI	Indy MIPS R4600	IRIX	5.3	30		
PC INTEL	Pentium 166MHz	Linux	2.0.24	43	g77	
SPARC	10	SOLARIS	2.5	70	SC3.0	

Of course, Thermo-Calc is compatible with recently released versions of these processors, and has improved calculation methods, algorithms and routines that decrease the time needed for complex systems. Thermo-Calc users are therefore recommended to refer to the latest reference with a benchmark calculation to see if there is any new number of the Thermo-Calc applicable hardware family.

1.7 Benefits of Using the Thermo-Calc Package

Thermo-Calc is one of the most powerful and flexible software/database/interface package in the field of Computational Thermodynamics and Kinetics. It has been widely used for all kinds of thermochemical calculations of complicated heterogeneous phase equilibria and multicomponent phase diagrams. Available for most platforms, the Thermo-Calc software provides you with basic thermodynamic necessities, such as equilibrium calculations, phase and property diagrams, and thermodynamic factors (driving forces) in multicomponent systems.

Thermo-Calc features a wide spectrum of models, making it possible to perform calculations on most complex problems involving thermodynamics.

Thermo-Calc consists of several basic and special modules for equilibrium calculations, phase and property diagram calculations, tabulation of thermodynamic quantities, database management, assessment of model parameters, experimental data manipulations, and post-processing of professional graphical presentations.

Thermo-Calc facilitates a comprehensive data bank of assessed thermochemical data for the phases in various systems, and uses many comprehensive databases covering a very wide range of industrial materials and applications.

Thermo-Calc enables you to efficiently establish your own reliable databases through critical assessment based on various experimental information.

Thermo-Calc utilizes a tremendously flexible user interface that is easy to use (TCC, Thermo-Calc Classic). Additionally, the complete GUI-driven (graphical user interface) version, *i.e.*, TCW (Thermo-Calc Windows), has been developed. The results are ready to import to high-quality documents.

Thermo-Calc presents the standard thermodynamic calculation engine that has the fastest and most stable mathematical and thermodynamic solutions. Any other software (user-written application program or third-party software package) that requires precisely calculated thermochemical quantities can be conveniently and efficiently plugged into the Thermo-Calc Engine (with DICTRA extensions). It currently has two powerful application programming interfaces (*i.e.*, TQ and TCAPI) and one comprehensive thermodynamic calculation toolbox in the MATLAB® software (*i.e.*, TC-MATLAB Toolbox).

It is available for most modern platforms, and allows user-preferred installations, updates and upgrades on individual computers and/or servers. Moreover, it associates with a very client-friendly license management system (*i.e.*, FLEXlm® License Manager).

The advantages of Thermo-Calc are its multiple applications. Several departments or divisions at the same company, institute or university can use the packages for different purposes. Proven application examples include industries such as communication, aerospace, transportation, and manufacturing. With the facilities provided by Thermo-Calc Software (TCS), you can optimize your materials processes to produce a higher yield, better product at a lower cost.

For a comprehensive list of the Thermo-Calc functionality and applications, please refer to the document *Thermo-Calc Software System*.

(This page is intended to be empty)

2 How to Become a Thermo-Calc Expert

TCS and its partners greatly appreciate all kinds of communication on using and improving the various software/database/interface products of the Thermo-Calc and DICTRA software family. Such communication includes success stories, publications, reported problems, opinions and improvement suggestions. All suitable information inspires new and improved features to future software/database/interface products and services.

In addition, technical supports are provided and consultancy services are available to help users become Thermo-Calc and DICTRA experts.

This part presents some hints for efficiently using Thermo-Calc, including:

- ◆ *how to easily use the Thermo-Calc manual set, including the TCCS User's Guide, TCCS Examples Book, as well as the Thermo-Calc Software System and Thermo-Calc Database Guide (which are intentionally separated from the TCCS User's Guide started from this version);*
- ◆ *how to properly install and maintain the Thermo-Calc software/database/interface package;*
- ◆ *how to efficiently use the Thermo-Calc software/database/interface package in your R &D activities;*
- ◆ *how to promptly obtain various supporting and consultancy services from TCS or its partners;*
- ◆ *how to constructively assist TCS to further improve various software/database/interface products.*

2.1 How to Use This User's Guide

This **TCCS User's Guide** is the first part of the **TCC Manual Set**, and it should be used together with the second part (the **TCCS Examples Book**). The current manual revision is primarily based on the TCCS (Thermo-Calc Classic, version S), which is available since June 2008.

IMPORTANT NOTE: The **TCC User's Guide** has become bigger and bigger in every later versions, and has thus turned to be more and more difficult to maintain and read. Moreover, some chapters are (in some senses) independent upon software-versions, and are really common for other software (such as TCW and DICTRA) and application programming interfaces (such as TQ, TCAPI and TC-MATLAB Toolbox). Therefore, started from the previous version TCCR, five individual documents have been separately prepared, extracted from the relevant chapters of the TCC User's Guide; however, for the purpose of easy reference to earlier editions, those five chapters are still kept in the structure of this edition but will only remain a very-brief introduction page.

The above-mentioned five separated documents (and their relations with relevant chapters in the original structure of the TCC User's Guide) are briefly described below:

- ☞ **TCCS User's Guide:** *(This document)*
- ☞ **Thermo-Calc Software System:** *(Extracted from Chapter 3 of the TCC User's Guide)*
It demonstrates a general overview of the entire Thermo-Calc software/database/interface package. Similar to a comprehensive reference, it is detailed in basic concepts/terms and thermodynamic framework, further development strategies, available functionality and possible applications, etc. Review of this document helps users obtain to an entire picture of the package. It is particularly useful when a user is looking for descriptions on some common terms as a specific model, database or module is interacted, and it can be used as a cross reference when the user reads other parts in the manuals of TCC, TCW and/or DICTRA software, as well as of TQ, TCAPI and TC-MATLAB Toolbox programming interfaces.
- ☞ **Thermo-Calc Database Guide:** *(Extracted from Chapters 4&6 of the TCC User's Guide)*
It gives a full description of all the available Thermo-Calc databases that can be used together with the Thermo-Calc software (TCC and TCW) and DICTRA software, as well as in various application programming interfaces. This also helps in deciding whether or not to further access to other databases is necessary for your R&D projects, as well as in planning your own database

developments. In this guide, you can also find all kinds of details regarding not only the data structures and formats of the Thermo-Calc databases but also the DICTRA database-extensions.

☞ **DATAPLOT User's Guide & Examples:** (*Extracted from Chapter 15 of the TCC User's Guide*)

It presents how the Thermo-Calc (TCC/TCW) and DICTRA software use a special data-handling language facility, namely the DATAPLOT Graphical Language (which was developed at the MSE department of KTH), for graphical presentations in their post-processors. The POST module (in TCC and DICTRA) and Diagram-Definition sub-window (in TCW) communicate with various calculation/simulation modules using this language. This document details how to arrange all kinds of graphical settings and how to handle the calculation and simulation results of various types. A highly professional graphic presentation of the TCC/TCW/DICTRA graphs can be obtained when a set of DATAPLOT expressions is enforced. Some examples of DATAPLOT files and appendices of the language are given as well.

☞ **Thermo-Calc Reference List:** (*Extracted from Chapter 16 of the TCC User's Guide*)

It documents some selective references of the Thermo-Calc software, databases, programming interfaces and applications, which you may find in your local libraries or on Internet. In addition, you are encouraged and appreciated to provide us with your own publications and manuscripts related to varied successful applications, so that we can include them in the future revisions of Thermo-Calc Reference List. If many users contribute their research papers and technical reports to Thermo-Calc Software (TCS), a *Thermo-Calc Reference Book* could also be compiled and supplied with this manual set in the near future.

This edition, *TCCS User's Guide*, is prepared in a structure of Chapter-Section-Subsection, same as for TCCR. There are 17 separate chapters in the entire document (structured in the same order as in the TCCQ/TCCP User's Guides); however, as mentioned above, four chapters have been extracted from this Guide and turned to be four separated documents whilst a single page is used to keep each of them in the structure. Among them, the first 4 chapters (two of them are detailed in two separated documents) give general introductions of the Thermo-Calc software/database/interface package and detailed descriptions on available database. The following 11 chapters (two of them are detailed in separated documents) primarily present various Thermo-Calc modules and facilities, and the last 2 chapters (again, one of them is detailed in a separated document) provide some principal references, useful appendices and helpful indices. Each specific module for model/data-handling, calculation/simulation or result presentation starts with a list of all implemented commands in the module for user's convenience.

Those who are not familiar with Thermo-Calc should begin with this chapter and the document *Thermo-Calc Software System*. Those who have already gained some experience can directly start with a specific module (i.e., Chapters 5-14 and DATAPLOT User's Guide & Examples). Reviewing Chapters 1-2 as well as the *Thermo-Calc Software System* and *Thermo-Calc Database Guide*, is also recommended to find updated information.

Chapter 1 gives a general introduction of the Thermo-Calc software/database/interface package, including its development history, general structure, and availability. It also briefly gives some examples from the package, makes suggestions to meet your requirements, and presents ideas to benefit your R&D activities.

Chapter 2 (this chapter) presents some hints on how to use the package in a general sense, how to become a Thermo-Calc expert to fully benefit the use of the package, and how to communicate with TCS and its partners.

Chapter 3 (a general overview of the entire Thermo-Calc software/database/interface package) has been extracted from this edition of the TCC User's Guide, and documented into the *Thermo-Calc Software System*.

Chapter 4 (a full description of all available Thermo-Calc databases that can be used together with the TCC/TCW/DICTRA software and various application programming interfaces) has been extracted from this edition of the TCC User's Guide, and (as a part) documented into the *Thermo-Calc Database Guide*.

Chapter 5 through *14* (among which Chapter 6 has extracted from this edition of the TCC User's Guide and moved into of the document *Thermo-Calc Database Guide*, and Chapter 15 into DATAPLOT User's Guide & Examples) extensively documents all kinds of Thermo-Calc modules and associated facilities. You can select

the appropriate chapters for your needs. These chapters are arranged in an order of utilization. Modules that are normally accessed for common calculations come first, while modules that for the understanding follow later. All available commands in each module of the Thermo-Calc Classic package are explained in detail. In most cases, why a command is needed and how it can be executed are described with help of some examples. Similarly to all modules, first layout the important and frequently used commands, followed by the remaining.

Chapter 5 describes guidelines on how to use the Database Module (TDB), for choosing/appending databases, obtaining information of selected databases, defining/rejecting/restoring systems (elements, species, phases, and phase constituents), retrieving data from databases, *etc.*

Chapter 6 (oriented to database managers or experienced users who want to create a user-specified database, dataset or data file; including some typical examples of how a Thermo-Calc database is organized, as well as some extensions for DICTRA mobility databases) has been extracted from this edition of the TCC User's Guide, and (as a part) documented into the *Thermo-Calc Database Guide*.

Chapter 7 shows, by using the Tabulation Module (TAB), how to tabulate all kinds of thermodynamic properties of various substances (stoichiometric or solution phases) or chemical reactions, and to graphically present the tabulated results.

Chapter 8 describes the most important module, the Equilibrium Calculation Module (POLY), for performing complicated thermodynamic calculations on multicomponent heterogeneous equilibria, either at a single point, with a specific special option, in one dimension (STEP) or in two dimensions (MAP). In this particular chapter, some further information on how to define various thermodynamic systems or problems in the Thermo-Calc equilibrium calculations and simulations will be given prior to going to details of all kinds of POLY commands. Specific examples show how to make various calculations with some very difficult phases or phenomena (such as ordering/disordering and aqueous solutions). This chapter also gives some useful hints on trouble-shooting, and provides answers on some frequently asked questions about the equilibrium and phase diagram calculations.

Chapter 9 presents the Post-Processor Module (POST) after the POLY Module to conveniently display successful calculations in a high professional graphical standard. This chapter illustrates how easy and efficient plotting phase diagrams and many other property diagrams from various calculations. The procedure for generating vrml (Virtual Reality Modelling Language) files and producing 3D graphs using one of the many plug-ins freely available for web browsers is also illustrated.

Chapter 10 (Some Special Modules) introduces some convenient and straightforward methods to calculate and plot diagrams. These modules simplify certain types of calculations and graphical presentations of phase diagrams and/or property diagrams. With such easy-to-use modules, the user just needs to answer a few simple questions in an automatic calculating and plotting procedure. They are: the BIN module for binary diagram calculations, the TERN module for ternary diagram calculations, the POT module for potential diagram calculations, the SCHEIL module for solidification simulations, the POURBAIX module for Pourbaix diagram and property diagram calculations, and the REACTOR module for steady-state reactor simulations. These special modules have been made available in some late versions of the Thermo-Calc package and more are going to be built up in the near future.

Chapter 11 (Gibbs Energy Module GES) provides rather comprehensive descriptions on all the already-implemented thermodynamic models. The implementations of reliable thermodynamic models for various phases under a wide physico-chemical conditions into the Thermo-Calc software/database/interface package are the essential steps to achieve a successful calculation or simulation on a complex system or problem in reality. Furthermore, this chapter shows how these models are constructed in the Thermo-Calc software core, in terms of the data structure, the thermodynamic quantity calculations, and the module's user interface. Information stored in the GES module are phase constitutions, thermodynamic properties, defined symbols, model selection, *etc.* There are extensive descriptions for all kinds of commands to interact the models and to access the data.

Chapter 12 introduces the Optimization Module (PARROT) to demonstrate its powerful assessment features, and to give some helpful hints. With rich experience of using various modules for data-handling, equilibrium and phase diagram calculations, and graphic presentations, this module conducts data assessment in order to

improve the available databases or customized datasets or data-files. The guide first shows how to prepare an experimental data file (POP), how to create a graphic experimental file (EXP), how to generate a system setup file (SETUP), and how to save and interactively access an optimization file (PARROT). Continuing, the examples demonstrate various procedures of optimizing with command, special modes, and treatments. The ED-EXP Module, as a PARROT command, is introduced and described in detail in the following chapter (*Chapter 13*). Also briefly described are the necessary connections of the module with other modules (such as GES, TAB, POLY, and POST), and its availability in the DICTRA software, the TQ and TCAPI application programming interfaces, the TCW graphic user interface, and the TC-MATLAB interface.

Chapter 13 documents the Edit-Experiment Module (ED-EXP), to show how to edit experimental data points in an appropriate way in order to make a good optimization, in terms of organizing data blocks, setting relative weights, inclusion and exclusion of data points, setting alternate conditions, and so on. This module is mainly connected with the PARROT and POLY modules, including some identical basic commands (See *Chapters 8* and *12* if necessary).

Chapter 14 (the System Utility Module SYS) troubleshoots problems to configure the O/S (operation system) dependent parameters for error message outputs and plotting environmental units. In addition, how to create, modify and maintain a MACRO file is fully explained. Some unique extra facilities, such as patch/trace/stop system debugging, HP-calculator, etc., are also illustrated.

Chapter 15 (a description of the special data-handling language facility, namely the DATAPLOT Graphical Language, for graphical presentations in the TCC/TCW/DICTRA software) has been extracted from this edition of the TCC User's Guide, and documented into the [DATAPLOT User's Guide & Examples](#).

Chapter 16 (a list of some primary references of the Thermo-Calc software, databases, programming interfaces, and applications) has been extracted from this edition of the TCC User's Guide, and documented into the [Thermo-Calc Reference List](#).

Chapter 17 presents some appendices to give some helpful hints, including the guidelines of installing the Thermo-Calc Classic package under different computer platforms, an overview of various Thermo-Calc file types and their relations, and a Thermo-Calc Classic quick reference card. Additionally, it indexes all available TCC commands with the references to their various modules and locations. Lists of figure and tables are also indexed at the end.

Finally, please note that some of the previously available but not so common or useful features or commands might have been disabled; thus, they are not longer included in this manual revision. Moreover, some models and databases, which have already been implemented but not yet extensively tested or fully released, may be described in this manual set. These extra features may not be fully available in the version released. For instance, the Flory-Huggin model for polymers is available but there is no database provided in the package. In such cases, please be patient for the subsequent patch releases of the current version (the executables will be made available at the TCS web site for downloading by those who have the valid and proper licenses for the current version), or for the pre-release of the next version (on-special-requests).

Editors of This Document:

Dr. Pingfang Shi
Thermo-Calc Software AB (TCSAB)
Stockholm Technology Park
SE-113 47 Stockholm, SWEDEN
E-Mail: pingfang@thermocalc.se

Prof. Bo Sundman
Dept. of Materials Science & Engineering
Royal Institute of Technology (KTH)
SE-100 44 Stockholm, SWEDEN
E-Mail: bosse@mse.kth.se

2.2 How to Install and Maintain the Thermo-Calc Package

2.2.1 License requirements

Previously, the installation of the Thermo-Calc and DICTRA software/database/interface packages was based upon a bilaterally-signed License Agreement, to prevent the packages from illegal copying and usage. Form June 2000 (*i.e.*, for TCCN, TCW1, DICTRA21, TQ3, TCAPI1 and TC-MATLAB Toolbox1), an appropriate **Software Licensing Management System** has been used. This also implies that the License Agreement for using the TCS-provided software, databases and programming interfaces normally do not require signature but, instead, the user must agree on general conditions set forth within the **Thermo-Calc Software END-USER LICENSE AGREEMENT (EULA)** upon the installation, and must promptly provide TCS with appropriate and sufficient information about the designated hardware and clearly-defined local settings at the designated site where the packages will be installed and used. This continues to apply for TCCS, TCW5, DICTRA25, TQ7, TCAPI5 and TC-MATLAB Toolbox5, as well as for all the commercial databases directly distributed by TCS.

Please note that the Software License Management System had been managed by FlexLM up to the TCCR/TCW4/DICTRA24 and TQ6/TCAPI4/Toolbox4 versions, but it has been switched to Sentinel RMS since TCCS/TCW5/DICTRA25 and TQ7/TCAPI5/Toolbox5 in June 2008).

The distribution normally comes first with a temporary license key file (which is usually valid for several weeks after the shipment) with the license features for all purchased software/database/interface items for the initial installations, on the condition that the End-User has provided TCS or its local agent the required **Licensed Installation Information** about designated hosting hardware and local setting at the Designated Physical Site of the End-User: *i.e.*, the so-called HOSTID (a unique single-computer identification number of 12 digits, *e.g.* “00065b307179”) for a SUNLL (Single-User Node-Locked License) installation, or a combination of Computer-Name (*e.g.* “XYZ_TC”) and HOSTID (*e.g.* “00065b307179”) of the designated License-Server, plus the IP-domain address range (*e.g.* “111.222.33.*”) of the designated Client-Computers within the local network for a SUFNL (Single-User Floating Network License) or Site (Multiple-User Network License, named as either ANSP or RNSP) installation]. Upon the prompt/complete settlement of related financial matters, TCS will shortly send to the End-User a revised license key file (on the basis of either permanent or annual usages, depending upon the types of valid license and/or valid maintenance subscription). If a user has a valid maintenance for a specific software/database/interface package, a new major-version release will be shipped together with a new permanent or annual license key file.

In most cases, the Software License Manager System is just a license key file (with the license features for all purchased software/database/interface items) that is always checked by the TCC/TCW/DICTRA software, TQ/TCAPI programming interfaces or TC-MATLAB Toolbox; this is also true when the MISCRESS™ or PrecipCalc™ software is installed/used, where either the TQ or TCAPI programming interface is an integrated part. If there is a SUFNL (Single-User Floating Network License) or Site (Multiple-User Network License) installation, the license key file must be installed on the designated License-Server. However, if an institution has several installations on different computers/license-servers with either the same or varied CPU types, each of such installations must have a unique license key file.

The users with valid software maintenance contracts must also install this license key file when they receive the new version. A license key file for the major versions of up to the TCCR/TCW4/DICTRA24 software and/or TQ6/TCAPI4/Toolbox4 programming interfaces also works properly for their relevant older versions (*e.g.*, a TCCR license feature normally works for TCCN/TCCP/TCCQ as well), but absolutely not the reverse way. However, a license feature of a particular database is dependent upon the major-version number of the database, and does not work at all if the major-version number is different (*e.g.*, the license features for the TCFE6, TCFE5, TCFE4, TCFE3, TCFE2, FEDAT and TCFE1 database-versions are completely independent). Please also note that the license key file provided for the major versions of TCCS/TCW5/DICTRA25 software and TQ7/TCAPI5/Toolbox5 programming interfaces does not work for any older versions, because of the change of Software Licensing Management System as mentioned above.

Like the previous FLEXLM license manager, the Sentinel RMS license manager can handle various kinds of hardware, *i.e.*, PC Microsoft Windows Vista/XP/2000/NT4, PC Linux, SUN Solaris, SGI, *etc.*, but a separate license is required for each CPU type.

The installation of the TCC/TCW/DICTRA software and databases, as well as of the TQ programming interfaces, are normally bonded together and distributed on the “TCS Standard Products CD”, so they can normally be installed in a single procedure, simultaneously; however, they (especially the TQ programming interface) can also be installed in separate procedures. An additionally-purchased database is usually distributed on a CD, which often bears an automatic-installation script (while sometimes arrives with a detailed installation guide for a special installation; see *Section 2.2.2.2*). The installation of the TC-API programming interface and/or TC-MATLAB Toolbox will have to be made separately, and the distribution media is usually provided separately. Please also note that a programming interface is often unnecessarily installed under the same installation tree (directory or folder) of the TCC/TCW/DICTRA software/database packages, and the user himself could properly define such interfacing pathways in his own application program. For the details of the installation and usage of programming interfaces, please also refer to their **Programming Guide and Examples**.

Please note that the copyright of all the TCS-provided manuals (User’s Guide, Examples Books, and Reference Books) and technical information sheets (Updated Software/Interface Development, Special Database Applications) are also bonded together general conditions set forth within the **Thermo-Calc Software END-USER LICENSE AGREEMENT (EULA)** which should be entirely agreed upon the installation. Therefore, any use of such copyrighted intellectual materials must not violate the conditions/terms restrictively stated in the General License Agreements and the FLEXLM Software Licensing Management System.

2.2.2 Installation procedures

Upon the receipt of the “TCS Standard Product CD” (nowadays installation tapes or floppy disks are not used anymore) for the packages consisting of the TCC/TCW/DICTRA software/databases and TQ programming interface, or alike (for TC-API programming interface, or TC-MATLAB Toolbox, or specific databases), the user can install the products with an installation script, which creates a number of directories on your hard disk, and expands the necessary files into the directories. Following the procedures described briefly below and detailed in the related *TCS Software Installation Guides*, the user can easily and appropriately complete the entire procedure of installing the TCC/TCW/DICTRA software, Thermo-Calc/DICTRA databases and programming interfaces onto the user-specified computer(s) or server(s).

For a PC Windows Vista/XP/2000/NT4 installation, the installation script will automatically start the installation, after having inserted the CD into a CD-driver (or opened the INSTALL.EXE program).

During the installation procedure, there is an enclosed individual program that you must run for a UNIX/Linux installation or is automatically switched on for a PC Windows Vista/XP/2000/NT4 installation, in order to find the Licensed Installation Information (*i.e.*, the unique identification number HOSTID) on your hardware (individual computer or server). The Licensed Installation Information for a designated computer and/or for a designated license-server (for which there must additionally include the license-server name and domain addresses range of the designated client computers that are locally connected with the designated license-server) must be sent back to us at TCS; we will then provide you (via surface mail or e-mail) with a proper license key file (being either temporary or permanent or annual).

The Licensed Installation Information for an individual computer or a server can also be easily found even before a physical installation of the packages, by using the LMHOSTID program and related supporting tools (for fixing problems during installation of packages and licenses). Such programs and tools are downloadable from our web site (http://www.thermocalc.se/DOWNLOAD_AREA/License_and_support_tools.html for an individual PC installation, and http://www.thermocalc.se/DOWNLOAD_AREA/License_server_files.html for a license-server installation).

A temporary license key file may be obtained from TCS along with the installation or afterwards, which allows the user to use the TCS-provided software/database/interface packages only for a certain and limited time period (normally a few weeks). After obtaining the permanent or annual license key file from TCS, you can run the packages on your specified computer(s) or license-server(s) on a permanent or annual basis.

2.2.2.1 *Installing the entire TCC software/database package*

The **TCS Software Installation Guides** give all the guidelines for installing the TCC/TCW/DICTRA software/database/interface packages (from a “TCS Standard Product CD” or alike) under PC Windows Vista/XP/2000/NT4 and UNIX platforms (Solaris, SGI, PC Linux, *etc.*). Most of the files on the distribution

CD-ROM (or tape or diskettes) are compressed or zipped, and should not be copied directly. The shipment normally comes with an installation script, which automatically creates a number of directories on your hard disk, and expands the necessary files into the directories.

Under PC Windows Vista/XP/2000/NT4 environments, a TCW installation shares most of the TCC directory-file structures (especially for the database subdirectories and files) or has its own directory-file structures, depending upon the user's choose in the installation procedure.

Similarly, a DICTRA installation also shares some of the TCC directory-file structure of the parallel TCC version (*e.g.*, DICTRA21 with TCCN, DICTRA22 with TCCP, DICTRA23 with TCCQ, DICTRA24 with TCCR, and DICTRA25 with TCCS) under PC Windows Vista/XP/2000/NT4 environments. Therefore, the installation procedure automatically allocates the DICTRA executable program, *readme*/license files and icon file at the same directory level as for the TCC installation, but adds the DICTRA database subdirectories and database files to the \DATA\ area, inserts the DICTRA help files to the \HELP\ area, and generates the \DIC_EXA\ directory and subdirectories for all DICTRA standard examples.

After a successful installation, a user may modify some of the textual files for setting his/her own preference and testing the enclosed standard examples. Occasionally, a user (as the administrator or locally-authorized user) can further modify the environmental parameter(s) of the installation: TCPATH on Windows Vista/XP/2000/NT4 (in the “**Registry Editor**” or “**System Properties**” window); TC_DATA, TC_TDBHLP, TC_GESHLP, TC_POLYHLP, *etc.* under Linux and UNIX platforms.

For all kinds of detail on installation and modifications, please refer to the related *TCS Software Installation Guides*.

2.2.2.2 Adding database(s) to an already installed TCC package

After the successful installation of the TCC/TCW/DICTRA software/database/interface packages on your computer or server (of certain CPU type), you may need to add another database, whether created by yourself or purchased from TCS or its agents, to the already installed TCC package. An additional database purchased from TCS or its agents is normally (but not always) distributed on a CD with an installation script, which will automatically and appropriately install the added database into the existing installation.

In case that a user-specified database is to be added manually, three things need to be performed:

- 1) Check that the additional database (in proper plain textual form) must have the same data format as the already-installed databases (especially the public databases, such as PBIN, PTERN, PFIRB, *etc.*).
- 2) Create another subdirectory for the added database, such as TCADD or MYOWN under the DATA directory level for PC Windows Vista/XP/2000/NT4 environments, or tcadd or myown under the data directory level for all kinds of UNIX platforms (Solaris, SGI, PC Linux, *etc.*). Then simply copy all the database file(s) into this subdirectory from your own database area (or from the distribution media). An added database may contain one single file (*i.e.*, the setup file), or several separate files but the setup file must contain the type definitions and correct paths for the internally connected files (*e.g.*, the parameter file, function file, FTP function random file, reference file, *etc.*). For details, please refer to the *Database Manager's Guide* part in the *Thermo-Calc Database Guide* document.
- 3) Modify the *database initiation file*, *i.e.*, TC_INITD under the DATA directory level for PC Windows Vista/XP/2000/NT4 environments, or initd.tdb under the data directory level for all kinds of UNIX platforms (Solaris, SGI, PC Linux, *etc.*), regarding all such added databases. For details on how a database initiation file is constructed, please refer to *Section 3.2 in the Database Manager's Guide* part in the *Thermo-Calc Database Guide* document. For each added database, it is usually recommended to define its short name, give its path to the setup file, and write a brief description, in one single line for Windows Vista/XP/2000/NT4 environments, *e.g.*,

```
TCADD  TCPATH\DATA\TCADD\ADDSETUP.TDB  TCS Additional Database !
MYOWN  TCPATH\DATA\MYOWN\OWNSETUP.TDB  My Own Additional Database !
```

In case, a TCADD database is provided/distributed as in the encrypted data form, the corresponding line should be:

```
TCADD  TCPATH\DATA\TCADD\ADDSETUP.TDC  TCS Additional Database !
```

For various UNIX/Linux platforms, such additional lines should be as *e.g.*,

```
tcadd  TC_DATA/tcadd/addsetup.tdb
      TCS Additional Database !
myown  TC_DATA/myown/ownsetup.tdb
      My Own Additional Database !
```

In case, a TCADD database is provided/distributed as in the encrypted data form, the corresponding line should be:

```
tcadd  TC_DATA/tcadd/addsetup.tdc
      TCS Additional Database !
```

2.2.3 Maintaining current and previous versions

The installation and performance of the Thermo-Calc (TCC/TCW) and DICTRA software/database packages are normally version sensitive; the DICTRA software/database package shares the same version structure as of the TCC/TCW software/database package. Different versions of the packages may have different directory-file structures and file contents. However, the automatic installation procedure of a later version will use the same upper-level directory structure as its previous version(s), but have a different version level (such as `VERS` or `vers` for both TCCS/TCW5 and DICTRA25) and all sub-levels below the version level.

Under PC Windows Vista/XP/2000/NT4 environments, the TCC installation procedure may also automatically creates a `\WORK\` directory parallel to the version level, and normally generates a TCC shortcut (pointing to its working area) with a shortcut icon shown on the screen. Then by clicking such an icon, the TCC (or TCW or DICTRA) is initialized, and all kinds of input files (except for the database files) are read from and output files are saved into this working area. The user may rename this working directory as another name, for instance, `\WORKS\` in order to identify the input and output files used together with the specific version `VERS`. The user may also copy such a shortcut icon to another one (or create a new shortcut of the TCC/TCW/DICTRA executable file) with any different ***Start-in*** path in the icon's property window, where the user can generate or save various input and output files for a specific purpose or research project. Of course, the easiest manner is to simply modify the ***Start-in*** path on an already created shortcut icon. In this way, the user can allocate various binary and textual files in different working directories (please refer to [Section 4.4](#) in the document *Thermo-Calc Software System*), on different drivers, or on a connected server.

Under various UNIX platforms (Solaris, SGI, PC Linux, *etc.*), an appropriate TCC/DICTRA installation allows all the assigned/authorized users at the designated site (group) to start the TCC/DICTRA software at certain directories on the designated computer/server. If there is any problem in doing so, please consult your local computer system manager.

Furthermore, different versions of the TCC/TCW/DICTRA software/database packages may have slightly different data structures (stored in dynamically-managed workspaces) for some binary data types (*e.g.*, `*.GES5`, `*.POLY3`, `*.PAR`, `*.DIC` or `*.TDC` files); however, the textual data types (*e.g.*, `*.TCM`, `*.LOG`, `*.TDB`, `*.DAT`, `*.TXT`, `*.SETUP`, `*.POP` or `*.EXP` files) normally remain the same or similar (for meanings of these file types, please refer to the relevant chapters and *Appendix B*). This implies the followings:

- (1) A binary data file saved by a later version may not be accepted by an earlier version; however, a newer version is sometimes able to read a binary file generated by an older version.
- (2) A textual data file generated for/by an earlier version is usually acceptable in a later version, and normally it is also true for the reversed way (except for those lately added or largely-revised legal GES/POLY/POST/PARROT/ED_EXP commands saved in such textual files). This is why it is highly recommended that the users shall necessarily save the important calculations as textual files (`*.TCM`, `*.TDB`, `*.TXT`, `*.SETUP`, `*.POP` and `*.EXP`) if possible, rather than the binary files, for the purposes of repeating the previous calculations in a later version and sometimes making comparisons among various versions.
- (3) The default output textual files (*e.g.*, `*.DAT`, `*.TXT` or `*.TAB`) and graphical files (with extensions of *e.g.* EMF, PNG, BMP, PDF, JPEG, TIFF, PostScript-PS5/6, HPGL-P7/8, *etc.*) are version independent, and can only be opened, edited or imported/exported outside the TCC/TCW/DICTRA packages anytime by using an appropriate textual, word-processing or graphical editing tool.

2.2.4 Making the TCC performance convenient for you

Under PC Windows Vista/XP/2000/NT4 environments, you can create several TCC (and TCW and DICTRA) shortcut icons (for various software versions, as well as for different purposes identifying certain systems or projects). The properties of such icons can be easily modified, by simply right-button clicking on the icon and choosing the “**Properties**” option.

For various versions of the TCC/TCW/DICTRA software, you can point to the executable program in the **Target** field on the “**Short-Cut**” bar of an icon’s property window; and as mentioned above, for different purposes to identify certain systems/projects, you can specify the working area in the **Start-in** field.

For the sake of making the TCC/DICTRA performance convenient for yourself, you can further modify the “**Options**” (for cursor size, display window size, buffer size and number of buffers of command history, and QuickEdit and Insert modes), “**Fonts**” (font type and font size), “**Layout**” (screen buffer size, window size, and window position) and “**Colors**” (color type and values for texts and background of screen and of popup windows), on the icon’s property under Windows Vista/XP/2000/NT4 environments, as illustrated in *Figure 2.1*. The “*QuickEdit Mode*” can make it easier to mark/copy/paste desired texts. Since TCCS, the “*height*” property of “screen buffer size”, which has been set as a large number (*e.g.* 8000) can always work appropriately, so that it is very convenient to scroll the consol window.

Under various UNIX platforms (Solaris, SGI, PC Linux, *etc.*), *your local system manager would always be happy to tell you all the tricks on how to set up your preferable working environment and to make TCC and DICTRA convenient for you to work with.*

However, you can also access a Linux/UNIX version of TCC or DICTRA from a Windows-based PC with an appropriate Tektronix-window program installed.

For instance, if you login to a Linux/UNIX server from your PC Windows Vista/XP/2000/NT4 using the TeraTerm program, and run the TCC or DICTRA software/database package, as illustrated in *Figure 2.2*. It is recommended to first modify some appearance setting of the TeraTerm window, by changing the some “**Setup**” options for *e.g.* **Terminal** (Terminal Size, Auto VT<->TEK switch), **Window** (Scroll buffer lines, and Text/Background Color), **Font**, *etc.*, and enabling Save setup for future uses.

For all tasks except for plotting diagrams on screen, it works perfectly. For generating graphical files (such as in PostScript formats) of calculated diagrams, it also works satisfactorily. If you want to plot diagrams on screen, you can either use the SYS command **SET_PLOT_ENVIRONMENT** with the device number of 1 (for Tektronix 4010) before conducting any TCC/DICTRA command, or use the POST command **SET_PLOT_FORMAT** with the device number of 1 (for Tektronix 4010) before typing the **PLOT_DIAGRAM** command (and sequentially accepting SCREEN as **PLOTFILE**). It is also recommend using the **Open TEK** option of the “**Control**” action prior to the **PLOT** command. Please note that when a diagram is plotted on the TEK sub-window, you can **Copy/Paste** and **Print** the contents on the sub-window, and also that afterwards you shall remember to **Exit** the TEK sub-window before executing any further TCC action on the main TeraTerm window.

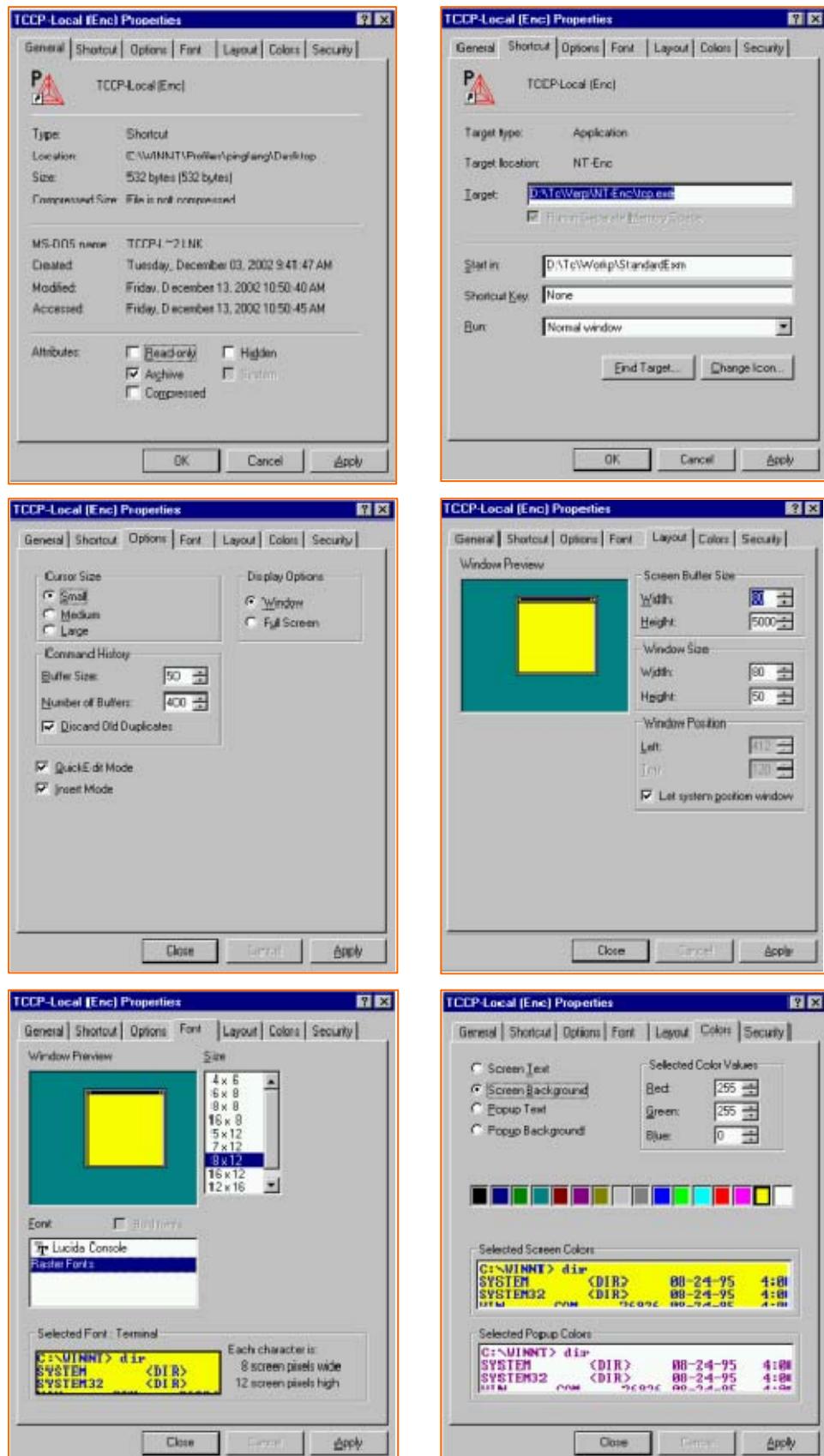


Figure 2-1. Modifying Your TCC Shortcuts (with recommended values)

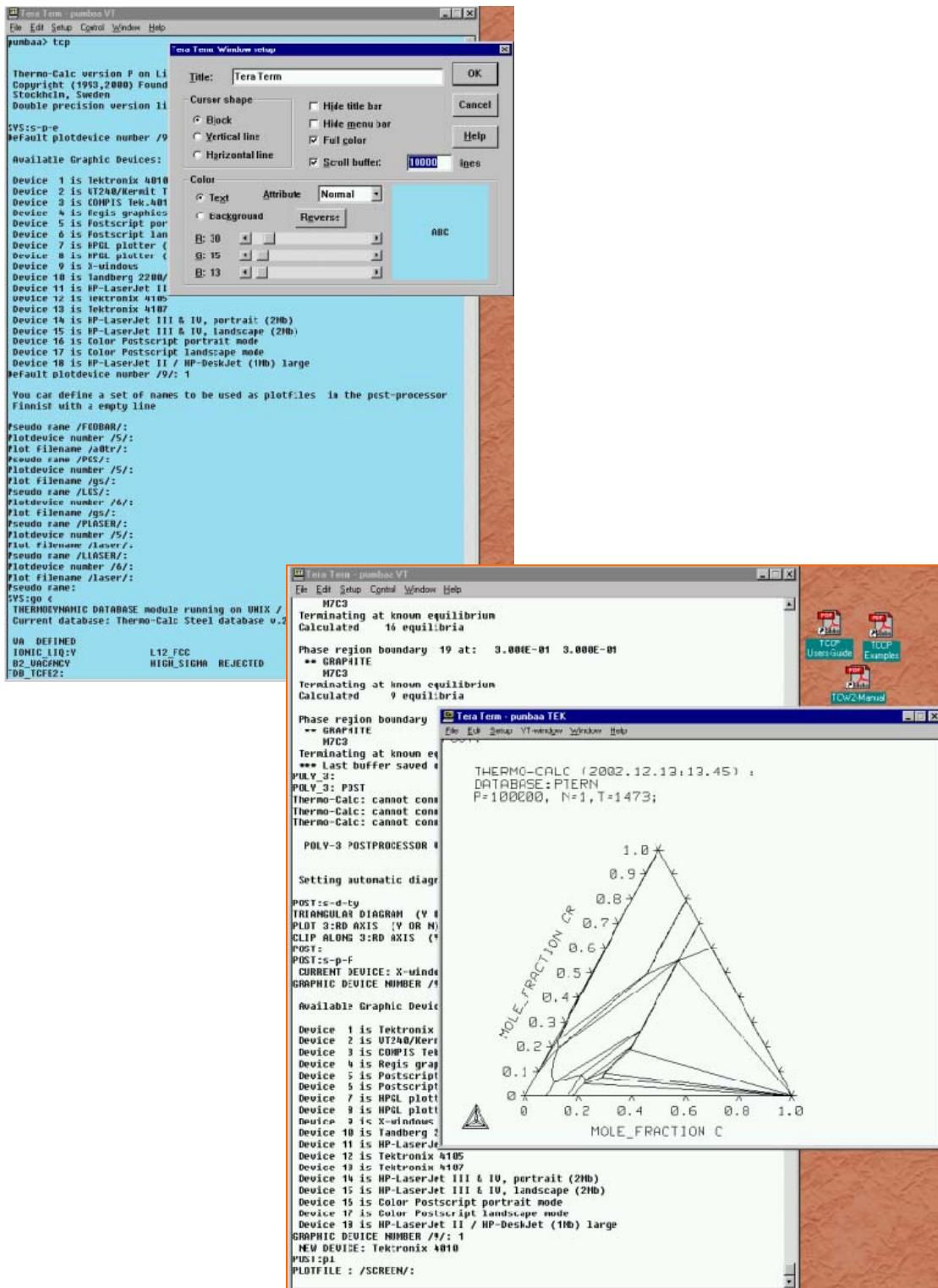


Figure 2-2. Working with a Linux/UNIX Version of TCC in TeraTerm Windows (Accessed from Windows Vista/XP/2000/NT4 computers)

2.3 How to Become a Thermo-Calc Expert

All users of the Thermo-Calc and DICTRA software/database/interface packages are, and always will be, the strength of TCS (Thermo-Calc Software).

- ☞ They utilize our professional products as efficient and reliable engineering tools in their R&D activities, and as advanced and state-of-art educational necessities in their teaching exercises.
- ☞ They become important contributors to our product development and service performance, by providing valued suggestions, reporting bugs and problems, participating in our development projects, conducting consultancy for TCS itself and our customers, arranging local training programs to our new and prospective clients, offering their journal publications, project reports and other types of references, joining our customer/users-group meetings, and so on.
- ☞ They act as valuable promoters of our professional products in universities, governmental institutions, private R&D and consultant companies, and industrial companies, and help us build our personal and professional networks of peers, problem solvers, advisors, and customers.
- ☞ They often take leadership roles in various scientific societies and industrial communities, and serve as champions and experts in computational thermodynamics and kinetics (applied in materials design and engineering, materials manufacturing and applications, automotive and aerospace industries, telecommunication, energy production and utility, heavy/precision instrument and lighting equipment, chemical engineering, mining industries, environmental protections, nuclear fuel and waste management, food production, and so forth).

We highly encourage, and will do our very best to assist, all our clients to become experts of applying these packages in various R&D and teaching activities. We would like to make the following recommendations:

2.3.1 Get prompt technical supports from TCS and its agents worldwide

TCS has, at present moment, around ten highly qualified agents all over the world. For the detailed addresses and contact persons of these agents, please visit our web site www.thermocalc.com.

TCS and its agents organize regular Thermo-Calc (TCC/TCW) and DICTRA training courses worldwide. For course descriptions and schedules, please also visit web sites of TCS and of your local agents, or send us a filled ***Thermo-Calc Course Application Form*** or ***DICTRA Course Application Form*** (which are promptly made available at our web page http://www.thermocalc.com/Service/Training_courses.html). Sometimes, a special training course can be arranged at a specific location upon the clients' particular requests; please contact us if there is a necessity and possibility.

All of our users are highly recommended and encouraged to regularly check our specific web page, http://www.thermocalc.com/DOWNLOAD_AREA/Download.html, where you can easily and freely download some possibly recently-updated publications of the following items (among others):

- 1) **Patches** (latest updates/subversions) of the most-current major versions of various software (TCCS, TCW5 and DICTRA25) and programming interfaces (TQ7, TCAP15 and TC-MATLAB Toolbox5); please follow the given instructions for downloading and for replacement installations (*requirements*: you must have a valid license for the corresponding major versions of the software and programming interfaces that had already been installed on the same Designated Computers and/or License-Server).
- 2) **Manuals** (User's Guides and Examples Books) of various software and programming interfaces.
- 3) **License and Support Tools** for node-locked single-computer installations.
- 4) **License Server Files** for networked license-server installations.
- 5) **Reference Lists and Representative Publications**.

For basic technical problems related to the installations and performances of TCS-provided software, databases and interfaces, prompt supports can be received from TCS (post: Thermo-Calc Software AB, Stockholm Technology Park, SE-113 47 Stockholm, Sweden; e-mail: support@thermocalc.se; phone: +46-8-5459 5930; fax: +46-8-673 3718; http: www.thermocalc.com) or your local agents. Our web site routinely publishes various technical updates and references, such as newsletters, technical information, FAQ list, software new features and improvements, special database applications, selected application references, software demo

download, updated software download (for valid license holders), updated manual download, forth-coming course information and registrations, and so on. Based on the client's software maintenance option, some in-deep technical assistance and problem-shooting supports can also be obtained from TCS or your local agents.

For inquiring more information about TCS software/database/interface products and support/consultancy services, please send e-mails to us (info@thermocalc.se).

TCS and its subsidiary/agents, as well as some TCS' special partners (experts in specific application fields), can conduct some consultancy (perform calculations, design databases or even make special software) on behalf of our clients. You may simply send us your requirements and specifications (consult@thermocalc.se). We will then immediately contact you for arranging the consultancy agreements and conducting the consultancy work.

If you have encountered any bug or problem in running any TCS-provided software, database or programming interface, we would very much appreciate that you would promptly report to us (support@thermocalc.se). We will do our very best to solve the problems in the corrected or updated versions, and provide you with the appropriate solutions on time.

2.3.2 Make daily use of various Thermo-Calc functionality

A new Thermo-Calc user may start with the special modules (BIN, TERN, POT, POURBAIX and SCHEIL) to make some calculations of phase diagrams and property diagrams. The standard examples, stored as MACRO files (*.TCM) in the installation area \TCEX\ and documented in the [TCCS Examples Book](#), give all kinds of instructions on various basic and special modules (SYS, TDB, GES, TAB, POLY, POST, PARROT, ED-EXP; BIN, TERN, POT, SCHEIL, POURBAIX, REACTOR). The user can copy such exemplary MACRO files to his/her own desired directories in order to conduct other desired calculations/simulations after having properly made the necessary modifications (on temperature, pressure, composition and other conditions) for the user-specified applications.

Along with your daily R&D and teaching activities, you can routinely use the Thermo-Calc and DICTRA software/database packages as reliable and efficient engineering tools, for various kinds of thermodynamic calculations and kinetic simulations. The powerful functionality of the packages will provide you with the most quick and precise descriptions of thermodynamic equilibria and kinetic paths, and help you understand complex problems and phenomena in varied materials systems and materials processes.

After gaining enough experience with using the calculation and simulation modules, you can step into performance of critical assessments on your interested systems, utilizing the PARROT module that is available within the TCC and DICTRA software. With your own experimental results and accessed literature information on particular materials systems and materials processes, you can optimally modify some existing data, and easily establish your own databases/datasets/data-files.

If you are interested in plugging the Thermo-Calc Engine (through the TQ/TCAPI application programming interfaces) with your R&D facilities, you may write your own programs for materials property calculations and materials process simulations, following the simple examples in the [TQ/TCAPI Programming Guide and Examples](#). You can also use the [TC-MATLAB Toolbox in the MATLAB® software package](#) for materials property calculations, materials process simulations, and materials engineering controls.

2.3.3 Present your results in professional and high quality standards

With some experience using the Thermo-Calc and DICTRA packages, you may soon establish your own favored routines and manners for the inclusions of your calculation and simulation results in various scientific publications, technical reports and conference presentations.

Upon special requests, TCS may provide you with some recommendations on how to present your calculation and simulation results (from using the Thermo-Calc and DICTRA packages) as various tabular and graphical forms in professional and high quality standards. With help of modern advanced software for word documenting, data handling and graphical processing, you can easily and efficiently edit/export the obtained

calculation/simulation tables (from the TAB and POST modules) and graphs (from the POST module) at your preference.

2.3.4 Exchange your experience with others through varied channels

TCS highly values your knowledge, expertise and suggestions on applying the Thermo-Calc and DICTRA software/database/interface packages in various R&D and teaching activities, and would greatly appreciate learning of your successful stories and unique examples. There are many good channels for significantly exchanging your experience with TCS, our agents and development/consultancy partners, as well as with other users all over the world.

Suggestions on improving and expanding the software functionality, user interfaces, programming interfaces, database designs and applications, as well as our technical supporting and consultancy services, are particularly welcomed.

With your research results promptly presented at various conferences of different societies, and frequently published on scientific journals and technical report series, many other existing users and prospects will be able to share your experience and to learn your stories. TCS and its agents often participant in various conferences (where we give presentations and demonstrations, and hold seminars/workshops and user-group meetings) and exhibitions. Detailed/updated information about such activities (organized/co-organized by TCS) can be found at our specific page http://www.thermocalc.se/News/News_Events.html.

Most importantly, for maintaining an updated and complete reference, TCS always expects to receive from you a reprint of your papers and reports, and a copy of your own programs and databases developed on the basis of the Thermo-Calc and DICTRA software/database/interface packages.

We highly encourage and very much appreciate you to send us (at info@thermocalc.se) your successful stories/examples, reprints of published papers, copies of research reports, and suggestions/comments.

3 Thermo-Calc Software System

Started from the TCCR version, this chapter had been extracted and separately prepared as an individual document “*Thermo-Calc Software System*”.

>>> Open the document [Thermo-Calc Software System](#)

(This page is intended to be empty)

4 Thermo-Calc Database Description

The Thermo-Calc software/database system covers more stoichiometric and non-ideal solution models and databases than any other available thermochemical software package. These models and databases can be used to describe steels, alloys, slags, salts, ceramics, solders, polymers, subcritical aqueous solutions, supercritical electrolyte solutions, non-ideal gases and hydrothermal fluids, organic substances, *etc.* over very wide ranges of temperature (up to 6000 K), pressure (up to 1 Mbar) and composition.

Started from the TCCR version, this chapter had been extracted and separately prepared as the Chapter 2 in the individual document “*Thermo-Calc Database Guide*”.

>>> Open the document [Thermo-Calc Database Guide](#)

(This page is intended to be empty)

5 Database Module (TDB)

5.1 Introduction

One of the problems with phase diagram calculations in multicomponent systems is finding valid parameters that describe the various equilibria. A related problem is obtaining parameters in such a way that the elements are referred to in the same state in all phases.

Transforming the available parameters to the same standard state is tedious work and errors are very common. It would be a great advantage to have this work completed to use the same transformed parameters in all future applications. This kind of work is performed by a few different groups around the world. One of the strongest groups in this field is the Scientific Groupe Thermodata Europe (SGTE) that has collected and assessed a huge amount of data which are available through two different database files.

For convenience, the **Thermodynamic DataBase module**, TDB \Rightarrow , to easily retrieve thermodynamic data has been developed.

The TDB module is included in the Thermo-Calc and DICTRA software packages to perform administration, selection, and retrieval of various types of thermodynamic and kinetic data.

The following commands are available in the TDB module:

TDB_SSOL: ?		
AMEND_SELECTION	EXIT	NEW_DIRECTORY_FILE
APPEND_DATABASE	GET_DATA	REJECT
BACK	GOTO_MODULE	RESTORE
DATABASE_INFORMATION	HELP	SET_AUTO_APPEND_DATABASE
DEFINE_ELEMENTS	INFORMATION	SWITCH_DATABASE
DEFINE_SPECIES	LIST_DATABASE	
DEFINE_SYSTEM	LIST_SYSTEM	
TDB_SSOL:		

Note that the command SET_AUTO_APPEND_DATABASE command is added in TCCP, whilst two previously available commands (EXCLUDE_UNUSED_SPECIES and MERGE_WITH_DATABASE) have been deleted in TCCN and TCCP, respectively.

\Rightarrow **Revision History of the TDB-Module User's Guide:**

Nov 1986	First release (Edited by Jan-Olof Andersson and Björn Jönsson)
Oct 1993	Second revised and extended release (Edited by Bo Sundman)
Jun 2000	Third revised and extended release (Edited by Pingfang Shi)
Nov 2002	Fourth revised release (Edited by Pingfang Shi); with minor changes in 2004/2006/2008

5.2 User Interface of the TDB Module

The user interface in the TDB module is constructed for an interactive computer session. The main feature of the interface is a command monitor to control the program action by typing an appropriate command. There is also an alternative interface that an application program can send the commands to the monitor.

The commands of TDB often consist of several words separated by a hyphen or an underscore character. The hyphen and underscore characters are considered identical. Some of the main keywords have a second keyword. This second keyword can either be joined to the main keyword with a hyphen or an underscore or separated with a space or a comma from the command keyword in the same way as for an ordinary parameter. The command may be abbreviated by an unambiguous group of letters to shorten each part between hyphens.

Remaining parts of a command may be omitted if they are not necessary to distinguish the commands. Many commands require that some parameter's values be specified. By pressing the <RETURN> key, the program will prompt for these values.

Usually, default command values are suggested between slashes and are chosen by pressing the <RETURN> key. A command (with its second keyword, if applicable) and all parameters may be entered on the same line, if a user knows such parameters required by the command; otherwise, they can be separately entered subsequently in ensuing prompt(s) after the command.

5.3 Getting Started

The TDB module used by the Thermo-Calc and DICTRA systems allows the user to define a system and retrieve thermodynamic and/or kinetic data for that system from a chosen database. A straightforward way to do this is to use the commands described below in order. Some commands are necessary, while some are optional but still useful and informative.

5.3.1 SWITCH_DATABASE

SWITCH_DATABASE is used to change the default database. The second part of the prompt TDB_XYZ indicates the present database, for instance, TDB_SSOL for the SSOL database.

5.3.2 LIST_DATABASE ELEMENT

LIST_DATABASE ELEMENT gives a list of elements available in the present database. The keyword ELEMENT can be substituted with SPECIES, PHASE or CONSTITUENT.

5.3.3 DEFINE_ELEMENTS

DEFINE_ELEMENTS <list of elements> is a command that prompts the user for input of all the elements that should be defined in the system.

5.3.4 LIST_SYSTEM CONSTITUENT

LIST_SYSTEM CONSTITUENT gives a list of the phases that can form from the defined system. The elements listed after the phase name are the constituents of that phase.

5.3.5 REJECT PHASE

REJECT PHASE <list of phases> tells the database not to retrieve any data for whichever phase is specified. The keyword SYSTEM can be used to reinitiate the database.

5.3.6 RESTORE PHASE

RESTORE PHASE <list of phases> acts as the reverse of the REJECT command, except that the keyword SYSTEM cannot be used for this command.

5.3.7 GET_DATA

GET_DATA searches the database and enters the system defined into the GES5 and/or DICTRA workspace. 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.

5.4 Description of All Available TDB Monitor Commands

All implemented commands in the TDB monitor are presented in alphabetical order (as shown in the *Introduction* in *Section 5.1*. Most commands will issue a sub-prompt asking for parameter values. Not all sub-prompts are explicitly explained in this text, only those for which it was thought necessary.

5.4.1 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, but not the constituents or the entire system, can be accepted or rejected. Ensuing prompt(s) for making other modifications can also be quitted by answering Q (Quit).

Synopsis: AMEND_SELECTION [keyword]

Ensuing 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

5.4.2 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 TDB module and GES5 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 file in the /DATA/ area for PC Windows Vista/XP/2000/NT4 environments, or the initd.tdb file in the \data\ area for all kinds of UNIX/Linux platforms), 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.

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.

Synopsis 1: APPEND_DATABASE <additional database name>

Synopsis 2: APPEND_DATABASE

Ensuing 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: Under the Windows Vista/XP/2000/NT4 environments, an **Open file** window will be popped up 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, so that the path (in the **Look in** box) and database name (in the **File name** box) can be appropriately specified. The user can then open the chosen USER database, or may cancel such a Window session; in the latter case, the program will list all predefined databases and the user can then specify one of them to be switched on or proceed the USER option again.

Under Windows Vista/XP/2000/NT4 environments, this command can be called more than once if appending two or more databases to the primarily switched database. Under various UNIX/Linux platforms, the second and further appending databases can only be connected using the SWITCH_DATABASE USER command sequence.

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.

5.4.3 BACK

Description: The control goes back to the most recent module. See also GOTO_MODULE.

Synopsis: BACK

5.4.4 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.

Synopsis: DATABASE_INFORMATION

5.4.5 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).

Synopsis: `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`.

5.4.6 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. Note that since the TCC version N, up to 1000 species can be defined into a single system (previously 400 species).

Synopsis: `DEFINE_SPECIE <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`.

5.4.7 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.

Synopsis 1: `DEFINE_SYSTEM <element1, element2, ...>`

Synopsis 2: `DEFINE_SYSTEM`

Ensuing Prompt: `ELEMENTS: <element1, element2, ...>`

Synopsis 3: `DEFINE_SYSTEM`

Ensuing Prompt: `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).

Note that since the TCC version N, up to 40 elements and 1000 species can be defined into a single system (previously 20 elements and 400 species).

5.4.8 EXCLUDE_UNUSED_SPECIES

Description: This command excludes species that do not enter any of your current defined phases from the species list (shown by the `LIST_SYSTEM SPECIES` command sequence), in order to save program workspace. *However, it has been disabled since the TCC version N, since computer memory is not a big issue nowadays.*

5.4.9 EXIT

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

Synopsis: `EXIT`

5.4.10 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 GES5 and/or DICTRA workspace. This command is necessary for retrieval of all information concerning a defined system from the databank.

Synopsis: `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.

5.4.11 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.

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing 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 unique module name will list all the available TCC modules, as below:

```
NO SUCH MODULE, USE ANY OF THESE:
SYSTEM UTILITIES
GIBBS ENERGY SYSTEM
TABULATION REACTION
POLY_3
BINARY DIAGRAM EASY
DATABASE RETRIEVAL
FUNC OPT PLOT
REACTOR SIMULATOR 3
PARROT
POTENTIAL DIAGRAM
SCHEIL SIMULATION
POURBAIX DIAGRAM
TERNARY DIAGRAM
MODULE NAME: <module name>
```

5.4.12 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

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

Specifying a unique TDB 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.

5.4.13 INFORMATION

Description: This command gives information on a few subjects concerning the Database Module and Database Management.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: ?

Specify a subject (or its abbreviation as long as it is unique, e.g., EA, EXT, EXA, etc.) on which information should be given, from the following subjects that are important to the use of the TDB Module:

PURPOSE

```
      ==> on what the INFORMATION command does for
HELP
      ==> on how to get help for various TDB commands
USER INTERFACE
      ==> on how the user interface of TDB module works
EASY TO USE GUIDE
      ==> on how to get started with the TDB module
EXTENDED COMMANDS
      ==> on how to use extended commands in the TDB module
DATABASES
      ==> on how the TC databases are developed and provided
MANAGEING DATABASES
      ==> on to how to maintain TC databases
INITIALIZATION OF TDB MODULE
      ==> on how to initialize the TDB module
SHORT USER-DEFINED DATABASE GUIDE
      ==> a short guide on user-defined database
LONG USER-DEFINED DATABASE GUIDE
      ==> a long guide on user-defined database
DICTRA EXTENSIONS
      ==> on DICTRA extensions in the TDB module
EXAMPLES OF DATABASE DEFINITION FILES
      ==> some examples of database definition files
WHICH SUBJECT /PURPOSE/: <a specified and unique subject>
```

Options:

One of the following subjects should be specified and unique:

```
PURPOSE -- gives an introduction on the TDB module, as shown in Section 5.1.
HELP -- instructs on how to get help for various TDB commands, as shown in Section 5.4.12.
USER INTERFACE -- describes the user interface of the TDB module, as shown in Section 5.2 (User interface).
EASY TO USE GUIDE -- lists all basic commands shown in Section 5.3 (Getting started).
EXTENDED COMMANDS -- lists all extended commands described in Sections 5.4 and 5.5.
DATABASES -- describes the database spectrum in Thermo-Calc, including those that are freely distributed, and others that are commercially available.
MANAGING DATABASES -- describes a Thermo-Calc database is generated and managed, as shown in Section 3.1 of the Thermo-Calc Database Guide.
INITIALIZATION OF TDB MODULE -- describes how to initialized the TDB module, as shown in Section 3.2 of the Thermo-Calc Database Guide.
SHORT USERDEFINED DATABASE -- gives a short description on how to construct a user database.
LONG USERDEFINED DATABASE -- gives a detailed description on how to construct a user database, as shown in Section 3.3 of the Thermo-Calc Database Guide.
DICTRA EXTENSIONS -- gives a detailed description on how to use DICTRA extensions in a user database, as shown in Section 3.4 of the Thermo-Calc Database Guide.
EXAMPLES OF DATABASE DEFINITION FILES -- demonstrates two examples of user-defined databases, as shown in Section 3.5 of the Thermo-Calc Database Guide.
```

5.4.14 LIST_DATABASE

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

Synopsis: 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.

5.4.15 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.

Synopsis: 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.

5.4.16 MERGE_WITH_DATABASE

Description: This command gets data from a file or additional database and merge them with 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. It replaces the old APPEND_DATABASE command, and is almost identical to the updated APPEND_DATABASE command except for that this command may enter parameters to already existing phases. **However, this command has been disabled since TCCP. It is recommended to use the APPEND_DATABASE command instead.** See Section 5.4.2 (APPEND_DATABASE command) for more details.

5.4.17 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 TCC version M.

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 PC Windows Vista/XP/2000/NT4 environments, or the initd.tdb file in /data/ area

that is under the directory defined by the TC_DATA parameter on PC Linux and various UNIX platforms (SUN Solaris, SGI, etc.). 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 TDB module among the original and additional database initiation files. TCC 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.

Synopsis 1: NEW_DICTORY_FILE <another database-initiation-file name>

Synopsis 2: NEW_DICTORY_FILE

Ensuing 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 TCC is used under a Windows Vista/XP/2000/NT4 environment, an *Open file* window is popped up on screen for specifying the new database initiation file if the file name or its path is not given on the same line of the NEW_DICTORY_FILE command, or if it is incomplete or incorrect. Then, the path (in the **Look in** box) and database initiation file name (in the **File name** box) can be appropriately selected. Canceling this widow may quit the program.

However, if TCC is run on an UNIX/Linux platform, this new database initiation file must be located in the current working directory (where TCC is started).

For details on how to construct an additional database initiation file, please refer to *Section 6.2 (Initialization of the TDB Module)*.

5.4.18 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 TDB module memory and GES5 workspace.

Synopsis 1: REJECT [keyword] (if keyword = ELEMENTS or SPECIES or PHASES)

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

Synopsis 2: REJECT [keyword] (if keyword = CONSTITUENT)

Ensuing Prompt: PHASE: <phase name>
SUBLATTICE NUMBER: <sublattice number in the phase>
CONSTITUENT: <constituent(s) in the sublattice of the phase>

Synopsis 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 TDB module is reset to its initial state; GES5 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.

5.4.19 RESTORE

Description: This command does the opposite of the command REJECT, *i.e.*, restoring already explicitly rejected elements, species, phases or constituents; but it can not restore a completely rejected system. Phases/species/constituents that are possible to form from the defined elements or species are entered to the list of system phases/species/constituents (shown by the command LIST_SYSTEM). 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.

Synopsis 1: RESTORE [keyword] (if keyword = ELEMENTS or SPECIES or PHASES)

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

Synopsis 2: RESTORE [keyword] (if keyword = CONSTITUENT)

Ensuing 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.

5.4.20 SET_AUTO_APPEND_DATABASE

Description: This command enforces an automatic action that will append thermodynamic data from an appending database (that is set by this command) for the same system with the same phases which is defined in the current database (*i.e.*, the default database in the TDB module, or set by the SWITCH_DATABASE command, see *Section 5.4.21*). It 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 (see *Section 5.4.3*), 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 will be 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.

Synopsis 1: SET_AUTO_APPEND_DATABASE <additional database name>

Synopsis 2: SET_AUTO_APPEND_DATABASE

Ensuing Prompt: 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: Although it is impossible to manually list, reject and restore any phase from the appending database, the TDB 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.

5.4.21 SWITCH_DATABASE

Description: This command switches (or changes) from the current database to a new one, and reinitializes the entire TDB 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 file in the /DATA/ area for PC Windows Vista/XP/2000/NT4 environments, or the initd.tdb file in the \data\ area for all kinds of UNIX/Linux platforms), 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.

Synopsis 1: SWITCH_DATABASE <new database name>

Synopsis 2: SWITCH_DATABASE

Ensuing 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 or initd.tdb of your installed TCC/TCW/DICTRA packages; please refer to *Section 2.2.2.2*, as well as to *Section 3.2* of the *Thermo-Calc Database Guide*.

If the USER option is selected, a file that contains the definition of the USER database, and its correct path, must be provided. Under various UNIX/Linux platforms, 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.

Under the Windows Vista/XP/2000/NT4 environments, an **Open file** window will be popped up 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, so that the path (in the **Look in** box) and database definition (setup) file name (in the **File name** box) can be appropriately specified. The user can then open the chosen USER database, or may cancel such a Window session; in the latter case, the program will list all predefined databases and the user can then specify one of them to be switched on or proceed the USER option again.

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 command. 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.

5.5 Extended Commands

All the following commands can accept a wildcard * as the first name in the <list of names>, to indicate a special meaning. The special names /? and /ALL are also available. The question mark gives an option to execute the command for each entry. These are particularly useful for the DEFINE, REJECT and other TDB commands.

```
DEFINE [keyword] <list of names>
RESTORE [keyword] <list of names>
REJECT [keyword] <list of names>
LIST_SYSTEM [keyword] <list of names>
LIST_DATABASE [keyword] <list of names>
```

Some examples:

DEF_SYS /?	(All elements are query defined)
DEF_SYS /ALL	(All elements are defined)
DEF_SYS * FROM va TO fe	(Elements from VA to FE are defined)
DEF_SYS * FIRST LAST	(All elements are defined)
DEF_SYS /? FIRST fe	(Elements up to FE are query defined)
DEF_SYS * va fe	(Elements from VA to FE are defined)
DEF_SYS * 1 5	(The five first elements are defined)

The order of elements *etc.* is as in the LIST_DATABASE command.

A * can also be used at the end of a name to indicate that all names starting with the letters up to the * are referred.

(This page is intended to be empty)

6 Database Manager's Guide

Started from the TCCR version, this chapter had been extracted and separately prepared as the Chapter 3 in the individual document “***Thermo-Calc Database Guide***”.

>>> Open the document [Thermo-Calc Database Guide](#)

(This page is intended to be empty)

7 Tabulation Module (TAB)

7.1 Introduction

The **TABulation module**, TAB \Leftrightarrow , in the Thermo-Calc software is designed to tabulate all kinds of thermodynamic functions of any type of substances, stoichiometric or solution phases, or reactions.

The current TAB module always evaluates the most stable species of a substance in either a homogeneous state or in a reaction. Results are listed in table forms or plotted as graphs.

This module can also tabulate thermodynamic functions of a pure stoichiometric phase or of a solution phase at a fixed composition.

Furthermore, this module can also tabulate various kinds of partial derivatives for a pure stoichiometric phase or solution phase at a given composition.

The following commands are available in the current TAB module:

TAB:?		
BACK	LIST_SUBSTANCES	SWITCH_DATABASE
ENTER_FUNCTION	MACRO_FILE_OPEN	TABULATE_DERIVATIVES
ENTERREACTION	PATCH	TABULATE_REACTION
EXIT	SET_ENERGY_UNIT	TABULATE_SUBSTANCE
GOTO_MODULE	SET_INTERACTIVE	
HELP	SET_PLOT_FORMAT	
TAB:		

\Leftrightarrow Revision History of the TAB-Module User's Guide:

Nov 1985	First release (Edited by Bo Sundman)
Oct 1993	Second revised release (Edited by Birgitta Jönsson and Bo Sundman)
Sept 1998	Third revised release (Edited by Bo Sundman)
Jun 2000	Fourth revised and extended release (Edited by Pingfang Shi)
Nov 2002	Fifth revised release (Edited by Pingfang Shi); with minor changes in 2004/2006/2008

7.2 General Commands

7.2.1 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

Notes: Pressing the <RETURN> key without typing a specific and unique command name will list all the available TAB commands.

Specifying a unique TAB 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.

7.2.2 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 (also see Section 5.4.1I).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing 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 unique module name will list all the available TCC modules, as below:

```
NO SUCH MODULE, USE ANY OF THESE:  
SYSTEM_UTILITIES  
GIBBS_ENERGY_SYSTEM  
TABULATIONREACTION  
POLY_3  
BINARY_DIAGRAM_EASY  
DATABASE_RETRIEVAL  
REACTOR_SIMULATOR_3  
PARROT  
POTENTIAL_DIAGRAM  
SCHEIL_SIMULATION  
POURBAIX_DIAGRAM  
TERNARY_DIAGRAM  
MODULE NAME: <module name>
```

7.2.3 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).

Synopsis: BACK

7.2.4 EXIT

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

Synopsis: EXIT

7.2.5 PATCH

Description: This is a command intended only for system debugging; only those who think they know what they are doing should use it!

Synopsis: PATCH

7.3 Important Commands

7.3.1 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 SSUB4, or from a specific solution database such as SSOL4 and TCFE6), or of a **solution phase with a fixed composition** (getting data from a specific solution database such as SSOL4 and TCFE6), 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 TDB 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 stoichiometric 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 TDB 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).

Synopsis: TABULATE_SUBSTANCE

Ensuing 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 TCC 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; SSOL2, SSOL4; TCFE4, TCFE5, TCFE6, *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 ensuing 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 TDB 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 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 (post-processor) 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				
2002.10.23 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

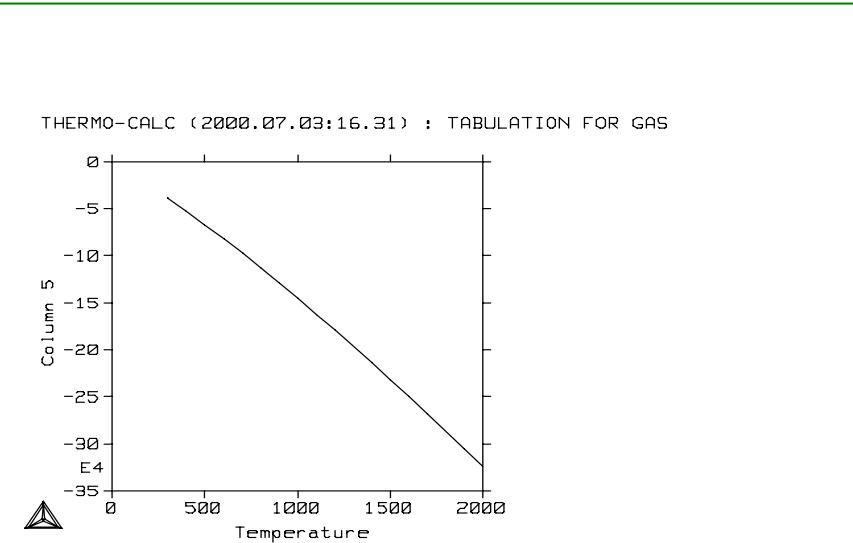
(Note that for a phase, as either an end-member or real solution, the listed specie name is irrelevant!).

Example Output 2: For the H2 species as a pure substance (using the SSUB3 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				
2002.10.23 14. 5.26				
Phase : GAS	Specie: H2	Pressure :	100000.00	

T (K)	Cp (Joule/K)	H (Joule)	S (Joule/K)	G (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.

7.3.2 TABULATEREACTION

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 SSUB database (by default), or the current database set by the SWITCH_DATABASE command (thus there is no need to use the TDB 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. Under UNIX/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 Vista/XP/2000/NT4, the TAB module only accepts upper cases, e.g., GA+S=GAS, 5AL+2FE=AL5FE2.
- 2) 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).
- 3) 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 can not be tabulated.
- 4) 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.

Synopsis: TABULATEREACTION

Ensuing Prompt: Same reaction? /Y/: <Y or N>

This prompt appears if there is at least one reaction already defined either by the command ENTERREACTION or TABULATEREACTION. If 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>
 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 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.

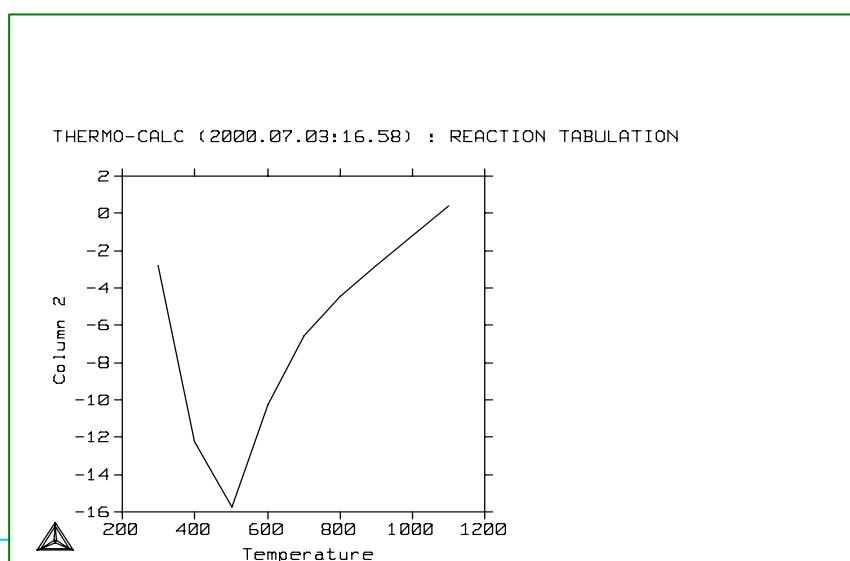
If answering Y (Yes), the program will both create a table with all thermodynamic functions as normal (which will be shown on screen) and generate a graph with temperature on the X-axis and a chosen property on a certain column in the table on the Y-axis (which will be plotted on screen and saved as 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 (*versus* temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (*i.e.*, the plotted column) for the graph 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 (post-processor) 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 SSUB 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
2000. 7. 3          16.54. 0
Reaction: S+GA=GA1S1
S stable as S_S
GA stable as GA_S
GA1S1 stable as GA1S1_S
*****
T      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.

7.3.3 ENTERREACTION

Description: This command is identical to TABULATEREACTION 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 SSUB database (by default), or the current database set by the SWITCHDATABASE command (thus there is no need to use the TDB module before this command). The user may select any other database by using the SWITCHDATABASE command in the TAB module.

The rules for specifying a reaction are the same as in the command TABULATEREACTION (see Section 7.3.2).

Synopsis: ENTERREACTION

Ensuing Prompt: Same reaction? /Y/: <Y or N>

This prompt appears if there is at least one reaction already defined either by the command ENTERREACTION or TABULATEREACTION. 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 (*versus* temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (*i.e.*, the plotted column) for the graph 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 (post-processor) 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.

7.3.4 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 solution phases, from the SSUB Substances Database. However, the user may select thermodynamic data from another database (for instance, SSOL Solutions Database, TCFE Steels/Fe-Alloys Database). This has been made possible since TCCN, 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 GES5 workspace for storing retrieved data. All the directly connected databases as predefined by the original database initiation file (TC_INITD file in the /DATA/ area for PC Windows Vista/XP/2000/NT4 environments, or the initd.tdb file in the \data\ area for all kinds of UNIX/Linux platforms) 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 ENTERREACTION or TABULATEREACTION for reactions.

When a predefined database or an appropriate USER database has been switched on, the TDB 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:
```

Synopsis 1: SWITCH_DATABASE <new database name>

Synopsis 2: SWITCH_DATABASE

Ensuing 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 show 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 front of one of the available predefined databases. For your convenience when switching/appending your own databases or lately purchase databases from TCSAB or its agents, you can simply add them to the predefined database list in the database initiation file TC_INITD or initd.tdb of your installed TCC/TCW/DICTRA packages; please refer to Sections 2.2.2.2 and 6.2.

If the USER option is selected, a file that contains the definition of the USER database, and its correct path, must be provided. Under various UNIX/Linux platforms, 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.

Under the Windows Vista/XP/2000/NT4 environments, an *Open file* window will pop up 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, so that the path (in the **Look in** box) and database definition (setup) file name (in the **File name** box) can be appropriately specified. The user can then open the chosen USER database, or may cancel such a Window session; in the latter case, the program will list all predefined databases and the user can then specify one of them to be switched on or proceed the USER option again.

Unlike the situation in the TDB 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.

7.3.5 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.

Synopsis: ENTER_FUNCTION

Ensuing 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/:
Low temperature limit /298.15/:
```

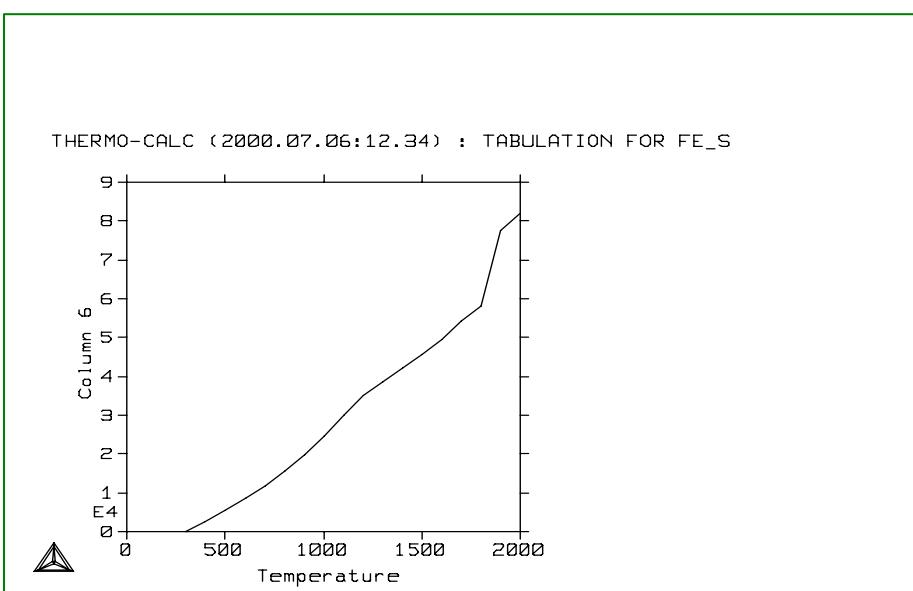
High temperature limit /2000/:
Step in temperature /100/:
Output file /try1/:
Graphical output? /Y/:
Plot column? /2/: 6

OUT PUT FROM THERMO-CALC
2000. 7. 6 12.31. 3

Column 6: InEnergy G+T*S-P*V

Phase : FE_S		Pressure : 100000.00			
Specie: FE					
T (K)	Cp (Joule/K)	H (Joule)	S (Joule/K)	G (Joule)	InEnergy

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



7.3.6 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 TDB 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).

Synopsis: TABULATE_DERIVATIVES

Ensuing 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 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
```

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/:
    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/:
Pressure /100000/:

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
    N            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
  O            in 1: 0.00000000E+00
  C            in 2: -1.36027071E+05
  N            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
  O            in 1: 0.00000000E+00
  C            in 2: -7.70620431E+04
  N            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
  O             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

```

TAB:

7.3.7 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.

Synopsis: LIST_SUBSTANCES

Ensuing Prompt: With elements /*/ <* or a set of elements>

The elements building up various species must be specified. * 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 SSOL database, various options gives different lists, as given below:

```

TAB:l-sub
With elements /*/:
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
N           NA          NB
ND          NI          NP
O            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/-1
F1/-       BR1/-        I1/-
H1/+1      H1O1/-1     SI1O2
CA1O1

TAB:l-sub
With elements /*/: fe cr ni c n o
Exclusively with those elements? /Y/:
C           CR          FE
N           NI          O
C1         C2          C3
C4         C5          C6
C7         N2          O2
TAB:l-sub fe cr ni c n o
Exclusively with those elements? /Y/: n
C           CR          FE
N           NI          O
C1         C2          C3
C4         C5          C6
C7         N2          O2
H2O         O1H1/-1    H1O1/-1
SI1O2
TAB:

```

7.4 Other Commands

7.4.1 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.

Synopsis 1: SET_ENERGY_UNIT <unit>

Synopsis 2: SET_ENERGY_UNIT

Ensuing Prompt: Energy unit (C=Cal, J=Joule) /J/: <unit>

Options: unit -- C (calories) or J (joule)

7.4.2 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 format; please refer to *Parts 12, 13 and 15*) and TAB file (as a simple text file as shown on screen). *Important Note:* 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 (1 for Windows Vista/XP/2000/NT4, and 9 for Linux and all types of UNIX platforms) may be altered with the SET_PLOT_ENVIRONMENT command in the SYS monitor or by your TC.INI file.

Synopsis: SET_PLOT_FORMAT <unit>

Options: unit -- the default set unit is 1 (Windows Vista/XP/2000/NT4) or 9 (Linux and all types of UNIX platforms) for graphical presentation on screen, or all other units for outputs as *.EXP and *.TAB files.

7.4.3 MACRO_FILE_OPEN

Description: MACRO is an extraordinary convenient and easier way of pre-defining sequences of various legal TCC (and DICTRA) commands stored in a so-called **MACRO file** (which is a simple textual file normally with the default extension “TCM” for the TCC software 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 TCC software; SYS, POLY, PARROT and DICTRA_Monitor module in the DICTRA software).

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 TCC software, if in the SYS module the SET_LOG_FILE command is used and a **LOG file** name is given before any other SYS/TDB/TAB/GES/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 current TCC session 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 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 TCC (and DICTRA) program.

All kinds of legal TCC (and DICTRA) commands and their required inputs can be stored into a MACRO file. A MACRO file must be terminated with the **EXIT** command, or can be interrupted (for the purposes of allowing further interactive operations by the user) in the SYS, GES, POLY, PARROT or POST modules with the **SET_INTERACTIVE** command.

Within a MACRO file (for previous versions up to TCCR), one could have as many as possible **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 TCC commands when the file is called by the TCC software.

However, this might be considered as somewhat tedious. Therefore, it is now (since TCCS) possible to have multiple-line **comment-blocks** inside a MACRO file, using the paired “@(” and “@)” signs (*i.e.*, the indicators that starting the lines for the “*begin comment*” and “*finish comment*”, respectively). 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 TCC 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 2nd 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, a textual copy of 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, *e.g.*

```
SET AXIS VAR 1 x(##3) 0 1,,,
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 TCC software from temporarily stopping at any pause by simply typing any character (except for the Y character) after specifying the name of a MACRO file.

Since TCCN, 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 TCC 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_Module command **SET_ECHO** at the every 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 TCC (and DICTRA) software, that are enforced according to the MACRO file(s).

Synopsis 1: MACRO_FILE_OPEN <name of a Macro file>

Synopsis 2: MACRO_FILE_OPEN

Ensuing Prompt: Macro filename: <name of a Macro file>
Specify the filename with the MACRO command. The default extension is “TCM”.

Notes: Under Windows Vista/XP/2000/NT4 environments, if an appropriate MACRO file is not given after the command, an **Open file** 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.*, TCM, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with executing various Thermo-Calc commands. The user may also cancel such an **Open file** window session, and thus the current MACRO file will not be opened.

If the MACRO file contains some SYS/TDB/TAB/GES/POLY/POST/PARROT/ED_EXP-module commands for setting *.LOG files, saving/reading GES5/POLY3/PARROT workspaces, switching USER databases, compiling experiments (from existing *.POP files), creating new *.PAR files, appending experimental data *.EXP files, plotting/dumping diagrams, *etc.*, a corresponding window (*e.g.*, **Save As**, **Open file**, **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.

More Notes: 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, *i.e.*, 1 (under Windows Vista/XP/2000/NT4) or 9 (under Linux and all types of UNIX platforms).

For an extensive example on using MACRO features for the TAB-Module, please refer to the TCEX12.TCM on the TCC installation area \TCEX\ and in the **TCCS Examples Book**. For more details on how a MACRO file is constructed, please refer to *Sections 8.10.26 and 14.2.8*.

7.4.4 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.

Synopsis: SET _INTERACTIVE

7.5 Plotting of Tables

As shown in Sections 7.3.1, 7.3.2, 7.3.3, and 7.3.5, it is convenient to plot thermodynamic properties for substances, pure or solution phases, or reactions (as columns in the tabulated tables) *versus* temperature.

To obtain a plot of a heat capacity or enthalpy of a substance, or of a pure or solution phase, or of a reaction among substances, the TABULATE_SUBSTANCE, TABULATE_REACTION or ENTER_REACTION command must first be typed, and the substance, phase and its fixed composition, or reaction must be configured. The questions about temperature limits, temperature step, pressure condition, *etc.*, must be answered, and the question “Output file” must be specified with a file name.

Two questions will then be answered as below:

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 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 (*versus* temperature as X-axis) as a graph on the screen. Simultaneously, all tabulated properties and the Y-axis setting (*i.e.*, the plotted column) for the graph 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 (post-processor) 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.

If you want a graphical plot, an *.EXP file with X/Y coordinates of plotted diagram, and an *.TAB file for the output table, just give the same TABULATE_SUBSTANCE or TABULATE_REACTION command twice (with two different file names under the “Output file” prompt).

8 Equilibrium Calculation Module (POLY)

8.1 Introduction

Knowledge of the thermodynamic equilibrium is an important factor for understanding properties of materials and processes. With a database of thermodynamic model parameters, it is possible to predict such properties and also to obtain driving forces for diffusion-controlled phase transformations and other dynamic processes.

With the comprehensive **Equilibrium Calculation module**, POLY[↓], it is possible to calculate many different kinds of equilibria and diagrams, in particular multicomponent phase diagrams. This is thus an important tool in developing new materials and processes. The current POLY module is its third version; this is why is often referred as POLY_3 in the Thermo-Calc software.

Different kind of databases can be used with the POLY module, and thus it can be used for alloys or ceramic system, as well as gaseous equilibria, aqueous solution involved heterogeneous interaction systems. Since TCCN (the Thermo-Calc Classic software, version N, up to 40 elements and 1000 species can be defined into a single system (previously 20 elements and 400 species) for equilibrium calculations.

Great care has been taken to provide the users with the most flexible tool. All normal thermodynamic state variables can be used to set as conditions in calculating equilibria, and as axes in plotting diagrams. A unique facility is to set the composition or any property of an individual phase as a condition. Any state variable can be varied along an axis in order to generate a diagram. During calculations of a diagram, complete descriptions of all calculated equilibria are stored, and in the diagram any state variable can be used as axis.

Together with the PARROT module, the POLY module is also used for critical assessment of experimental data in order to develop thermodynamic databases. The POLY module uses the Gibbs Energy System (GES) for modeling and data manipulations of the thermodynamic properties of each phase.

The following commands are available in the POLY module:

POLY_3:?		
ADD_INITIAL_EQUIlibRIUM	EXIT	REINITIATE_MODULE
ADVANCED_OPTIONS	GOTO_MODULE	SAVE_WORKSPACES
AMEND_STORED_EQUIlibRIA	HELP	SELECT_EQUIlibRIUM
BACK	INFORMATION	SET_ALL_START_VALUES
CHANGE_STATUS	LIST_AXIS_VARIABLE	SET_AXIS_VARIABLE
COMPUTE_EQUIlibRIUM	LIST_CONDITIONS	SET_CONDITION
COMPUTE_TRANSITION	LIST_EQUIlibRIUM	SET_INPUT_AMOUNTS
CREATE_NEW_EQUIlibRIUM	LIST_INITIAL_EQUIlibRIA	SET_INTERACTIVE
DEFINE_COMPONENTS	LIST_STATUS	SET_NUMERICAL_LIMITS
DEFINE_DIAGRAM	LIST_SYMBOLS	SET_REFERENCE_STATE
DEFINE_MATERIAL	LOAD_INITIAL_EQUIlibRIUM	SET_START_CONSTITUITION
DELETE_INITIAL_EQUILIB	MACRO_FILE_OPEN	SET_START_VALUE
DELETE_SYMBOL	MAP	SHOW_VALUE
ENTER_SYMBOL	POST	STEP_WITH_OPTIONS
EVALUATE_FUNCTIONS	READ_WORKSPACES	TABULATE
POLY_3:		

Note that, since TCCS, the commands SPECIAL_OPTIONS and SET_MINIMIZAION_OPTIONS (the later one was introduced in the TCCR version) has been merged into the new command ADVANCED_OPTIONS; and the command RECOVER_START_VALUES has been removed, due to that is not relevant to the POLY module anymore.

[↓] Revision History of the POLY Module User's Guide:

Mar 1991	First release (Edited by Bo Jansson and Bo Sundman)
Oct 1993	Second revised release (with version J) (Edited by Bo Jansson and Bo Sundman)
Oct 1996	Third revised release (with version L) (Edited by Bo Sundman)
Nov 1998	Fourth revised release (with version M) (Edited by Bo Sundman)
Jun 2000	Fifth revised and extended release (Edited by Pingfang Shi)
Nov 2002	Sixth revised and extended release (Edited by Pingfang Shi)
June 2004	Seventh revised release (Edited by Pingfang Shi)
May 2006	Eighth revised and extended release (Edited by Pingfang Shi)
Apr 2008	Ninth revised and extended release (Edited by Pingfang Shi)

One of the major improvements since the TCCR/TCW4 software version is that the recently-implemented **Global Minimization Technique** is used to assure that the present minimum in an equilibrium calculation is the most stable minima for the specified conditions. This new technique, which is based on the traditional GEM (Gibbs Energy Minimization) Technique (*i.e.*, the ordinary POLY Minimization routines used in previous versions, where pre-knowledge of miscibility gaps in involved phases are necessary, otherwise, metastable equilibria instead of the stable equilibria may be obtained), will ultimately prevent a calculation from reaching an undesired metastable or unstable (local) equilibrium in a defined system, and automatically detect possible miscibility gap(s) and automatically create additional composition sets in a solution phase if needed for handling single or multiple miscibility gaps. Therefore it is no longer necessary for the user to specify additional composition sets in advance.

A **Direct Global Minimization** can be performed on conditions: N, n (comp), B, b (comp), w (comp), x (comp), T, and P, but not when combined conditions as *e.g.* $w(a) - 3 \cdot w(b) = 1$ are used or when an activity or potential condition is used. For all other types of conditions where regular minimization converges, **Indirect Global Minimization**, *i.e.* global test and corrections, if necessary, are performed until the lowest minimum is found.

- **Direct Global Minimization:** From the mesh of Gibbs energy, find the set of grid points that gives the lowest energy solution under the specified conditions. This set of grid points provides starting combination of phases and their constitutions for regular minimization to find the exact equilibrium solution. This solution will be then subject to a global test as described below.
- **Indirect Global Minimization:** Under certain conditions, direct approach is impossible. In this case, regular minimization is performed first and then a check is performed in order to see if the found local minimum is a global one by checking if all grid points are above the equilibrium Gibbs energy plane. If not, then recalculate by including these grid points until no grid point is above the equilibrium Gibbs energy plane from the previous step.

The full-scale and full-scope usage of the Global Minimization Technique has been extended from for only single-point calculations within TCCR/TCW4 to for all types of calculations (of single-points, property diagram stepping and phase diagram mapping) within TCCS/TCW5.

The use of Global Minimization Technique may increase the computation time, while it is not an issue at all, thanks for the rapid developments of computer hardware nowadays.

- The main cost in time comes from the calculation of Gibbs energy at each grid point generated by properly meshing the composition space for each entered phase. In a typical multicomponent system calculation, about 100MB of RAM memory is needed in storing the mesh of Gibbs energies.
- An additional (but much smaller) cost in time comes from finding the set of grid points in the above mesh that give the lowest energy solution. This solution is where POLY starts its ordinary minimization. When POLY has found an equilibrium, the equilibrium Gibbs energy surface is compared to the mesh to assure that no grid point is below the surface, *i.e.* a global minimization has been reached.

Global Minimization is now performed by default in single-point or stepping or mapping equilibrium calculations, but can of course be turned *off* (and *on* again by repeating the command-sequence of ADVANCED_OPTIONS GLOBAL_MINIMIZATION) by the user for specific purposes. This means that truly stable equilibrium should be guaranteed for single-points, stepping and mapping calculations.

- A completely new stepping and mapping procedure that ensures Global Minimization everywhere it is critical has been developed and been made available in TCCS/TCW5. These newly re-written STEP/MAP routines are very important for stepping/mapping calculations in multicomponent systems where there are complex miscibility gaps in some phases, and it does not require having any “good” guess of starting points. Therefore, TCCS/TCW5 can automatically handle complex solution phases with single or multiple miscibility gaps [for instance, a solution phase that is thermodynamically described as a single phase in a Thermo-Calc database, such as FCC, BCC or HCP phases, may be split into two or several composition-sets/phases that are presented in an equilibrium state as metallic phase(s), carbide(s), nitride(s), carbonitride(s), nitrocarbide(s), and so on], and can thus ensure the correct and complete phase diagrams and property diagrams in multicomponent systems, without bothering staring points.

Please note that since TCCSTCW5, the IWSW dimension in the PLOY workspace has been increased from 40000 to 200000, and thus one can not read POLY3 files created with TCCS/TCW5 in earlier versions.

8.2 Getting Started

The basic steps to calculate an equilibrium or a diagram are described below. For more details, please refer to the specific parts in this manual.

Retrieving Data:

Thermodynamic data for the system must first be retrieved from appropriate database(s) in the TDB module. In the specialized and advanced modules like BINARY, TERNARY, POURBAIX and SCHEIL, there is an automatic retrieval of the necessary thermodynamic data. One may also use the `DEFINE_MATERIAL` command in the POLY module. If a calculation has already been made and `SAVED`, one may instead `READ` from the `SAVED` file.

Setting Conditions:

After getting all available thermodynamic data for a defined system, the user must give a number of conditions in the POLY module, in order to make the degree of freedom in the multicomponent system equal to zero. Conditions are set by the `SET_CONDITION` command. One may also use the `CHANGE_STATUS` command if there is a condition that a specific phase is in the `FIXED` status, *i.e.*, as a stable phase. Definitions of system components, and their reference states, can be changed by the `DEFINE_COMPONENT` and `SET_REFERENCE_STATE` commands, respectively. Besides the standard state and derived variable, additional variables, functions and tables can be defined by the `ENTER_SYMBOL` command.

Calculating Equilibria:

The `COMPUTE_EQUILIBRIUM` command in the POLY module tries to find the stable phase assemblage at the given equilibrium conditions for the heterogeneous interaction system. The result can be obtained by the `LIST_EQUIlibrium` command. If a calculation does not converge, you can try to modify the conditions to be as simple as possible. The simplest types of conditions are fixed temperature and pressure and an overall composition. Even if that is not what you want to calculate, it may help to calculate an initial equilibrium with such conditions as a first step. After that, you may try to play with phase status, enthalpy conditions, compositions of individual phases, *etc.* It is easy to set non-sensible conditions and the program will not tell you that the conditions are impossible in any other way than failure to calculate the equilibrium.

Generating Diagram Properties:

When one equilibrium calculation has converged, the next steps are to set stepping/mapping axis variables and to perform the stepping or mapping calculations in the POLY module. All kinds of predefined standard state and derived properties, as well as user-entered symbols (variables/functions/tables), which can be used for plotting phase diagrams and property diagrams, are generated with the `STEP` or `MAP` commands.

Plotting Diagrams:

Phase diagrams and property diagrams are plotted in the post-processor, namely the POST module. After a `STEP` or `MAP` command, axes are automatically set, using some default variables (normally the axis variables set before stepping or mapping) as the X/Y axis variables. Otherwise, one must first use the command `SET_DIAGRAM_AXIS` to define the X/Y-axes. Using various POST-module commands can further refine the diagram appearances, and can export the generated diagrams in various graphical forms.

8.3 Basic Thermodynamics

Extensive introduction of various thermodynamic terms used in the Thermo-Calc software/database/interface package has been given in the separate document [Thermo-Calc Software System](#).

8.3.1 System and phases

In thermodynamics, one always deals with a SYSTEM that can be closed or open for exchange of matter, heat and work with its surroundings. In a multicomponent system, the matter will always appear in one or more stable PHASES. A phase means a homogeneous part of the system. Homogeneous means that it is uniform in composition, temperature and pressure, and has the same structure everywhere. The same phase may often appear in many separate places in a volume, e.g., dust particles in air. If there are several homogeneous phases, the system becomes heterogeneous, and thermodynamics governs the heterogeneous interactions in the system.

40 elements (i.e., 40 components) and 1000 species can be defined into a single system (previously 20 elements and 400 species before TCCN) for equilibrium calculations. An individual solution phase can have up to 10 sublattices. An ideal mixture phase (e.g., GAS) can cover at most 1000 species, whilst a non-ideal solution phase (e.g., AQUEOUS, LIQUID, SLAG) can contain maximum 200 species (since TCCP, increased from 80 in TCCN).

8.3.2 Species and constituents

As mentioned in [Section 2.2 of the document Thermo-Calc Software System](#), the composition dependence of the thermodynamic properties of a phase may be modeled by using an expression showing how the Gibbs energy depends on the fractions of the *constituents* in the phase. The phase constituents can either be the elements [e.g., Fe, Si, C, O], or molecular-like aggregates which can be neutral [e.g., Fe₃C, FeSi, Fe_{0.87}O, SiO₂, H₂O, CH₃COOH] or charged [e.g., Fe⁺², Fe⁺³, HO₂⁻¹, OH⁻¹, Fe₂(OH)₂⁺⁴].

All such constituents are called SPECIES. A species can be defined as a constituent of one or several phases, and its presence in a particular phase can be real or fictive (postulated from a corresponding thermodynamic model for the phase).

A special species used for modeling is the vacancy, always denoted as VA. The vacancy is used as a constituent on sublattices where there are sites that normally are empty or have defects. VA is by default defined as a system component, and its chemical potential is always set to zero.

Another special species is for the electron, denoted as either /- (in gaseous, liquid or solid phases) or ZE (in an aqueous solution phase), in order to appropriately express the stoichiometry of a charged species in a phase.

Both vacancy (VA) and electron (/ - in gaseous, liquid or solid phases or ZE in an aqueous solution phase) can be entered as “special elements” in definitions of some databases. Whenever ZE and VA (but not /-) are defined in a system, they are by default also regarded as **special components** in the defined system, while only ZE has a unique chemical potential and other thermodynamic properties, calculated by the GES module using an appropriate model associated with the aqueous solution phase and REF_ELECTRODE phase (i.e., the hypothetical electrode); see [Section 8.12](#).

Therefore, a system component is normally an element in a defined system (especially for alloys), but it can also be an existing species in a defined system: e.g., for an aqueous-free Cr-Fe-O-H system, the system components (in addition to the default defined VA for vacancy) can be defined as [Fe, Cr, O and H], or [Fe, Cr, O₂ and H] or [FeO, Cr₂O₃, O₂ and H₂] or another set; for an aqueous-bearing Cr-Fe-O-H system, the system components (in addition to the default defined VA for vacancy) are normally defined as [H₂O, H+1, Cr, Fe and ZE]. One can even define a system component as an appropriate combination of defined elements (e.g., CrC and FeC in an Cr-Fe-C system) and have electron (/ - or ZE) as a part of chemical formula for an element-combined component (e.g., H₁O₂/-1 and FE₁/+2 in a gaseous mixture phase, or H₁O₂ZE+1 and FE₁ZE-2 in an aqueous solution phase), but the reference state of such a component must always be SER (Stable Element Reference).

Please note that for a heterogeneous interaction system without considering ZE as a special component, the number of components is equal to the number of elements in the system; while for an aqueous bearing heterogeneous interaction system, ZE is considered as an additional system component, and thus the number of components is equal to the number of elements in the system plus one.

8.3.3 Standard state variables and partial derivatives

Classic Thermodynamics deals only with systems that are in equilibrium, *i.e.*, in a state stable against internal fluctuations in a number of variables, such as temperature and composition. These variables that have defined values or properties at the equilibrium state are called **state variables**. Other examples of state variables are pressure (P), and chemical potential (μ). Thermodynamics provides a number of relations between these state variables that make it possible to calculate the value of any other variable at equilibrium.

A state variable can be of two types, extensive or intensive. The value of an **extensive variable**, *e.g.*, volume, depends on the size of the system, whereas the value of an **intensive variable**, *e.g.*, temperature, is independent of the size of the system. Each type of state variable has a complementary variable of the other type. The variable complementing the volume is pressure, while the variable complementing the composition of a component is its chemical potential.

It is worth mentioning here that the activity of a component can always be obtained from its chemical potential using a simple mathematical relationship. It is also possible to choose any convenient reference state for the activity or the chemical potential. One of the advantages with a thermodynamic databank on a computer is that, in most cases, such reference state changes can be handled internally without troubling the user.

If the work that can be exchanged with the surroundings is limited to pressure-volume work, the state of equilibrium of a system can be obtained by assigning values to exactly $N+2$ state variables where N is the number of components of the system.

Note that the Thermo-Calc software distinguishes between *components* of a system and *constituent* (*i.e.*, *species*) of a phase in the system. Many state variables require one or the other. By default, the elements are defined as the system components, but this definition can be changed with the POLY command `DEFINE_COMPONENT`. For instance, if the elements are Ca, Si and O, the other set of components can be defined as CaO, SiO and O2; in a pure water system, the components are normally defined as H2O and H+. However, one cannot change the number of components when using this command.

A state variable is a defined thermodynamic quantity either for the whole system, or for a component in the system, or a species in a specific substitutional phase, or a constituent (*i.e.*, a species on a specific sublattice site) in a specific phase.

The POLY module operates on a thermodynamic system in a stable or meta-stable or partial/local equilibrium state that is described by state variables. Examples of state variables are temperature, mole fraction, chemical potential and activity of a component (in the system or in a specific phase), enthalpy (of the system or a specific phase), *etc.* In the POLY module, a general notational method has been designed for the important set of standard state variables and their partial derivatives (or called as derived variables).

Common examples of state variables in a defined system are:

MNEMONIC	MEANING
T	Temperature (in K)
P	Pressure (in Pascal)
N	Total system size (in moles)
B	Total system size (in grams)
N (H)	Overall number of moles of the hydrogen component
B (H ₂ O)	Overall mass (grams) of the H ₂ O component
X (FE)	Overall mole fraction of the Fe component
W (AL ₂ O ₃)	Overall mass fraction of the AL ₂ O ₃ component
Y (HCP, CR#1)	Site fraction of the Cr species on the first sublattice site in the HCP phase
X (LIQUID, FE)	Mole fraction of the Fe component in the liquid phase
W (FCC, C)	Mass fraction of the C component in the FCC phase
NP (BCC)	Mole number of the BCC phase

BP (BCC)	Mass (grams) of the BCC phase
VP (BCC)	Volume (m^3) of the BCC phase
DGM (BCC)	Driving force of the BCC phase per mole of components
QF (SIGMA)	Phase stability function of the SIGMA phase
MUR (C)	Chemical potential of the C component (wrt its defined reference state)
ACR (C)	Activity of the C component (wrt its defined reference state)
LNACR (C)	Natural logarithm of activity of the C component [$\ln ACR(C) = MUR(C) / RT$]
MUR (FE+3, AQ)	Chemical potential of the Fe^{+3} species related to the aqueous solution phase
ACR (FE+3, AQ)	Activity of the Fe^{+3} species related to the aqueous solution phase
LNACR (FE+3, AQ)	Natural logarithm of activity of the Fe^{+3} species related to the aqueous solution phase [$\ln ACR(FE+3, AQ) = MUR(FE+3, AQ) / RT$]
HM	Total enthalpy per mole of components in the system (in J/mol)
HM (FCC)	Enthalpy per mole of components of the FCC phase (in J/mol)
HM . T	Isobaric heat capacity per mole of components in the system (in J/mol/K)
HM (FCC) . T	Isobaric heat capacity per mole of components of the FCC phase (in J/mol/K)
P . T	The slope $\partial P / \partial T$ of a phase boundary on a $P-T$ phase diagram. Note that the equilibrium with a phase assemblage must have been calculated first.
T . W(SI)	The slope $\partial T / \partial W(Si)$ of a phase boundary on a $T-W(Si)$ phase diagram with regard to mass of the component in the system
T . X(LIQ, CR)	The slope $\partial T / \partial X(Liq, Cr)$ of a phase boundary on a $T-X(Liq, Cr)$ phase diagram with regard to mole fraction of the component in the phase
Many more see details below and in the document of Thermo-Calc Software System

Note that the state variables involving components can be used for the defined components, but not for any species. To define new components in a defined system, the `DEFINE_COMPONENT` command should be used.

The basic intensive and extensive variables which are suitable in the POLY module of the Thermo-Calc software system are listed and briefly described in *Table 1* (of the document *Thermo-Calc Software System*), and are also be dealt with in the following subject-keywords when using the on-line help feature of the POLY module:

- `INFO INTENSIVE VARIABLES` [for various intensive state variables]
- `INFO EXTENSIVE VARIABLES` [for various extensive state variables]

Derivatives of state variables can be evaluated using a dot “.” between two state variables. Many derived variables of a defined system, or of a certain system component, or of a given phase, or of a specific component in a defined phase, can be easily obtained using appropriate partial derivatives of state variables, such as heat capacity, thermal expansivity, isothermal compressibility, among others. For the details of various derived variables which are suitable in the POLY and POST modules of the Thermo-Calc software system, please refer to *Section 2.6* in the document *Thermo-Calc Software System*, and also refer to the on-line help feature of the POLY module:

- `INFO DERIVED VARIABLES` [for various derived variables (partial derivatives)]

The lists below, of basic intensive and extensive state variables, as well as of derived variables (partial derivatives), are not exhaustive, but many other remaining state variables and partial derivatives can be obtained through direct calls or by using appropriate combinations of the predefined ones. For more details, please refer to [Sections 2.5 \[Equilibrium State and State Variables\]](#) and [2.6 \[Derived Variables \(Partial Derivatives\)\]](#) in the document *Thermo-Calc Software System*.

8.3.3.1 Intensive properties of the system, or of a component

MNEMONIC	MEANING	Notes
T	Temperature	
P	Pressure	
MU (comp)	Chemical potential of a component	(using the default reference state)
MUR (comp)	Chemical potential of a component	(using a re-set reference state)
AC (comp)	Activity of a component	(using the default reference state)
ACR (comp)	Activity of a component	(using a re-set reference state)
LNAC (comp)	In(activity) of a component	(using the default reference state)
LNACR (comp)	In(activity) of of a component	(using a re-set reference state)

Note: $\ln ac = MU/RT$, and $\ln acr = MUR/RT$

The reference states for activities and chemical potentials of system components are default defined by a chosen database (which is usually the default SER- “Stable Element Reference” state), but can be changed by pre-definition in special modules (such as SCHEIL and POURBAIX module) or by the user utilizing the SET_REFERENCE_STATE command. Remember that after such a change of reference state, $MU(\text{comp}) \neq MUR(\text{comp})$, $AC(\text{comp}) \neq ACR(\text{comp})$, and $LNAC(\text{comp}) \neq LNACR(\text{comp})$.

The units for the above variables in the POLY module are always as:

- K for T
- Pascal for P (1 bar = 100000 Pa; 1 atm = 101325 Pa)
- J/mol for various chemical potential quantities [MU or MUR, for a system component]
- dimensionless for various activity quantities [AC or ACR, or LNAC or LNACR, for a system component]

However, when listing out the equilibrium results (through the LIST_EQUILIBRIUM command) the temperature condition is additionally shown up in °C unit. Moreover, in the POST module one may simply use the “T_C” quantity as an axis variable; and in both the POLY and POST module one can additionally enter some appropriate functions in order to convert such variables to other units through the ENTER_SYMBOL command.

8.3.3.2 Intensive properties of a species relative to a specific phase

For systems with some solution/mixture phases which have been defined as *single-substitutional-lattice solution phases* (such as a gaseous mixture, aqueous solution, metallic liquid solution, slag solution, MeO solid solution, etc., in some existing databases), it is sometimes also convenient to directly use a condition as of an activity or chemical potential for a specified phase constituent (such as a gaseous species) *relative to* a specific solution phase. This is possible by using the following state variables:

MNEMONIC	MEANING	Notes
MU (sp, ph)	Chemical potential of the species <i>relative to</i> the specific phase	$MU(\text{sp}, \text{ph}) = MUR(\text{sp}, \text{ph})$
MUR (sp, ph)		
AC (sp, ph)	Activity of the species <i>relative to</i> the specific phase	$AC(\text{sp}, \text{ph}) = ACR(\text{sp}, \text{ph})$
ACR (sp, ph)		
LNAC (sp, ph)	In(activity) of the species <i>relative to</i> the specific phase	$LNAC(\text{sp}, \text{ph}) = LNACR(\text{sp}, \text{ph})$
LNACR (sp, ph)		

Note: $\ln ac = MU/RT$, and $\ln acr = MUR/RT$

The reference state of a phase constituent (species) is the specified *single-substitutional-lattice* solution/mixture phase itself consisting of only the <species> at the current temperature and 1 bar (100000 Pascal). Any solution/mixture phase with a *single substitutional lattice* description can be used, but the <species> must of course be a constituent of the phase. Like all other state variables, these species-related state variables could also be used in the post-processor.

The units for the above variables in the POLY module are always as:

- J/mol for various chemical potential quantities [MU or MUR, for a species related to a specific phase]
- dimensionless for various activity quantities [AC or ACR, or LNAC or LNACR, a species related to a specific phase]

Please note that the concepts of chemical potentials and activities (and activity coefficients) are only appropriately and meaningful for some *single-substitutional-lattice solution phases* (such as gaseous mixture, aqueous solution, metallic liquid solution, slag solution) as they may have clear/definitive physical meanings.

In the following, let's take the gaseous mixture and aqueous solution phases as examples, to how these quantities are really handled inside the Thermo-Calc software:

- Ideal Gas mixture:

$$\begin{aligned} \text{ACR(sp, gas)} &= P_{\text{sp}} / P \\ &= \exp [\text{MUR(sp, gas)}] / RT \\ &= \exp [\text{MU(sp, gas)} - \mu_{\text{pure species}}^{\circ}] / RT \end{aligned}$$

where P_{sp} is partial pressure of the species in gaseous mixture;
 P is total pressure in the system;
 $\mu_{\text{pure species}}^{\circ}$ is the chemical potential at its reference state (*i.e.*, the standard molar Gibbs free energy of a pure gas under the defined reference P-T condition).

- Non-ideal gas mixture:

$$\begin{aligned} \text{ACR(sp, gas)} &= f_{\text{sp}} / f_{\text{sp}}^{\circ} \\ &= \exp [\text{MUR(sp, gas)}] / RT \\ &= \exp [\text{MU(sp, gas)} - \mu_{\text{pure species}}^{\circ}] / RT \end{aligned}$$

where f_{sp} is model-dependent fugacity of the species in gaseous mixture;
 f_{sp}° is model-dependent fugacity of the pure gas (at same P-T condition), which is determined its EOS (equation of state) and equals to its fugacity coefficient times total pressure;
 $\mu_{\text{pure species}}^{\circ}$ is the chemical potential at its reference state (*i.e.*, the standard molar Gibbs free energy of a pure gas under the defined reference P-T condition).

- Aqueous solution:

$$\text{ACR(H}_2\text{O, aq)} = \gamma_{\text{H}_2\text{O}} * y(\text{aq, H}_2\text{O}) \quad (\text{for solvent})$$

$$\begin{aligned} &= \alpha_{\text{H}_2\text{O}} \\ &= \exp [\text{MUR(H}_2\text{O, aq)}] / RT \\ &= \exp [\text{MU(H}_2\text{O, aq)} - \mu_{\text{pure H}_2\text{O}}^{\circ}] / RT \end{aligned}$$

$$\text{ACR(sp, aq)} = \gamma_{\text{sp}} * y(\text{aq, sp}) / y(\text{aq, H}_2\text{O}) \quad (\text{for solute})$$

$$\begin{aligned} &= \alpha_{\text{sp}} * y(\text{aq, sp}) / \text{ML(aq, sp)} / y(\text{aq, H}_2\text{O}) \\ &= \alpha_{\text{sp}} / 55.508435 = \alpha_{\text{sp}}^m \\ &= \exp [\text{MUR(sq, aq)}] / RT \\ &= \exp [\text{MU(sq, aq)} - \mu_{\text{hypothetical species}}^{\circ}] / RT \end{aligned}$$

where $\gamma_{\text{H}_2\text{O}}$ and $\alpha_{\text{H}_2\text{O}}$ are model-dependent acvity coefficients and activity of solvent (H_2O) in aqueous solution;

γ_{sp} and α_{sp} are model-dependent acvity coefficients and activity (*both as of mole-fraction based*) of solute species in aqueous solution, while α_{sp}^m is the model-dependent acvity (*as of molality based*) of solute species in aqueous solution;

$\mu_{\text{pure H}_2\text{O}}^{\circ}$ is the chemical potential of pure water [*i.e.* the standard molar Gibbs free energy of the solvent H_2O at its reference state (*according to the Raoult's Law*) under the defined reference P-T condition];

$\mu_{\text{hypothetical species}}^{\circ}$ is the chemical potential at its reference state [*i.e.*, the standard molar Gibbs free energy of the “*hypothetical solute species*” at a *hypothetical state at unit molality* (*1 mole of solute dissolved into 1.0 kg of solvent*) but in which the environment of each molecular is the same as at infinite dilution, *according to the Henry's Law*) under the defined reference P-T condition].

Please also note that the wild card “*” or the dollar “\$” sign normally does not work properly for the activity and/or chemical potential properties. For examples, one should avoid using *e.g.* $\text{ACR}(*, *)$ or $\text{MUR}(*, *)$ for all the species *relative to* all phase, or $\text{ACR}(*, \text{phase})$ or $\text{MUR}(*, \text{phase})$ for all the species *relative to* a specific phase, or $\text{ACR}(\text{species}, *)$ or $\text{MUR}(\text{species}, *)$ for a specific species *relative to* all phases; this is because of the fact that the $\text{ACR}(\text{species, phase})$ and or $\text{MUR}(\text{species, phase})$ quantities can only be *appropriately and meaningfully applied* to substitutional solution phases (such as GAS, AQUEOUS, LIQUID)

and to solution phases without interstitial sublattice set. But one can use specific names for species and (applicable) phase in such cases. For example,

```
SHOW_VALUE ACR (FE+2 , AQUEOUS) ... lists activity of the Fe+2 species in AQUEOUS solution phase
SHOW_VALUE MUR (C2 , GAS) ... lists chemical potential of the C2 species in GAS mixture phase
SET_AXIS_VAR Y AC(O2 , GAS) ... sets activity of the O2 species in GAS mixture phase as Y-axis
```

8.3.3.3 Normalizing of extensive properties

As described in *Sections 2.5 and 2.6* in the document *Thermo-Calc Software System*, the Thermo-Calc software can handle many different types of extensive variables (energetic, compositional, or constitutional composition).

For all extensive properties, a suffix can be added to the mnemonic name to indicate a normalized extensive property. A mnemonic name is a combination of letters that makes it easy to remember what the combination stands for.

The general rules for normalizing a specific extensive property (*and for corresponding unit changes*) are:

Extensive property Z for the whole system:		Examples
Z	Extensive property Z in the whole system	G
ZM	Extensive property Z per moles of components in the system	GM
ZW	Extensive property Z per mass (gram) in the system	GW
ZV	Extensive property Z per volume (m^3) in the system	GV
Extensive property Z for a specific phase:		Examples
Z(phase)	Extensive property Z for the current amount of a phase	G (AB)
ZM(phase)	Extensive property Z per mole of a phase	GM (AB)
ZW(phase)	Extensive property Z per mass (gram) of a phase	GW (AB)
ZV(phase)	Extensive property Z per volume (m^3) of a phase	GV (AB)
ZF(phase)	Extensive property Z per mole formula unit of a phase	GF (AB)
Extensive property Z for a specific component in the system:		Examples
Z(comp)	Extensive property Z of a component in the whole system	N (Fe)
ZM(comp)	Extensive property Z of a component per moles of all components in the system	NM(Fe)
ZW(comp)	Extensive property Z of a component per mass (gram) of the system	NW(Fe)
ZV(comp)	Extensive property Z of a component per volume (m^3) of the system	NV(Fe)
Extensive property Z for a specific component in a phase:		Examples
Z(ph, comp)	Extensive property Z of a component in a phase in the system	N (AB, Fe)
ZM(ph, comp)	Extensive property Z of a component in a phase per mole of all components in the system	NM(AB, Fe)
ZW(ph, comp)	Extensive property Z of a component in a phase per mass (gram) of a system	NW(AB, Fe)
ZV(ph, comp)	Extensive property Z of a component in a phase per volume (m^3) of a system	NV(AB, Fe)

Notes: In order to have clearer physical meanings for various normalizations of system quantities and phase quantities, the following general rules have been enforced since TCCS/TCW5/DICTRA25:

- For system quantities, they should be normalized by the total system size (in terms of N, B or V), *as usual*;

- For phase quantities, they should be normalized by the phase amount [in terms of NP(ph), BP(ph) or VP(ph)], except for the normalized NPM(ph), BPW(ph) and VPV(ph) quantities [as well as for the normalized DGM(ph), DGW(ph), DGV(ph) and DGF(ph) properties] where the suffixes M, W, and V still mean the normalizations are performed with regard to the total system size (in terms of N, B and V, respectively).

8.3.3.4 Energetic extensive properties of the system or of a phase

The following mnemonic names are used for the energetic extensive properties of the system or a phase (with a phase name of *e.g.* AB):

MNEMONIC	MEANING	Examples (Possible Normalizations)				
H	Enthalpy of the system	H	HM	HW	HV	
H(phase)	Enthalpy of a phase	H (AB)	HM (AB)	HW (AB)	HV (AB)	HF (AB)
S	Entropy of the system	S	SM	SW	SV	
S(phase)	Entropy of a phase	S (AB)	SM (AB)	SW (AB)	SV (AB)	SF (AB)
V	Volume of the system	V	VM	VW		
V(phase)	Volume of a phase	V (AB)	VM (AB)	VW (AB)		VF (AB)
G	Gibbs energy of the system	G	GM	GW	GV	
G(phase)	Gibbs energy of a phase	G (AB)	GM (AB)	GW (AB)	GV (AB)	GF (AB)
A	Helmholtz energy of the system	A	AM	AW	AV	
A(phase)	Helmholtz energy of a phase	A (AB)	AM (AB)	AW (AB)	AV (AB)	AF (AB)
U	Internal energy of the system	U	UM	UW	UV	
U(phase)	Internal energy of a phase	U (AB)	UM (AB)	UW (AB)	UV (AB)	UF (AB)

Notes: Suffixes M, W, V and F can be applied to these quantities (as shown in the *Examples* column). The normalizations of such system quantities should be preformed with regard to the total system size (in terms of N, B or V), and of such phase quantities should be conducted with regard to the phase amount [in terms of NP(ph), BP(ph) or VP(ph)] of the phase.

The units for the above variables (when without any suffix) in the POLY module are always as:

- J for H, H (phase); G, G (phase); A, A (phase); and U, U (phase)
- J/K for S, S (phase)
- m³ for V, V (phase)

When a suffix (M/W/V/F) is used in such properties, their units should be alternated according to the suffix-implication [M (per mole), W (per mass in grams), V (per volume in m³) or F (per mole formula unit)]. For instance, for the M-suffixed quantities, their units should be:

- J/mol for HM, HM (phase); GM, GM (phase); AM, AM (phase); and UM, UM (phase)
- J/mol/K for SM, SM (phase)
- m³/mol for VM, VM (phase)

In both the POLY and POST module one can additionally enter some appropriate functions in order to convert such variables to other units through the ENTER_SYMBOL command.

Some energetic quantities (such as heat capacity, thermal expansion, isothermal compressibility), as well as variations of some energetic and/or compositional quantities with controlling conditions, can be defined as partial derivatives (derived variables). For details, refer to *Section 8.3.3.9* below and *Section 2.6 in the document Thermo-Calc Software System*.

As a unique feature, the Thermo-Calc software can calculate **chemical driving force** (*i.e.*, **thermodynamic factor**) for phase transformation (precipitation of specific phase in a defined system). This energetic quantity can only be applied to a phase and cannot be used as conditions. Moreover, it should always be used with a specific suffix (M/W/V/F), which are normalized with regard to the total system size [in terms of N or B or V] in the whole system:

MNEMONIC	MEANING	Notes
DGM (phase)	Driving force of a phase per mole of components	
DGW (phase)	Driving force of a phase per mass of components	
DGV (phase)	Driving force of a phase per volume of components	
DGF (phase)	Driving force of a phase per formula unit of components	All of them are always divided by RT , and are thus dimensionless.

8.3.3.5 Total size of the system

The following mnemonic names are used for the total amounts of compounds in the system (*i.e.*, total size):

MNEMONIC	MEANING	Examples (Possible Normalizations)			
N	Mole number of components in the system	N		NW	NV
B	Mass (gram) of components in the system	B	BM		BV

Note: Suffixes M, W and V may be applied to these quantities (as shown in the Examples column).

Remark: The combinations NM and BW are illegal (and not very interesting as they will always be unity)!
BV is the density of the system.

The units for the above variables and their suffixed quantities in the POLY module are always as:

- N in moles; NW in mol/g; NV in mol/m³
- B in grams; BM in g/mol; BV in g/m³

In both the POLY and POST module one can additionally enter some appropriate functions in order to convert such variables to other units through the ENTER_SYMBOL command.

8.3.3.6 Amount of compounds in the system or in a phase

The following mnemonic names are used for the amount of a compound (*e.g.* Fe) in the system or in a phase (with a phase name of *e.g.* AB):

MNEMONIC	MEANING	Examples (Possible Normalizations)			
N(comp)	Mole number of a component in the system	N(Fe)	NM(Fe)	NW(Fe)	NV(Fe)
N(ph, comp)	Mole number of a component in a phase	N(AB, Fe)	NM(AB, Fe)	NW(AB, Fe)	NV(AB, Fe)
B(comp)	Mass (gram) of a component in the system	B(Fe)	BM(Fe)	BW(Fe)	BV(Fe)
B(ph, comp)	Mass (gram) of a component in a phase	B(AB, Fe)	BM(AB, Fe)	BW(AB, Fe)	BV(AB, Fe)
X(comp)	Mole fraction of a component in the system	<i>Note that:</i> X(Fe) = NM(Fe)			
X(ph, comp)	Mole fraction of a component in a phase	<i>Note that:</i> X(AB, Fe) = NM(AB, Fe)			
W(comp)	Mass fraction of a component in the system	<i>Note that:</i> W(Fe) = BW(Fe)			
W(ph, comp)	Mass fraction of a component in a phase	<i>Note that:</i> W(AB, Fe) = BW(AB, Fe)			

Notes: Suffixes M, W, V and F can be applied to these quantities (as shown in the Examples column). The normalizations of such system quantities should be preformed with regard to the total system size (in terms of N, B or V), and of such phase quantities should be conducted with regard to the phase amount [in terms of NP(ph), BP(ph) or VP(ph)] of the phase.

Remark: Instead of the mnemonics NM and BW, X and W can be used to designate mole fraction and mass fraction, i.e.,

NM (comp) = X (comp)	BW (comp) = W (comp)
NM (ph, comp) = X (ph, comp)	BW (ph, comp) = W (ph, comp)

The units for the above variables and their suffixed quantities in the POLY module are always as:

- N (comp) and N (ph, comp) in moles; NM (comp) and NM (ph, comp) as *mole fraction*; NW (comp) and NW (ph, comp) in mol/g; NV (comp) and NV (ph, comp) in mol/m³
- B (comp) and B (ph, comp) in grams; BM (comp) and BM (ph, comp) in g/mol; BW (comp) and BW (ph, comp) as *mass fraction*; BV (comp) and BV (ph, comp) in g/m³
- X (comp) and X (ph, comp) as mole fraction; W (comp) and W (ph, comp) as mass fraction.

In both the POLY and POST module one can additionally enter some appropriate functions in order to convert such variables to other units through the ENTER_SYMBOL command.

Important Note 1: Other two derived quantities for system components, i.e., X% (comp) [=X (comp) *100] and W% (comp) [=W (comp) *100], can be used as axis variables for plotting in the POST module, but they must be instead specified as Mole-Percent <comp> and Weight-Percent <comp>, respectively. Please remember that neither of these two variables or relevant syntaxes can be used as conditions in the POLY module.

Important Note 2: When plotting mole-fraction or mass-fraction of a component in the system as an axis available in the POST module, it is recommended to use Mole-Fraction <comp> or Weight-Fraction <comp>, rather than X (comp) or B (comp).

Furthermore, as another extensive composition variable, the concept of so-called **u-fraction** concept represents an important quantity that is often shown up during special single-point or stepping paraequilibrium calculations within the POLY module of the Thermo-Calc software [or during local/para-equilibrium calculations within the DICTRA monitor of the DICTRA software]. The u-fraction, denoted as u_i for the i -th system component (it can be applied to the substitutional matrix component or substitutional alloying components, or to the interstitial component) in a whole system or in a specific phase under some various equilibrium states (such as full-equilibria,

partial-equilibria, para-equilibria, local-equilibria, etc.), is generally defined as: $\frac{x_i}{\sum_{j \in S} x_j}$, where the sum of x_j (in

the denominator) is taken for the substitutional matrix element and all substitutional alloying elements in the whole system or in the specific phase, and x_i (*in the numerator*) stands for the mole-fraction of the i -th component itself in the whole system or in the specific phase. Please remember that: the u-fraction is not really a standard state variable, and thus can not be directly used inside the POLY module (as no POLY syntax stands for it); however, in the POST module of the Thermo-Calc (and DICTRA) software, the u-fraction quantity is always assigned with the special syntax of u-f which can also be directly used as axis variables.

For instance, in the Fe-M-C ternary alloy system, it very frequently happens that one of the elements (the interstitial solute C) diffuses very much faster than the other two (the substitutional matrix element Fe and the substitutional alloying element M). It is often possible that a new phase forms with a different content of the mobile element (C) but without a change of the relative contents of the other two (Fe and M). Therefore, partial equilibria will remain in two phases, and a phase transformation will be partly partitionless through the locally equilibrated two-phase interface. At the interface, there is no driving force. The chemical potential of the mobile element (μ_C) has the same value on both sides, but the chemical potential for the non-mobile elements (μ_{Fe} and μ_M) have different values. Instead, the products of the chemical potentials and the u-fractions of the non-mobile elements [u_{Fe} and u_M , defined as $N_i / (N_{Fe} + N_M)$ or $x_i / (x_{Fe} + x_M)$] have the same values. Under the paraequilibrium state, $dT = dP = d\mu_C = du_{Fe} = du_M = 0$, the driving force should be zero, and T , P , μ_C and $u_{Fe}\mu_{Fe} + u_M\mu_M$ (the combined chemical potentials of Fe and M) must have the same values on both sides. In a molar Gibbs energy diagram (as illustrated in *Figure 8-1*), the tie-line between the two phases in paraequilibrium is directed towards the C corner, which falls on a common tangent line to the two Gibbs energy surfaces but not on the common tangent plane.

8.3.3.7 Amount of a phase in the system

The following mnemonic names are used for the amount of a phase (with a phase name of *e.g.* AB) in the system:

MNEMONIC	MEANING	Examples (Possible Normalizations)			
NP (phase)	Mole number of a phase	NP (AB)	NPM (AB)	NPW (AB)	NPV (AB)
BP (phase)	Mass (gram) of a phase	BP (AB)	BPM (AB)	BPW (AB)	BPV (AB)
VP (phase)	Volume (m^3) of a phase	VP (AB)	VPM (AB)	VPW (AB)	VPV (AB)

Note: Suffixes M, W and V can be applied to these quantities (as shown in the *Examples* column). The normalizations of such system quantities should be performed with regard to the total system size (in terms of N, B or V), and of such phase quantities should be conducted with regard to the phase amount [in terms of NP (ph), BP (ph) or VP (ph)] of the phase, except for the normalized NPM (ph), BPW (ph) and VPV (ph) quantities where the suffixes M, W, and V still mean the normalizations are calculated with regard to the total system size (in terms of N, B and V, respectively).

The units for the above variables and their suffixed quantities in the POLY module are always as:

- NP (phase) in moles; NPM (phase) as *mole fraction*;
 NPW (phase) in mol/g;
 NPV (phase) in mol/ m^3
- BP (phase) in grams; BPM (phase) in g/mol;
 BPW (phase) as *mass fraction*;
 BPV (phase) in g/ m^3
- VP (phase) in m^3 ; VPM (phase) in m^3/mol ;
 VPW (phase) in m^3/g ;
 VPV (phase) as *volume fraction*

In both the POLY and POST modules, one can additionally enter some appropriate functions in order to convert such variables to other units through the ENTER_SYMBOL command.

8.3.3.8 Special quantities of a phase

The constitution of a phase is expressed using site fractions (which is dimensionless):

MNEMONIC	MEANING	Notes
QF (phase)	Phase stability function of a phase	<ul style="list-style-type: none"> • Being negative when the phase composition is inside a spinodal and positive everywhere else, in an equilibrium state; • Dimensionless
TC (phase)	Curie temperature of a phase	<ul style="list-style-type: none"> • Being functions of phase composition; • Unit: K
BMAG (phase)	Bohr magneton number of a phase	<ul style="list-style-type: none"> • Being functions of phase composition; • Dimensionless

Note: These special quantities for a specific phase are dependent upon the thermodynamic model chosen for the phase. They cannot have any normalizing suffix (M/W/V/F); while the reference-state suffix R is legal to be added to such special variables but a R-suffixed quantity is always identical to its original variable [*e.g.*, QFR (ph) = QF (ph)] under a specific condition in a defined system, *i.e.*, the phase stability functions, Curie temperature and Bohr magneton number of various phases do not change along with any possibly re-defined reference states of system components, when the definitions of the system conditions remain unchanged. They cannot be used as conditions, as they can only be calculated under a certain type of equilibrium state.

8.3.3.9 Constitution of a phase

The constitution of a phase is expressed using site fractions (which is dimensionless):

MNEMONIC	MEANING	Examples
Y(ph, species#subl)	Site fraction of a species on specific sublattice site in a phase	Y(SIGMA, Fe#1), Y(SIGMA, Fe#3), Y(FCC, C#2)

Note: This quantity is dependent upon the thermodynamic model chosen for the specific phase.

8.3.3.10 Partial derivatives (derived variables)

For some material systems (such as heterogeneous interactions involving aqueous solutions, gaseous mixtures, liquids, etc.), it is necessary to utilize some additional derived variables which are defined as functions of state variables. The Thermo-Calc software (especially some advanced, easy-to-use modules) predefines some derived variables as assigned symbols (variables, functions and tables) for some phases. A user can also define any derived variable, as he prefers for his systems.

The state variables can be used to define additional functions or variables, through some mathematical expressions of state variables or other functions. Such functions or variables are referred as partial derivatives or derived variables.

Many derived variables *of a defined system, or of a certain system component, or of a given phase, or of a specific component or species in (related to) a given phase*, can be easily obtained using appropriate partial derivatives of state variables, such as heat capacity, thermal expansivity, isothermal compressibility, among others.

Derivatives of state variables can be evaluated using a dot “.” between two state variables , for example:

HM.T for heat capacity of the system.

Important Note: The variable after the dot sign “.” MUST be a condition that has already set in the equilibrium calculation of the defined system.

Actually, some variables listed in *Table 1* (of the document *Thermo-Calc Software System*) are derived variables, e.g., HM.T and HM(ph).T, where a dot “.” notation is used. As an important feature of the Thermo-Calc and DICTRA software, any partial derivatives of one state variable with respect to another can be defined/evaluated as a derived variable, by using the dot “.” notation between the two state variables.

One may use many more elaborate derivatives, e.g., T.W(LIQUID, C) means the slope of the phase boundary of the liquid with respect to carbon; after the liquidus temperature or solidus temperature (under a certain composition condition) has been calculated, the T.W(C) value implies the changing direction/scale of the liquidus temperature or solidus temperature along with each unit of the varying carbon composition condition W(C) in the defined system.

Some typical examples are listed below:

MNEMONIC	MEANING	Notes and Examples
HM.T	Heat capacity of the system	$\partial H / \partial T = C_p$ (if P is a condition) $\partial H / \partial T = C_v$ (if V is a condition)
HM(ph).T	Heat capacity of a phase	$\partial H(\text{ph}) / \partial T = C_p(\text{ph})$ (if P is a condition) $\partial H(\text{ph}) / \partial T = C_v(\text{ph})$ (if V is a condition)
H.T	Heat capacity of the system multiplied by total mole number of components	$\partial H / \partial T = C_p * N$ (if P is a condition in a close system) $\partial H / \partial T = C_v * N$ (if V is a condition in a close system)
H(ph).T	Heat capacity of a phase multiplied by the mole number of the phase, and plus a $\partial H(\text{ph}) * \partial N(\text{ph}) / \partial T$ term	$C_p(\text{ph}) * N(\text{ph}) + H(\text{ph}) * \partial N(\text{ph}) / \partial T$ (if P is a condition) $C_v(\text{ph}) * N(\text{ph}) + H(\text{ph}) * \partial N(\text{ph}) / \partial T$ (if V is a condition)
V.T	Thermal expansivity of the system (already multiplied by the total volume)	$\partial V / \partial T = \alpha * V$
VM.T	Thermal expansivity of the system (already multiplied by the total molar volume)	$\partial VM / \partial T = \alpha * VM$

$V(\text{ph}) \cdot T$	Thermal expansivity of a phase (<i>already multiplied by the phase volume</i>), and plus a $\text{VM}(\text{ph}) * \partial N_P(\text{ph}) / \partial T$ term.	$\partial V(\text{ph}) / \partial T = \alpha(\text{ph}) * V(\text{ph})$ $= \alpha(\text{ph}) * \text{VM}(\text{ph}) * N_P(\text{ph}) + \text{VM}(\text{ph}) * \partial N_P(\text{ph}) / \partial T$
$\text{VM}(\text{ph}) \cdot T$	Thermal expansivity of a phase (<i>already multiplied by its molar volume</i>)	$\partial \text{VM}(\text{ph}) / \partial T = \alpha(\text{ph}) * \text{VM}(\text{ph})$
$V \cdot P$	Isothermal compressibility of the system (<i>already multiplied by the total volume</i>)	$\partial V / \partial P = -\kappa * V$
$\text{VM} \cdot P$	Isothermal compressibility of the system (<i>already multiplied by the total molar volume</i>)	$\partial \text{VM} / \partial P = -\kappa * \text{VM}$
$V(\text{ph}) \cdot P$	Isothermal compressibility of a phase (<i>already multiplied by the phase volume</i>), and plus a $\text{VM}(\text{ph}) * \partial N_P(\text{ph}) / \partial P$ term.	$\partial V(\text{ph}) / \partial P = -\kappa(\text{ph}) * V(\text{ph})$ $= -\kappa(\text{ph}) * \text{VM}(\text{ph}) * N_P(\text{ph}) + \text{VM}(\text{ph}) * \partial N_P(\text{ph}) / \partial P$
$\text{VM}(\text{ph}) \cdot P$	Isothermal compressibility of a phase (<i>already multiplied by its molar volume</i>)	$\partial \text{VM}(\text{ph}) / \partial P = -\kappa(\text{ph}) * \text{VM}(\text{ph})$
$P \cdot T$	The slope $\partial P / \partial T$ of a phase boundary on a $P-T$ phase diagram. Note that the equilibrium with phase assemblage must have been calculated first.	
$T \cdot X(\text{comp})$	The slope $\partial T / \partial X(\text{comp})$ of a phase boundary on a $T-X(\text{comp})$ phase diagram with respect to mole fraction of the component in the system	
$T \cdot W(\text{comp})$	The slope $\partial T / \partial W(\text{comp})$ of a phase boundary on a $T-W(\text{comp})$ phase diagram with respect to mass of the component in the system	
$T \cdot X(\text{ph, comp})$	The slope $\partial T / \partial X(\text{ph, comp})$ of a phase boundary on a $T-X(\text{ph, comp})$ phase diagram with respect to mole fraction of the component in the phase	
$T \cdot W(\text{ph, comp})$	The slope $\partial T / \partial W(\text{ph, comp})$ of a phase boundary on a $T-W(\text{ph, comp})$ phase diagram with respect to mole fraction of the component in the phase	
Many more ...		

Restrictedly speaking, any of energetic extensive variables o of *the entire system*, of *a specific system component in the whole system*, of *a given phase*, or of *a specific system component (or species) in (or related to) a given phase*, with normalizing suffixes like M (per mole of component), W (per mass in grams), V (per volume in m^3) or F (per mole formula unit) as specified above and listed in [Table 1 of the document Thermo-Calc Software System](#), are derived variables, which have been explicitly defined inside the program by using expressions as either first or second derivatives of some state variables with respect to other independent state variables. In case a phase is not stable in the system, many standard variables and derived quantities for the phase will be still calculated by the program and saved in the workspace. However, when showing a partial derivative (after an equilibrium calculation), the state variable used after the dot sign “.” must have been defined as a condition prior to the calculation. For instance, the derived variable $G \cdot B$ is exactly the same as the W-suffixed state variable GW when B (total mass of all components in the system) has been defined as an equilibrium condition in the system.

Moreover, those compositional extensive variables with normalizing suffixes like M (per mole), W (per mass in gram), V (per volume in m^3), F (per mole formula unit), or % (percent) of *the entire system*, *any specific or all system components in the whole system*, or *a given phase*, or *any specific or all system components in a given phase*, as listed in [Table 1 of the document Thermo-Calc Software System](#), are also derived variables, which are defined by expressions of some independent state variables. Therefore, the derived variable $N(\text{comp}) \cdot N$ is exactly the same as the M-suffixed state variable $N_M(\text{comp})$ or $X(\text{comp})$ when N (total moles of all components in the system) has been defined as an equilibrium condition in the system. For instance,

$$\begin{aligned}
 N \cdot N &= NM = \partial N / \partial N && (\text{always as unity so it is unnecessary!}) \\
 N \cdot B &= NW = \partial N / \partial B && \text{moles per mass of the system (mol/g)} \\
 N \cdot V &= NV = \partial N / \partial V && \text{moles per volume of the system (mol/m}^3\text{)} \\
 B \cdot N &= BM = \partial B / \partial N && \text{mass per mole of the system (g/mol)} \\
 B \cdot B &= BW = \partial B / \partial B && (\text{always as unity so it is unnecessary!}) \\
 B \cdot V &= BV = \partial B / \partial V && \text{density of the entire system (g/m}^3\text{)}
 \end{aligned}$$

$N(\text{comp}) \cdot N = NM(\text{comp}) = X(\text{comp})$	mole fraction of a component in the system
$= \partial N(\text{comp}) / \partial N$	
$N(\text{ph, comp}) \cdot NP(\text{ph}) = NM(\text{ph, comp}) = X(\text{ph, comp})$	mole fraction of a component in a phase
$= \partial N(\text{ph, comp}) / \partial NP(\text{ph})$	
$X\%(\text{comp}) = X(\text{comp}) * 100$	mole percent of a component in the system
$B(\text{comp}) \cdot B = BW(\text{comp}) = W(\text{comp})$	mass fraction of a component in the system
$= \partial B(\text{comp}) / \partial B$	
$B(\text{ph, comp}) \cdot BP(\text{ph}) = BW(\text{ph, comp}) = W(\text{ph, comp})$	mass fraction of a component in a phase
$= \partial B(\text{ph, comp}) / \partial BP(\text{ph})$	
$W\%(\text{comp}) = W(\text{comp}) * 100$	mass percent of a component in the system
$NP(\text{ph}) \cdot N = NPM(\text{ph}) = \partial NP(\text{ph}) / \partial N$	moles of a phase per mole of the system (<i>mole fraction</i>)
$NP(\text{ph}) \cdot B = NPW(\text{ph}) = \partial NP(\text{ph}) / \partial B$	moles of a phase per mass of the system (mol/g)
$NP(\text{ph}) \cdot V = NPV(\text{ph}) = \partial NP(\text{ph}) / \partial V$	moles of a phase per volume of the system (mol/m ³)
$NPF(\text{ph}) = NP(\text{ph}) / NA$	moles of a phase per mole formula unit
$BP(\text{ph}) \cdot N = BPM(\text{ph}) = \partial BP(\text{ph}) / \partial N$	mass of a phase per mole of the system (g/mol)
$BP(\text{ph}) \cdot B = BPW(\text{ph}) = \partial BP(\text{ph}) / \partial B$	mass of a phase per mass of the system (<i>mass fraction</i>)
$BP(\text{ph}) \cdot V = BPV(\text{ph}) = \partial BP(\text{ph}) / \partial V$	mass of a phase per volume of the system (g/m ³)
$BPF(\text{ph}) = BP(\text{ph}) / NA$	mass of a phase per mole formula unit
$VP(\text{ph}) \cdot N = VPM(\text{ph}) = \partial VP(\text{ph}) / \partial N$	volume of a phase per mole of the system (m ³ /mol)
$VP(\text{ph}) \cdot B = VPW(\text{ph}) = \partial VP(\text{ph}) / \partial B$	volume of a phase per mass of the system (m ³ /g)
$VP(\text{ph}) \cdot V = VPV(\text{ph}) = \partial VP(\text{ph}) / \partial V$	volume of a phase per volume of the system (<i>volume fraction</i>)
$VPF(\text{ph}) = VP(\text{ph}) / NA$	volume of a phase per mole formula unit

All the state variables listed in *Table 1* can be used to define additional functions or variables for quantities that the user may be interested in, by utilizing the ENTER_SYMBOL command. Functions are saved in the Thermo-Calc workspaces, and at anytime a function value is requested all functions will be evaluated (for the reason that they may depend on each other). Variables are evaluated only when they are entered or if they are explicitly named in an EVALUATE command, and it is possible to enter a variable with a new expression anytime. Variables can be used as values of conditions in an SET_CONDITION command. Please also note that such derived variables, either predefined by the GES, PLOY or PARROT modules, or by an advanced module (e.g., POURBAIX and SCHEIL), or entered by the users, are saved as special symbols (variables, functions or tables). Each symbol has a unique name that must start with a letter and can have maximum 8 legal characters [including letters (either UPPER or lower case), digits and underscore “_”, but not any other special character such as parentheses “(” and “)”, plus “+”, minus “-”, slash “/” or “\”, period “.”].

There are many different ways to define additional derived variables and functions depending on users' purposes. For instance, the **activity coefficient of a component** in the system, and the **partition coefficient of a component** between two phases, can be defined as follows:

$$\begin{aligned} Rc<\text{name}> &= ACR(\text{component}) / X(\text{component}) \\ P_{\text{c}}<\text{name}> &= X(\text{phase1, component}) / X(\text{phase2, component}) \end{aligned}$$

It is important to note that the activity coefficient of a species in a solution phase depends upon the model definitions on the reference states and standard states for the species (see more descriptions in the *Section 2.13, Reference state and standard state*). When the standard state of a species is defined as the same as the “*pure species*” in a substitutional phase (such as gas and liquid, where all the dissolved species in the phase can substitute for each other and occupy equivalent sites, and the phase is possible to exist as pure with only the species), the *activity coefficient of the species* can then be calculated by:

$$RC<\text{name}> = ACR(\text{species, phase}) / Y(\text{phase, species})$$

If there are more than one sublattice in the phase and thus the reference states for the phase species can not be presented by the “*pure species*”, e.g., C in FCC [(Fe)₁(C, N, VA)₁] as the model may end at 50% C plus 50% Fe, the chemical potential and activity of all phase constituents will not be well defined. Therefore, one can not properly define the activity coefficients of the species in the phase.

For an aqueous solution phase, no matter what model (e.g., SIT, HKF, PITZ or another) is used, the reference state and standard state for both the solvent and solutes are properly defined in a special way in the Thermo-Calc software. The reference state for the solvent is set as “*pure water*”, the same as its standard state (according to the

Raoult's Law). The standard state for a solute species is set the “*hypothetical solute species*”, whilst its reference state is defined as the “*hypothetical state at unit molality (one mole dissolved into 1.0 kg of solvent) but in which the environment of each molecular is the same as at infinite dilution*” (according to the **Henry's Law**). Under these definitions, the *activity coefficients for both solvent and solute species* can be calculated, as follows:

$$\begin{aligned} \text{RC}_{\text{H}_2\text{O}} &= \text{ACR}(\text{H}_2\text{O, aqueous}) / \gamma(\text{aqueous, H}_2\text{O}) && (\text{for solvent}) \\ \text{RC}_{\text{spec}} &= \text{AI}(\text{species, aqueous}) / \text{ML}(\text{aqueous, species}) && (\text{for solute}) \\ &= \text{ACR}(\text{species, aqueous}) / \gamma(\text{aqueous, species}) * \gamma(\text{aqueous, H}_2\text{O}) \end{aligned}$$

where AI is the activity of solute species calculated from the model, and ML is the molality of the species.

The aqueous solution phase is treated in a comprehensive way within the Thermo-Calc software, due to its special requirements on presenting the calculated quantities that are of particular interest for aqueous chemistry, materials corrosion, chemical engineering, geochemistry, environmental engineering, etc. All the standard state variables used for other phases (see *Table 1 of the document Thermo-Calc Software System*) can be directly applied to the aqueous solution phase. Moreover, some additional derived variables that are defined as functions of state variables are necessary for the phase. The software (especially in the POURBAIX module) predefines some derived variables as assigned symbols (variables, functions and tables) for the aqueous solution phase.

Some examples of the Thermo-Calc predefined derived variables for the aqueous solution phase are listed and briefly described in *Table 2 of the document Thermo-Calc Software System*. Because the EOS (Equation of State) expressions, standard thermodynamic properties and transport properties for the pure solvent H_2O must be identical to those for the pure gaseous species H_2O , some derived variables for a gaseous mixture phase are also listed in the table. Under a certain temperature-pressure-composition condition, an aqueous solution phase may be in equilibrium with a stable gaseous mixture phase or a saturated vapour.

Another frequently-used concept regarding the constitution of a phase is **solvability**, that is applied where one or several of the constituents are dominant (dissolving, highly concentrated) while the rest species remain at minor levels of contents (dissolved, less concentrated or even dilute). Please note that the solubility concept is usually used when there exists a heterogeneous equilibrium between a dissolving (terminal) solution/mixture phase (e.g., liquid, gas, aqueous or solid; which has a dissolving capacity) and another stoichiometric/solution phase (e.g., metallic/non-metallic solids/liquids/gases, carbide/carbonate/nitride/nitrate/hydride/oxide/hydroxide/silicate/sulphide/sulphate/... solids; from which some constituents tend to be dissolved into the dissolving solution/mixture phase). One should also keep in mind that the application of the solubility concept is usually related to a saturation status between the dissolving (terminal) solution phase and dissolved phase (which is not completely-dissolvable for the terminal phase). For instance,

- A Fe-/Cr-dominant BCC phase may dissolve certain amounts of e.g. Ni and C from a carbide phase (such as M23C6, M7C3 and M3C) under a specific condition of temperature-pressure and defined Fe-Cr composition; so the Ni and C elements in the carbides will have their defined solubility in the BCC phase;
- An aqueous solution phase is always dominant by the solvent water H_2O , and any other element (such as Fe and C) or substance [such as a pure SO_2 gas, stoichiometric phase Cu_2S and solution phase $(\text{Fe}, \text{Ni})_1(\text{O}, \text{Va})_1$] may have a solubility limit under a specific condition of temperature-pressure and other aqueous compositions;
- An O_2 -dominant gaseous mixture phase may dissolve certain amounts of $\text{Fe}^{+2}/\text{Fe}^{+3}$ species from magnetite (Fe_3O_4) under a given condition of temperature-pressure and concentrations of other dissolved species in the mixture; therefore, the magnetite solid will have a defined solubility in the gaseous mixture under the given condition;
- A Fe-dominant liquid mixture phase may dissolve certain amounts of e.g. Cr and O under a specific condition of temperature-pressure and concentrations of other dissolved species in the liquid phase; this is why that the Cr and O components will have their defined solubility in the liquid mixture.

For more details on state variables and derived variables (partial derivatives), please refer to *Sections 2.5 and 2.6 in the document Thermo-Calc Software System*.

8.3.4 Components

A component is a system wide entity; sometimes calling it a *system component* emphasizes this fact. A component has a unique name with some characteristic thermodynamic properties, such as amount, activity or chemical potential. At the equilibrium state, the activity and the chemical potential of the components are constants in the entire system.

By default, the defined chemical elements are usually used as the system components in the POLY module. In some cases, *e.g.*, in oxide/sulfide/silicate and aqueous systems, such a default setting of elements as system components is rather inconvenient or inappropriate. However, the set of components may be changed with the POLY command `DEFINE_COMPONENT`. The new system components (corresponding to each of the defined elements in the system) which shall be defined in the POLY module must also be species that are available in the currently defined system. Examples of non-element component definitions are:

- ❖ `DEF_COMP CAO MGO FEO AL2O3 SiO2 O2`
for the CaO-MgO-FeO-Al₂O₃-SiO₂ oxide system;
- ❖ `DEF_COMP H2O H+1 ZE FE CR NI NA CL C N S`
for the Fe-Cr-Ni-C-N-S-H₂O-NaCl aqueous-heterogeneous interaction system.

Normally, when setting the equilibrium conditions, the standard state variables `AC`, `MU`, `N`, `X`, `W`, *etc.* can only be given for system components (see *Section 8.3.3*)! Thus, the command

```
SET_CONDITION N(H2)=100
```

is illegal unless a component has already been defined as `H2` rather than the default `H`. One may circumvent this inconvenience by using the command:

```
SET_INPUT_AMOUNT N(H2)=100
```

which only requires that `H2` is an existing species in the defined system, and the program subsequently translates this to the condition

```
SET_CONDITION N(H)=200
```

A system component can have the following extensive and intensive variables, which can be set as parts of equilibrium conditions or be obtained from equilibrium calculation results:

- ✚ `N(comp)` or `N(ph,comp)` Mole of a component in the system, or in a specific phase;
- ✚ `B(comp)` or `B(ph,comp)` Mass (grams) of a component in the system, or in a specific phase;
- ✚ `X(comp)` or `X(ph,comp)` Mole fraction of a component in the system, or in a specific phase;
- ✚ `W(comp)` or `W(ph,comp)` Mass (*weight*) fraction of a component in the system, or in a specific phase;
- ✚ `MU(comp)` Chemical potential of a component in the system;
- ✚ `AC(comp)` Activity of a component in the system;
- ✚ `LNAC(comp)` Natural logarithm of activity of a component in the system.

Please note the following general regulations:

- The extensive variables `N` and `B` quantities of *a component in the system* can be normalized using the normalizing suffixes like `M` (per mole), `W` (per mass in gram) or `V` (per volume in m³), with regard to the total system size [in terms of `N` or `B` or `V`] in the whole system;
- The extensive variables `N` and `B` quantities of *a component in a specific phase* can be normalized using the normalizing suffixes like `M` (per mole), `W` (per mass in gram) or `V` (per volume in m³), with regard to the phase amount [in terms of `NP(ph)`, `BP(ph)` or `VP(ph)`] of the phase;
- The intensive variables `MU`, `AC` and `LNAC` quantities of *a component in the system* can be suffixed by the so-called *reference state suffix R*, in order to get their values that are appropriately calculated with respect to a chosen reference state, *e.g.*, `MUR(comp)`, `ACR(comp)` and `LNACR(comp)`.

8.3.5 Conditions

In order to make an equilibrium calculation, one must have zero degree of freedom according to the Gibbs phase rule. The degree of freedom is equal to the number of components plus two (temperature and pressure).

The degree of freedom is reduced, one by each condition defined with the SET_CONDITION command, and one for each phase with the CHANGE_STATUS PHASE command.

The simplest conditions are to set temperature, pressure and the amounts of the components. But one may alternatively set conditions as the activity or chemical potential of a component (of the system or in a specific phase), or enthalpy, or any intensive or extensive state variable. Use the INFO STATE VARIABLE command-sequence, as well as in [Section 2.5](#) in the document *Thermo-Calc Software System*, to understand how a state variable is denoted.

Simple examples are, for the Fe-Cr-C system:

```
SET_CONDITION T=1200, P=1E5, X(CR)=0.18, W(C)=0.0013, N=1
```

meaning that temperature is 1200 K, pressure 1 bar, mole fraction of Cr 0.18 (*i.e.*, 18 mole percent), mass fraction C 0.0013 (*i.e.*, 0.13 weight percent), and the total amount of material equals to 1 mole (meaning the Fe is the rest). A set of conditions for the C-H-O system is

```
SET_CONDITION T=2173, P=101325, N(H2)=14, N(O2)=18, N(C)=5
```

meaning that temperature is 2173 K, pressure is 1 atm, moles of H₂ is 14, moles of O₂ is 18, and moles of C is 5. This assumes that the user has already set both H₂ and O₂ as components.

For user's convenience, inputs of non-component species are also possible to be used in conditions, but these have to be done with the SET_INPUT_AMOUNT command, which will translates them into acceptable conditions in terms of the defined components in the system. See information on the SET_INPUT_AMOUNT command.

It is possible to prescribe a nonsensical set of conditions, and thus the user must be careful. Normally, the program can detect a nonsensical set of condition only by failure to converge during the calculation.

An important feature of the POLY module is the possibility to set a condition on the composition of a phase. Most equilibrium program can only handle conditions on the overall composition; but with POLY, one may specify the composition or enthalpy or entropy of an individual phase. That phase should be stable, of course. An example is

```
SET_CONDITION W(LIQ,C)=0.012
```

meaning that the mass fraction of C in the liquid phase shall be 0.012 (*i.e.*, 1.2 weight percent).

It is possible to give a linear expression of state variables as a condition. For an example,

```
SET_CONDITION X(LIQ,S)-X(PYRR,S)=0
```

could be used to compute the congruent melting temperature of pyrrhotite in the Fe-S system.

The FIXED phase status is useful to directly calculate invariant or monovariant points or extrema. Note that the value set for the FIXED phase status is "NPF(phase)" rather than NP(phase). For an example, in order to calculate the melting temperature, one can replace the temperature condition by changing the status of the LIQUID phase status to be fixed as 0 moles in the system:

```
SET_CONDITION T=NONE  
CHANGE_STATUS PHASE LIQUID=FIX 0
```

The FIXED phase state can also be used to calculate diagrams where the system is in equilibrium with one or several given stable phases.

8.4 Different Types of Calculations

8.4.1 Calculating a single equilibrium

The user normally starts by calculating a single equilibrium. This is done by setting all necessary conditions in order to reduce the degree of freedom to zero. See information on the `SET_CONDITION` command.

When the necessary conditions have been properly set, the first equilibrium is calculated by giving a `COMPUTE_EQUILIBRIUM` command (For more details on this command, refer to *Sections 8.4.9 and 8.7.4*). Traditionally, this command enforces the so-called **Gibbs Energy Minimization Technique (GEM)**; however, the techniques involved in the calculation routines have been continuously improved/modified in order *not only* to make easier to find the equilibrium states *but also* to be confident in getting the most-stable equilibrium states in the entire defined systems:

- ☞ Some advanced algorithms had been used in some later versions (particularly TCCN/TCCP/TCCQ), through using the `COMPUTE_EQUILIBRIUM *` command-combination, in cases that the equilibrium states in some rather complicated heterogeneous interaction systems are not easy to calculate.
- ☞ Since the previous version TCCR and TCW4, as well as the previous version of various Thermo-Calc programming interfaces (TQ6, TCAPI4 and TC-MATLAB Toolbox4), the recently-implemented so-called **Global Minimization Technique** has been used to assure that the present minimum in an equilibrium calculation is the most stable minima under specified conditions. This new technique (as described in *Section 9.4.9*) will ultimately prevent a calculation from reaching an undesired metastable or unstable (local) equilibrium in a defined system, and automatically create additional composition sets in a solution phase if needed for handling single or multiple miscibility gaps (*i.e.* there will no longer be necessary for the user to specify these in advance). Accordingly, there has been some necessary changes in how to use this command, as described below (and in *Section 8.7.4*):
 - `COMPUTE_EQUILIBRIUM` Using the new Global Minimization Technique
 - `COMPUTE_EQUILIBRIUM -` Using the traditional GEM Technique
 - `COMPUTE_EQUILIBRIUM *` *Since the Global Minimization Technique is even more robust, when the Global Minimization mode is on, it is not that useful anymore! It can be appropriately used ONLY after the Global Minimization mode has already been disabled.*

Please note that the Global Minimization Technique is normally used as default in a single-point equilibrium calculation and (since TCCS/TCW5) in a stepping or mapping calculation, but can of course be turned off by the user (see details in *Sections 8.7.4 and 8.10.22*). In cases that it has been switched off, the traditional GEM Technique will always be applied.

The POLY program can provide automatic start values for the compositions of the phases, and this normally works well, but in some cases there are problems to calculate the first equilibrium. Some hints on how to handle these are given in *Section 8.12 (Trouble Shooting)*.

If the calculations do not converge, try to use as simple conditions as possible. The simplest types of conditions are fixed temperature and pressure, and a given overall composition. Even if that is not what you want, it may help to calculate an equilibrium with such conditions first. After that, you may try to play with phase status, enthalpy conditions, compositions of individual phases, *etc.* Note that it is easy to set conditions that represent an impossible equilibrium, and the program will not tell you that in any other way than failure to calculate this impossible equilibrium. Therefore, the user must know what is relevant to the calculating equilibrium for each of the conditions.

After calculating the first single equilibrium, the user may be satisfied, and may also continue to calculate other single equilibria by changing the conditions, before calculate a phase diagram or property diagram. Setting one of the defined conditions as the axis variable and use the `STEP` command (see information on the `STEP` command) can calculate a property diagram; setting two or more conditions as the axis variables can calculate a phase diagram with the `MAP` command (see information on the `MAP` command). For more details, see *Sections 8.4.2 to 8.4.5*.

8.4.2 Stepping calculations of property diagrams

In the POLY module, it is possible to compute a series of equilibria with one condition as the stepping axis variable. This is achieved by first calculating a single equilibrium, and then selecting one of the conditions as the independently-varied axis variable. The minimum and maximum value and the incremental step along the axis must also be given. Since TCCS/TCW5, the Global Minimization Technique is by default used for stepping calculations.

There are seven options with the STEP_WITH_OPTIONS command:

- NORMAL Only the step axis value is changed between each step.
- INITIAL_EQUILIBRIA An initial equilibria is added at each calculated point. This can be useful, for instance, to generate a set of isothermal calculations.
- EVALUATE When additional conditions (rather than the stepping variable) should be changed during a stepping calculation. With this option, one may simulate Scheil-Gulliver solidification processes by changing the overall composition to the new liquid composition after each step. See information on *Section 8.4.3, Solidification path simulations*.
- SEPARATE_PHASES The entered phases will be calculated separately at each step. In this way, one may calculate how the Gibbs energy for a number of phases varies for varying compositions.
- T-ZERO The so-called T_0 (*T-zero*) line, where the Gibbs energies of two phases are equal, can be calculated. See information on *Section 8.4.4, Paraequilibrium and T_0 temperature simulations*.
- PARAEQUILIBRIUM The so-called *paraequilibrium lines*, where the chemical potential for one or more interstitial components but not for the substitutional components in two partially equilibrated phases are equal, can be calculated. See information on *Section 8.4.4, Paraequilibrium and T_0 temperature simulations*.
- MIXED_SCHEIL Added since TCCQ. With this option, a stepping calculation of partial-equilibrium Scheil-Gulliver simulation on solidification with back diffusion of one or more interstitial species (such as C, N, O, S, etc.) in solid phases will be performed. However, users are always recommended to go directly to the extended SCHEIL_SIMULATION module and make the simulation in an automatic way. *This has been available since TCCQ.*

Note the T-ZERO and PARAEQUILIBRIUM options are only available since TCCP, and last one option only since TCCQ.

8.4.3 Solidification path simulations

Thermo-Calc is intended only for equilibrium calculations, and thus time-/space-dependent transformation processes cannot be simulated. However, some non-equilibrium or partial-equilibrium transformations can be simulated within the TCC/TCW software, for example, the Scheil-Gulliver solidification where the diffusion in the liquid phase is very fast and the diffusion in the solid phases are so slow that it can be ignored.

Under the Scheil approximation, the condition at the liquid/solid interface can be described with a local equilibrium. By stepping with small decrements of the temperature (or enthalpy or amount liquid phase), one can determine the new composition of the liquid and then remove the amount solid phase formed by resetting the overall composition to the new liquid composition before taking the next step. This is achieved with the special option EVALUATE, of the POLY command STEP, *i.e.*, the command-sequence STEP EVALUATE.

In order to conduct such a solidification simulation in a more user-friendly way, a special module, *i.e.*, the SCHEIL module, was implemented in the Thermo-Calc package. More detailed description of the SCHEIL module can be found in *Sections 10.8*, and two cases using the module are included in the in the **TCCS Examples Book** (*i.e.*, Examples 15 and 30).

The Scheil-Gulliver simulation often gives reasonable results for the normal solidification behaviour in Al and Ni-based alloys, but is by no means appropriate for steels that contain fast diffusing interstitial elements C or N. In

such a case, the Scheil-Gulliver scheme can be modified to allow equilibrium back diffusion of the interstitial element(s) in solid phases at each temperature step, i.e. to redistribute interstitial element(s) in liquid and solid by attaining the same chemical potential. More detailed description of this partial-equilibrium approximation can be found in Chen and Sundman, Mater. Trans., 43 (2002) p551.

Since TCCQ, the SCHEIL module has been extended and is able to handle the partial-equilibrium solidification approximation. A demonstration of this partial-equilibrium simulation using the extended SCHEIL module and the comparisons with the Scheil-Gulliver and lever-rule calculations can be found in the **TCCS Examples Book** (i.e., Example 48).

8.4.4 Paraequilibrium and T_0 temperature simulations

Beside the SCHEIL-module simulations of solidification processes, two types of partial equilibria, namely the paraequilibrium and T_0 (T-zero) temperature, can also be simulated by the Thermo-Calc software system.

A paraequilibrium means a partial equilibrium where one interstitial component (such as carbon C and nitrogen N) in a multicomponent alloy can diffuse much faster than the other components (the substitutional elements, including the matrix element and alloying elements), and consequently the chemical potential for the interstitial component but not for the other components in two partially equilibrated phases are equal. Under such a paraequilibrium state, it is possible to have a partly partitionless transformation where a new phase can form with different content of the mobile component but with the same composition of the slow diffusing components. A paraequilibrium calculation is useful e.g. when studying phase transformations in systems with large differences in the diffusivities of different elements. Transformations occurring under paraequilibrium states can be much more rapid than if full local equilibrium holds at the phase interface.

For instance, in a ternary alloy system (e.g., Fe-M-C), it very frequently happens that one of the elements (the interstitial solute C) diffuses very much faster than the other two (the substitutional matrix element Fe and the substitutional alloying element M). It is often possible that a new phase forms with a different content of the mobile element (C) but without a change of the relative contents of the other two (Fe and M). Therefore, partial equilibria will remain in two phases, and a phase transformation will be partly partitionless through the locally equilibrated two-phase interface. At the interface, there is no driving force. The chemical potential of the mobile element (μ_C) has the same value on both sides, but the chemical potential for the non-mobile elements (μ_{Fe} and μ_M) have different values. Instead, the products of the chemical potentials and the so-called u-fractions of the non-mobile elements [u_{Fe} and u_M , defined as $N_i / (N_{Fe} + N_M)$ or $x_i / (x_{Fe} + x_M)$] have the same values. The so-called u-fractions (a composition variable), u_i , for the i -th component (it can be the substitutional matrix component Fe or a substitutional alloying components or the interstitial component) in a whole system or in a specific phase in the

multicomponent system is generally defined as: $\frac{x_i}{\sum_{j \in S} x_j}$, where the sum of x_j is taken for the substitutional matrix

element and all substitutional alloying elements in the whole system or in the specific phase, and x_i stands for the mole-fraction of the i -th component itself in the whole system or in the specific phase. Under the paraequilibrium state, $dT = dP = d\mu_C = du_{Fe} = du_M = 0$, the driving force should be zero, and T , P , μ_C and $u_{Fe}\mu_{Fe} + u_M\mu_M$ (the combined chemical potentials of Fe and M) must have the same values on both sides. In a molar Gibbs energy diagram (as illustrated in *Figure 8-1*), the tie-line between the two phases in paraequilibrium is directed towards the C corner, which falls on a common tangent line to the two Gibbs energy surfaces but not on the common tangent plane.

A T_0 -temperature (T-zero) is defined as the temperature where two phases have the same Gibbs energy for a certain composition in a multicomponent system. The T_0 -temperature is located within the two-phase field between the phases and it is the theoretical limit for a diffusionless transformation. Calculations of T_0 -temperatures are thus of interest e.g. when studying diffusionless transformations.

The T_0 temperature in a multicomponent system with a fixed composition is a single point which locates at the common tangent line where the Gibbs energies of two phases in the partial equilibrium of a diffusionless transformation are equal, rather than chemical potentials of components. If the composition of one or two components varies, the common Gibbs energy for the two phases in partial equilibrium of a diffusionless transformation becomes a plane or surface, and T_0 becomes a line or plane, accordingly.

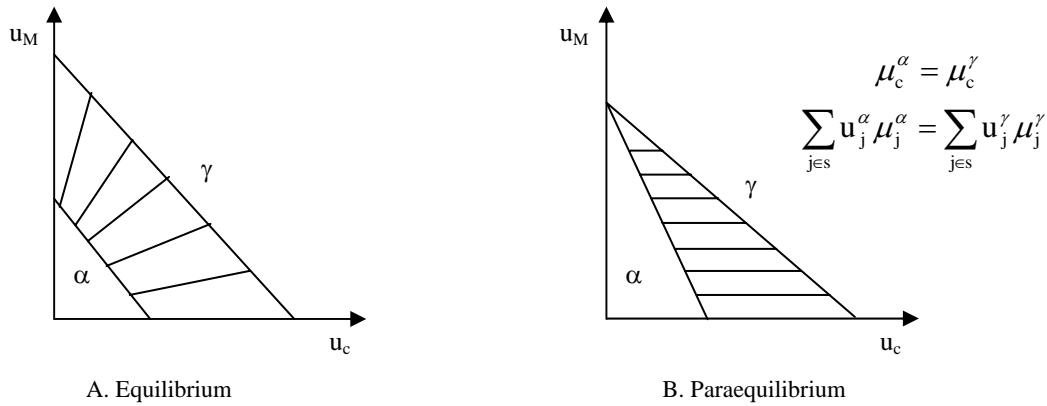


Fig. 8-1. Equilibrium and paraequilibrium states: Ternary Fe-M-C diagram where M is a substitutional alloying element and C is the fast diffusing carbon. Figure A shows the isothermal section and the two phase field $\alpha+\gamma$ for equilibrium conditions, while figure B is the same isothermal section and two-phase field for a paraequilibrium state.

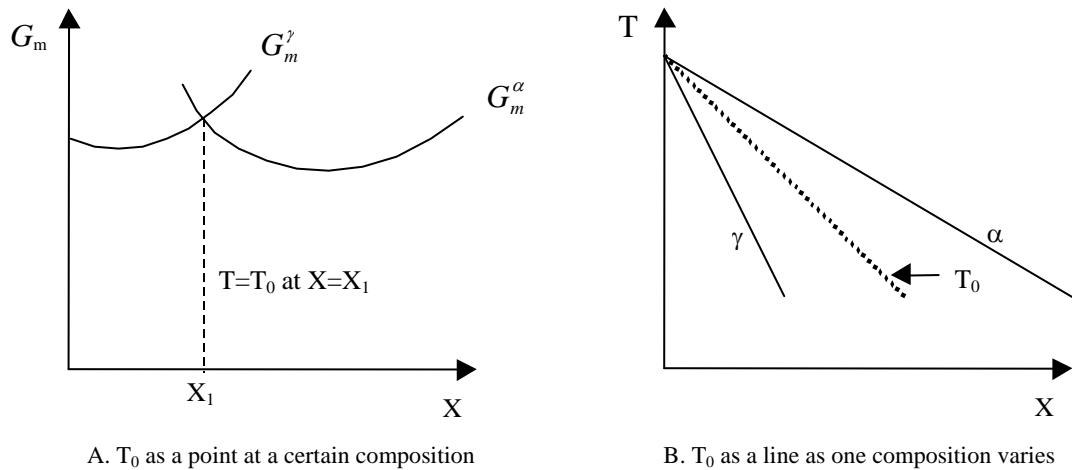


Fig. 8-2. The principle for calculating T_0 temperature in a multicomponent system. Figure A shows that the Gibbs energies at T_0 for phases α and γ are equal at a certain composition X_1 , while figure B indicates that T_0 changes with the system composition (note that the variation of T_0 is not necessary as a straight line).

The principle for calculating the T_0 -temperature is illustrated in Figure 8.2.

For more descriptions on these two concepts, please refer to Hillert (1998), pages 358-366.

Previously, these two types of partial equilibrium simulations can only be performed by using the PARROT module (see Example 23 in the **TCCS Examples Book**). Since TCCP, two advanced options have been made available in the POLY module for simulations for paraequilibrium and T_0 temperature in multicomponent systems:

- ☞ with the ADVANCED_OPTIONS command [i.e., utilizing the command-sequences ADVANCED T-ZERO and ADVANCED PARAEQUILIBRIUM, respectively] in single equilibrium calculations;
- ☞ with the STEP_WITH_OPTIONS command [i.e., utilizing the command-sequences STEP T-ZERO and STEP PARAEQUILIBRIUM, respectively] for calculations with one axis variable.

Figure 8-3 presents an example of the calculation results of paraequilibrium state and T_0 -temperature line for an Fe-1Cr-0.5Mn-0.5Si-0.2C (wt%) steel, that are imposed on the equilibrium phase diagram.

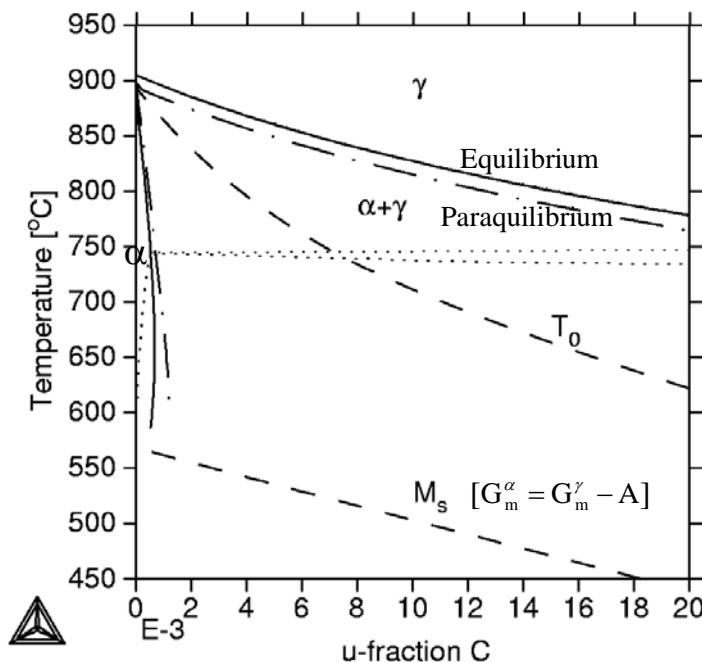


Fig. 8-3. Phase diagram for an Fe-1C-0.5Mn-0.5Si-0.2C (wt%) steel. The calculated paraequilibrium state and T_0 temperature line are also imposed on the diagram.

For more details on how they are performed, please refer to *Section 8.10.16* (the `ADVANCED_OPTIONS` command) and *8.9.4* (the `STEP_WITH_OPTIONS` command), as well as *Examples 42* and *43* in the **TCCS Examples Book**.

8.4.5 Mapping calculations of phase diagrams

Phase diagrams have two or more independent axis variables. Any already-defined condition (*Section 8.3.5*) can be used as the mapping variable for a mapping calculation and then as the axis variable for a phase diagram. In addition to the axis variables, there may be many more conditions if the system has more than two components. Since TCCS/TCW5, the Global Minimization Technique is by default used for mapping calculations. From a mapping calculation, many types of phase diagrams can also be plotted, with one of the mapped variables as one axis variable, and with other mapped variables or any varied property (state or derived variables) or entered symbol (variables, functions or table values) as the other axis variables. Therefore, very general types of phase diagrams can be generated.

Binary, ternary and multicomponent phase diagrams are calculated in much the same way, not matter how many additional conditions there are in the system (see *Sections 8.5.1*, *8.5.2* and *8.5.4*). One may also have fixed phases during a mapping procedure (see *Section 8.5.3*). There are two special modules, namely the `BIN` and `TERN` modules, developed for automatically calculating binary and ternary phase diagrams, respectively; for more detailed descriptions, please refer to *Sections 10.4* and *10.5*.

Note that all phase diagrams consist of lines where the amount of a phase is zero (*zero phase fraction lines*). But there are two distinct types of phase diagrams: those with the tie-lines in the plane of the diagram and those where the tie-lines are not in the plane. The formers are the traditional types of binary phase diagrams and ternary isotherms. The latter are more general isopleth diagrams where at least one extensive variable, normally a composition, is fixed.

Potential diagrams and Pourbaix diagrams are special types of phase diagrams, where the tie-lines are not in the plane. Two special modules, namely, the `POTENTIAL` and `POURABIX` module, were implemented in Thermo-Calc to automatically calculate such phase diagrams and other related phase diagrams. See details in *Section 8.4.6* and *Section 10.6* for potential diagram calculations, and *Section 8.4.7* and *Section 10.7* for Pourbaix diagram calculations.

8.4.6 Potential diagram calculations

The so-called potential diagram in a metal-oxide/sulfide-gas interaction system at certain temperature and pressure conditions can be calculated by using the chemical potentials or activities of two major species in the gaseous mixture phase as the mapping variables. Then, the potential diagram is plotted, usually by utilizing the activities (*i.e.*, fugacities; normally in common logarithm) of such two species as the X/Y-axes. The phase relations, between the gaseous mixture and various metal forms, metal-oxides, metal-sulfides, or other metal-bearing solids, are represented by different fields controlled by these two potential quantities. Many other properties are possible to be used as axis variables in various property diagrams for the interaction system at a specific temperature and pressure.

The special module, POTENTIAL, was developed for automatically calculating such potential diagrams and other related diagrams; for more detailed descriptions, please refer to *Sections 10.6*.

8.4.7 Pourbaix diagram calculations

The so-called Pourbaix diagram (*i.e.*, pH-Eh diagram) for an aqueous solution involving heterogeneous interaction system at certain temperature and pressure conditions is a special type of phase diagram, where the tie-lines are not in the plane. The chemical potentials or activities of two system components, H⁺ and EA, are used as the mapping variables, and the pH and Eh quantities are normally defined as the X/Y axes. The phase relations, between aqueous solution and various metal forms, oxides, hydroxides, sulfides, sulphates, nitrates, silicates, carbonates or other solids, or gaseous mixtures, are represented by different fields controlled by the acidity and electric potential. Many other properties are possible to be used as axis variables in various property diagrams for the interaction system at a specific temperature and pressure.

The special module, POURBAIX, was developed for automatically calculating the Pourbaix diagrams and other related phase diagrams; for more detailed descriptions, please refer to *Sections 10.7*.

8.4.8 Plotting of diagrams

If a STEP or MAP command has been conducted, the post-processor is then independently utilized in order to generate and refine a high-quality graphical output of the calculated results, which can be further imposed with experimental data for comparisons. The POST command transfers control to the post-processor (*i.e.*, the POST module). A detailed description of the commands in the post-processor is given in *Part 9*.

8.4.9 From Gibbs Energy Minimization to Global Minimization

The **Traditional GEM Technique (Gibbs Energy Minimization Technique)** employed in the Thermo-Calc software system as the so-called **Ordinary POLY Minimization** has been proved to be one of the best available for thermochemical calculations, due to the high efficiency and convenience in interactively accessing to the Gibbs energy expressions in the Gibbs Energy System (GES), to the calculated phase stability, assemblage and speciation in the Heterogeneous Equilibrium Calculation Module (POLY), to the derived thermodynamic functions (including reaction constants) in the Tabulation Module (TAB), and to the assessed thermodynamic variables in the Parameter Optimization Module (PARROT). It ensures, *mathematically and thermodynamically*, the internal consistency of thermodynamic data in a specific materials system (thanks to its direct use of a basic thermodynamic function, the Gibbs energy) and the highly efficiency of thermodynamic calculations for multicomponent heterogeneous systems/processes.

One of the major improvements since the previous version of Thermo-Calc software, TCCR and TCW4 [as well as the previous version of various Thermo-Calc programming interfaces (TQ6, TCAPI4 and TC-MATLAB Toolbox4)], is that the newly-implemented **Global Minimization Technique** is used to assure that the present minimum in an equilibrium calculation is the most stable minima for the specified conditions. In other words, this will prevent that you end up with an undesired metastable or unstable (local) equilibrium. Additional composition sets are automatically created in a solution phase if needed (for handling single or multiple miscibility gaps), *i.e.* there will no longer be necessary for the user to specify these in advance.

The technique used may be demonstrated with the example present in *Figure 8-4*. In this example, the Fe-Cr-Ni system is considered, and we seek a stable minima for an alloy with the bulk composition of $x_{\text{Fe}}=0.5$, $x_{\text{Cr}}=0.4$ and $x_{\text{Ni}}=0.1$ (shown as the red star), under the conditions of $T=700$ K and $P=1$ bar (100000 Pascal). When the COMPUTE-EQUILIBRIUM command in the POLY module (see [Section 8.7.4](#)) is executed in the new versions, a set of meshes is first computed; these meshes consist of discretized Gibbs energy functions for all the individual phases present in the system. Using these meshes and the defined equilibrium conditions, we find an approximate solution for the common tangent plane that has the lowest Gibbs energy. This plane is in this example defined by three different points which are indicated by yellow squares (which present BCC#1+BCC#2 miscibility gap and FCC phase). These points and also the amounts calculated for each of the phases are used as a start point in a subsequent POLY optimization in order to reach a truly global minimum (the most stable equilibrium state in the defined alloy).

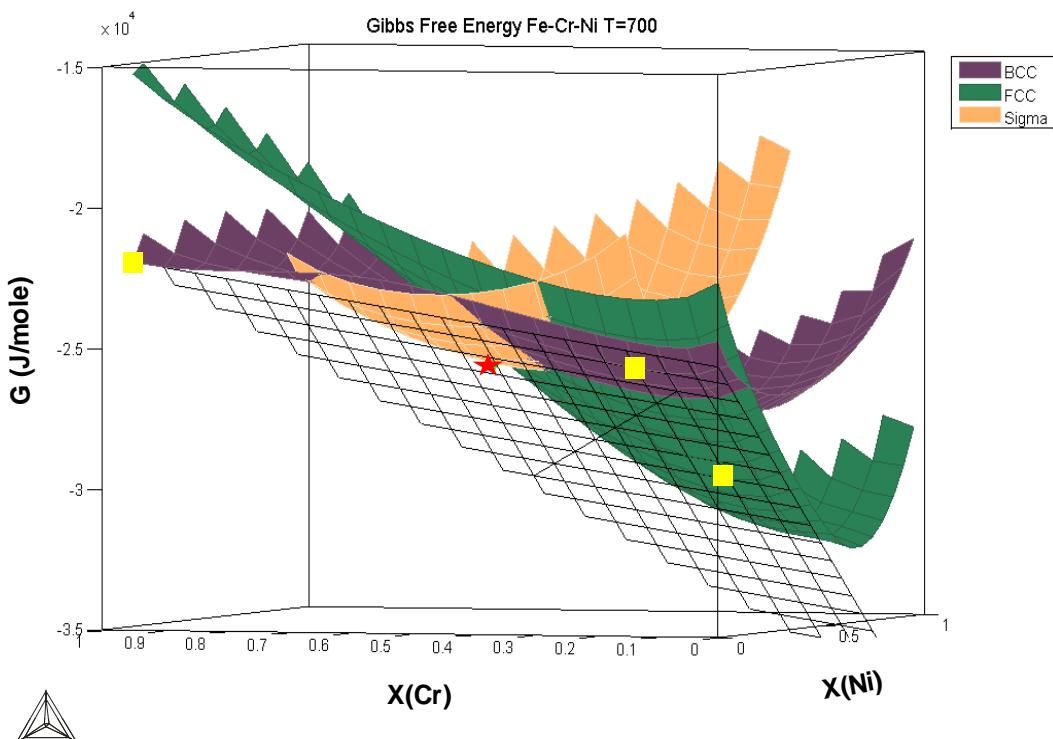


Figure 8-4. Global Minimum vs Local Minimum

The diagram illustrates the Gibbs energy surfaces for BCC (α), FCC (β) and SIGMA (σ) phases in the Fe-Cr-Ni system (with a bulk composition of $X(\text{Fe})=0.5$, $X(\text{Cr})=0.4$ and $X(\text{Ni})=0.1$, shown as the red star), under the conditions of 700 K and 1 bar.

The Global Minimization Technique finds the lowest Gibbs energy plane that is defined by three different points (that are indicated by yellow squares which present BCC#1-BCC#2 miscibility gap and FCC phase). This phase and the amounts calculated for each of phases are used as a start point in a further POLY optimization in order to finally find a truly global minimum (the most stable equilibrium state in the alloy). Without a global minimization, a calculation may end up a local minimum where the SIGMA (σ) phase is possibly in meta-stable equilibrium with BCC and FCC phases.

Since TCCR/TCW4, fully-supported equilibrium conditions in the Global Minimization Technique are:

T	temperature in the system (in K)
P	pressure in the system (in Pascal)
N	system size (mole number in moles)
$N(<\text{component}>)$	mole number of a component in the system
$X(<\text{component}>)$	mole fraction of a component in the system
B	system size (mass in grams)
$B(<\text{component}>)$	mass of a component in the system
$W(<\text{component}>)$	mass fraction of a component in the system

If other types of equilibrium conditions are used, a Global Minimization test and corrections are performed until the lowest minimum is found after the initial POLY optimization (using the Traditional GEM Technique). Some of such conditions are listed below as examples:

- Chemical potentials and activities, *and their logarithms*:

<code>MU (<component>)</code> or <code>MUR (<component>)</code>	chemical potential of a component in the system
<code>AC (<component>)</code> or <code>ACR (<component>)</code>	activity of a component in the system

- Compositions of a given component in a specific phase:

<code>N (<phase>, <component>)</code>	mole number of a component in a phase
<code>X (<phase>, <component>)</code>	mole fraction of a component in a phase
<code>B (<phase>, <component>)</code>	mass (grams) of a component in a phase
<code>W (<phase>, <component>)</code>	mass fraction of a component in a phase

- Chemical potentials or activities (*and their logarithms*) of a given species in a specific phase:

<code>MUR (<species>, <phase>)</code>	chemical potential of a species in a solution phase
<code>ACR (<species>, <phase>)</code>	activity of a species in a solution phase

- Energies of the system or of a particular phase, for instance:

<code>H</code>	enthalpy in the system (in J)
<code>HM (<phase>)</code>	enthalpy of a phase (in J/mol)
<code>G</code>	Gibbs free energy in the system (in J)
<code>GM (<phase>)</code>	Gibbs free energy of a phase (in J/mol)

- A condition has been defined as a value of a linear expression involving more than one state variable.
- When a particular phase has been set at the `FIXED` phase-status (either explicitly through the `CHANGE_STATUS PHASE` command or inexplicitly through the `COMPUTE_TRANSITION` command).
- When a partial derivates of state variable (derived variable) has been used as conditions.

Please note that the price to pay for assuring a Global Minimum in a calculation is an increase in the computational time.

The Global Minimization Technique is normally used as default in a single-point equilibrium calculation when performing the `COMPUTE_EQILIBRIUM` command, but can be turned *off* by the user using the POLY command-sequence of `ADVANCED_OPTIONS GLOBAL_MINIMIZATION` (which, of course, can also turn *on* the Global Minimization mode, if the user decisively repeats this command-sequence later on); see more details in [Sections 8.4.1, 8.7.4 and 8.10.22](#).

Since TCCS/TCW5, for stepping and/or mapping calculations, the Global Minimization Technique is performed not only during the initialization (for finding the starting points) but also during the actual stepping/mapping. The `STEP` and `MAP` routines have been completely rewritten and have been made available in TCCS/TCW5.

Please also note that the Global Minimization Technique is normally not used in the PARROT module for data assessments/evaluations, while the Ordinary POLY Minimization (GEM) is used for equilibrium calculations during optimizations; this is simply because that experimental data points used during a PARROT optimisation procedure are usually for specific phases (stoichiometric or solution phases), for well-defined phase boundaries, and for certain phase assemblages in a given low-order subsystem (which is often a unary, binary, ternary, quaternary or another subsystem within a multicomponent system) under specified experimental setups, and due to that probably some experimental data used in an assessment/evaluation are truly at some local/partial equilibrium states (that are not always necessarily at the global Gibbs energy minima state for a larger-scaled system). However, since TCCS it is possible to turn Global Minimization Technique on for (some) individual experimental equilibrium points during optimizations using the `POLY/ED_EXP` command-sequence `ADVANCED-OPTION TOGGLE-ALTERNATE`.

8.5 Diagrams

8.5.1 Binary phase diagrams

Binary phase diagrams are by far the most common found in the literature and calculated by computers. However, most real systems have more than two components and thus binary diagrams are more a kind of exercise in order to calculate more realistic phase diagrams.

Note that there is a special BIN module for calculating binary diagrams (see *Section 10.4*). However, one must use a special database (such as TCBIN, SBIN and PBIN), which is adaptable to the module. One example is given as the [Example 1 in the TCCS Examples Book](#).

One may calculate many different types of binary phase diagrams with POLY, but the traditional one has one composition axis and the temperature on the other axis. This is calculated by first specifying a single equilibrium by setting conditions like

```
SET_CONDITION T=1200, P=1E5, W(C) = .02, N=1
```

This means that the temperature should be 1200 K, pressure 1 bar, mass fraction of C is 0.02 (*i.e.*, 2 weight percent) and the system contains one mole of atoms. Note that one does not set a condition on the amount of the second component. This is simply taken to be the rest of the system.

After setting these conditions one can calculate the equilibrium by giving the command:

```
CALCULATE_EQUILIBRIUM
```

When the calculation has converged, one continues by setting the axes for the diagram calculation. In most cases, one is interested in the complete composition range but the temperature range may vary.

```
SET_AXIS_VARIABLE 1 W(C) 0 .1 0.002
SET_AXIS_VARIABLE 2 T 900 1900 25
```

This means that the axis number 1 is taken to be the mass fraction of C which may vary between 0 and 0.1 with a maximum increment of 0.002. The axis number 2 is the temperature varied between a minimum 900 and maximum 1900 K with a maximum increment of 25.

Before starting the mapping, it is a good rule to **SAVE** the workspaces. If this is done just before the **MAP** command, and the **MAP** command fails for some reason, the user can restart from the situation just before mapping and try to find a better start point.

```
SAVE
MAP
```

The diagram is plotted in the post-processor. The post-processor will automatically set the same axes for plotting as used for mapping.

```
POST
PLOT
```

[Example 4 in the TCCS Examples Book](#) demonstrates the traditional way of calculating a binary phase diagram.

For more alternatives in plotting the results, look for information in the POST module (*Chapter 9*).

8.5.2 Ternary phase diagrams

There are many types of ternary phase diagrams but perhaps the most common is the isothermal section. In this case, the temperature is constant, which means that the tie-lines are in the plane of the calculation. Isothermal sections and monovariant lines on the liquidus surface can be calculated in traditional way (See [Example 37 in the TCCS Examples Book](#)). They can also be calculated in the advanced TERN module (*Section 10.5*), but this normally requires a special database (such as PTERN) while any solution database with assessed ternary subsystems can also be used within this module; also see [Example 3 in TCCS Examples Book](#) for more details.

In order to set up calculation of a ternary isothermal section in the system Al-Mg-Si, do the following (after retrieving the data from the database module):

```
SET_CONDITION T=823, P=1E5, N=1, X(MG)=.01, X(SI)=.01
C_E
```

Note that there is no condition on the amount of Al as that will be the “rest”. N=1 specifies that the system is closed. This calculation gives one point inside the system. Now define the axis as below:

```
S-A-V 1 X(MG) 0 1 0.01
S-A-V 2 X(SI) 0 1 0.01
```

Of course, one may not be interested in the whole section, and in such cases one may specify different lower and upper limits for the axes. Then give a SAVE command before the MAP command:

```
SAVE almgisi-823
MAP
```

The result can be plotted in the POST module.

Ternary isopleths are equally simple to calculate. We may use the same conditions as above if we are interested in a section of Al-Si at 1 mole percent of Mg. We just define a different set of axis:

```
S-A-V 1 X(SI) 0 1 .01
S-A-V 2 T 500 2000 25
SAVE almgisi-1mg
MAP
```

In the resulting diagrams, the tie-lines will not be in the plane of the diagram, thus the lever rule *etc.* cannot be applied. But one may calculate more elaborate sections. An interesting diagram is the section from pure Al to Mg₂Si, which is almost a quasi-binary section. In order to calculate this, we must set up the conditions a bit differently, as shown below:

```
SET-CONDITION T=823, P=1e5, N=1, X(MG)-2*X(SI)=0, W(MG)=.01
C_E
```

Note that there is no condition on the amount of Si as this is given by the relation X(MG)-2*X(SI)=0. Also note that one may mix conditions in X and in W. Then set the axis, save the workspace and map.

```
S-A-V 1 W(MG) 0 .1 .01
S-A-V 2 T 500 1500 10
SAVE al-mg2si
MAP
```

A special type of ternary phase diagrams are liquidus surfaces. This is still not simple to calculate as a separate start point is needed for each separate line at each temperature.

However, one may easily calculate the monovariant lines where the liquid is stable together with two condensed phases in the following way:

```
SET-CONDITION T=800, P=1e5, N=1, X(MG)=0.01, X(SI)=.01
C_E
C-S P LIQ=FIX 0.3
S-C T=NONE
C_E
```

Now one should have an equilibrium where liquid is just stable. Set the axis to be the compositions.

```
S-A-V 1 X(MG) 0 1 .01
S-A-V 2 X(SI) 0 1 .01
SAVE almgisi-luni
MAP
```

Table 8-1 summarizes the general rules for setting equilibrium conditions and mapping variables for various types of phase diagrams in ternary systems. It also gives hints on how to define different vertical sections.

Table 8-1. Handling of various phase and property diagrams for ternary systems in the TCC software

Phase Diagram Type	Diagram Description	Triangular Diagram	Perpendicular Diagram	Mapping Calculation ^ and Graphical Plotting
Isothermal section	Phase boundaries over the entire or a partial composition space, at a specific T (and P)	X (A) - X (B) - X (C)	X (A) - X (B)	mapped in X (A) and X (B), under specific T and P plotted for phase boundaries at a specific T (and P)
Liquidus surface	Monovariant lines	X (A) - X (B) - X (C) [-T]	X (A) - X (B) [-T]	mapped in X (A) and X (B), under various T and a specific P, where the liquid phase has a fixed phase status (with a fixed amount 0–1.0 mole) plotted for the monovariant lines on the liquidus surface, with or without the varied T (in K or °C) plotted as tickmarks on the lines
	Isothermal surface projection	X (A) - X (B) - X (C) [-T]	X (A) - X (B) [-T]	mapped in X (A) and X (B), under various T and a specific P, where the liquid phase has a fixed phase status (with a fixed amount 0–1.0 mole). And there must be an individual start point for each separate T contour plotted for the monovariant lines on the liquidus surface, together with some projected liquidus T contours (isothermal invariant lines, in K or °C)
Isopleth	Stable phases in relation with T and one of composition variables (with another one is kept constant), at a specific P. It is a special case of vertical section (see below)		X (A) - T X (B) - T	mapped in X (A) or X (B) and T, under a constant X (C) and at a specific P plotted stable phase relations under various X (A) or X (B) and T, at a specific X (C) and P
	Stable phases in relation with T and a specific composition section [expressed by a correlation among two of the composition variables ^①] at a specific P		X (A) - T X (B) - T X (C) - T	mapped in X (A) or X (B) or X (C) and T, under a defined composition relation and at a specific P plotted stable phase relations along the defined composition section and T, under a specific P
<ul style="list-style-type: none"> ▲ In defining the equilibrium calculation conditions and mapping variables regarding compositions in the POLY module, one can not only mole fractions of components X(component), but also mole number N(component), mass B(component), mass fractions W(component), chemical potential MUR(component), activity ACR(component). Accordingly, many varied phase diagrams of these types can be generated. However, in this table, it is only referred to X(component). ① <u>Compositional relationship for vertical sections in a ternary system:</u> <p>General cases: The correlation of expressing the composition variables (\mathbf{X}_A, \mathbf{X}_B, \mathbf{X}_C) along a specific vertical section ranged from a known composition point ($X_{A'}$, $X_{B'}$, $X_{C'}$) to another ($X_{A''}$, $X_{B''}$, $X_{C''}$) in a ternary system (A-B-C) is as follows:</p> $[X_{B'} - X_{B''}] \mathbf{X}_A - [X_{A'} - X_{A''}] \mathbf{X}_B - [X_{B'} - X_{B''}] \mathbf{X}_{A''} + [X_{A'} - X_{A''}] \mathbf{X}_{B''} = 0$ $[X_{C'} - X_{C''}] \mathbf{X}_A - [X_{A'} - X_{A''}] \mathbf{X}_C - [X_{C'} - X_{C''}] \mathbf{X}_{A''} + [X_{A'} - X_{A''}] \mathbf{X}_{C''} = 0$ $[X_{C'} - X_{C''}] \mathbf{X}_B - [X_{B'} - X_{B''}] \mathbf{X}_C - [X_{C'} - X_{C''}] \mathbf{X}_{B''} + [X_{B'} - X_{B''}] \mathbf{X}_{C''} = 0$				

8.5.3 Quasi-binary and quasi-ternary phase diagrams

Quasi-binary diagrams, as well as quasi-ternary diagrams, *etc.*, have often been misunderstood and misused. A real quasi-binary phase diagram is for a ternary system where one component has a fixed activity or chemical potential; and a real quasi-ternary phase diagram is for a quaternary system where one component has a fixed activity or chemical potential.

A typical example of quasi-ternary system is a quaternary system Fe-Cr-Ni-C calculated at fixed carbon activity. The condition can be set as follows:

```
SET-REF-STATE C GRAPH ,,,,
SET-CONDITION T=1273, P=1e5, N=1, X(CR)=.1, X(NI)=.1, ACR(C)=.002
```

In some cases, the activity may vary in a quasi-binary section if one has a phase fixed instead (and where the phase may vary in composition). The advantage with the quasi-binary section is that the tie-lines will be in the plane of the diagram and thus the lever rule *etc* is obeyed.

A quasi-binary section in Ca-Fe-O when the liquid oxide is in equilibrium with liquid Fe can be specified as follows

```
DEFINE-COMPONENTS CAO FE FEO
SET-CONDITION T=1850, P=1e5, N=1, X(CAO)=.1
CHANGE-STATUS FE-LIQ=FIX 0
```

and the mapping can be done with the following axis

```
S-A-V 1 X(CAO) 0 1 .01
S-A-V 2 T 1500 2000 25
```

However, it is a problem as FE-LIQ is not the stable phase for pure iron below 1811. One will have to calculate separate sections depending on the stable modification of Fe.

Example 17 in the **TCCS Examples Book** gives some hints on calculating a quasi-binary phase diagram.

8.5.4 Higher order phase diagrams

Flexible calculations of multicomponent phase diagrams are a unique feature of the Thermo-Calc software. Although some software can calculate particular sections through multicomponent systems, Thermo-Calc is the only software that can calculate an arbitrary two-dimensional section through a composition space. However, there is not anything special about these diagrams, they are calculated as simply as binary diagrams but one more condition has to be added for each component. One may combine activity conditions, fixed phase status and fraction conditions in any way (as long as they describe a legal equilibrium). For example a steel with (in weight-percentage) 15% Cr, 5% Co, 2% Mo, 5% Ni, 1% V, 0.2% N rest Fe for 0-1.5% C and 700-1500°C can be calculated as follows:

```
SET-CONDITION T=1000, P=1e5, N=1, W(CR)=.15, W(CO)=.05, W(MO)=.02
SET-CONDITION W(NI)=.05, W(V)=.01, W(N)=.002, W(C)=.01
C-E
S-A-V 1 W(C) 0 .015,
S-A-V 2 T 800 1800 25
SAVE isop1
MAP
```

8.5.5 Property diagrams

These types of diagrams are calculated using only one independent axis. In the Thermo-Calc software, they are calculated normally by the STEP command (see the information in [Section 8.4.2 “Stepping calculation of property diagrams”](#) and [Section 8.4.3 “Solidification path simulations”](#)).

A common type of property diagrams is when all compositions are fixed and only temperature varies. In the POST module, one may then plot many different quantities as functions of temperature, for example, the amounts of the stable phases, the composition of a specific phase, or the activity of a component (in the entire system or in a specific phase).

Note that the predominance diagram is a special type of property diagram normally with temperature, pH, Eh or the composition of one major component as one of the axis variables, in which the curves separate the areas with dominant species (usually determined in terms of activity).

When a MAP calculation or a special module (BIN, TERN, POTENTIAL and POURBAIX) calculation has been made, many properties of the entire system, stable phases, components in the system or a specific phase, species in a specific phase can be used as axis variables to plot diagrams. Sometimes, such diagram might have been called as “property diagrams”. However, they must be regarded as special types of phase diagrams, because they represent variations of plotted properties along the phase boundaries, even only one of the mapped variables is used as axis variable in such diagrams.

8.6 General Commands

8.6.1 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

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

8.6.2 INFORMATION

Description: Basic information about various POLY subjects (concepts and models) can be obtained with this command for a number of subjects, as they are described in different parts of this chapter.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: <a subject and unique subject>

The name of a specific subject (or its abbreviation as long as it is unique, e.g., SIN, SIT, SOL, SPE, STATE, STEP, SYM, SYS, SUB, etc.) must be given. Extensive information is available for various subjects as listed below (this list can be seen if typing a question mark "?"):

PURPOSE	GETTING STARTED	USER INTERFACE
HELP	MACRO FACILITY	PRIVATE FILES
BASIC THERMODYNAMICS	SYSTEM AND PHASES	CONSTITUENTS AND SPECIES
SUBLATTICES	COMPONENTS	SITE AND MOLE FRACTIONS
COMPOSITION AND CONSTITUTION		CONCENTRATION
STATE VARIABLES	INTENSIVE VARIABLES	EXTENSIVE VARIABLES
DERIVED VARIABLES	UNITS	BASIC UNITS
SYSTEM UNITS	COMPONENT UNITS	PHASE UNITS
PHASE-COMPONENT UNITS	SPECIES UNITS	USER UNITS
SYMBOLS	REFERENCE STATES	METASTABLE EQUILIBRIUM
CONDITIONS	AXIS-VARIABLES	ADVANCED OPTIONS
CALCULATIONS TYPES	SINGLE EQUILIBRIUM	INITIAL EQUILIBRIUM
STEPPING	SOLIDIFICATION PATH	PARAEQUILIBRIUM AND TO
MAPPING	PLOTTING OF DIAGRAMS	TABULATION OF PROPERTIES
DIAGRAM TYPES	BINARY DIAGRAMS	TERNARY DIAGRAMS
QUASI-BINARY DIAGRAMS	HIGHER ORDER DIAGRAMS	PROPERTY DIAGRAMS
POTENTIAL DIAGRAMS	POURBAIX DIAGRAMS	AQUEOUS SOLUTIONS
ORDER-DISORDER	TROUBLE SHOOTING	FAQ

8.6.3 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 (also see *Section 5.4.11*).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing Prompt: MODULE NAME:
NO SUCH MODULE, USE ANY OF THESE:
SYSTEM UTILITIES
GIBBS ENERGY SYSTEM
TABULATION REACTION
POLY 3
BINARY DIAGRAM EASY
DATABASE RETRIEVAL
REACTOR SIMULATOR 3
PARROT
POTENTIAL DIAGRAM
SCHEIL SIMULATION
POURBAIX DIAGRAM
TERNARY DIAGRAM
MODULE NAME: <module name>

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

8.6.4 BACK

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

Synopsis: BACK

8.6.5 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.

Synopsis: SET_INTERACTIVE

8.6.6 EXIT

Description: This command terminates the program and returns to the operating system. Unless a SAVE command has been executed (in either the GES, POLY3 or PARROT module), all data and results will be lost.

Synopsis: EXIT

8.7 Basic Commands

8.7.1 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; *see Section 8.10.3*), it is equivalent to one condition (implying that particular phase is stable in the defined system).

Synopsis 1: 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. See details for conditions described in *Section 8.3.5 (on Conditions)* and more descriptions given below.

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.

Synopsis 2: SET_CONDITION

Ensuing 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 (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 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.

8.7.2 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.

Synopsis 1: SET_CONDITION <condition>=<new value>

Synopsis 2: SET_CONDITION <condition>=NONE

Example: SET_CONDITION T=1673.15

This will reset the condition T (temperature) as 1673.15 K.

```
SET_COND T=NONE
```

This will remove the condition on T (temperature).

More: It is now possible to remove all conditions by giving the command as below:

```
SET_CONDITION *=none
```

8.7.3 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. With such an information, the user can figure out if what could/should be further done:

- ☞ 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 FXIED status of a phase to another type of phase status, ENTERED or DORMANT or SUSPENDED).

Synopsis: LIST_CONDITIONS

Example Output: P=100000, T=800, N(NI)=1E-1, N=1
FIXED PHASES
FCC_A1=1 LIQUID=0
DEGREE OF FREEDOM 0

8.7.4 COMPUTE_EQUIlibrium

Description: The full equilibrium state is calculated for the given set of conditions. The number of iterations and the CPU time is listed after the command has finished.

Important Note: Since TCCR, 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 subsequent 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 (see details in [Section 8.10.22](#)).

For some details of the new Global Minimization Technique and of the traditional GEM Technique (*i.e.*, the ordinary POLY minimization routines used in versions up to the previous version TCCR/TCW4), refer to [Section 8.4.9](#); and for some strategies of single-point equilibrium calculations, refer to [Section 8.4.1](#).

Synopsis 1: COMPUTE_EQUIlibrium

Synopsis 2: COMPUTE_EQUIlibrium *

Synopsis 3: COMPUTE_EQUIlibrium -

Notes: Since TCCR, the “**C_E -**” command-combination can be used, in order to enforce the ordinary POLY minimization routines (*i.e.*, the traditional GEM Technique) in an equilibrium calculation; this is because of that the ordinary “**C_E**” command is now associated with the Global Minimization Technique (if it has not been turned off prior to the current calculation), 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.

However, since TCCR, 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 (for some details of such conditions, refer to [Section 8.4.9](#)) are used, after the initial POLY optimization (using the Traditional GEM Technique), a Global Minimization test and corrections are performed until the lowest minimum is found (we call it “**Indirect Global Minimization**”).

In previous versions before TCCR, 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 (that is more robust than the traditional GEM 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/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. Since TCCP, the “C_E **” command-combination has been further improved. BUT, since TCCR, this command-combination is not that useful any more, because the newly-implemented 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 (Section 8.10.7), 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.

Started from TCCP, if there is any convergence difficult in finding a stable solution at any stage of an ordinary equilibrium calculation performed by this 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
```

implying 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 PC Windows, as 2.00E-14). For other subsequent POLY-module calculations in the current TCC run, you can use the 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 (see Section 8.10.7 on the SET_NUMERICAL_LIMITS command).

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, before consulting Section 8.12 (*Trouble shooting*).

8.7.5 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.

Synopsis 1: LIST_EQUILIBRIUM <Return or file name> <option(s)>

Synopsis 2: LIST_EQUILIBRIUM

Ensuing 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

<i>Fraction type:</i>	W means MASS FRACTION X means MOLE FRACTION
<i>Composition:</i>	C means only COMPOSITION N means CONSTITUTION and COMPOSITION
<i>Phase:</i>	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).

Useful Hints: 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 ,,,
```

8.7.6 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 TDB module and this is still the recommended way. But the growing number of “occasional” TCC 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 this way. One may also use this command for USER databases.

Synopsis: DEFINE_MATERIAL

Ensuing 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.

Note: This 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 “*”.

Note: This 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.

.

Notes:

Since TCCM, it is possible to use this command to select an “alloy” from a database (*e.g.*, the TCNI Ni-based Superalloys Database). The alloys are predefined by the `OPTION` keyword in the database, and have their default major elements and composition limits of their alloy elements (for detailed information, see *Section 6.3.16*).

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 Thermo-Calc 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.

8.7.7 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 (see Section 8.7.5) 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.

Synopsis: DEFINE_DIAGRAM

Ensuing Prompt: 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 X-axis variable.

Minimum value /XXX/: <minimum value for X-axis>
Specify the minimum value of the chosen X-axis variable. A default value is shown automatically by the program; press <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.

Save file /RESULT/: <file name>
The file name where the calculations will be stored (saved as an *.POLY3 file); the default file name is RESULT.POLY3 (under Windows XP/2000/NT4) or RESULT.poly3 (under UNIX and Linux).

```

8.8 Commands to SAVE and READ the POLY Data Structure

8.8.1 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 TCC run for any reason.

The POLY3 and GES5 workspaces are saved on a file with this command. In the GES5 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 GES5 workspace. 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 POLY3 and GES5 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 SAVE. You may append several results obtained by sequential MAP or STEP calculations without destroying the previous results, whilst SAVE 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 TCC run.*

To suspend some of the MAP or STEP results, use the AMEND_STORED_EQUILIBRIA command.

Synopsis 1: `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 Windows XP/2000/NT4) or “*.poly3” (under UNIX and Linux), while the user can have any other extension as wished.

Synopsis 2: `SAVE_WORKSPACES`

Ensuing Prompt: File name /RESULT/: <file name>

By pressing <RETURN>, one can save the POLY3 and GES5 workspaces onto a file under the default name RESULT and with a default extension of “*.POLY3” (under Windows XP/2000/NT4) or “*.poly3” (under UNIX and Linux).

Or a user can specify a desired file name, for which the default extension is “*.POLY3” or “*.poly3”, while the user can also have any other extension as wished.

`Overwrite current file content /N/: <Y or N>`

(under Windows XP/2000/NT4 environments)

or

`Proceed with save /N/: <Y or N>`

(under PC Linux and various UNIX platforms)

This question is asked only if there is already a file with the same name, and the following message appears on screen:

This file contains results from a previous STEP or MAP command.
The SAVE command will save the current status of the program but destroy the results from the previous STEP or MAP commands.

If you answer Y, the previous content will be overwritten. Note that results from the STEP or MAP commands are destroyed by SAVE. You may append several results by MAP or STEP without destroying the previous results but SAVE will erase them all. To suspend some of the MAP or STEP results, use the AMEND command.

If you answer N, nothing will be saved, and the previous content will not be overwritten. You may later use the SAVE command with an unspecified name to save the POLY3 and GES5 workspaces.

Notes:

Under Windows XP/2000/NT4 environments, a **Save As** 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 8-4*. The file type (*i.e.*, POLY3, as in the **Save as type** box) can not be changed. By pressing the **Save** button, the program proceeds with saving the POLY3 and GES5 workspace in the specified *.POLY3 file. The user may also cancel such a **Save As** window session, and thus the current POLY3 and GES5 workspaces will not be saved.

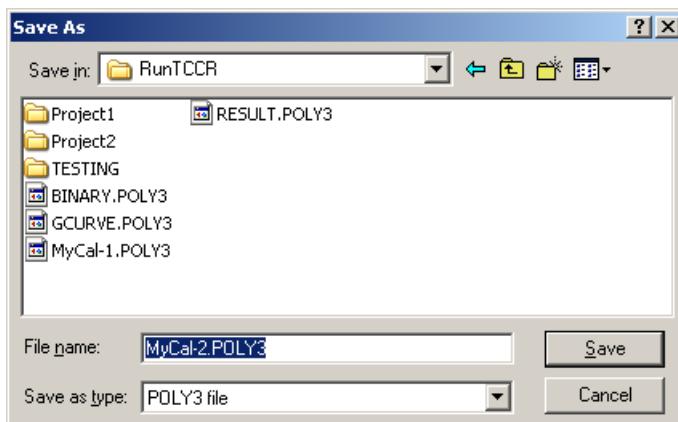


Figure 8-4. The “Save As” window: Saving the POLY3/GES5 workspaces as an *.POLY3 file.

However, if there is already a file with the same name under the directory (as indicated by the **Save in** box), a warming message will pop up on the screen, as shown in *Figure 8-5*. If clicking on the **No** button, the program returns to the **Save As** window, so that the user may choose another **Save in** path or a different **File name**. If clicking on the **Yes** button, the program will then ask the question “Overwrite current file content /N/ :”, so that the user can decide if overwriting the current POLY3/GES5 workspaces on the existing *.POLY3 file (see above).



Figure 8-5. The warming message: If saving the POLY3/GES5 workspaces on an existing *.POLY3 file.

Under UNIX and Linux platforms, by pressing <RETURN>, one can save the POLY3 and GES5 workspaces onto a file under the default name RESULT with a default extension of “*.poly3” (if the SAVE command is used for the first time), or under the previously-specified file name with a default extension of “*.poly3” (if the SAVE command has already been used at least once).

Useful Hints:

When saving a POLY3 workspace under a name that already exist under the current work area, which has saved by default (after running a special module, *e.g.*, BIN, TERN, POT, SCHEIL and POURABIX) or in an earlier stage of the current TCC 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 should be avoided if some results from previous MAP or STEP calculations shall not be destroyed.

8.8.2 READ_WORKSPACES

Description: The POLY3 and GES5 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.

Synopsis 1: READ_WORKSPACES <file name>

Options: file name -- The name of a previously saved POLY3-file where the POLY3 and GES5 workspaces shall be read from must be specified. The user does not need to type the extension if it is the default “*.POLY3” (under Windows XP/2000/NT4) or “*.poly3” (under UNIX and Linux), otherwise the user must type the whole POLY-file name.

Synopsis 2: READ_WORKSPACES

Ensuing Prompt: File name /ABCDEF/: <file name>

The program shows up the “ABCDEF” as the most lately specified POLY3-file name or as “RESULT” if there is no workspace already saved in the run. By pressing <RETURN> or typing a specific file name, one can read the POLY3 and GES5 workspaces from the file.

Notes: Under Windows XP/2000/NT4 environments, a **Open file** 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, as shown in *Figure 8-6*. The file type (*i.e.*, POLY3, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with opening the POLY3 and GES5 workspaces from the specified, previously-saved *.POLY3 file. The user may also cancel such an **Open file** window session, and thus no previously-saved POLY3 and GES5 workspaces will be opened.

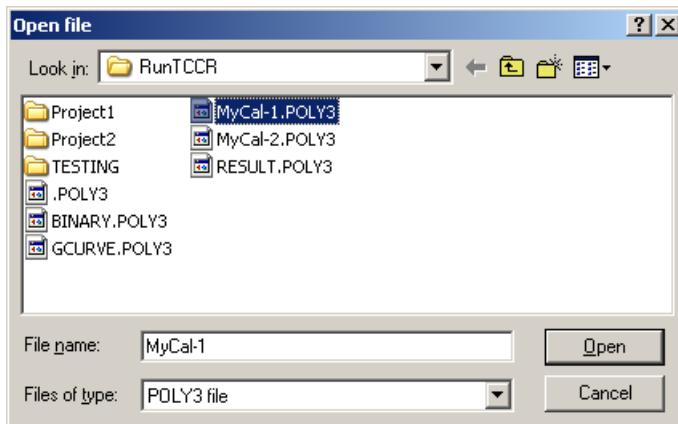


Figure 8-6. The “Open file” window: Reading the POLY3/GES5 workspaces from an *.POLY3 file.

Useful Hints: When reading back an original POLY3 workspace that has already saved as an *.POLY3 file in the current TCC run or previously 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 TCC run) or any diagram has been plotted in the POST module, one can simply type “READ,,,” or the following line:

READ_WORKSPACE , ,

8.9 Commands to Calculate and Plot Diagrams

8.9.1 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 extremely unique and very powerful feature of the TCC software, up to 5 independent axis variables can possibly be used in a mapping calculation of a multicomponent 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.

Synopsis 1: SET_AXIS_VARIABLE <axis number> <condition> <min> <max> <length>

Synopsis 2: SET_AXIS_VARIABLE

Ensuing 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.

8.9.2 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.

Synopsis: LIST_AXIS_VARIABLE
Then all the axis variables that have been set in the current POLY3 workspace will show up on screen, like:

Axis No 1: W(C)	Min: 0.001	Max: 0.010	Inc: 0.001
Axis No 2: X(Cr)	Min: 0.001	Max: 0.100	Inc: 0.010
Axis No 3: T	Min: 1073.15	Max: 2073.15	Inc: 25

Notes: When there is only one axis variable defined in the current POLY3 workspace, the user can only conduct a stepping calculation (STEP_WITH_OPTIONS). However, if there are two or more (up to five) axis variables defined appropriately, the user should perform a mapping calculation (MAP).

8.9.3 MAP

Description: This command maps a phase diagram from one or more initial equilibria. Note that a phase diagram consists of lines where the amount of a phase is zero. All different types of phase diagrams are generated by this command (see information in *Section 8.4.5*).

Important Note: Since TCCS, 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 (see details in *Section 8.10.22*).

Synopsis: MAP

Notes: During a MAP calculation, the values of the mapping axis variables for each calculated equilibrium will be listed and also the set of stable phases. Since TCCN, the lengthy output of calculated equilibria has been removed. Please note that the special option “OUTPUT_AT_MAP_AND_STEP” enforced the previously-available SPECIAL_OPTION command for setting back a long listing has been removed since TCCS (see *Section 8.10.22 – ADVANCED_OPTIONS*).

The user may terminate the mapping of a line by pressing a single CTRL-A (on PC Windows) or CTRL-C (on UNIX or PC Linux). This can be useful in order to stop a longish calculation without loosing what is already calculated.

Since TCCP, 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
```

implying 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 PC 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 (see *Section 8.10.7* on the SET_NUMERICAL_LIMITS command).

The procedure during the MAP command to calculate phase diagrams has been continuously improved in all new releases! And it will be improved in the forthcoming releases!!.

In particular, diagrams with tie-lines in the plane, *i.e.*, most binary systems and ternary isotherms, have now 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 gives smoother curves and also better stability in finding adjacent regions.

Miscibility gaps will be detected during mapping if a phase has two or more composition sets. It may be necessary to use other values than the default axis increment in some diagrams.

To get a complete phase diagram, sometimes it may be necessary to have multiple start points.

8.9.4 STEP_WITH_OPTIONS

Description: This command initiates the stepping procedure. Before it can be evoked, one must have calculated an equilibrium and given a SET_AXIS_VARIABLE command. The program will list the current values of the axis variable for each calculated equilibrium and also when the set of stable phases changes.

Important Note: Since TCCS, 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 (see details in [Section 8.10.22](#)).

There are seven options with the STEP command. These are:

- **NORMAL** which means that just the axis variable will change.
- **INITIAL_EQUILIBRIA** which means that an initial equilibrium will be stored at each calculated equilibria. (Not implemented yet).
- **EVALUATE** which means that the variables given will be evaluated after each step, which may change other conditions.
- **SEPARATE_PHASES** which means that each entered phase will be calculated separately at each step.
- **T-ZERO** which means that T_0 (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). *This option has been available since TCCP.*
- **PARAEQUILIBRIUM** which 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). *This has been available since TCCP and has been greatly improved since TCCQ.*
- **MIXED_SCHEIL** which means that a stepping calculation of partial-equilibrium Scheil-Gulliver simulation on solidification with back diffusion of one or more interstitial species (such as C, N, O, S, etc.) in solid phases will be performed. However, users are always recommended to go directly to the extended SCHEIL_SIMULATION module and make the simulation in an automatic way. *This option has been available since TCCQ.*

Note that one may terminate the stepping in a controlled way by pressing a single CTRL-A (on Windows XP/2000/NT4) or CTRL-C (on UNIX or Linux).

Synopsis 1: STEP_WITH_OPTIONS <a chosen option, N or I or E or S or T or P>

Synopsis 2: STEP_WITH_OPTIONS

Ensuing Prompt: Option? /NORMAL/: <a chosen option, N or I or E or S or T or P>

One has to choose one of the following four available options:

NORMAL	<i>Stepping with given conditions</i>
--------	---------------------------------------

INITIAL_EQUILIBRIA	<i>An initial equilibrium stored at every step</i>
--------------------	--

This is not implemented yet. It has been intended to generate a “matrix” of calculated equilibria. One may thus use one axis variable, *e.g.*, temperature, and give a STEP_INITIAL command. This will calculate a number of equilibria and automatically ADD each as an initial equilibrium for another STEP_NORMAL or MAP command. Before the second STEP command, one may chose composition as axis variable, and the STEP command will then use the start points at different temperatures and step in composition and this will give a “matrix” of values. Actually one may carry this even deeper.

EVALUATE	<i>Specified variables evaluated after each step</i>
----------	--

This is an advanced command for experts of Thermo-Calc. It allows stepping 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. One example of using this command is described in the INFORMATION command with the subject as SOLIDIFICATION (Solidification Path Calculations).

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 calculation in some earlier versions, it used to calculate all kinds of property diagram, phase fraction plots, *etc.*, all kinds of problems can 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 command 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 the STEP command the MC carbide will first become metastable with a composition different from the metallic FCC phase and later also stable. Previously such calculations had to be made from low temperature.

SEPARATE_PHASES	<i>Each phase calculated separately</i>
-----------------	---

This is used when the user wants to plot Gm curves versus composition for a given temperature.

T-ZERO	<i>T0 lines between two specific phases calculated</i>
--------	--

This option, which has been available since TCCP, allows calculating the T_0 (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 can not 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_WITH_OPTION 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 **TCCS Examples Book**.

PARAEQUILIBRIUM *Paraequilibrium state between two specific phases calculated*

Paraequilibrium state between two specific phases is calculated with this option which has been available since TCCP. It allows a stepping calculation of paraequilibrium states in a partly partitionless transformation between two specific phases in an alloy system with interstitial component(s) as fast diffusion species. Note that within TCCP only carbon C was possible to be considered as the interstitial component which was always set as default; however, this option had been greatly modified and improved since TCCQ, and it now allows more than one interstitial components treated as fast diffusion species (such as C, N, O, S, etc., as *individual or combined*) in a paraequilibrium stepping calculation.

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.

Since TCCQ this option works not only for alloy systems with carbon C but also with other interstitial components (such as N, O, S, etc.), furthermore it allows combined cases such as C and N both treated as interstitial species. Please also note that the composition condition for the interstitial components can not 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:

- 1) 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 can not find the paraequilibrium state (as it is impossible to correctly calculate u-fractions).
- 2) 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 starting bulk composition for the system.
- 3) Always set the status of the chosen interstitial components as SPECIAL 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.

- 4) Always make a single-point paraequilibrium calculation by the command-sequence of ADVANCED_OPTIONS PARAEQUILIBRIUM, prior to performing this STEP_WITH_OPTION 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 in terms of calculating a paraequilibrium state between the two specified phases. Specifically, 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 component(s) (C as the default single component) must be given at the above prompts subsequently or at the same (first) prompt. Since TCCQ, it is possible to specify one or more interstitial component(s) as fast diffusion species.

Please pay attentions on the followings 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 <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 including:

- ❖ 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., $u-f(\text{phaseB}, \text{component(s)})]$];
- ❖ the LNACR (component) value(s) for the interstitial(s).

The output format during the stepping calculation is:

```
axis value, phase amounts of phases A & B,
u-fractions of interstitial(s) in phases A & B,
LNACR of interstitial(s)
```

Example 1: Stepping in W(Ni), with C as interstitial in BCC/FCC:

```
Phase Region from 1.000000E-02 for:
BCC_A2
FCC_A1#1
1.000000E-02 0.155 0.845 2.634103E-02 6.644555E-04 -1.799147E+00
7.500000E-03 0.144 0.856 2.810506E-02 7.187681E-04 -1.729148E+00
.
.
.
2.500000E-03 0.125 0.875 3.153940E-02 8.286421E-04 -1.604024E+00
2.500000E-09 0.117 0.883 3.321170E-02 8.841991E-04 -1.547684E+00

Phase Region from 1.000000E-02 for:
BCC_A2
FCC_A1#1
1.000000E-02 0.155 0.845 2.633047E-02 6.643965E-04 -1.800276E+00
1.250000E-02 0.169 0.831 2.453376E-02 6.104901E-04 -1.876313E+00
.
.
.
```

Example 2: Stepping in T, with C & N as interstitials in BCC/FCC#2:

```
Phase Region from 1.000000E+03 for:
BCC_A2
FCC_A1#2
1.000000E+03 0.809 0.191 5.499567E-04 2.351291E-04 2.204353E-02
4.245957E-03 -1.986402E+00 -1.170203E+01
9.900000E+02 0.834 0.166 5.975632E-04 2.519434E-04 2.500321E-02
4.758404E-03 -1.762858E+00 -1.158479E+01
.
.
.
8.200000E+02 0.959 0.041 1.036210E-03 3.509956E-04 8.922062E-02
1.619333E-02 1.646691E+00 -1.029484E+01
8.100000E+02 0.961 0.039 1.033397E-03 3.457079E-04 9.330950E-02
1.704517E-02 1.846164E+00 -1.024367E+01
8.000000E+02 0.962 0.038 1.027487E-03 3.393415E-04 9.739766E-02
1.792068E-02 2.047286E+00 -1.019454E+01

Phase Region from 1.000000E+03 for:
BCC_A2
FCC_A1#2
1.000000E+03 0.809 0.191 5.499567E-04 2.351291E-04 2.204353E-02
4.245957E-03 -1.986402E+00 -1.170203E+01
1.010000E+03 0.779 0.221 5.019002E-04 2.175302E-04 1.925546E-02
3.756088E-03 -2.215848E+00 -1.182725E+01
.
.
```

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 an 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 **TCCS 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.

MIXED_SCHEIL*Scheil-Gulliver type solidification simulated*

This option, which has been available since TCCQ, leads a stepping calculation of partial-equilibrium Scheil-Gulliver simulation on solidification with back diffusion of one or more interstitial species (such as C, N, O, S, etc.) in solid phases will be performed.

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 axis variable (by the **SET_AXIS_VARIABLE** command, with a minimum and maximum temperature points as well as an appropriate temperature step for the cooling process);
- It is equally important to set the composition condition(s) of the fast diffusion interstitial components in terms of mole fraction or weight fraction, e.g., **X(C)** and **W(N)**. Otherwise, the stepping calculation with this option will fail.
- You must have made a starting-point equilibrium calculation in the liquid stability region (*i.e.*, at a relatively higher temperature condition);

In cases where fast diffusion interstitial components have been specified as fast diffusion interstitial components, there is a possibility to choose an option to allow **BCC**→**FCC** phase transformation during the solidification process.

However, users are always recommended to go directly to the extended SCHEIL_SIMULATION module and make the Scheil-Gulliver simulation of alloy solidification processes in an automatic way.

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 fast diffusion species; always on the same line at once (separated by an empty space, e.g., "C N").

Please note that 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.; otherwise, such a Scheil-Gulliver simulation does not make sense at all.

Allow BCC → FCC ? /Y/: <Y or N>

If there is/are more fast diffusion component(s) for the current paraequilibrium calculation, then type the appropriate interstitial species; otherwise, just press <RETURN> key to start the paraequilibrium point calculation.

During the STEP MIXED_SCHEIL calculation procedure, the solidus temperature of the alloy is first calculated and shown on screen, and then the calculated solidification path, including the amounts (in the unit of mass fraction) of remaining liquid phase and overall formed solid phases, as well as the heat of latent along the solidification process, are shown along the corresponding scanned conditions (of the stepping temperature variable), 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 plot various property diagrams for the alloy solidification process according to the Scheil-Gulliver model, or to impose existing relevant experimental information or other types of calculated solidification results [e.g., Scheil-Gulliver simulation with or without considering fast diffusion interstitial component(s), equilibrium approach, or DICTRA-type simulation with moving-boundary conditions] onto the plotted property diagrams.

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. Since TCCN, the lengthy output of calculated equilibria has been removed. Please note that the special option “OUTPUT_AT_MAP_AND_STEP” enforced the previously-available SPECIAL_OPTION command for setting back a long listing has been removed since TCCS (see *Section 8.10.22 – ADVANCED_OPTIONS*).

The user may terminate the mapping of a line by pressing a single CTRL-A (on PC Windows) or CTRL-C (on UNIX or PC Linux). This can be useful in order to stop a longish calculation without loosing what is already calculated.

Since TCCP, 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

implying 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 PC Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current TC run, you can use the POLY command SET_NOMERICAL_LIMITS to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits (see *Section 8.10.7* on the SET_NOMERICAL_LIMITS command).

The procedure of STEP calculations (of property diagrams) has been continuously improved in all new releases! And it will be improved in the forthcoming releases!!.

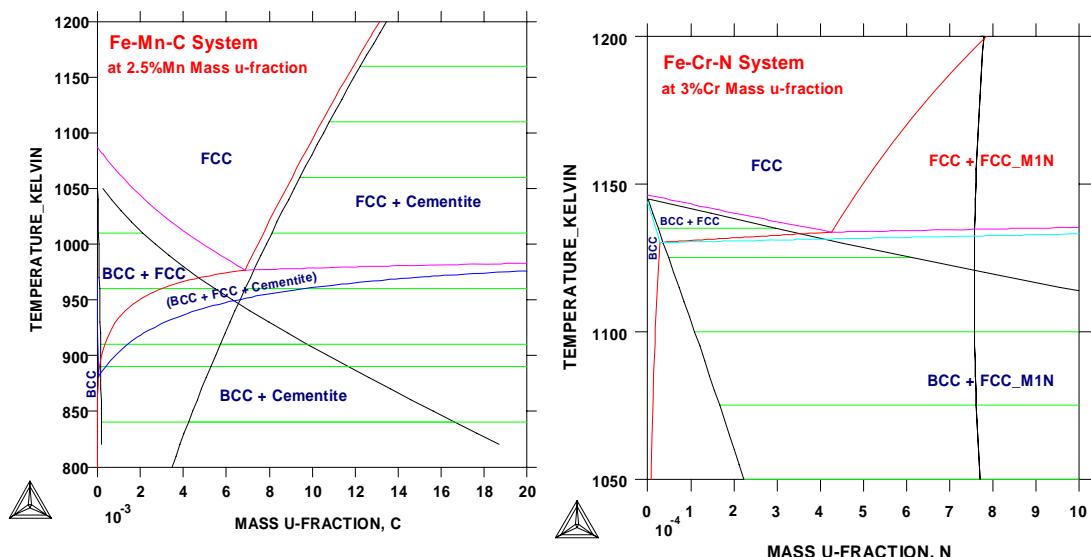


Figure 8-7. Paraequilibrium calculation of isopleths: Formation of para-pearlite in two Fe-based alloys, with equilibrium results appended. Note that the calculated paraequilibrium phase boundaries (black lines) are shown with tie-lines (in green-colour). In the Fe-2.5Mn-C alloy system, C is considered 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.

8.9.5 ADD_INITIAL_EQUILIBRIUM

Description: This command is used to add initial equilibrium points from which a phase diagram is calculated (through the MAP command). It is not needed for calculating a property diagram (through the STEP command). In many cases, the ADD_INITIAL_EQUIILBRIUM 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 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.

Since TCCS/TCW5, this command becomes unnecessary in most cases, as the re-written 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 BIN or TERNARY module for calculating a binary or ternary phase diagram.

Synopsis 1: ADD_INITIAL_EQUIILBRIUM <direction code>

Synopsis 2: ADD_INITIAL_EQUILIBRIUM

Ensuing 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 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.

All the initial equilibrium points generated by the ADD_INITIAL_EQUIILBRIUM command (previously and presently; saved in the current POLY workspace) can be easily listed out on screen by issuing the LIST_INITIAL_EQUIILBRIUM command (see *Section 8.10.18*). A certain initial equilibrium point (including its conditions and equilibrium results) can be loaded into the current equilibrium, if needed, by the LOAD_INITIAL_EQUIILBRIUM command (see *Section 8.10.19*). Furthermore, any specific or all of the initial equilibrium points can be deleted from current POLY workspace, if desired, by the DELETE_INITIAL_EQUIILBRIUM command (see *Section 8.10.20*).

8.9.6 POST

Description: This command switches to the post-processor, the POST module, which has its own command repertoire.

Synopsis: 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 files (but this feature is only available since TCCR, only in the MS Excel format XLS, and only limited to results from stepped calculations)].

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.*, curve-labeling 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).

For all kinds of details of the POST module, please refer to *Chapter 9 – Post-Processor Module*.

8.10 Other Useful and Advanced Commands

8.10.1 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.

NOTE that this command implies a REINITIATE command and thus it should be given as the first command in the POLY module.

Synopsis 1: `DEFINE_COMPONENTS <all new components>`

Synopsis 2: `DEFINE_COMPONENTS`

Ensuing Prompt: Give all new components /existing components/: <new components>

The new components must be given (all on one line) to replace the existing component definitions. The number of components cannot be changed. If one wishes to change the number of components, the appropriate command is `CHANGE_STATUS`.

If some of the existing component definition shall be kept, it is recommended to type them again in the line; otherwise, the program might be confused in defining the new components, especially when some components are built by several elements.

The new components must already be present as species. For example, in the system Fe-Si-O, one may wish to define the components FEO, FE2O3 and SIO2 instead of the default FE, SI and O.

8.10.2 SET_REFERENCE_STATE

Description: The reference state for a component is important when calculating activities, chemical potentials and enthalpies. See information in [Section 2.13 in the document Thermo-Calc Software System](#). 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).

[Appendix D \(Units for State Variables and Derived Variables\) in the TCCS User’s Guide](#) gives an extensive overview and comprehensive examples on how to set reference states for various system components inside the Thermo-Calc software system and how to explain the calculated results.

Synopsis 1: SET_REFERENCE_STATE <component> <phase> <temperature> <pressure>

Synopsis 2: SET_REFERENCE_STATE

Ensuing 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.

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 /*/: <temperature for the reference state>

One may select the temperature (in K) for the reference state. The 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 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
```

8.10.3 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 “u-fractions” 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 statuses:

- **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 already-retrieved from the chosen database(s). An ENTERED phase-status is always associated with an initially-estimated amount (in mole number; normally, as 0 if the phase is not likely to be stable, and as 0.5 or 1 or any positive number if the phase could be stable) but it is only used as the rough starting value in the equilibrium calculations.
- **SUSPENDED** 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.

Synopsis 1: CHANGE_STATUS <keyword> <name(s)>=<status> <value, optional>

Synopsis 2: CHANGE_STATUS

Ensuing 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 * directly followed by an S, 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 species(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 key word, 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 * directly followed by an S, 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 “u-fractions” 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 it is only used as the rough starting value 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.

Important Note: 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)] can not be directly applied in any POLY command, implying that it can not be directly evaluated or listed/shown. If intended to show such a normalized phase amount value in an equilibrium state, one should instead use a properly-entered symbol (function or variable), for instance,

$$\text{NPFabc} = \text{NP(abc)} / \text{NA}$$

$$\text{or } \text{NPFabc} = \text{NPM(abc)} / \text{NA} * \text{N}$$

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
the evaluation of its NA value becomes even more complicated,

$$\text{NA} = p + q * y_C^2 + q * y_N^2 + q * y_O^{-2} + q * y_{\text{FeO}_{3/2}}^2$$

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 multiple-sublattice model with ionic constituents, such as SPINEL and HALITE phases in some databases.

- Obviously, 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 $N / [p + q * y_C^2 + q * y_N^2 + q * y_O^{2-} + q * y_{FeO_{3/2}}^2]$.

In the following example, two calculated equilibrium states, in which the SIGMA and FCC phases have set as FIXED phase-status, respectively, in a Fe-Cr-Ni-C system under certain conditions of pressure and bulk alloy compositions, are listed:

```
*** Example: Equilibrium State 1:
POLY_3: c-st ph SIGMA=fix .137726659E-4
          = NP(SIGMA)/NA
          = NPM(SIGMA)/NA*N

POLY_3: 1-c
P=100000, X(CR)=1E-1, X(NI)=1E-1, X(C)=1E-4, N=2
FIXED PHASES
SIGMA=1.377267E-05
DEGREES OF FREEDOM 0
POLY_3: c-e
      15 ITS, CPU TIME USED 0 SECONDS
                                     SIGMA phase is stable !
                                     (with fixed NPF at .137726659E-4)

POLY_3: 1-e
OUTPUT TO SCREEN OR FILE /SCREEN/:
Options /VWCS/:
Output from POLY-3, equilibrium number = 1, label A0

Conditions:
P=100000, X(CR)=1E-1, X(NI)=1E-1, X(C)=1E-4, N=2
FIXED PHASES
SIGMA=1.377267E-05
DEGREES OF FREEDOM 0

Temperature 577.14, Pressure 1.000000E+05
Number of moles of components 2.000000E+00, Mass 1.11484E+02
Total Gibbs energy -3.92693E+04, Enthalpy 1.90411E+04, Volume 1.29025E-05

Component          Moles      W-Fraction Activity Potential Ref.stat
C                 2.0000E-04 2.1548E-05 3.3550E-07 -7.1537E+04 SER
CR                2.0000E-01 9.3280E-02 5.2568E-02 -1.4135E+04 SER
FE                1.5998E+00 8.0141E-01 1.9013E-02 -1.9015E+04 SER
NI                2.0000E-01 1.0529E-01 1.9130E-03 -3.0035E+04 SER

BCC_A2#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 1.7831E+00, Mass 9.9047E+01    Mass fractions:
FE   8.47657E-01 CR  1.04366E-01 NI  4.79771E-02 C   1.71162E-12

FCC_A1#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 2.1549E-01, Mass 1.2372E+01    Mass fractions:
NI  5.64585E-01 FE  4.34618E-01 CR  7.96782E-04 C   7.54907E-13

M23C6#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 9.6667E-04, Mass 4.2322E-02    Mass fractions:
CR  9.24289E-01 C   5.67603E-02 FE  1.86006E-02 NI  3.50241E-04

SIGMA#1           Status FIXED      Driving force 0.0000E+00
Number of moles 4.1318E-04, Mass 2.2147E-02    Mass fractions:
CR  5.92227E-01 FE  3.66819E-01 NI  4.09539E-02 C   0.000000E+00

POLY_3: 1-st p
*** STATUS FOR ALL PHASES
PHASE            STATUS      DRIVING FORCE MOLES
SIGMA            FIXED      0.0000000E+00 4.13179977E-04
M23C6            ENTERED   0.0000000E+00 9.6666595E-04
FCC_A1            ENTERED   0.0000000E+00 2.15489190E-01
BCC_A2            ENTERED   0.0000000E+00 1.78313096E+00
HCP_A3            ENTERED   -3.96334108E-02 0.0000000E+00
FCC_A1#2          ENTERED   -4.90774776E-02 0.0000000E+00
CHI_A12           ENTERED   -2.51869838E-01 0.0000000E+00
CEMENTITE         ENTERED   -4.03402719E-01 0.0000000E+00
HCP_A3#2          ENTERED   -4.07818999E-01 0.0000000E+00
FE4N              ENTERED   -5.18975862E-01 0.0000000E+00
CR3SI             ENTERED   -5.92805912E-01 0.0000000E+00
M7C3              ENTERED   -6.53940414E-01 0.0000000E+00
CRSI2             ENTERED   -6.81955340E-01 0.0000000E+00
```

```

CBCC_A12           ENTERED   -1.14394354E+00  0.00000000E+00
ENTERED PHASES WITH DRIVING FORCE LESS THAN   -1.15
CUB_A13 LIQUID M3C2 ALNI_B2 AL3NI2 KSI_CARBIDE FECN_CHI M5C2 V3C2 GRAPHITE
DIAMOND_A4
POLY_3: show NP(SIGMA) NPM(SIGMA)
NP(SIGMA)=4.13179977E-4
NPM(SIGMA)=2.06589989E-4
POLY_3:

*** Example: Equilibrium State 2:
POLY_3: c-st ph FCC=fix 2.15659726E-01
= NP(FCC)/NA
= NPM(FCC)/NA*N

POLY_3: l-c
P=100000, X(CR)=1E-1, X(NI)=1E-1, X(C)=1E-4, N=2
FIXED PHASES
FCC_A1=.2156597
DEGREES OF FREEDOM 0
POLY_3: c-e
12 ITS, CPU TIME USED 0 SECONDS
                                         FCC phase is stable !
                                         (with fixed NPF at 2.15659726E-01)

POLY_3: l-e
OUTPUT TO SCREEN OR FILE /SCREEN/:
Options /VWCS/:
Output from POLY-3, equilibrium number =      1, label A0

Conditions:
P=100000, X(CR)=1E-1, X(NI)=1E-1, X(C)=1E-4, N=2
FIXED PHASES
FCC_A1=.2156597
DEGREES OF FREEDOM 0

Temperature 577.00, Pressure 1.000000E+05
Number of moles of components 2.00000E+00, Mass 1.11484E+02
Total Gibbs energy -3.92547E+04, Enthalpy 1.90290E+04, Volume 1.29009E-05

Component          Moles      W-Fraction Activity Potential Ref.stat
C                 2.0000E-04 2.1548E-05 3.3404E-07 -7.1540E+04 SER
CR                2.0000E-01 9.3280E-02 5.2615E-02 -1.4127E+04 SER
FE                1.5998E+00 8.0141E-01 1.9023E-02 -1.9008E+04 SER
NI                2.0000E-01 1.0529E-01 1.9130E-03 -3.0028E+04 SER

BCC_A2#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 1.7825E+00, Mass 9.9015E+01     Mass fractions:
FE 8.47809E-01  CR 1.04267E-01 NI 4.79241E-02 C 1.69116E-12

FCC_A1#1          Status FIXED      Driving force 0.0000E+00
Number of moles 2.1566E-01, Mass 1.2382E+01     Mass fractions:
NI 5.64613E-01  FE 4.34593E-01 CR 7.94157E-04 C 7.48497E-13

M23C6#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 9.6667E-04, Mass 4.2322E-02     Mass fractions:
CR 9.24313E-01  C 5.67604E-02 FE 1.85777E-02 NI 3.49412E-04

SIGMA#1          Status ENTERED     Driving force 0.0000E+00
Number of moles 8.2636E-04, Mass 4.4294E-02     Mass fractions:
CR 5.92327E-01  FE 3.66758E-01 NI 4.09149E-02 C 0.00000E+00

POLY_3: l-st p
*** STATUS FOR ALL PHASES
PHASE             STATUS    DRIVING FORCE MOLES
FCC_A1            FIXED    0.00000000E+00 2.15659726E-01
SIGMA             ENTERED  0.00000000E+00 8.26359318E-04
M23C6             ENTERED  0.00000000E+00 9.66666596E-04
BCC_A2            ENTERED  0.00000000E+00 1.78254725E+00
HCP_A3            ENTERED  -3.97480337E-02 0.00000000E+00
FCC_A1#2          ENTERED  -4.93283385E-02 0.00000000E+00
CHI_A12           ENTERED  -2.51973421E-01 0.00000000E+00
CEMENTITE         ENTERED  -4.03528585E-01 0.00000000E+00
HCP_A3#2          ENTERED  -4.07616231E-01 0.00000000E+00
FE4N              ENTERED  -5.19312878E-01 0.00000000E+00
CR3SI             ENTERED  -5.92513532E-01 0.00000000E+00
M7C3              ENTERED  -6.54249240E-01 0.00000000E+00
CRSI2             ENTERED  -6.81614967E-01 0.00000000E+00
CBCC_A12          ENTERED  -1.14437450E+00 0.00000000E+00

ENTERED PHASES WITH DRIVING FORCE LESS THAN   -1.15
CUB_A13 LIQUID M3C2 ALNI_B2 AL3NI2 KSI_CARBIDE FECN_CHI M5C2 V3C2 GRAPHITE
DIAMOND_A4
POLY_3: show NP(FCC) NPM(FCC)
NP(FCC_A1)=2.15659726E-1
NPM(FCC_A1)=1.07829863E-1
POLY_3:

```

8.10.4 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.

Synopsis 1: LIST_STATUS <keyword(s)>

Synopsis 2: LIST_STATUS

Ensuing Prompt: 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, in the versions later than TCCN, 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.

Example:

```
POLY_3:1-st
Option /CPS/:
*** STATUS FOR ALL COMPONENTS
COMPONENT      STATUS    REF. STATE      T (K)      P (Pa)
VA            ENTERED   SER
C             ENTERED   GRAPHITE    *
FE            ENTERED   SER
NI            ENTERED   SER
*** STATUS FOR ALL PHASES
PHASE          STATUS    DRIVING FORCE    MOLES
FCC_A1         FIXED     0.00000000E+00  1.00000000E+00
BCC_A2         ENTERED   0.00000000E+00  0.00000000E+00
HCP_A3         ENTERED   -2.69336869E-01 0.00000000E+00
CEMENTITE      ENTERED   -2.86321394E-01 0.00000000E+00
M23C6          ENTERED   -3.44809821E-01 0.00000000E+00
LIQUID          ENTERED   -4.95421844E-01 0.00000000E+00
CBCC_A12       ENTERED   -6.16764645E-01 0.00000000E+00
M7C3           ENTERED   -6.56332559E-01 0.00000000E+00
M5C2           ENTERED   -6.83594326E-01 0.00000000E+00
GRAPHITE        ENTERED   -1.02142788E+00 0.00000000E+00
DIAMOND_A4     ENTERED   -1.73225646E+00 0.00000000E+00
ALNI_B2         ENTERED   -4.79816887E+00 0.00000000E+00
ENTERED PHASES WITH DRIVING FORCE LESS THAN    -4.80
AL3NI2 GAS
HCP_A3          DORMANT   -2.69336869E-01
SUSPENDED PHASES:
V3C2 KSI CARBIDE FECN CHI FE4N CUB_A13
*** STATUS FOR ALL SPECIES
C  ENTERED      C2 ENTERED      C4 ENTERED      C6 ENTERED      FE ENTERED      VA ENTERED
C1 ENTERED      C3 ENTERED      C5 ENTERED      C7 ENTERED      NI ENTERED
```

8.10.5 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.*** It was available since TCCN, and had been improved further since TCCP.

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.

Since TCCP, 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.

Synopsis: COMPUTE_TRANSITION

Ensuing 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
```

Note that since TCCP, 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 assigned 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
```

Note that since TCCP, 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 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 -.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 assigned 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
```

8.10.6 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 convert this into conditions for the current set of components. In the case when the elements are 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!

Synopsis 1: SET_INPUT_AMOUNTS <N(<specie>) or B(<specie>)>=<value>

Synopsis 2: SET_INPUT_AMOUNTS

Ensuing 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.

8.10.7 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.

This command will first list on screen various limitations of system definitions, as below:

```
LIMITATIONS of the present version of Thermo-Calc
Max number of elements : 40
Max number of species : 1000
Max number of sublattices in a phase : 10
Max number of constituents in a phase: : 200
Max number of constituents in an ideal phase : 1000
```

Please notice that the maximum number of constituents in a non-ideal solution phase, such as the AQUEOUS solution using the SIT, HKF or PITZ models, has been increased from 80 to 200 since TCCP.

Synopsis: SET_NUMERICAL_LIMITS

Ensuing Prompt: Maximum number of iterations /200/: <xxx>

By default, the program will try 500 iterations (increased from 200 as in previous versions) 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.

Note that since TCCS/TCW5, the default value for the smallest site-fractions has been changed from 1E-30 to 1E-12 for all phases, except for IDEAL phase with one sublattice site (such as the GAS mixture phase in many databases *e.g.* SSUB4, SSOL4, TCFE6, SLAG2, TCMP2, TCES1) for which the default value is still 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:

Since TCCP, 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
```

implying 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 PC Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current TC 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.

8.10.8 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.

Synopsis: SET_ALL_START_VALUES

Ensuing 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 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/: <.YYYYYY>

The current value (.XXXXXXXXXX) is default. The user may accept the default by pressing <RETURN> or give a new value (.YYYYYY).

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.

8.10.9 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.

Synopsis: SET_START_VALUE

Ensuing 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.

8.10.10 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.

Synopsis: SET_START_CONSTITUITION

Ensuing 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.

8.10.11 RECOVER_START_VALUES

Description: This command can recover the start values in equilibrium calculations. *But it is not implemented yet.*

Synopsis: RECOVER_START_VALUES

8.10.12 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 – Post-Processor Module (POST)*.

Synopsis 1: ENTER_SYMBOL <keyword> <name>=<value, expression or variables>

Synopsis 2: ENTER_SYMBOL

Ensuing 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: 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. 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.

8.10.13 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.

Synopsis: LIST_SYMBOLS

8.10.14 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.

Synopsis 1: DELETE_SYMBOL <name of a symbol>

Synopsis 2: DELETE_SYMBOL

Ensuing Prompt: Name: <name of a symbol>

Specify the name of the symbol to be deleted. Only one symbol can be deleted each time.

8.10.15 SHOW_VALUE

Description: This command is used to show on the screen the current value of any state variable, function or variable.

Synopsis 1: SHOW_VALUE <name(s) of state variable(s) or symbol(s)>

Synopsis 2: SHOW_VALUE

Ensuing 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 wild card “*” can be used to denote all phases or all components. One may also use the dollar “\$” 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.

8.10.16 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.

Synopsis 1: EVALUATE_FUNCTIONS <name(s) of defined function(s)>

Synopsis 2: EVALUATE_FUNCTIONS

Ensuing 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 *, all functions and variables are evaluated.

8.10.17 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.

Synopsis 1: TABULATE <name of a defined table> <Return or a file name>

Synopsis 2: TABULATE

Ensuing 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).

8.10.18 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 STEP and MAP commands.

Synopsis: LIST_INITIAL_EQUILIBRIA

8.10.19 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 results are lost.

Synopsis 1: LOAD_INITIAL_EQUILIBRIUM <number of an initial equilibrium>

Synopsis 2: LOAD_INITIAL_EQUILIBRIUM

Ensuing 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.

8.10.20 DELETE_INITIAL_EQUILIBRIUM

Description: This command deletes an initial equilibrium or all initial equilibria. The initial equilibria are used as start points for the MAP and STEP commands, see also the ADD-INITIAL-EQUILIBRIUM command.

Synopsis 1: DELETE_INITIAL_EQUILIBRIUM <number of an initial equilibrium>

Synopsis 2: DELETE_INITIAL_EQUILIBRIUM

Ensuing Prompt: Number /ALL/: <number of an initial equilibrium>

Specify the number of an initial equilibrium to be deleted from the POLY3 workspaces. Use the LIST-INITIAL-EQUILIBRIA command to find out the numbers for all initial equilibria. By default, ALL are deleted.

8.10.21 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)

Synopsis: AMEND_STORED_EQUILIBRIA

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

Options: L(ist) S(uspend) Q(uey 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 wild card “*” (pressing <RETURN>).

In case of the option L, a block (if a block number is specified) or all the blocks (if the wild card “*” 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 wild card “*” is accepted/used) will be shown up on screen and the user will be asked if suspending or keeping it (see later):

S (suspend) 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 wild card “*” 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 wild card “*” to suspend or restore all ranges in the specified block. In order to know the ranges the LIST option must first be used.

S (suspend) 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.

8.10.22 ADVANCED_OPTIONS

Description:

Since TCCS, this new command **ADVANCED_OPTIONS** replaces and combines the **SPECIAL_OPTIONS** command (that had been available in earlier versions up to TCCR) and **SET_MINIMIZATION_OPTIONS** command (that was added in TCCR).

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 ambiguities when specifying the first few letters and thus tying only fewer letters is faster and easier*):

“SET_MISCIBILITY_GAP” option is replaced by “**NEW_COMPOSITION_SET**”,
 “SET_MAJOR_CONSTITUENTS” by “**MAJOR_CONSTITUENTS**”,
 “SET_PRESENT_PHASE” by “**PRESENT_PHASE**”,
 “SET_PHASE_ADDITION” 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”

Synopsis: ADVANCED_OPTIONS

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

Option: Choose one of the following 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 POLY module in TCCS/TCW5 uses the Global Minimization Technique for various single-point equilibrium calculations and for stepping or mapping calculations. 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 (which is similar to that in previous versions *e.g.* TCCQ/TCW3, TCCP/TCW2 and so on), 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 than 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, which is similar to that in previous versions (*e.g.* TCCQ/TCW3, TCCP/TCW2 and so on).

Notes: 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, *i.e.*, $G_m(X) = G_m(T, P; y_1, y_2, \dots, y_n)$, and we may compute a Hessian matrix as $\partial^2 G_m / \partial y_i \partial y_j$ which describes the curvature of the Gibbs energy curve of this phase at the defined composition X. Diagonalize this matrix and call the Eigenvalues as e_1, e_2, \dots, 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: $QF(ph) = \min(\{e_1, e_2, \dots, e_n\}) / \min(\{se_1, se_2, \dots, se_n\})$, where se_1, se_2, \dots, se_n 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 in TCCS/TCW5 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 (which is similar to that in previous versions *e.g.* TCCQ/TCW3, TCCP/TCW2 and so on), 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 in TCCS/TCW5.

➤ STEP_AND_MAP

The advanced option STEP_AND_MAP that has been available since TCCS/TCW5 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 nth 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 (TCCS/TCW5) 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

This advanced option replaces the SET_MISCIBILITY_GAP option that was available in versions prior to TCCS/TCW5. A solution phase that can exist with two (or more) different compositions simultaneously must have two (or more) composition sets in its phase descriptions. Previously this had to be done in the Gibbs Energy System (GES module), but in the version TCCJ the command SET_MISCIBILITY_GAP was introduced into the POLY module. Now it has been made 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.

Since TCCR/TCW4, 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.

In the previous versions up to TCCR/TCW4, it is often necessary to have two or more composition sets in phases that have order/disorder transformations, but the users have often forgotten to create composition sets for both the ordered and disordered phases, and this leads to program crashes. This advanced 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 GES 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

The previously-available SET_MAJOR_CONSTITUENT option has been renamed as to this advanced option since TCCS/TCW5. 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

The previously-available SET_PHASE_ADDITION option has been renamed as to this advanced option since TCCS/TCW5. 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 (TDB) or Gibbs Energy Module (GES), 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 previously-available SET_PRESENT_PHASE option has been renamed as to this advanced option since TCCS/TCW5. 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 T_0 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.

Note that within TCCP only carbon C was possible to be considered as the interstitial component which was always set as default; however, this advanced option had been greatly modified and improved since TCCQ, and it now allows more than one interstitial components treated as fast diffusion species (such as C, N, O, S, *etc.*, as *individual or combined*) in a paraequilibrium calculation.

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 can not 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 starting bulk composition for the system.
- Always set the status of the chosen interstitial components as SPECIAL 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. Since TCCQ, 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
```

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). 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 gap 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=

Since TCCS, this new 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.

Notes:

Since TCCP, 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

implying 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 PC Linux, as 2.00E-14). For other subsequent POLY-module calculation in the current TC run, you can use the POLY command SET_NUMERICL LIMITS to restore or reset the smallest site fraction to the previous or another preferred value, as well as to reset other numerical limits (see *Section 8.10.7* on the SET_NUMERICL LIMITS command).

Further Notes: Since TCCS, 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

A miscibility gap test has been added to all calculations. This works both in the normal compute equilibrium as well as in STEP and MAP. Normally the database will create all necessary composition sets for possible miscibility gaps of a phase, but the user may also enter them, see other ADVANCED_OPTION options. The test will try the default composition of all composition sets and calculate if it is possible for the set to exist, stable or metastable, with a composition different from any other composition sets of that phase. If this is successful, it means that one is within a stable or metastable miscibility gap. If all composition sets have different compositions, the test will be skipped, otherwise there will be a message for each phase tested. This message will have probably been suppressed in some later versions.

Miscibility gap test frequency /5/: <#>

This rather awkward command can be useful to control the miscibility gap test during STEP and MAP. By default a test is made every *fifth* calculation but the user may change this to another numerical number # by this option if there are problems during a MAP or STEP command.

➤ SET_BREAK_CONDITION

The condition for terminating a STEP command can be set.

Break Condition: <a break condition>

The user can specify a break condition that can cause the STEP calculation to terminate before it has reached the end of the axis. The break condition can look like NP(LIQ) < 0.001 or W(FCC, CR) >.13 or REST=1 where REST is a user defined function or variable.

➤ OUTPUT_AT_MAP_AND_STEP

Normally, the information given on screen during a MAP or STEP calculation is compressed, and only the first few values of the mapping or stepping variables are shown for each phase assemblage. However, if one wishes to see all the details of a MAP or STEP calculation as in a long list, he can set this advanced option on.

On? /Y/: <Y or N>

The user may answer Y (Yes) or N (No). The default is Y to switch on the extensive listing during all subsequent MAP or STEP calculations. By answering N, the listing is set back to normal.

➤ NEVER_ADJUST_MINIMUM_Y

Since TCCS/TCW5, the default value for the smallest site-fractions has been changed from 1E-30 to 1E-12 for all phases, except for IDEAL phase with one sublattice site (such as the GAS mixture phase) for which the default value is still as 1E-30.

With this special option on, the smallest site-fractions for non-gaseous species in all subsequent single-point equilibrium and MAP/STEP calculations will be permanently set as the possibly lowest value [*i.e.*, 1E-12 by default, or another lower value (*e.g.*, 1E-30) set by the SET_NUMERICAL_LIMITS command]. Even if there is a convergence problem, this special option will not allow any automatic adjustment in this numerical limit, unless the SET_NUMERICAL_LIMITS command is once-again used afterwards.

Note that the smallest site-fractions for gaseous species modelled by the *IDEAL EOS/Mixing Model* (as in the so-called GAS mixture phase in many databases *e.g.* SSUB4, SSOL4, TCFE6, SLAG2, TCMP2, TCES1) are always set as 1E-30, regardless if one has re-set the site-fraction limit to a higher value (*e.g.* 1E-12) using the command SET_NUMERICAL_LIMITS. However, one has re-set the limit to a lower value (*e.g.* 1E-45), then such a site-fraction limit will be enforced to all the phases in the system.

➤ SHOW_OUTPUT_FILE

With this advanced option, a textual-file name will be asked for and all the results output from the command SHOW_VALUE will be written to this file. Since TCCQ this advanced option has been slightly improved, so that the output will still appear on screen.

8.10.24 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.

Synopsis: CREATE_NEW_EQUIlibrium

Ensuing 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.

8.10.25 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.

Synopsis 1: SELECT_EQUIlibrium <choice on equilibrium>

Synopsis 2: SELECT_EQUIlibrium

Ensuing 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 equilibria.

8.10.26 MACRO_FILE_OPEN

Description: MACRO is an extraordinary convenient and easier way of pre-defining sequences of various legal TCC (and DICTRA) commands stored in a so-called **MACRO file** (which is a simple textual file normally with the default extension “TCM” for the TCC software 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 TCC software; SYS, POLY, PARROT and DICTRA_Monitor module in the DICTRA software).

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 TCC software, if in the SYS module the **SET_LOG_FILE** command is used and a **LOG file** name is given before any other SYS/TDB/TAB/GES/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 current TCC session 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 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 TCC (and DICTRA) program.

All kinds of legal TCC (and DICTRA) commands and their required inputs can be stored into a MACRO file. A MACRO file must be terminated with the **EXIT** command, or can be interrupted (for the purposes of allowing further interactive operations by the user) in the SYS, GES, POLY, PARROT or POST modules with the **SET_INTERACTIVE** command.

Within a MACRO file (for previous versions up to TCCR), one could have as many as possible **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 TCC commands when the file is called by the TCC software.

However, this might be considered as somewhat tedious. Therefore, it is now (since TCCS) possible to have multiple-line **comment-blocks** inside a MACRO file, using the paired “@(*” and “@*)*” signs (*i.e.*, the indicators that starting the lines for the “begin comment” and “finish comment”, respectively). 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 TCC 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 2nd 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, a textual copy of 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, e.g.

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

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 TCC software from temporarily stopping at any pause by simply typing any character (except for the Y character) after specifying the name of a MACRO file.

Since TCCN, 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 TCC 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_Module command **SET_ECHO** at the every 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 TCC (and DICTRA) software, that are enforced according to the MACRO file(s).

Synopsis 1: MACRO_FILE_OPEN <name of a Macro file>

Synopsis 2: MACRO_FILE_OPEN

Ensuing Prompt: Macro filename: <name of a Macro file>

Specify the filename with the MACRO command. The default extension is “TCM”.

Notes: Under Windows Vista/XP/2000/NT4 environments, if an appropriate MACRO file is not given after the command, an **Open file** 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 8-7*.

The file type (i.e., TCM, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with executing various Thermo-Calc commands. The user may also cancel such an **Open file** window session, and thus the current MACRO file will not be opened.

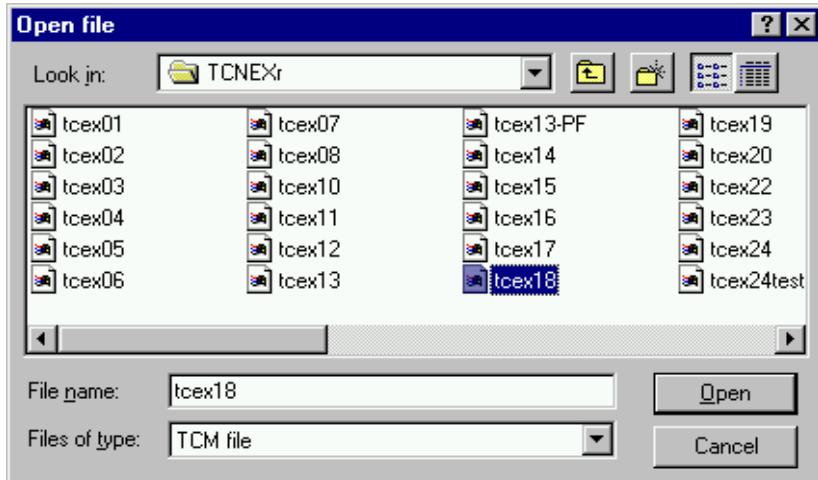


Figure 8-7. The “Open file” window: Opening an existing TCM file.

If the MACRO file contains some SYS/TDB/TAB/GES/POLY/POST/PARROT/ED_EXP-module commands for setting *.LOG files, saving/reading GES5/POLY3/PARROT workspaces, switching USER databases, compiling experiments (from existing *.POP files), creating new *.PAR files, appending experimental data *.EXP files, plotting/dumping diagrams, etc., a corresponding window (e.g., *Save As*, *Open file*, *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.

More Notes:

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, i.e., 1 (under Windows Vista/XP/2000/NT4) or 9 (under Linux and all types of UNIX platforms).

For more extensive examples, please refer to various TCM files on the TCC installation area \TCEX\ and in the *TCCS Examples Book*.

8.10.27 REINITIATE_MODULE

Description: The module is reinitiated to the state it was when first entered. All reset conditions, changed status, entered symbol are removed. The saved file name is restored to the default.

Synopsis: REINITIATE_MODULE

8.10.28 PATCH

Description: This command is only for those who think they know what they are doing! It has been deleted since TCCN!

Synopsis: PATCH

8.11 Aqueous Solutions

Due to the complexity of heterogeneous interactions involving aqueous solutions, some special assignments have to be enforced in the Thermo-Calc software/database/interface package for aqueous solution phases, regarding phase definitions, reference states, additional derived variables, models and data manipulations, as well advanced easy-to-use modules.

8.11.1 Phase definitions and reference states for the aqueous phase

The aqueous solution phase is always referred as the so-called AQUEOUS phase, no matter what aqueous solution model is applied to the phase. One would never mix different models for the phase, because that whenever a database containing an aqueous solution phase is retrieved, a specific aqueous solution model (which is predefined by the database) would be connected, *e.g.*, the SIT Model with the TCAQ and PAQ Aqueous Solution Databases, and the complete Revised HKF Model with the AQS Aqueous Solution Database.

As extensively described in the GES module (Gibbs Energy System, in *Part 11*), in order to handle various charged aqueous species (free ions or complexes) and internal electron balance in the aqueous solution phase, a special element (named “ZE”) is used in chemical formulas of aqueous species. For instance, H1ZE1 for H^+ , H1O1ZE-1 for OH^- , C2H1O4ZE-1 for HC_2O_4^- , C2H3O2ZE-1 for CH_3COO^- , C6FE1H2N6ZE-2 for $\text{FeC}_6\text{H}_2\text{N}_6^{2-}$.

Furthermore, for the purpose of calculating standard electronic potential (or redox potential) in the aqueous solution phase, a hypothetical electrode phase (named “REFERENCE_ELECTRODE”, referring to standard hydrogen electrode) is predefined in the database and thus in all Thermo-Calc calculations. Of course, the REFERENCE_ELECTRODE phase shall never be stable together with the aqueous solution phase, and is thus always assigned as a suspended phase in the GES5 and POLY3 workspaces.

The component definitions for the basic elements O (oxygen), H (hydrogen) and ZE (electron) in an aqueous solution bearing system should be always as H₂O, H+ and ZE, in order to construct the aqueous solution phase, and to calculate the aqueous activities, concentrations, acidity (pH), electronic potential (Eh) and so on. For other elements defined in the interaction system, their components can be defined as either the corresponding elements (such as FE, NI, C, S, NA and CL) or other interested formulas (such as FECL3, NICL2, CO2/CO3-2, H2S, NACL and CL-1).

The reference state of the H₂O component is usually set as the solvent (*i.e.*, the constituent H₂O) in the AQUEOUS phase at the current temperature and 1 bar. For the convenience of calculating aqueous solution concentration, normally in molality (mole of solute in 1 kg of water), the initial bulk composition of the H₂O component is usually set as 1.0 kg, *i.e.*, B(H₂O)=1000 or N(H₂O)=55.5084. Of course, one may also set the weight of water differently, *e.g.*, 2 kg [B(H₂O)=2000 or N(H₂O)=111.0168] or 100 grams [B(H₂O)=100 or N(H₂O)=5.55084].

8.11.2 Connected models, databases and modules for the aqueous phase

There have been some improvements and modifications for thermodynamic calculations for aqueous solutions since TCCL (released in 1996). Such improvements and modifications have been made on the following directions:

- 1) Implementation and verifications of aqueous solution models in the Gibbs Energy System (*e.g.*, SIT, HKF and PITZ models; see *Chapter 11*)
- 2) Development and enlargement of aqueous solution databases in the Thermodynamic Database spectrum (*e.g.*, AQ/PAQ, TCAQ, AQS, *etc.*; see *Chapter 4*)
- 3) Establishment and enhancement of some advanced modules with easy-to-use interface (*e.g.*, POURBAIX; see *Chapter 10*)

Some more comprehensive models, reliable databases, advanced modules and enhanced features for aqueous solution involved systems are under further development at Thermo-Calc Software.

8.11.3 Additional derived variables for the aqueous phase

The AQUEOUS solution phase is treated in a comprehensive way within the GES and POLY modules, due to its special requirements on presenting the calculated quantities which are of particular interest for aqueous chemistry, materials corrosion, chemical engineering, geochemistry, environmental engineering, etc.

All the standard state variables used for other phases (see *Section 8.3.3* and *Section 2.5* in the document *Thermo-Calc Software System*) can be directly applied to the AQUEOUS solution phase. Moreover, some additional derived variables that are defined as functions of state variables are necessary for the phase.

The Thermo-Calc software (especially some advanced, easy-to-use modules) predefines some derived variables as assigned symbols (variables, functions and tables) for some phases, such as aqueous solutions, gaseous mixtures, liquids, etc. A user can also define any derived variable, as he prefers for his systems.

Some examples of the Thermo-Calc predefined derived variables for the AQUEOUS solution phase are listed and briefly described in *Table 8-2*. Because the EOS (Equation of State) expressions and standard thermodynamic properties for the pure solvent H₂O must be identical to that for the pure gas species H₂O, some derived variables for a GAS mixture phase are also listed in the table. Under a certain temperature-pressure-composition condition, an aqueous solution phase may be in equilibrium with a stable gaseous mixture phase or a saturated vapour.

Please note that some of derived variables for such phases as listed in the table may have been automatically predefined by the GES and POLY modules, as well by some advanced modules (*e.g.*, POURBAIX).

Please also note that such derived variables, either predefined by the GES, PLOY or POURBAIX module, or entered by the users, are saved as special symbols (variables, functions or tables). As clearly mentioned in *Section 8.10.7*, each symbol has a unique name that must start with a letter and can have maximum 8 legal characters [including letters (either UPPER or lower case), digits and underscore “_”, but not any other special character such as parentheses “(” and “)”, plus “+”, minus “-”, slash “/” or “\”, period “.”]. Therefore, those quantities for various individual aqueous species (and gaseous species) are defined as a name followed by a digital number, *e.g.*, NS#, ML#, TIC#, RC#, logAI#, AI#, (and RA#, RF# and FUG#).

8.11.4 Modification of output from LIST_EQUILIBRIUM

In order to provide as much information as possible for the AQUEOUS solution phase involved in various heterogeneous equilibria, the standard output format of equilibrium result list in POLY module has been expanded and modified slightly. Such modifications were made in the following aspects:

- 1) More equilibrium information for the solvent (water) and each of the aqueous solutes are list in table, including: site fraction, mole numbers, molality, activity and its common logarithm (log₁₀Act).
- 2) Some overall properties for the entire aqueous solution phase are list as well: acidity, electronic potential, thermodynamic affinity and activity of hypothetical electron, ionic strength, total molality, titration alkalinity, water activity and osmotic coefficient, and so on.
- 3) The lowest site-fraction limit for aqueous solution species can be altered using the SET_NUMERICAL_LIMIT command.

Accordingly, the POLY command LIST_EQUILIBRIUM always automatically presents some modified and additional information for an aqueous solution phase.

Table 8-2. Examples of derived variables for aqueous solution and gaseous mixture phases

Name	Mnemonic	Possibly Units	Meaning	Comments
For Aqueous Solution Phase				
pH	PH	dimensionless	Acidity	of the aqueous solution phase, $pH = -\log_{10}(AC(H^+)) = -\log_{10}(ACR(H^+, AQ))$
Eh	EH	V, mV	Hypothetical electric potential	of the aqueous solution phase, $Eh = u(ZE) / 96485.309$
pe	PE	dimensionless	Logarithm of the hypothetical electron activity	of the aqueous solution phase, $pe = u(ZE) / (2.3025851 * RT)$
Ah	AH	kJ, kcal	Thermodynamic affinity per electron	of a redox couple wrt the standard hydrogen electrode in the aqueous solution phase, $Ah = u(ZE)$
y _w	YH2O	dimensionless	Mole fraction	of the solvent H ₂ O in the aqueous solution phase, $YH_2O = Y(AQ, H_2O)$
N _w	AH2O	mole	Mole number	of 1.0 kg of solvent H ₂ O, $AH_2O = 55.508435$
N _{sp}	NSH2O	mole	Mole number, NS (AQ, H ₂ O)	of the solvent (H ₂ O) in the aqueous solution phase, $NS(AQ, H_2O) = YH_2O * NP(AQ)$
	NS#		Mole number, NS (AQ, sp)	of a solute species in the aqueous solution phase, $NS(AQ, sp) = Y(AQ, sp) * NP(AQ)$
m	ML#	mol/kg_H ₂ O	Molality, ML (AQ, sp)	of a solute species in the aqueous phase, $ML(AQ, sp) = Y(AQ, sp) * AH_2O / YH_2O$
m*	TIM	equivalent molality	Total molality	of all solute species in the aqueous solution phase $TIM = \sum [ML(AQ, sp)]_{ions} + \sum [ML(AQ, sp)]_{complexes}$
m _i	TIC#	equivalent molality	Total ionic concentration, TIC (AQ, sp)	of a cation I in the aqueous p solution base, $TIC(AQ, spI) = \sum [ML(AQ, spJ)] * V(spI-in-spJ)$
I	ISTR	dimensionless	Ionic strength	of the aqueous solution phase, $ISTR = 1/2 * \sum [ML(AQ, sp)] * Z(AQ, sp) ** 2$
γ	RCH2O	dimensionless	Activity coefficient, RC (H ₂ O, AQ)	of the solvent (H ₂ O), $RC(H_2O, AQ) = ACR(H_2O, AQ) / YH_2O$
	RC#		Activity coefficient, RC (sp, AQ)	of a solute species, $RC(sp, AQ) = ACR(sp, AQ) / Y(AQ, sp) * YH_2O$
α _i	AIH2O	dimensionless	Activity, AI (H ₂ O, AQ)	of the solvent (H ₂ O), $AI(H_2O, AQ) = ACR(H_2O, AQ)$
	AI#		Activity, AI (sp, AQ)	of a solute species related to the aqueous phase, $AI(sp, AQ) = ACR(sp, AQ) * AH_2O$
	LogAI#		Log10 Activity, LogAI (H ₂ O, AQ), LogAI (sp, AQ)	of the solvent or a solute species related to the aqueous solution phase in common logarithm, $LogAI(H_2O, AQ) = \log_{10}[AI(H_2O, AQ)]$ $LogAI(sp, AQ) = \log_{10}[AI(sp, AQ)]$
α _w	AW	dimensionless	Activity	of H ₂ O in the aqueous solution phase, $AW = ACR(H_2O, AQ)$
ϕ	OS	dimensionless	Osmotic coefficient	of aqueous solution phase, $OS = -55.508435 * \ln AW / TIM$
At1	AT1	equil_mol/kg_H ₂ O	Titration alkalinity (definition 1)	of the aqueous solution phase, generally defined as the equivalent molality of carbonate and bicarbonate at the methyl orange endpoint (pH=4.5)
At2	AT2	equil_mol/kg_H ₂ O	Titration alkalinity (definition 2)	of the aqueous solution phase, generally defined as the equivalent molality of carbonate and bicarbonate, plus sulfide, at the methyl orange endpoint (pH=4.5)
For Gaseous Mixture Phase				
γ	RA#	dimensionless	Activity coefficient, RA (sp, GAS)	of a gaseous species in the gaseous mixture, $RA(sp, GAS) = function(Y, T, P)$
γ*	RF#	dimensionless	Fugacity coefficient, RF (sp, GAS)	of a pure gaseous species under TP, $RF(sp, GAS) = function(T, P/V)$
f	FUG#	pa, bar, psi	Fugacity, FUG (sp, GAS)	of a gaseous species in the mixture, $FUG(sp, GAS) = RA(sp, GAS) * RF(sp, GAS) * Y(GAS, sp) * P$
f _t	TFUG	pa, bar, psi	Total gas fugacity	of the gaseous mixture phase, $TFUG = \sum [FUG(sp, GAS)]$

The *first part* of output table for the AQUEOUS solution phase is essentially the same as normal cases, *i.e.*, it contains the Phase status, Driving force, Number of moles, Mass and Mole fractions of system components.

The *second part* for the phase consists of the details of phase constitution: corresponding to each aqueous species, there exist Site Fraction, Mole number, Molality, Activity and its common logarithm ($\log_{10}\text{Act}$).

The definitions of and relations among these properties are as follows:

- ❖ SiteFraction is the mole fraction for each aqueous species (the constituent in the aqueous solution phase), since the AQUEOUS phase is regarded as a single-sublattice solution; in other words the equation stands for all aqueous species:

$$X(\text{AQUEOUS}, \text{sp}) = Y(\text{AQUEOUS}, \text{sp})$$

As a common state-variable in Thermo-Calc, it is calculated in the POLY module using Gibbs free energy relationship implemented in the GES models and TDB databases.

- ❖ Mole is the mole number for each aqueous species in a heterogeneous equilibrium which is related to SiteFraction as:

$$NS(\text{AQUEOUS}, \text{sp}) = Y(\text{AQUEOUS}, \text{sp}) * NP(\text{AQUEOUS})$$

where $NP(\text{AQUEOUS})$ is the mole number of AQUEOUS phase in the interaction system.

- ❖ Molality is the molal concentration for each aqueous species, *i.e.*, mol/kg. For the solvent H₂O, the number is always as 5.55084E+01 which implies the molal concentration is referred to the mole number per kg of water; For aqueous solutes (cations, anions and neutral molecules), molality is calculated from:

$$ML(\text{AQUEOUS}, \text{sp}) = Y(\text{AQUEOUS}, \text{sp}) * 55.5084 / Y(\text{AQUEOUS}, \text{H}_2\text{O})$$

- ❖ Activity is the activity for each aqueous species with the specific standard state (which will be further described later). Note that Thermo-Calc has a more general definition about the component activity (AC or ACR) which uses the common standard state of component for all kinds of phases in a system. The relation between them is given as below:

$$AI(\text{H}_2\text{O}, \text{AQUEOUS}) = ACR(\text{H}_2\text{O}, \text{AQUEOUS})$$

$$AI(\text{sp}, \text{AQUEOUS}) = ACR(\text{sp}, \text{AQUEOUS}) * 55.5084$$

- ❖ $\log_{10}\text{Act}$ is the common logarithm of activity for each aqueous species, *i.e.*,

$$\log_{10}\text{Act}((\text{sp}, \text{AQUEOUS})) = \log_{10}(AI(\text{sp}, \text{AQUEOUS}))$$

Notes for the definitions of standard state and activity for aqueous species: It is important to be aware of the different definitions of standard states and activities. As generally preferred in the aqueous chemistry society, we use the followings in Thermo-Calc:

- For the solvent (the major component, H₂O, in aqueous solution phase), use the Raoult's Law, and take the standard state as the *pure solvent*:

$$\mu_A = \mu_A^0 + RT \ln a_A$$

where $a_A = \gamma_A X_A$ with $\gamma_A \rightarrow 1$ as $X_A \rightarrow 1$ (*i.e.*, pure solvent)

- For the solute (the minor component), use Henry's Law, and take the standard state as the *hypothetical state at unit molality but in which the environment of each molecular is the same as at infinite dilution*:

$$\mu_B = \mu_B^\Theta + RT \ln a_B$$

where $a_B = \gamma_B m_B / m^\Theta$ with $\gamma_B \rightarrow 1$ as $m_B \rightarrow 0$, $m^\Theta \stackrel{\text{def}}{=} 1 \text{ mol/kg}$ (*i.e.*, infinite dilution)

For further readings on the definitions, please see Atkins (1982): *Physical Chemistry* (2nd Edition), pp. 243-248 and pp. 314-316.

(The applications of such terms are MUCH LESS CONFUSING than their definitions.)

The *third part* for the phase gives some additional information on solution properties for the entire aqueous solution phase, as below:

Solution Properties:	pH = 8.8472	Eh = -0.5263 V	I = 3.0067
	pe = -8.8962	Ah = -50.7796 kJ	m* = 6.0143
	Aw = 0.9010	Os = 0.9622	
	At1= 3.9785E-23	At2= 1.2452E-05	(equiv-mol/kg-H ₂ O)

The definitions of and relations among these properties are as follows:

- pH is the acidity of aqueous solution:

$$\text{pH} = -\log_{10}(\text{AC}(\text{H}^+))$$

$$= -\log_{10}(\text{ACR}(\text{H}^+, \text{AQUEOUS}))$$
 [if the reference state for H⁺ is that in aqueous solution]
- Eh is the hypothetical electrical potential of the equilibrium:

$$\text{Eh} = \text{u}(\text{ZE}) / 96485.309$$
- Ah is the thermodynamic affinity per electron of a redox couple with respect to the standard hydrogen electrode, *i.e.*, the chemical potential of the hypothetical electron in aqueous solution:

$$\text{Ah} = \text{u}(\text{ZE})$$
- pe is the logarithm of the hypothetical electron activity:

$$\text{pe} = \text{u}(\text{ZE}) / (2.302585 * \text{RT})$$
- I is the ionic strength of aqueous solution:

$$\text{I} = 1/2 * \text{sum}(\text{ML}(\text{AQUEOUS}, \text{sp}) * \text{Z}(\text{AQUEOUS}, \text{sp})^{**2})$$
- m* is the total molality (in equivalent molality) of all solute species (including free ions and complexes) in the aqueous solution in equilibrium:

$$\text{m}^* = \text{sum}[\text{ML}(\text{AQUEOUS}, \text{ions})] + \text{sum}[\text{ML}(\text{AQUEOUS}, \text{complexes})]$$
- Aw is the water activity in aqueous solution:

$$\text{Aw} = \text{ACR}(\text{H}_2\text{O}, \text{AQUEOUS})$$
- Os is the osmotic coefficient of aqueous solution:

$$\text{Os} = -55.5084 / \text{m}^* \ln \text{Aw}$$
- At is the titration alkalinity of aqueous solution, generally defined as the *equivalent molality* of carbonate and bicarbonate (in some cases plus sulfide) at the methyl orange endpoint (*i.e.*, pH=4.5):

$$\text{At1} = \text{m}(\text{HCO}_3^-) + \text{m}(\text{CaHCO}_3) + \dots + 2\text{m}(\text{CO}_3^{2-}) + 2\text{m}(\text{CaCO}_3) + \dots$$

$$\text{At2} = \text{m}(\text{HCO}_3^-) + \text{m}(\text{CaHCO}_3) + \dots + 2\text{m}(\text{CO}_3^{2-}) + 2\text{m}(\text{CaCO}_3) + \dots$$

$$+ \text{m}(\text{OH}^-) + \text{m}(\text{CaOH}) + \dots + \text{m}(\text{HS}^-) + 2\text{m}(\text{S}^{2-}) + \dots$$

8.11.5 Calculations with aqueous solution involving systems

The heterogeneous equilibrium calculations involving aqueous solutions in the Thermo-Calc package is nevertheless the same as other types of equilibrium calculations. However, the setups of equilibrium conditions and reference states in the calculation systems must be made in an appropriate way, the setups of equilibrium conditions and reference states in the calculation systems must be made in an appropriate way, mainly due to the four respects given as below:

- the complexity of the aqueous solution models;
- the specific choices of the standard and reference states of aqueous species;
- the unique definitions of activities of water and solutes; and
- the requirements on proper definitions for aqueous concentrations.

In practice, the calculations can be performed in the traditional command-lines or in special advanced-modules, with the above four respects kept in mind or in design.

8.11.5.1 Command-line calculations

Normally, the following statements have to necessarily specified before an equilibrium for a system involving an aqueous solution phase is calculated:

```
DEFINE_COMPONENT H2O, H+, ZE, <others>
```

to make the aqueous solution phase is acceptable for the models and databases in the Thermo-Calc software system. The first three components, H₂O, H⁺ and ZE, are always necessary for the aqueous solution phase, and they are the only correct component definitions for the basic elements O (oxygen) and H (hydrogen) plus the specially-assigned “element” ZE (electron). Otherwise, it will impossible to construct the aqueous solution phase, and to calculate the aqueous activities, concentrations, acidity (pH) and electronic potential (Eh). For other elements defined in the interaction system, the <others> components can be defined as either the corresponding elements (such as FE, NI, C, S, NA and CL) or other interested formulas (such as FECL₃, NICL₂, CO₂/CO₃-2, H₂S, NACL and CL-).

```
SET-CONDITION B(H2O)=1000
```

to make the aqueous solution phase always consisting of 1 kg of solvent (water), which will be easy to calculate aqueous solution concentration, normally in molality (mole of solute in 1 kg of water). Of course, one may set the weight of water differently (e.g., 2000 grams or 100 grams); one may even set the H₂O constituent in mole unit, e.g., N(H₂O)=55.5084, or N(H₂O)=111.0168 and N(H₂O)=5.55084.

```
SET-CONDITION T=298.15, P=1E5, <bulk composition for other components>
```

to set the temperature and pressure as room condition or other different values, and to set the bulk composition for all other available components in the interaction system.

```
SET-REFERENCE-STATRE H2O AQUEOUS * 100000
```

to set the reference state for the component H₂O in the interaction system as the constituent H₂O in the AQUEOUS phase, and to set the temperature and pressure for the reference state as the current temperature and 1 bar.

```
SET-REFERENCE-STATRE ZE REF_ELECTRODE * 100000
```

to set the reference state for the specially-assigned component ZE in the interaction system as the REFERENCE_ELECTRODE phase, and to set the temperature and pressure for the reference state as the current temperature and 1 bar.

```
CHANGE-STATUS PHASE REF_ELECTRODE=SUSPEND
```

to set the phase status of the REFERENCE_ELECTRODE phase as suspended, for the reason that this is only a hypothetical phase and it is always unstable in the interaction system. This hypothetical phase is used for calculating the standard electronic potential.

Other POLY commands might need to be used to specify the interaction system in more details, especially for pure solid, solid solution and gas mixture phases. For details, see the corresponding parts in this chapter.

8.11.5.2 Advanced-module calculations

Having all the special requirements for calculating an aqueous solution involved heterogeneous interaction system considered in programming, some advanced and easy-to-use modules, and enhanced options, have been developed in the Thermo-Calc software. At present moment, the so-called POURBAIX module can be used to calculate the pH-Eh diagrams and many other property diagrams for complex heterogeneous interaction systems. Other advanced modules for aqueous involving systems would be made available in forth coming releases of Thermo-Calc in the near future.

With such special advanced modules, the user will not need to specify the complicated system/component definitions and the basic calculation conditions; instead, the software just asks the user to answer some simple questions, and then conducts the entire calculation and post-processing automatically.

For more details on such advanced modules, see *Section 10.7*.

8.12 Trouble Shooting

8.12.1 First step

When a calculation fails to converge check first that the set of conditions are reasonable. This may not be trivial and the following checks are recommended:

8.12.1.1 Check conditions

Use only conditions on the temperature, pressure and the overall amount or fraction of components for the first calculation. If you use conditions on activities or partial Gibbs energies make sure that the values are reasonable (Check the reference states!). Note that W and X are mass (weight) fractions and mole fractions, not percent! The sum of the fractions must not exceed unity, the program does not check that, it just fails to converge!

Note that if you have conditions on the overall amounts of fractions of the components, the amount of one of the components should not be given explicitly but be given by the condition N=1 (*i.e.*, that the system is closed with one mole of components). Note that if you have only “mass balance” conditions this means that the system is closed.

8.12.1.2 Simplify the system

If you use a condition on the fraction of a component in a phase try to replace this by a condition on the overall composition (if the phase with the fraction is not stable such a condition may give strange results), try to replace this by a condition on the overall composition in the first calculation. When this has converged and the phase is indeed stable you may change back to the condition on the fraction in the phase.

8.12.1.3 Phases

Some conditions are identical although they may not seem so. Of course setting the activity of C equal to unity and the partial Gibbs energy of C to zero is the same thing but it may not be obvious that changing the status of graphite to fix is also the same thing. (All these conditions require that C has graphite at the current temperature as reference state). Note that in the SGTE solution and substance databases the default reference state is the stable state at 298 K and 1 bar. Using this reference state carbon will always have activity lower than unity at higher temperatures even if it is in equilibrium with graphite. But, it is not necessary to have one condition on each component. It is perfectly legal to have conditions both on the amount and on the activity of carbon.

8.12.1.4 Zero fractions

Normally you should not use a condition that the fraction of a component is zero. It works in a few cases when you do not have the elements as components. You should use the command CHANGE_STATUS COMP in order to suspend the component instead. However, the CHANGE_STATUS does not always work properly for other components than the elements.

8.12.2 Second step

If you have checked and double-checked your conditions, you can try the SET_ALL_START_VALUES F command-sequence. The option F stands for “FORCE” and by this command the program will set automatic start values even if some phase is stable. Give another C_E command.

If this is not successful you must provide explicit manual start values before giving a new C_E command. If you answer N to the question of Automatic start values this command will first ask you for a start value of T and P unless they are set as conditions. Secondly it will for all entered phases ask if the phase should be stable.

The selection of initial set of stable phases is usually the most important for the convergence. In particular when some phases do not dissolve all components. The calculation will invariably fail unless the initial set of phases can span the composition space. Usually this is not a big problem as the gas phase or liquid phase usually exist over

the whole composition space and by assuming only gas or only liquid one will always be able to start the iteration cycle.

The command will also ask you for major constituents of the phases which may vary in composition. There are several answers to this (as is displayed by a ? when asked this question). By giving * the default major constituents will be set (usually a good choice).

By giving \$ the current constitution is not changed, by giving NONE the program will ask for the site fraction of each constituent. The constitution of the phases are usually less important than the initial set of stable phases, unless you have miscibility gaps. If you have a miscibility gap you must enter both composition sets as stable and give initial compositions on the different sides of the gap.

8.12.3 Third step

If everything else fails, you may have to SUSPEND all phases except one. This should be the one you think should be dominant but there are exceptions. If you calculate an equilibrium with stoichiometric phases at low temperature, the program may have difficulties to find the set of stable phases. In that case it may be advantageous to suspend all phases and just have the gas phase entered (provided the gas phase dissolves all components!), even if the gas may not be stable at the final equilibrium. The calculation with just the gas phase MUST converge if it dissolves all components and the conditions are reasonable. When it has converged you may set the phases back to dormant by the command

```
C-S P *S=D  
C-E  
L-ST P
```

The *S means all suspended phases and =D means that they are set dormant. After a new C-E give an L-ST P command and this will display the phases ordered in decreasing driving force. The phases that would like to be stable have positive driving forces. Try to set the phase with highest positive value back as entered with the amount one mole.

```
C-S P <phase>=E 1  
C-E  
L-ST P
```

This should also converge. If not there is probably something wrong with the data for <phase>. The new L-ST P command will show which phase now has the highest driving force. Change this to be entered and continue like this until all dormant phases have a negative driving force. Save the result after each successful calculation. Note that some of the phases you have set entered may become unstable (negative driving force) as you add new phases. Then set all dormant phases back as entered with zero amount, and you have calculated the first equilibrium! **DO NOT FORGET TO SAVE THIS ON A FILE!**

Another case which is more troublesome is calculations with the SLAG database. In this database the gas phase only dissolves O₂, the SLAG phase exists only for high oxygen content and the FE_LIQUID phase exists only for low oxygen content. Thus, both the SLAG and the FE_LIQUID phase must be stable at the initial equilibrium, and the FE_LIQUID phase must be high in Fe. In this case, a special strategy is recommended shown in the examples on the use of the SLAG database (*i.e.*, Example 10 in the *Thermo-Calc Examples Book*).

8.13 Frequently Asked Questions

8.13.1 Why do I only get half of the lines in my diagram?

Answer: You have used the command `SET_DIAGRAM_AXIS X X(<element>)` when you should have used `SET_DIAGRAM_AXIS X MOLE_FRACTION <element>`!

Explanation: Note that MOLE-FRACTION, MOLE-PERCENT, WEIGHT-FRACTION and WEIGHT-PERCENT behave all in the same way. In principle, the explanation is quite simple. When you have “tie-lines in the plane” you must use MOLE-FRACTION as axis variable in the post processor in order to get the composition of both end points of the tie-lines plotted. The plot variable MOLE-FRACTION means that you get the composition of all stable phases plotted. If you use `X(element)` or `W(element)`, you will only plot the overall composition of `<element>` as this is what `X(element)` means.

If you have calculated an isopleth, it does not matter what to use as axis variable as the program must calculate all lines. It is only when you have tie-lines in the plane (as in binary or ternary systems) it matters.

You may play a lot with this. In principle you should use MOLE-FRACTION after a MAP command and `X(element)` after a STEP command. But sometimes you can also use MOLE-FRACTION with STEP, even in cases when the overall composition has been constant! And sometimes you should use `X(element)` or `X(phase, element)` after a MAP command, the latter for example to get the liquid composition along monovariant lines! Thus you should not be rigid about what axis you use in the post processor but try various options. Just remember that MOLE-FRACTION is the same as `X(*, element)` and not `X(element)` because * means all stable phases.

8.13.2 Why can not I plot the diagram after I have SAVED it?

Answer: You should never use the SAVE command after a MAP or STEP because that actually destroys the results of these commands. You should use SAVE before STEP or MAP!

From version J and on, there is a better message explaining what happens when you give a SAVE command on a file that contains results from a MAP or STEP. Results are automatically saved on a file during MAP and STEP. If you afterwards give a SAVE command that will save the current workspaces but destroy any previous results.

8.13.3 Why is G.T not always the same as -S?

Answer: The entropy S is always calculated for constant pressure and composition whereas G.T is calculated for the current conditions.

POLY recognizes a large number of state variables like G, H, S, T, N, MU, AC, etc. You can use them as conditions or obtain the value of any of these variables with a command like:

```
SHOW S
```

This will give the entropy for the whole system which is calculated as the partial derivative of the Gibbs energy with respect to temperature at constant pressure and composition. One may also use normalizing suffices like M for “per mole”, W for “per gram”.

But POLY also provides a possibility to generate your own functions of these state variables, in particular by using the facility to take partial derivatives by using the dot “.” notation. Thus you may obtain the (negative of the) entropy by the command:

```
SHOW G.T
```

One may only have one level of dots. Thus `G.T.T` is not legal but one may instead use `-S.T`.

But do not be surprised if $G \cdot T$ is different from $-S$ because this derivative is calculated for the current conditions. And if you for example has a condition on the activity or a phase fixed, *etc.*, that means that you do not have fixed composition.

8.13.4 How do I get the partial enthalpy for a component?

Answer: You have to enter an expression of state variables as described here.

You should know that the partial Gibbs energy of a component, like Fe, can be obtained by

```
SHOW MUR(FE)
```

You should also know that the partial entropy is the negative of the derivative of this with respect to temperature, *i.e.*, “ $S(FE)$ ” = $-\text{MUR}(FE) \cdot T$. However, you cannot use the command `SHOW S(FE)` as the state variable S can only take a phase name as argument. Thus $S(\text{LIQUID})$ is a legal state variable but not $S(FE)$. The only way to obtain the (negative of the) partial entropy is by the command:

```
SHOW MUR(FE) . T
```

But you wanted the partial enthalpy. The relation between Gibbs energy, enthalpy and entropy is $G = H - TS$. This is valid also for partial quantities. Thus the partial enthalpy of a component $H(FE) = \text{MUR}(FE) + TS(FE)$. The formula for $S(FE)$ was given above and you may thus obtain the partial enthalpy by entering the following commands

```
ENTER VARIABLE HFE = MUR(FE) - T*MUR(FE) . T;
SHOW HFE
```

Note that you cannot give an expression after `SHOW`, it has to be a single state variable or symbol. The reason why you should enter `HFE` as `VARIABLE` rather than `FUNCTION` is that the calculation of the derivative, `MUR(FE) . T`, may have some unwanted side effects.

8.13.5 Why is $H(\text{LIQUID})$ zero but not $HM(\text{LIQUID})$?

Answer: H depends on the amount, HM is normalized to one mole.

The value of H depends on the size of the system or amount of phase. If you have set $N=1$ in a calculation and obtain a value of H , then this value will be twice as large if you just change $N=2$ and calculate again. The same is true when you want the enthalpy of a specific phase rather than the total enthalpy of the system. $H(\text{LIQUID})$ is the enthalpy of the current amount of liquid. If there is no liquid present, or its amount zero, then $H(\text{LIQUID})$ is also zero. If you ask for a normalized amount, HM for example, then this value is independent of the current amount as its value is for one mole of components.

8.13.6 Why is my carbon activity less than unity even if graphite is stable?

Answer: Think about your reference state.

The chemical potential and activity of a component must always have a reference state. In the `LIST_EQUIlibrium` command this state is called “*default*” as the reference state is different in different databases.

In the KP and TCFE databases, each element has a selected phase as reference phase and this phase has zero thermodynamic properties. For carbon, the reference phase is graphite and the activity of graphite will thus be unity at all temperatures.

In the SGTE databases and most other databases, the reference state is a selected phase at 298.15 K and 1 bar. That means that only at 298.15 K and 1 bar the activity will be unity unless you redefine the reference state with the command `SET_REFERENCE_STATE`. This command takes four arguments, the component, the phase, the temperature and the pressure. It is possible to give * as answer to the question of temperature and this indicates that the reference state should be at the current temperature, whatever that is. For example

```
SET REFERENCE STATE C GRAPHITE * 1E5
SHOW ACR(C)
```

Note that in order to get the activity with the reference state you defined you must append a suffix R to the state variable AC. The command `SHOW ACR(C)` will not be changed by the change of reference state and neither the value at `LIST_EQUILIBRIUM`. We have plans to change that for a later version.

8.13.7 How can I obtain the value of the excess Gibbs energy?

Answer: There is no way to obtain the excess Gibbs energy from POLY as this is a model dependent quantity.

You may obtain the Gibbs energy of mixing, and for a substitutional regular solution model that is the same thing. In order to have for example the enthalpy of mixing in the liquid for a ternary system A-B-C give the following commands:

```
SET_REFERENCE_STATE A LIQUID * 1E5
SET_REFERENCE_STATE B LIQUID * 1E5
SET_REFERENCE_STATE C LIQUID * 1E5
SHOW HMR(LIQUID)
```

The suffix R requires that the user defines which reference states should be used. But if you have sublattices with mixing on two or more sublattices the Gibbs energy of mixing is different from the Excess Gibbs energy.

8.13.8 What is wrong when I get crossing tie-lines but no miscibility gap?

Answer: The crossing tie-lines on a phase diagram are signs of a miscibility gap. However, you have to inform Thermo-Calc in advance that you know, or suspect, that there is a miscibility gap in a specific phase. This can be done with the POLY command `ADVANCED_OPTION` (with the `NEW_COMPOSITION_SET` option) or GES command `AMEND_PHASE_DESCRIPTION` (with the `COMPOSITION_SET` option), where you can give a phase two or more composition sets. Then you must start the calculation where the miscibility gap is stable and use the `SET_ALL_START_VARIABLES` command to make both composition sets stable but with different initial compositions.

8.13.9 How can I calculate the maximum of a miscibility gap directly?

Answer: Not with Thermo-Calc.

You can only calculate arbitrarily close. This is different from a congruent transformation which can be calculated directly by setting the composition of the two phases equal.

Note: This FAQ list has not been updated for some time, but is planned to be updated irregularly.
Please promptly visit the TCS web site (www.thermocalc.com) for revised editions.

(This page is intended to be empty)

9 Post-Processor Module (POST)

9.1 Introduction

After a Thermo-Calc calculation or DICTRA simulation, the results are handled in the post-processor, namely, the POST module [✉]. It has been developed to produce various graphical presentations of phase diagrams, property diagrams, diffusion profiles, and many other types of plots a user would expect from the Thermo-Calc calculations or DICTRA simulations. The POST module is usually monitored as a special POLY or DICTRA command rather than the normal way a module is entered (*i.e.*, GOTO <module name>), but sometimes it is interfaced automatically, such as using the TABULATEREACTION or TABULATESUBSTANCE command in the TAB module, or using the DEFINE_DIAGRAM command in the POLY module, or through some advanced modules (*e.g.*, BIN, TERN, POTENTIAL, SCHEIL and POURBAIX).

With this unique module, the user can easily define various diagrams, including their diagram types, axis variables, axis texts, symbols, arrows, and so on. Normally, from a single Thermo-Calc or DICTRA calculation, many diagrams can be generated to present the internal relationships among various quantities of the calculated system. Some specific POLY or DICTRA commands (*e.g.*, DEFINE_DIAGRAM), as well some advanced easy-to-use modules (*e.g.*, BIN, TERN, POTENTIAL, POURBAIX and SCHEIL), will automatically define the first diagram after a MAP or STEP calculation. One can choose any state variable or any entered symbol (functions or variable) as the X/Y-axis. In the POST module, one can also define desired variables, which will be evaluated based on the existing calculation results in the POLY3 or DICTRA workspaces, and select them as X/Y-axis. Since TCCP, it is possible to create vrml (Virtual Reality Modelling Language) graphical files in order to generate 3D plots. The TCCR/TCCQ versions have a very-much improved functionality and flexibility on the **Thermo-Calc Graph** window for further graphical refinement (see *Section 9.3.7*).

The following commands are available in the POST module:

POST:?		
ADD_LABEL_TEXT	PLOT_DIAGRAM	SET_LABEL_CURVE_OPTION
APPEND_EXPERIMENTAL_DATA	PRINT_DIAGRAM	SET_PLOT_FORMAT
BACK	QUICK_EXPERIMENTAL_PLOT	SET_PLOT_OPTIONS
CREATE_3D_PLOTFILE	REINITIATE_PLOT_SETTINGS	SET_PLOT_SIZE
DUMP_DIAGRAM	RESTORE_PHASE_IN_PLOT	SET_PREFIX_SCALING
ENTER_SYMBOL	SET_AXIS_LENGTH	SET_RASTER_STATUS
EXIT	SET_AXIS_PLOT_STATUS	SET_REFERENCE_STATE
FIND_LINE	SET_AXIS_TEXT_STATUS	SET_SCALING_STATUS
HELP	SET_AXIS_TYPE	SET_TIC_TYPE
LIST_DATA_TABLE	SET_COLOR	SET_TIELINE_STATUS
LIST_PLOT_SETTINGS	SET_CORNER_TEXT	SET_TITLE
LIST_SYMBOLS	SET_DIAGRAM_AXIS	SET_TRUE_MANUAL_SCALING
MAKE_EXPERIMENTAL_DATAFILE	SET_DIAGRAM_TYPE	SUSPEND_PHASE_IN_PLOT
MODIFY_LABEL_TEXT	SET_FONT	TABULATE
PATCH_WORKSPACE	SET_INTERACTIVE_MODE	
POST:		

Note that the DUMP_DIAGRAM and PRINT_DIAGRAM commands are only valid under PC Windows environments.

When a diagram has been plotted, varied appearance parameters/options for defining/refining a high-standard graph can be specified/used, *e.g.*, curve-labeling options, phase-region labeling, property-line identifying, 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

[✉] Revision History of the POST Module User's Guide:

Jan 1989	First release (Edited by Lars Höglund)
Oct 1993	Second revised release (Edited by Lars Höglund)
Oct 1996	Third revised release (Edited by Lars Höglund)
Jun 2000	Fourth revised and extended release (Edited by Lars Höglund and Pingfang Shi)
Nov 2002	Fifth revised release (Edited by Lars Höglund and Pingfang Shi)
Jun 2004	Sixth revised release (Edited by Lars Höglund and Pingfang Shi)
May 2006	Seventh revised release (Edited by Lars Höglund and Pingfang Shi); modified in May 2008

formats, text fonts, colors, raster plots, symbols, arrows, *etc.* Calculated results (from stepping calculations) can also be directly tabulated (as to simple textual files or MS Excel spreadsheet files) in the POST module. A user can easily append experimental data onto a plotted phase diagram or property diagram. One can also save the coordinates onto a textual file, which can be edited and used as an experimental file to be merged onto another diagram or as a part of a `setup` file for PARROT-module assessments. Phases can be suspended or restored in a plotted diagram. The reference states for components can also be modified for the resulted diagrams.

Please note that since TCCS/TCW5 the `IDGMAR` dimension in the POST workspace has been increased from 500 to 2000, so that one can now plot extremely complex diagrams with many phase boundaries.

9.2 General Commands

9.2.1 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

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

9.2.2 BACK

Description: This command gives the control back to the most recent module. From the POST module, BACK always goes back to the POLY module. However, if the POST module has been called by the TAB module, it returns to the TAB module.

Synopsis: BACK

9.2.3 EXIT

Description: This command terminates the program and returns to the operating system. Unless a SAVE command has been executed (in either the GES, POLY3 or PARROT module), all data and results will be lost.

Synopsis: EXIT

9.3 Important Commands

9.3.1 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 Thermo-Calc, 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, then the composition axis must be “mole-fraction”, “weight-percent” or the other two combinations of these. You must not use the stable variable “ $x(c)$ ” 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 “ $x(*, c)$ ”.

When you plot the result of a STEP calculation and want a composition axis variable, you should normally use “ $x(c)$ ” and not “mole-fraction”.

Also when you plot a phase diagram where the tie-lines are not in the plane, “mole-fraction” and “ $x(c)$ ” are identical.

Synopsis 1: SET_DIAGRAM_AXIS <axis name> <variable name & specification>

Synopsis 2: SET_DIAGRAM_AXIS

Ensuing 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:

- Basic state variables:

TEMPERATURE-CELSIUS	as temperature in °C
TEMPERATURE	as temperature in K
PRESSURE	as pressure in Pa
ACTIVITY	for a component
LNACTIVITY	for a component [$\ln(\text{ACR}(\text{component}))$]
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 such as $x(*, \text{component})$. If a wildcard is used, the program will also ask for columns.
- Any entered function or variable.
- Any entered table. In this case, a column or column range will also be asked for.

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.

Examples:

1	column 1
2, 3	columns 2 and 3
2, 3 >5	column 2,3 and all columns above column 5
*	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.

Special Notes: 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 calculation. The facility to set any set of state variables as diagram axes after a 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 $x(z)$, $x(*, z)$ and “mole-fraction of z ”.

If the state variable $x(z)$ has been used in MAP, then mole-fraction of the z 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.

9.3.2 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. For more information, see *Section 9.7*.

Synopsis: SET_DIAGRAM_TYPE

Ensuing 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 3:rd 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.

9.3.3 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.

Synopsis 1: SET_LABEL_CURVE_OPTION <curve option>

Synopsis 2: SET_LABEL_CURVE_OPTION

Ensuing 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. Both E and F provide varied colors for different curves. The option N (NONE) will disable all curve labels and lists.

If, for example, one has used T as the variable axis in a STEP command and then plots the amount of 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.

9.3.4 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.

Since TCCQ, 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 - graph** window, are stored in the workspace, and they may later be saved onto an EXP file (through the MAKE_EXPERIMENTAL_DATAFILE command).

Synopsis: ADD_LABEL_TEXT

Ensuing 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.

Text size: /.44/: <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 .44 if the command is called for the first time.

9.3.5 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.

Since TCCQ, 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 - graph** window, are stored in the workspace, and they may later be saved onto an EXP file (through the MAKE_EXPERIMENTAL_DATAFILE command).

Synopsis: MODIFY_LABEL_TEXT

Ensuing 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.

9.3.6 SET_PLOT_FORMAT

Description: With this command the user can set the format of the graphical output to different graphical devices. The default device is normally for screen output (remembered with reverence!), *e.g.*, X-Window numbered as 9 for UNIX or PC Linux, and MS-Windows numbered as 1 for PC Windows. This default device value may be altered with the SET_PLOT_ENVIRONMENT command in the system monitor or by your TC.INI file.

Since TCCR, under PC Windows NT4/2000/XP/Vista environments, an editable graphical format, *i.e.*, the EMF (Enhanced Windows Metafile) format has been made available [its corresponding device number is 19]. A diagram plotted in this format (*i.e.*, saved as a graphical file with the default extension of EMF) can be edited and manipulated using an appropriate graphical editor (such as in MS PowerPoint Editor).

Since TCCS, under PC Windows NT4/2000/XP/Vista environments, another PostScript graphical format for inclusions in LaTeX documents has been added [its corresponding device number is 20]. A diagram plotted in this format (*i.e.*, saved as a graphical file with the default extension of ps) can be directly included in LaTeX documents.

Synopsis: SET_PLOT_FORMAT

Ensuing 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 UNIX and Linux (Note that the last column will not be shown up; these are given here only for user's reference to indicate the default file extensions when a graphic file is saved later on by the PLOT_DIAGRAM command):

GRAPHIC DEVICE NUMBER /9/ : ?	
AVAILABLE GRAPHIC DEVICES:	
DEVICE 1 is Tektronix 4010	p1
DEVICE 2 is VT240/Kermit Tek.4010 emulation	p2
DEVICE 3 is COMPIS Tek.4010 emulation	p3
DEVICE 4 is Regis graphics	p4
DEVICE 5 is Postscript portrait mode	ps
DEVICE 6 is Postscript landscape mode	ps
DEVICE 7 is HPGL plotter (HP7475 landscape A4)	p7
DEVICE 8 is HPGL plotter (HP7475 portrait A4)	p8
DEVICE 9 is X-Windows	p9
DEVICE 10 is Tandberg 2200/9S, Tek.4010	p10
DEVICE 11 is HP-LaserJet II / HP-DeskJet (1Mb)	p11
DEVICE 12 is Tektronix 4105	p12
DEVICE 13 is Tektronix 4107	p13
DEVICE 14 is HP-LaserJet III & IV, portrait (2Mb)	p14
DEVICE 15 is HP-LaserJet III & IV, landscape (2Mb)	p15
DEVICE 16 is Color Postscript portrait mode	p16
DEVICE 17 is Color Postscript landscape mode	p17
DEVICE 18 is HP-LaserJet II / HP-DeskJet (1Mb) large	p18

For PC Windows XP/2000/NT4 (Note that the last column will not be shown up; these are given here only for user's reference to indicate the default file extensions when a graphic file is saved later on by the PLOT_DIAGRAM command):

GRAPHIC DEVICE NUMBER /1/ : ?	
AVAILABLE GRAPHIC DEVICES:	
Device 1 is MS-Windows	P1
Device 2 is VT240/Kermit Tek.4010 emulation	P2
Device 3 is COMPIS Tek.4010 emulation	P3
Device 4 is Regis graphics	P4
Device 5 is Postscript portrait mode	ps
Device 6 is Postscript landscape mode	ps
Device 7 is HPGL plotter (HP7475 landscape A4)	P7
Device 8 is HPGL plotter (HP7475 portrait A4)	P8
Device 9 is Tektronix 4010	P9

Device 10 is Tandberg 2200/9S, Tek.4010	P10
Device 11 is HP-LaserJet II / HP-DeskJet (1Mb)	P11
Device 12 is Tektronix 4105	P12
Device 13 is Tektronix 4107	P13
Device 14 is HP-LaserJet III & IV, portrait (2Mb)	P14
Device 15 is HP-LaserJet III & IV, landscape (2Mb)	P15
Device 16 is Color Postscript portrait mode	P16
Device 17 is Color Postscript landscape mode	P17
Device 18 is HP-LaserJet II / HP-DeskJet (1Mb) large	P18
Device 19 is EMF	EMF
Device 20 is Postscript for include in LaTeX	ps

For some formats (*e.g.*, PostScript, HP-LaserJet), 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.

The graphical device number 19 [*i.e.*, the EMF (Enhanced Windows Metafile) format]; has been made available since TCCR, but only under PC Windows NT4/2000/XP/Vista environment. A diagram plotted in this format (*i.e.*, saved as a graphical file with the default extension of EMF) can be edited and manipulated using an appropriate graphical editor (such as in MS PowerPoint Editor).

USE HARD COPY FONTS /YES/: <Y or N>

Under Windows NT4/2000/XP/Vista, if a non-screen format has already been set on (*e.g.*, a PostScript font for a hard copy of any previous graph) and if the MS-Windows format (default as device 1), this will prompt for a decision on if using the hard copy fonts for various texts as plotting on screen. Please note that some hard copy fonts may look strange for the sequentially plotted graphs on screen.

USE POSTSCRIPT FONTS /YES/: <Y or N>

If a non-screen format is be set on, this will prompt for a decision on if using PostScript fonts on the plotted graphs in hard copies. Further prompts will appear for choosing the font number and font size.

SELECT FONTPRINT /9/:

Specify an valid font number, or by giving a question mark "?" a list of all available fonts for the specified device will be listed on screen.

Note that the available fonts for various formats are normally different. As an example, the following listed fonts are valid for the PostScript format (device 5 or 6):

```
USE POSTSCRIPT FONTS /YES/: y
SELECT FONTPRINT /9/: ?

AVAILABLE FONTS:
FONT 1 IS AvantGarde-Book
FONT 2 IS AvantGarde-BookOblique
FONT 3 IS AvantGarde-Demi
FONT 4 IS AvantGarde-DemiOblique
FONT 5 IS Courier
FONT 6 IS Courier-Oblique
FONT 7 IS Courier-Bold
FONT 8 IS Courier-BoldOblique
FONT 9 IS Helvetica
FONT 10 IS Helvetica-Oblique
FONT 11 IS Helvetica-Bold
FONT 12 IS Helvetica-BoldOblique
FONT 13 IS LubalinGraph-Book
FONT 14 IS LubalinGraph-BookOblique
FONT 15 IS LubalinGraph-Demi
FONT 16 IS LubalinGraph-DemiOblique
FONT 17 IS NewCenturySchlbk-Roman
FONT 18 IS NewCenturySchlbk-Italic
FONT 19 IS NewCenturySchlbk-Bold
FONT 20 IS NewCenturySchlbk-BoldItalic
FONT 21 IS Souvenir-Light
FONT 22 IS Souvenir-LightItalic
FONT 23 IS Souvenir-Demi
```

```

FONT 24 IS Souvenir-DemiItalic
FONT 25 IS Times-Roman
FONT 26 IS Times-Italic
FONT 27 IS Times-Bold
FONT 28 IS Times-BoldItalic
FONT 29 IS Symbol
FONT 30 IS Helvetica-Narrow
FONT 31 IS Helvetica-Narrow-Bold
FONT 32 IS Helvetica-Narrow-Oblique
FONT 33 IS Helvetica-Narrow-BoldOblique
FONT 34 IS Bookman-Demi
FONT 35 IS Bookman-DemiItalic
FONT 36 IS Bookman-Light
FONT 37 IS Bookman-LightItalic
FONT 38 IS Palatino-Roman
FONT 39 IS Palatino-Bold
FONT 40 IS Palatino-Italic
FONT 41 IS Palatino-BoldItalic
FONT 42 IS ZapfChancery-MediumItalic
FONT 43 IS ZapfDingbats

```

**FONT SIZE / .35833/: **
Specify the relative size for the specified font.

Example:

```

SET_PLOT_FORMAT
CURRENT DEVICE: X-windows
GRAPHIC DEVICE NUMBER /9/: 5
NEW DEVICE: Postscript portrait mode
USE POSTSCRIPT FONTS /YES/: y
SELECT FONTPNUMBER /9/: 27
NEW FONT: Times-Bold
FONT SIZE / .35833/: 0.32

```

9.3.7 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!

Since TCCN, the functionality of this command has been split up into two:

- one remained in the **PLOT_DIAGRAM** command for normal screen showing and file saving; and
- the other implemented in the **PRINT_DIAGRAM** command (under PC Windows environment) for direct hard-copy printing (see *Section 9.3.7*).

Therefore, with this command, a diagram can be shown on screen (if using the default graphic device), or be saved on a desired file. However, when the diagram is plotted on screen under Windows NT4/2000/XP/Vista, one can also directly print the graph to a connected printer and save the graph as an EMF file which can also be easily inserted into a document (see below).

Synopsis 1: **PLOT_DIAGRAM <name of a file or RETURN for SCREEN>**

Synopsis 2: **PLOT_DIAGRAM**

Ensuing Prompt: **OUTPUT TO SCREEN OR FILE /SCREEN/: <name of a file or RETURN>**

Specify the name of the desired file (Note that under Linux/UNIX, 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 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 extension (*i.e.*, “`P1`” for MS-Windows on Windows, or “`p9`” for X-Windows on UNIX/Linux) will be automatically given.

9.3.8 Thermo-Calc Graph Window

This is not a POST command; the idea of having this special section is to emphasize on various features of the Thermo-Calc Graph Window which is automatically opened on screen after conducting the PLOT_DIAGRAM command in the POST module.

Important Notes: For the PC Windows NT4/2000/XP/Vista versions of the TCC software, there is an additional and very helpful feature to assist the users to export the calculated Thermo-Calc graph to printers for hard copies or to documents. If the diagram has been chosen to be plotted using the default graphic format (*i.e.*, `1` for `IBM-vga`, by pressing <RETURN> as in the `SET_PLOT_FORMAT` command, or the `SET_PLOT_FORMAT` command has not been called at all), a *Thermo-Calc - graph* window will pop up on screen. On the upper-left corner of this window, three interactive bar (“**Menu**”, “**Settings**” and “**Layers**”) and several function buttons can be activated or disabled by clicking (as illustrated in *Figure 9.1*).

Since TCCP, it can also be possible to *directly* add, edit and/or delete various curve-labeling texts, phase-region-labeling texts, diagram-identifying texts, diagram titles, subtitles, special notes, axis-variable texts, symbols, arrows, or any other types, on the plotted diagrams on the *Thermo-Calc - graph* window (as demonstrated in *Figures 9.2 through 9.7*). While moving the mouse cursor on the plotted diagram, the current X/Y-coordinates (in terms of currently plotted X/Y-axis variables) of the cursor are promptly shown at the left-bottom corner of the *Thermo-Calc - graph* window, which are useful for deciding where to adding labeling texts/symbols on the plotted diagram. By editing such texts, symbols and arrows, their sizes, fonts, colors, styles, coordinates, rotation angles, *etc.*, can be easily changed, or simply deleted from the plotted diagrams. Manually added texts, symbols or arrows may also appear on diagrams that are subsequently plotted.

Since TCCQ, labeling texts and their coordinates, either added by the `ADD_LABEL_TEXT` command or modified by the `MODIFY_LABEL_TEXT` command or manually edited on the *Thermo-Calc - graph* window, are stored in the workspace, and they may later be saved onto an `EXP` file (through the `MAKE_EXPERIMENTAL_DATAFILE` command). Moreover, on the *Thermo-Calc - graph* window since TCCQ, all kinds of curve/region labeling texts/symbols, diagram identifying texts/symbols and axis-variable texts/symbols can be manually moved, directly edited and easily deleted, as well.

Since TCCR, by just simply moving the mouse on a plotted diagram in the *Thermo-Calc - graph* window, it is very easy to identify various standard-variable property curves (on a property diagram after a stepping calculation) or stable-phase regions (on a phase diagram after a mapping calculation), as illustrated in *Figures 9.9 and 9.10*. It is also possible to directly and automatically write the stable phase-assemblages onto a plotted phase diagram, by moving the mouse into a phase region followed by right-clicking the mouse and then choosing the “Add Label” option, as shown in *Figure 9.11*. Furthermore, using the mouse-right-button, one can add any text item on the plotted diagram, print the diagram to a connected printer, save the diagram as an EMF graphical file, or copy the EMF code of the diagram into clipboard (as shown in *Figure 9.8*).

Since TCCR, under PC Windows NT4/2000/XP/Vista environment, a graphical file can be plotted in the EMF format [after having specified the plot-format as `19` (for EMF) using the `SET_PLOT_FORMAT` command]; remember that it is needed to specify the file name with the correct extension, *e.g.*, `MyGraph3.EMF`. Such a plotted EMF file can be edited and manipulated using an appropriate graphical editor (such as in MS PowerPoint).

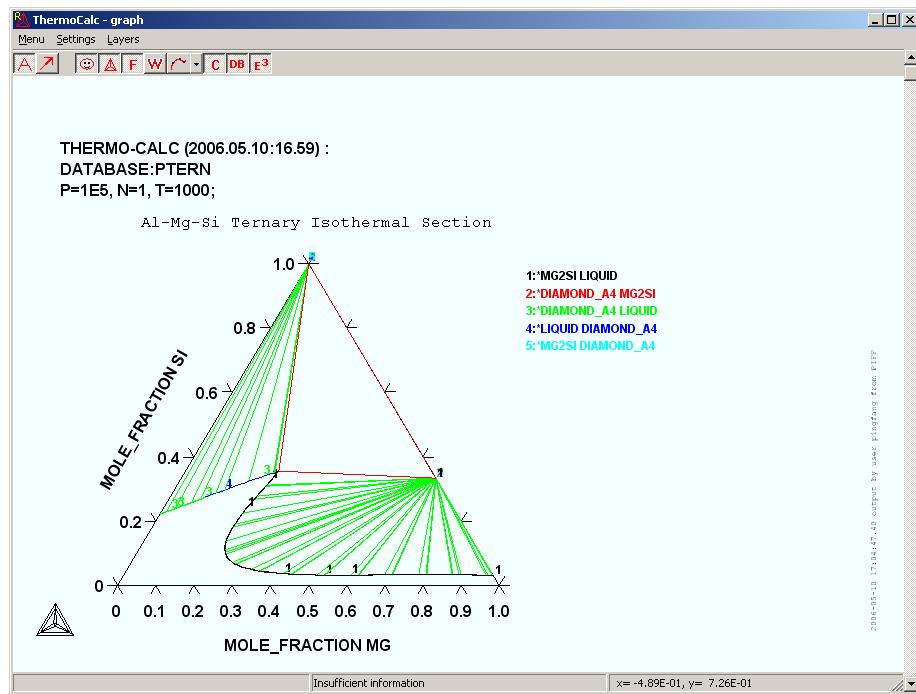
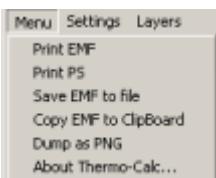


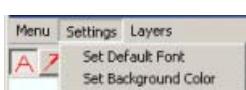
Figure 9.1. The “Thermo-Calc - graph” window: Options for graphical outputs.

From the **Menu** bar, one chose one of the following four options:



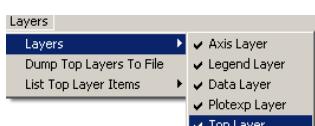
- Print EMF to directly print the graph (in EMF format/quality) on a connected printer.
- Print PS to directly print the graph (in PostScript format/quality) on a connected printer.
- Save EMF to file to save the EMF (Enhanced Windows Metafile) graphical code onto a desired file specified in the “Save Metafile As” window (as shown in *Figure 9.2*), which can later be inserted as a Picture File into any document using some editing application software (e.g., Microsoft Word, PowerPoint, Excel, CorelDraw, Adobe Illustrator, etc.).
- Copy EMF to Clipboard to copy the EMF graphical code to the Windows Clipboard that can be directly pasted into any document using some editing application software.
- Dump as PNG to dump the PNG graphical code onto a desired PNG file specified in the “Save As” window. This option is actually the short-cut of the DUMP_DIAGRAM command with the PNG option; see details in the DUMP_DIAGRAM command.
- About Thermo-Calc Graph to view the description about the “Thermo-Calc Graph” window.

From the **Settings** bar, one chose one of the following two options:



- Set Default Font to globally alternate the font for all texts plotted on diagram, by selecting the desired one in the “Font” window (as shown in *Figure 9.3*),
- Set Background Color to add/refine the background color for the plotted diagram by specifying the desired color on the “Color” window (as shown in *Figure 9.4*).

From the **Layers** bar, one chose one of the following three options and their sub-options:



- Layers to decide on which layers (axis, legend, data, plotexp and/or top layers) to be plotted on diagram,
- Dump Top Layer to File to choose to dump the top layer of the plotted diagram to a file,
- List Top Layer Items to list out top layer items.

Various function buttons can be activated or disabled by clicking, as described below:



- | | | |
|--|------------------------------|--|
| | Add/Edit Text Items: | For manually and directly adding and editing user's desired labeling/identifying texts (and their appearance settings further specified in an "Add Text to Diagram" window as shown in <i>Figure 9.5</i>). |
| | Add Arrow Items: | For manually and directly adding user's desired pointing arrows. For editing the appearance settings of an added arrow, you can first check on the button and then click on the arrow to open "Edit Arrow" window (as shown in <i>Figure 9.6</i>) in order to further modify/adjust it. |
| | Plot Header: | For activating or disabling the header plotting. |
| | Plot Logo: | For activating or disabling the Thermo-Calc logo plotting. |
| | Plot Footer: | For activating or disabling the footer plotting. |
| | Write Contoured PS Text: | For activating or disabling the writing of contoured PostScript text. |
| | Plot Symbols at Node Points: | For activating or disabling the symbol plotting at node points. |
| | Set Symbol Number & Size: | For setting symbol numbers/types and sizes in the "Set Symbol Number and Size" window (as shown in <i>Figure 9.7</i>) for symbols to be plotted at node points. |
| | Condition Texts: | For activating or disabling the condition plotting. |
| | Database Name: | For activating or disabling the database-name plotting. |
| | Remote Exponent: | For activating or disabling the remote-exponent plotting. |

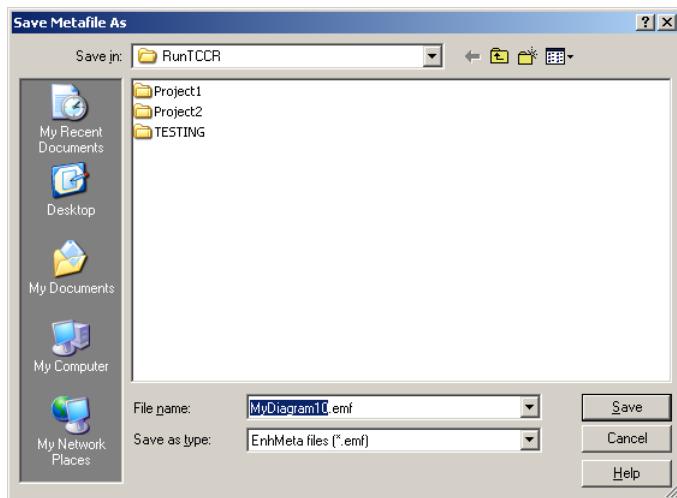


Figure 9.2. The “Save Metafile As” window: Save plotted diagram as an EMF file.

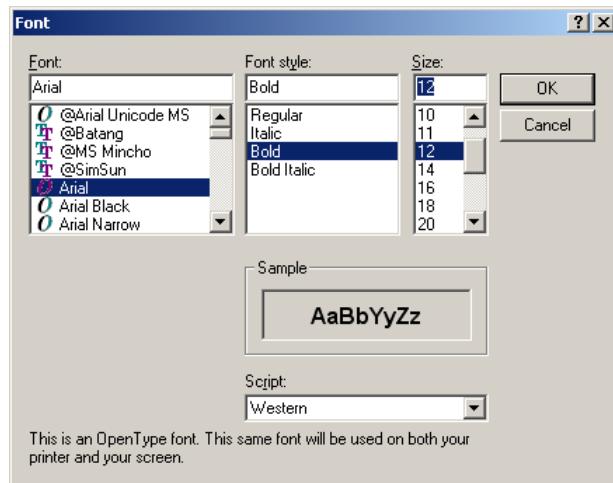


Figure 9.3. The “Font” window: Globally change font for plotted diagram.

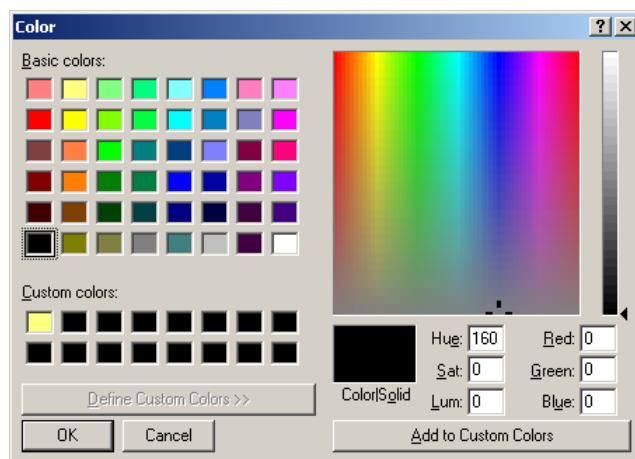


Figure 9.4. The “Color” window: Adjust color for background and other purposes.

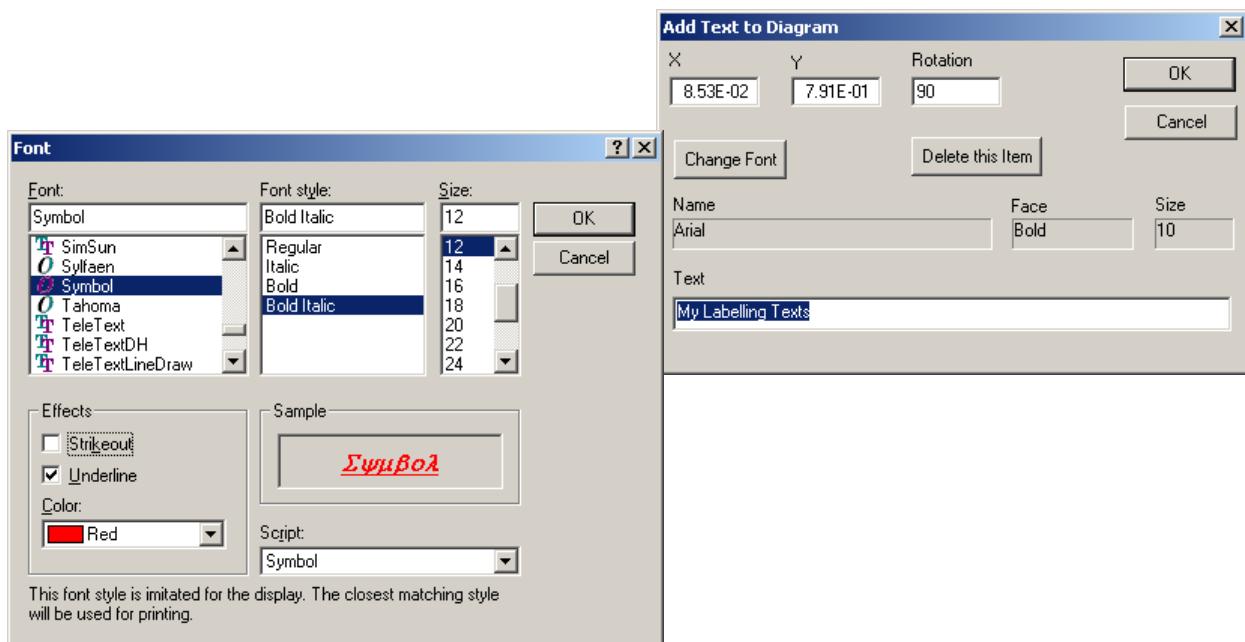


Figure 9.5. The “Add Text to Diagram” window and its sub-window “Font”: Easily and flexibly edit desired labeling/identifying texts.

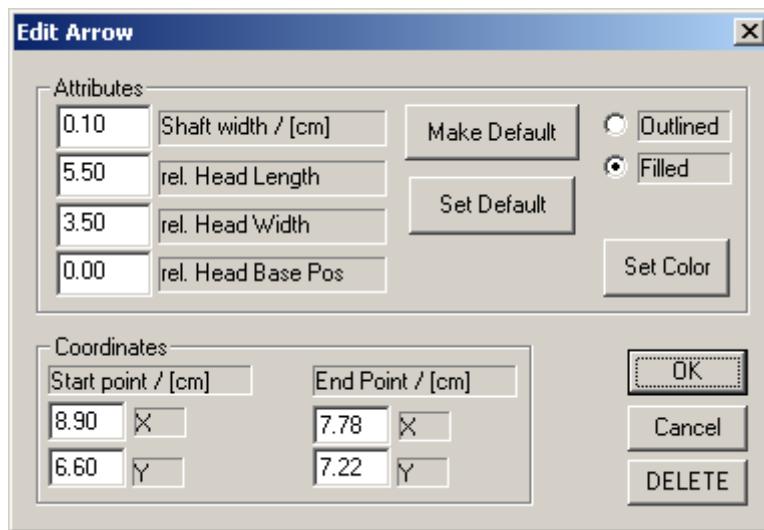


Figure 9.6. The “Edit Arrow” window: Edit an arrow appearance (for color change it opens a sub-window “Color” as shown in Figure 9-4).

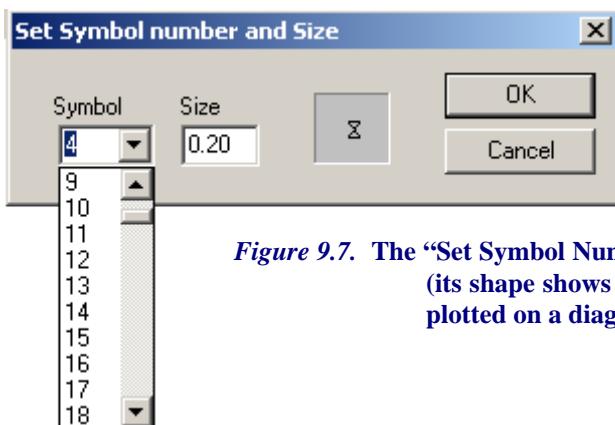


Figure 9.7. The “Set Symbol Number and Size” window: Choose a desired symbol (its shape shows up once the number is selected) and its size to be plotted on a diagram.

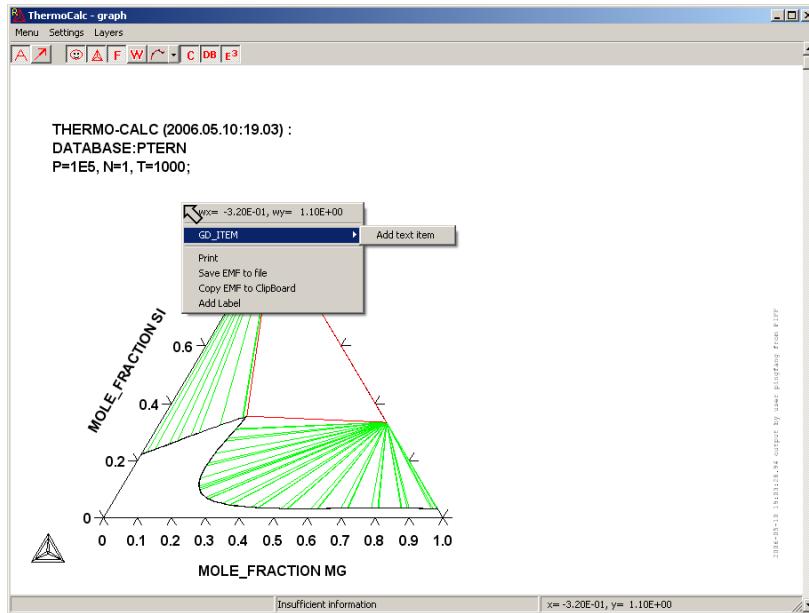


Figure 9.8. The mouse-controlled editing function of the *Thermo-Calc - graph* window: For editing a user-specified text or symbol. Clicking the right-mouse-button and then choosing the “Add text item” option will pop up a “Add Text to Diagram” window to adding/editing the desired text and font/style/size, etc. (see *Figure 9.5*).

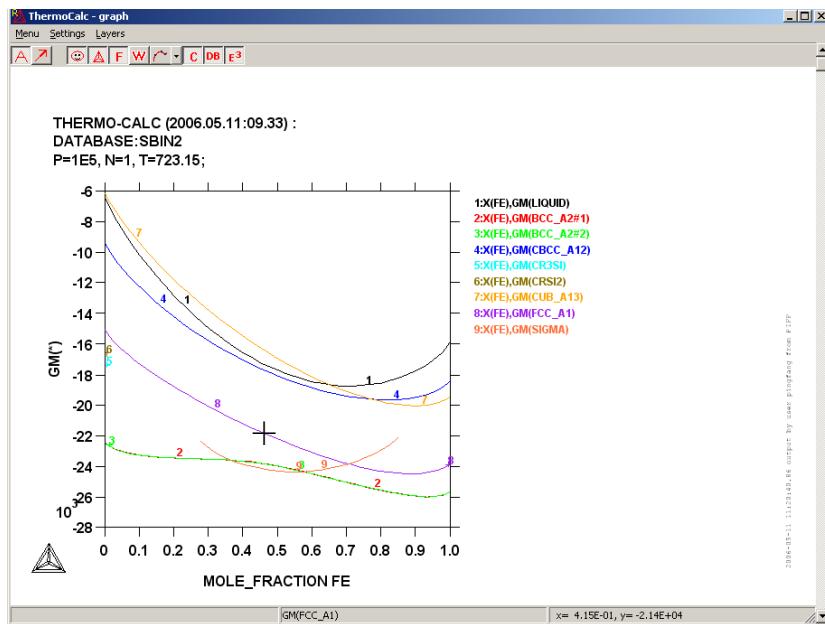


Figure 9.9. The mouse-controlled editing function of the *Thermo-Calc - graph* window: For identifying a property curve on a plotted property diagram. The identified property at the mouse position [in this case, GM(FCC_A1)] is shown inside the middle field at the bottom of the window.

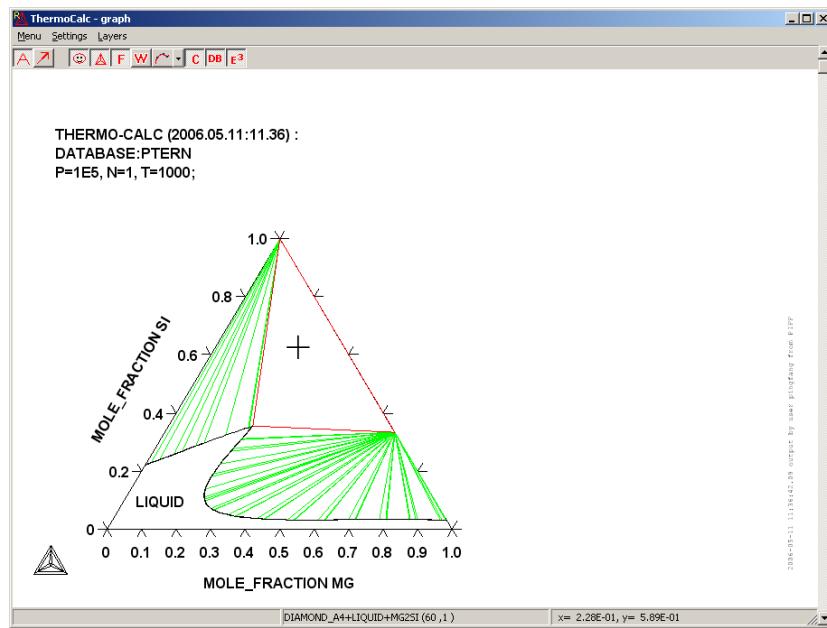


Figure 9.10. The mouse-controlled editing function of the *Thermo-Calc - graph* window: For identifying a phase region on a plotted phase diagram. The identified stable phase-assemblage in the phase region at the mouse position [in this case, DIAMOND_A4+LIQUID+MG2SI] is shown inside the middle field at the bottom of the window.

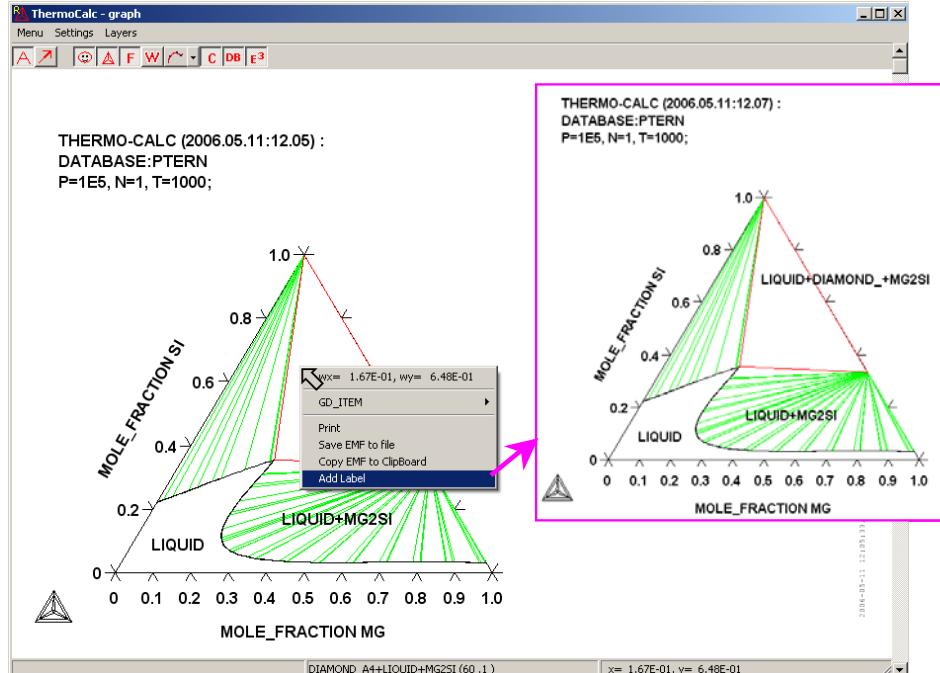


Figure 9.11. The mouse-controlled editing function of the *Thermo-Calc - graph* window: For identifying and labeling a phase region on a plotted phase diagram. Clicking the mouse-right-button and then choosing the “Add Label” option will write the phase-assemblage that is stable in the plotted phase region in which the mouse pointer locates.

9.3.9 PRINT_DIAGRAM

Description: This is a very useful command for immediate printing with high quality, but it is only available for TCC under Windows NT4/2000/XP/Vista environments. 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 newly implemented command.

Synopsis: PRINT_DIAGRAM

Ensuing Window: After this command, a **Print** window (as illustrated in *Figure 9.12*) is popped up, in which the user can choose his preferred connected printer around and if desired the properties for the printing process can be altered.

As confirming the selected printer and property by clicking the **OK** button on the **Print** window, another window, the **Ghostscript** window (as shown in the *Figure 9.13*), will pop up for a short period, which shows the proceeding printing percentage. In this small window, one can cancel the printing if really wanted; otherwise, the window will disappear immediately after the printing is over.

Important Note: It does not matter what graphic device has been set up for the graph, this command will always be able to convert it to the Ghostscript-compatible graphic format (*i.e.*, Postscript portrait mode, or Color Postscript portrait mode) and a pre-selected Postscript-related font (*e.g.*, Helvetica or Times-Bold), through which the graph is printed on a connected printer.

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.

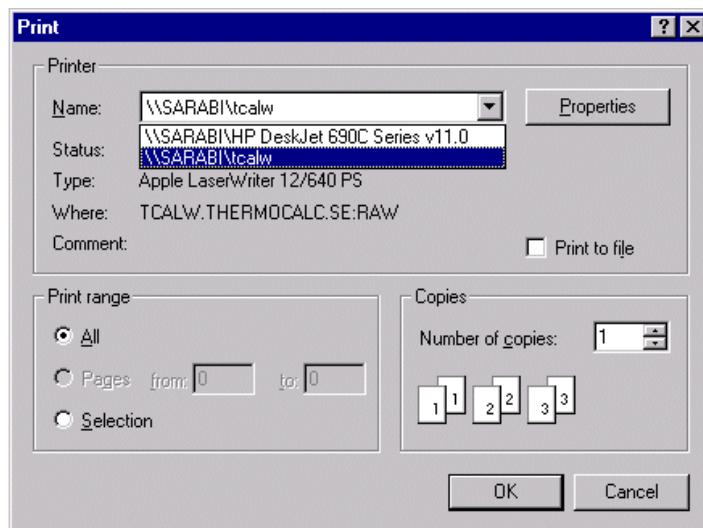


Figure 9.12. The “Print” window: Choices on connected printers and properties.

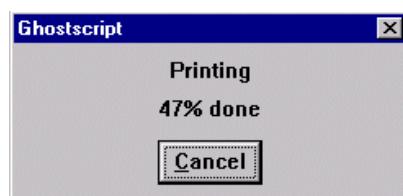


Figure 9.13. The “Ghostscript” window: Proceeding printing.

9.3.10 DUMP_DIAGRAM

Description: This is also a very useful command, but for quick printing with average quality. 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 newly implemented command to save (dump) the diagram to a file with the available graphical formats list in *Table 9.1*.

Table 9.1. List of major graphic formats used for dumping a Thermo-Calc graph

Graphic Format	Default File Extension	Suitable Editor(s)
PNG (Portable Network Graphics)	.PNG	MS Photo Editor
BMP (Windows Bitmap)	.BMP	MS Photo Editor
PDF (Adobe Portable Document File)	.PDF	Acrobat Reader, CorelDraw
JPEG (File Interchange Format)	.JPG	MS Photo Editor
TIFF (Tagged Image File Format)	.TIF	MS Photo Editor

As shown in the table, the saved file (*e.g.*, PNG format) can then be opened and edited by *e.g.* Microsoft Photo Editor, Adobe Reader or CorelDraw, from which the graph can be directly printed the diagram on a connected printer. With such graphical editors, the Thermo-Calc graph can be cut and pasted into another word/graphical processing documents (*e.g.*, Microsoft Word or PowerPoint files).

Note that this command is only available for TCC under Windows NT4/2000/XP/Vista environments, while it has been disabled since TCCN under PC Linux and UNIX platforms.

Synopsis: DUMP_DIAGRAM

Ensuing Prompt: OUTPUT FORMAT (PNG,BMP,PDF,JPEG,TIFF) /PNG16/: <format or RETURN>
Specify an appropriate graphic format for dumping the graph or pressing <RETURN> to accept the default format (PNG16).

RESOLUTION (LOW,MEDIUM,HIGH) /MEDIUM/: <resolution choice>
One of the resolution levels should be specified. Medium level is default.

FILE NAME: <name of a graphic file>
A **Save As** window as shown in *Figure 9.14* will be popped up instead. 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 **Save as type** box is the proper one for the graphic format selected in the previous prompt.

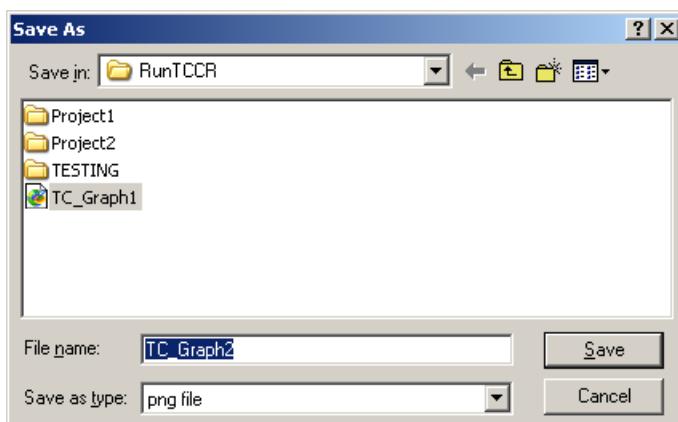


Figure 9.14. The “Save As” window: Saving the dumped graphical file.

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.

9.3.11 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.

Synopsis 1: SET_SCALING_STATUS <name of an axis> <Y or N> <min> <max>

Synopsis 2: SET_SCALING_STATUS

Ensuing 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.

9.3.12 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 TCC.

Synopsis 1: SET_TITLE <title>

Synopsis 2: SET_TITLE

Ensuing 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.

9.3.13 LIST_PLOT_SETTINGS

Description: The present values of most parameters specifying the type of diagram to be plotted will be listed on the terminal.

Synopsis: LIST_PLOT_SETTINGS

9.4 Experimental Data File Plot Commands

9.4.1 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 program according to syntax of a graphical language, *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 may be created by an ordinary textual editor (such as NotePad, PFE, etc.).

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**.

Synopsis: APPEND_EXPERIMENTAL_DATA

Ensuing 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.

EXPERIMENTAL DATAFILE: <name of an experimental datafile>

Specify the name of the file with the experimental data; this prompt is only valid under Linux and UNIX platforms. The default file extension is “exp”.

Under Windows NT4/2000/XP/Vista environments, an **Open file** window will pop up on the screen instead, 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 9.15*.

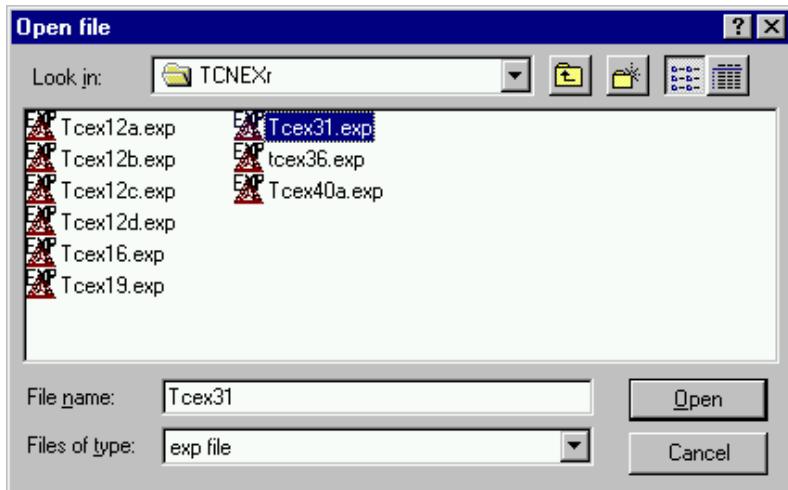


Figure 9.15. The “Open file” window: Opening an existing *.EXP file.

The file type (*i.e.*, EXP, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with reading the experimental data file into PARROT workspace. The user may also cancel such an **Open file** window session, and thus the current experimental data file will not be opened.

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.

9.4.2 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 format (see the document [DATAPLOT Graphical Language](#)). 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 (such as NotePad, PFE, etc.).

Synopsis 1: MAKE_EXPERIMENTAL_DATAFILE <name of a file or RETURN>

Synopsis 2: MAKE_EXPERIMENTAL_DATAFILE

Ensuing 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 PC Windows) or “exp” (under UNIX and Linux).

9.4.3 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.

See the document [DATAPLOT Graphical Language](#) for more information about the format of an experimental datafile.

Synopsis 1: QUICK_EXPERIMENTAL_PLOT <name of an experimental data file>

Synopsis 2: QUICK_EXPERIMENTAL_PLOT

Ensuing Prompt: EXPERIMENTAL DATAFILE: <name of an experimental data file>
Specify the name of the file with the experimental data; this prompt is only valid under Linux and UNIX platforms. The default file extension is “exp”.

Under Windows NT4/2000/XP/Vista environments, an **Open file** window (which is the same as in the APPEND_EXPERIMENTAL_DATA command) will pop up on the screen instead, 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 9.5*.

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.

9.5 Other Commands

9.5.1 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.

Synopsis: ENTER_SYMBOL

Ensuing 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 “=”, for instance, 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)
H.T/4.184

The Gibbs energy of liquid per mole component
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.

9.5.2 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 “%”.

Synopsis: LIST_SYMBOLS

9.5.3 SET_AXIS_LENGTH

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.

Synopsis: SET_AXIS_LENGTH

Ensuing 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.

9.5.4 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.

Synopsis: SET_AXIS_TEXT_STATUS

Ensuing 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.

9.5.5 SET_AXIS_TYPE

Description: This command is to change the axis type among linear, logarithmic and inverse axis.

Synopsis: SET_AXIS_TYPE

Ensuing 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 LINEar (default), LOGarithmic or INVerse. Only the three first characters are relevant.

9.5.6 SET_COLOR

Description: Some devices that support colors (e.g., MS-Windows) or line thickness (e.g., X-Windows or HP-LaserJet). 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.

Synopsis: SET_COLOR

Ensuing 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
- RED
- GREEN
- 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
VERY_THICK
THIN_
THICK
VERY_THIN
DASHED
DOTTED
```

The Default option for either Color or LineType are listed below:

<i>Text/Line</i>	<i>Color</i>	<i>LineType</i>
Text and axis	FOREGROUND	NORMAL
Invariant equilibria	RED	VERY_THICK
Tie-line	GREEN	THIN
All other lines	FOREGROUND	NORMAL

9.5.7 SET_CORNER_TEXT

Description: This command writes texts in the corners on a plotted diagram. Normally, one can write such texts as a subtitle.

Synopsis: SET_CONER_TEXT

Ensuing 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: As a new feature since TCCP, 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.

9.5.8 SET_FONT

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 (*e.g.*, PostScript), there may be other fonts available and these are selected by the SET_PLOT_FORMAT command.

Synopsis: SET_FONT

Ensuing 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.

Under Linux/UNIX platforms, the following 10 fonts are available:

```
FONT 1 IS Cartographic Roman
FONT 2 IS Bold Roman Script
FONT 3 IS Bold Roman
FONT 4 IS Bold Italic
FONT 5 IS Script
FONT 6 IS Bold Script
FONT 7 IS UNCIAL
FONT 8 IS Bold Greek
FONT 9 IS Gothic English
FONT 10 IS Gothic Greek
```

Under Windows NT4/2000/XP/Vista environments, the following 43 fonts are available:

```
FONT 1 IS AvantGarde-Book
FONT 2 IS AvantGarde-BookOblique
FONT 3 IS AvantGarde-Demi
FONT 4 IS AvantGarde-DemiOblique
FONT 5 IS Courier
FONT 6 IS Courier-Oblique
FONT 7 IS Courier-Bold
FONT 8 IS Courier-BoldOblique
FONT 9 IS Helvetica
FONT 10 IS Helvetica-Oblique
FONT 11 IS Helvetica-Bold
FONT 12 IS Helvetica-BoldOblique
FONT 13 IS LubalinGraph-Book
FONT 14 IS LubalinGraph-BookOblique
FONT 15 IS LubalinGraph-Demi
FONT 16 IS LubalinGraph-DemiOblique
FONT 17 IS NewCenturySchlbk-Roman
FONT 18 IS NewCenturySchlbk-Italic
FONT 19 IS NewCenturySchlbk-Bold
FONT 20 IS NewCenturySchlbk-BoldItalic
FONT 21 IS Souvenir-Light
FONT 22 IS Souvenir-LightItalic
FONT 23 IS Souvenir-Demi
FONT 24 IS Souvenir-DemiItalic
FONT 25 IS Times-Roman
FONT 26 IS Times-Italic
FONT 27 IS Times-Bold
FONT 28 IS Times-BoldItalic
FONT 29 IS Symbol
FONT 30 IS Helvetica-Narrow
FONT 31 IS Helvetica-Narrow-Bold
FONT 32 IS Helvetica-Narrow-Oblique
FONT 33 IS Helvetica-Narrow-BoldOblique
FONT 34 IS Bookman-Demi
FONT 35 IS Bookman-DemiItalic
FONT 36 IS Bookman-Light
FONT 37 IS Bookman-LightItalic
FONT 38 IS Palatino-Roman
FONT 39 IS Palatino-Bold
FONT 40 IS Palatino-Italic
FONT 41 IS Palatino-BoldItalic
FONT 42 IS ZapfChancery-MediumItalic
FONT 43 IS ZapfDingbats
```

FONT SIZE /.34/: <##>

The size of the chosen font is specified. A value around 0.34 is recommended.

9.5.9 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.

Synopsis: **SET_INTERACTIVE_MODE**

9.5.10 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.

Since TCCS, two new options, *i.e.*, “Always initiate POST on re-entering” and “Always solid line”, have been added to the setting of plot options:

- 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 a 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.

Synopsis: SET_PLOT_OPTIONS

Ensuing 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>

As a new feature since TCCP, 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>

As a new feature since TCCS, 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 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 a 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>

As a new feature since TCCS, 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.

9.5.11 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 #.

Synopsis: SET_PREFIX_SCALING

Ensuing 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.

9.5.12 SET_REFERENCE_STATE

Description: The reference state for a component is important when calculating activities, chemical potentials and enthalpies. See detailed information in Sections 8.10.7 and 3.2.13.

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)].

Synopsis: SET_REFERENCE_STATE

Ensuing 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 /1E/: <reference pressure>

One may select the pressure (in pa) for the reference state. The value * means the current pressure is used for the calculation.

9.5.13 SET_TIELINE_STATUS

Description: If the tie-lines are in the plane of the calculation, one may select to plot the tie-lines in two-phase fields with this command.

Synopsis: SET_TIELINE_STATUS

Ensuing 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.

9.5.14 SET_TRUE_MANUAL_SCALING

Description: The tie-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).

Synopsis: SET_TRUE_MANUAL_SCALING

Ensuing 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.

9.5.15 FIND_LINE

Description: This new command (since TCCR) can be used to 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).

Synopsis: FIND_LINE

Ensuing 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); for instance:

found line : GM(FCC_A1)
found line : LIQUID+MG2SI

(for a property diagram)
(for a phase diagram)

9.5.16 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.

Synopsis: TABULATE

Ensuing 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.

OUTPUT 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.

9.5.17 LIST_DATA_TABLE

Description: This is also a new command since TCCR. It 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 - graph* 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 9.16*.

Please note that it works only for a property diagram after a stepping calculation (but not for any phase diagram after a mapping calculation).

Synopsis: LIST_DATA_TABLE

Ensuing 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.

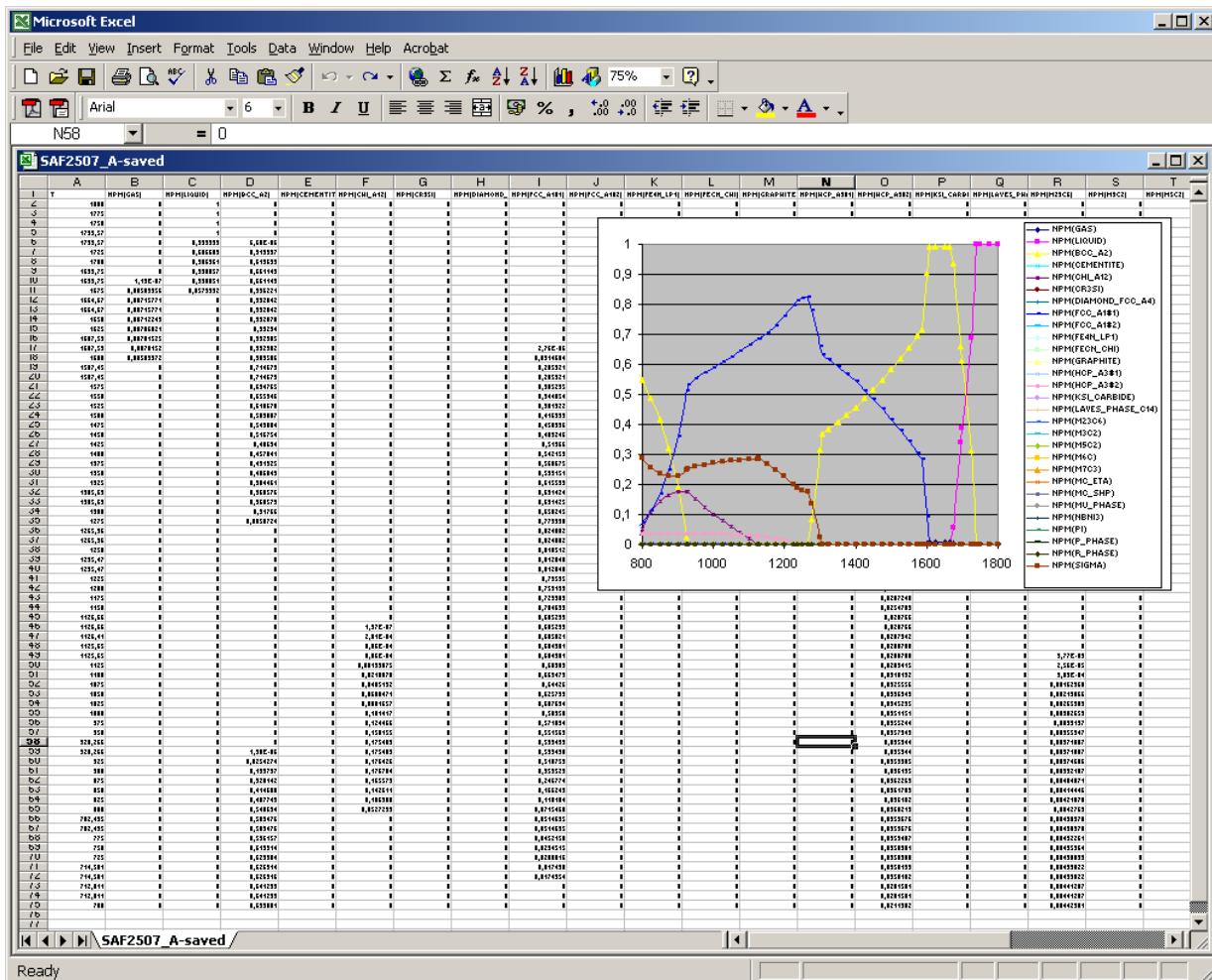


Figure 9-16. An Example of Listed Property Table (of a defined property diagram), output as a XLS file. Such a file can be further handled using the Microsoft Excel program, in order to generate other types of graphical presentations.

9.6 Odd Commands

9.6.1 PATCH_WORKSPACE

Description: This command is only for those who think they know what they are doing!

Synopsis: PATCH_WORKSPACE

9.6.2 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.

Synopsis 1: RESTORE_PHASE_IN_PLOT <name of a phase>

Synopsis 2: RESTORE_PHASE_IN_PLOT

Ensuing Prompt: PHASE NAME: <name of a phase>

Give the name of the phase to be restored on the diagram.

9.6.3 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.

Synopsis: REINITIATE_PLOT_SETTINGS

9.6.4 SET_AXIS_PLOT_STATUS

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.

Synopsis 1: SET_AXIS_PLOT_STATUS <Y or N>

Synopsis 2: SET_AXIS_PLOT_STATUS

Ensuing Prompt: Axis plot (Y or N) /Y/: (Y or N)

The user can specify to plot axis texts (Y) or not (N).

9.6.5 SET_PLOT_SIZE

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.

Synopsis 1: SET_PLOT_SIZE <relative scaling factor>

Synopsis 2: SET_PLOT_SIZE

Ensuing 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.

9.6.6 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.

Synopsis 1: SET_RASTER_STATUS <Y or N>

Synopsis 2: SET_RASTER_STATUS

Ensuing Prompt: RASTER PLOT (Y or N) /N/: <Y or N>
Enables (Y) or disables (N) the raster plot.

9.6.7 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 can not be altered by this command, but they are adjusted based on the selected graphic device and the defined relative scaling factor (global plot size).

Synopsis 1: SET_TIC_TYPE <1 or -1 or 0>

Synopsis 2: SET_TIC_TYPE

Ensuing 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.

9.6.8 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.

Synopsis 1: SUSPEND_PHASE_IN_PLOT

Synopsis 2: SUSPEND_PHASE_IN_PLOT <name of a phase>

Ensuing Prompt: PHASE NAME: <name of a phase>
Specify the name of the phase to be suspended from the diagram.

9.7 3D-Graphical Presentations: Command and Demonstration

Since TCCP, it has made possible to view Thermo-Calc diagrams in 3-dimensions with high-quality, through a VRML (Virtual Reality Modelling Language) viewer such as plug-ins to web browsers or standalone programs. Such viewers include the **Cortona VRML Client** for Windows NT4/2000/XP/Vista environments (which is developed by ParallelGraphics, and downloadable from www.parallelgraphics.com), **VRMLView** for Windows (NT4/2000/XP/Vista), Mac, Linux and various UNIX platforms (which is developed by SIM - Systems in Motion, and downloadable from www.sim.no), and alternative viewers listed at the VRLM Repository at www.web3d.org (Wed3D Consortium). In some browsers, it is needed to set correct background color or to turn on the so-called Headlight for best viewing. The headlight always shines directly on the 3D diagram.

This new feature is suitable for presentation of diagrams such as ternary phase diagrams represented by two compositional axis and a third Z-axis for temperature. In such cases, the diagram can be plotted as a triangular diagram or a square (tetrahedron) diagram.

It can also be used for presentation quaternary diagrams where the temperature and pressure has been fixed and lines in the compositional space are drawn. In such cases, the diagram should be plotted as tetrahedron diagram with all three axes set to compositional variables.

For this particular purpose, a new POST command (CREATE_3D_PLOTFILE) has been added, which can automatically convert all the default/predefined graphical definition information and selected data points (expressed and stored in specific tables or *.TAB files, which are taken from the current and/or previous MAP/STEP calculations) into an appropriate format and create an *.WRL (Virtual Reality Modelling Language) file for viewing 3D diagrams by an appropriate VRML viewer.

The command takes graphical definition information set by other POST processor commands, such as:

SET_DIAGRAM_AXIS	to set the axis variables of the 3D diagram.
SET_TITLE	to set the title of the 3D diagram.
SET_DIAGRAM_TYPE	to set the type of 3D diagram to be plotted. The implemented options are for diagrams with square or triangular base or for tetrahedron diagrams. No clipping of drawn lines will be done. Note that for tetrahedron diagrams, the user has to use the command sequence SET_DIAGRAM_TYPE N Y (see details in <i>Section 9.3.2</i>).
SET_RASTER_STATUS	to specify whether short tic-marks or full raster lines shall be drawn in bottom and backside faces of the square or triangular 3D diagrams.
SET_AXIS_TEXT_STATUS	to set the X-, Y- and Z-axis text labels used in the 3D diagrams.
SET_SCALING_STATUS	to set manual or automatic scaling of the 3D axis and in the case of manual scaling prompts for the min and max values.

There are some limitations to the scaling possibilities in 3D diagrams.

- Square diagrams
 - The X and Y minimum values must be 0.0
- Triangular diagrams
 - The X and Y minimum values must be 0.0
 - The X and Y maximum values must be the same for both axis
- Tetrahedron diagrams
 - The X,Y and Z minimum values must be 0.0
 - The X and Y maximum values must be the same for both axis
 - The Z maximum can be selected freely

For instance, if one is interested in the Fe-rich corner of the Fe-Cr-Ni-C system, the axis variables (Note that temperature is not allowed!) should be selected as:

 - X = Cr content
 - Y = Ni content
 - Z = C content, probably the C content values are much lower than Cr and Ni and by using this variable on the Z-axis a more suitable scaling can be selected.
- No clipping of drawn lines will be done.

The CREATE_3D_PLOTFILE command can use specified data (beside the graphical definition information) in the following three different modes:

- WS reads data stored as tables that exist in the current workspace of the calculation just carried out by the POLY command MAP or STEP, or of previously calculated results that have been loaded from an existing *.POLY3 file,
- TAB reads data stored as tables from one or more *.TAB files (all such files must locate under the current work directory), in which the table data for other previous calculations for related subsystems have already been saved with the TABULATE command, and
- BOTH reads data stored as tables from both the current workspace and previously saved *.TAB files, implying that the table data from the current workspace input data can be combined with other previous MAP/STEP calculation results if appropriate table(s) are already written onto the *.TAB file(s) by using the TABULATE command (all such files must locate under the current work directory).

In all such modes, one must have already defined some tables (entered by the ENTER_SYMBOL command in the POST or PLOY module), which can be further used within the TABULATE command to generate the *.TAB files. To simplify this task, two additional helping symbols have been defined in the POST module and will be automatically shown up once the CREATE_3D_PLOTFILE command is used:

- TEMP_C=T-273.15 for temperature in Celsius (as a variable)
- ZERO =0 for the value 0.0 (as a constant)

These two symbols can be directly called when defining tables. See examples below:

```
ENTER-SYMBOL TABLE demotb1 = x(fcc,ni),x(fcc,cr),TEMP_C;
ENTER-SYMBOL TABLE demotb2 = x(fcc,ni),ZERO,TEMP_C;
ENTER-SYMBOL TABLE demotb3 = ZERO,x(fcc,cr),TEMP_C;
ENTER-SYMBOL TABLE demotb4 = x(fcc,ni),x(fcc,cr),ZERO;
ENTER-SYMBOL TABLE demotbA = x(fcc,ni),x(fcc,cr),x(fcc,c);
ENTER-SYMBOL TABLE demotbB = x(fcc,ni),x(fcc,fe),x(fcc,c);
ENTER-SYMBOL TABLE demotbC = x(fcc,ni),x(fcc,cr),TEMP_C,x(fcc,c);
ENTER-SYMBOL TABLE demotbD = x(fcc,ni),ZERO,x(fcc,c),TEMP_C;
```

In the case of WS or BOTH modes, the CREATE_3D_PLOTFILE command can take such defined tables as the inputting data.

Such defined table(s) may have also been used in creating various *.TAB file(s) using the TABULATE command in the POST or PLOY module, for instance:

```
TABULATE demotb1 crni.TAB
```

In the case of TAB or BOTH modes, the CREATE_3D_PLOTFILE command can take one or more such previously created *.TAB files as the inputting data; however, all such *.TAB files must locate under the current work directory.

Note that to further simplify the use of TAB files, it is possible to combine several such *.TAB files in Windows DOS session by the following DOS command:

```
Copy Ter.tab+Bin1.tab+Bin2.tab+Bin3.tab Tmp.tab
```

(or similar commands in other operating systems), or use a textual editor to merge the individual files into one single file. Then, one can use the Tmp.tab file as the single TAB input file as the input data in the CREATE_3D_PLOTFILE command.

In the following sections, the CREATE_3D_PLOTFILE command will be described, and then a demonstration on how to view a 3D diagram will be illustrated. Furthermore, three standard examples on how to generate an *.WRC file in TCCP are given in *Examples 45, 46 and 47*, respectively, in the **TCC Examples Book**.

9.7.1 CREATE_3D_PLOTFILE

Description: This POST command automatically converts all the default/predefined graphical definition information and selected data points (expressed and stored in specific tables or *.TAB files, which are taken from the current and/or previous MAP/STEP calculations) into an appropriate format, and creates an *.WRL (Virtual Reality Markup Language) file for viewing by a VRML viewer such as plugins to web browsers or standalone programs. For more information, refer to the introduction above.

Synopsis: CREAT_3D_PLOTFILE

Ensuing Prompt: Use current WORKSPACE (WS), TAB files or BOTH: <##>

Choose one of the three available modes: WS for reading table data in the current workspace, TAB for reading table data from existing *.TAB file(s), or BOTH for reading combined table data from both the current workspace and existing *.TAB files. Note for the BOTH mode, you have to type the whole word as “BOTH” or “both”; otherwise the command may take it as for the TAB mode.

In the WS and BOTH modes, all the previously defined tables in the current workspace are listed out on screen, for the purpose of selecting desired table data into the output *.WRL file.

Table Name: <table name>

In the WS and BOTH modes, give the name of an available table, which has been previously defined by the TABULATE command with at least 3 variables that will be used as data points for the 3D diagram. The first column is always the X-axis value, the second is the Y-axis value and the third is the Z-axis value. A table with more than 3 columns can also entered, and the columns beyond the third one will also be converted into the output *.WRL file, while they will not be used when the 3D diagram is viewed in a VRML viewer.

Give TAB filename: /Cancel_to_finish/: <file name(s)>

In the TAB and BOTH modes, specify the file name(s) of already prepared *.TAB file(s) to be included in the single output *.WRL file for viewing the 3D diagram. One or more *.TAB files can be specified on the same line or on subsequent line(s); while all such files must be located under the current working directory. This is particularly useful, for instance, to add binary sides onto a ternary diagram.

In the TAB mode, if there is no specific *.TAB file name given under the first appearance of this prompt, the command is terminated.

This prompt will be repeated for additional *.TAB file(s) until no file name is given (*i.e.*, just pressing <RETURN>).

Output file: /3Dplot/: <file name>

Specify the output file name of the *.WRL file (with the extension .WRL, *e.g.*, FeCrNiC-a.WRL) or press <RETURN> to accept the default file name 3Dplot, in which all the graphical definition information and selectively specified table data to be viewed in the 3D diagram will be written. All the data will be automatically converted to a proper format for VRML viewers. Note that if the extension .WRL is missing in the specified file name, the output file is only a simple textual file.

X-AXIS SCALING FROM 0.0 TO XMAX: <##>

Give the maximum value on the X-axis. Note that the X and Y maximum values must be the same for both X- and Y-axis.

Y-AXIS SCALING FROM 0.0 TO YMAX: <##>

Give the maximum value on the Y-axis. Note that the X and Y maximum values must be the same for both X- and Y-axis.

Z-AXIS SCALING, GIVEN ZMIN: <##>

Give the minimum value on the Z-axis for the 3D diagram. Note that in the case of tetrahedron diagrams the Z minimum value must be 0.0.

Z-AXIS SCALING, GIVEN ZMAX: <##>

Give the maximum value on the Z-axis for the 3D diagram. For tetrahedron diagrams, the Z maximum value can be selected freely.

Notes:

In all modes, a table defined in the current workspace or saved in an *.TAB file to be selected as the input data must have at least 3 columns: the first one is always the X-axis value, the second is the Y-axis value and the third is the Z-axis value. All possible extra columns beyond the third one in the table(s) will also be converted into the output *.WRL file, but they will not be used when the 3D diagram is viewed in a VRML viewer.

In both the TAB and BOTH modes, all the (first) three columns in all specified *.TAB files and in the selected table from the current workspace must have been similarly defined, in order to properly view the 3D diagram for a specific system. By “similarly defined”, it means that if there is no data on one certain column from a previous calculation, it should be entered as the ZERO constant, and any other column should always have the same variable (or ZERO). If the definitions for all the (first) three columns in all tables are not similar, the output *.WRL file will have data blocks with different data structures in various parts, although the conversion of data points can be done within this command. Therefore, one can have the following four tables (saved in *.TAB files or defined in the current workspace) for the data conversion into the same *.WRL file:

```
ENTER-SYMBOL TABLE demotb1 = x(fcc,ni),x(fcc,cr),TEMP_C;
ENTER-SYMBOL TABLE demotb2 = x(fcc,ni),ZERO,TEMP_C;
ENTER-SYMBOL TABLE demotb3 = ZERO,x(fcc,cr),TEMP_C;
ENTER-SYMBOL TABLE demotb4 = x(fcc,ni),x(fcc,cr),ZERO;
```

And they should not be mixed with other tables defined as below:

```
ENTER-SYMBOL TABLE demotbA = x(fcc,ni),x(fcc,cr),x(fcc,c);
or
ENTER-SYMBOL TABLE demotbB = x(fcc,ni),x(fcc,fe),x(fcc,c);
```

Moreover, if you have typed a name that does not exist in the current working directory, the command will still be possible to process, but a new empty file (with that name) would be created in the directory. Therefore, one should be very careful when specifying the *.TAB file names.

For the TAB and BOTH modes, some informative message may be given after specifying all the details in this command, such as below:

```
It is possible to combine files by:
Copy Ter.tab+Bin1.tab+Bin2.tab+Bin3.tab Tmp.tab

Processing crnic.TAB
2.64306994E-08<X< 0.491403997
0.0539233014<Y< 0.241238996
6.70927989E-07<Z< 0.000157856004
```

9.7.2 View 3D Diagrams in Cortona VRML Client Viewer and SIM VRMLView

The *.WRL files created by the CREATE_3D_PLOTFILE command can be viewed in 3-dimensions through a VRML viewer, such as plug-ins to web browsers or standalone programs. The user can download a preferred one from the VRML Repository at www.web3d.org (Wed3D Consortium).

For Windows (NT4/2000/XP/Vista) environments, the **Cortona VRML Client** (which is developed by ParallelGraphics, and downloadable from www.parallelgraphics.com) is recommended, and is demonstrated here.

It is a pluggable application for MS Internet Explorer and Netscape Navigator, capable of browsing virtual worlds written in VRML. Cortona provides navigation paradigms (such as walking or flying) which enable the user to move the viewer through the virtual world. Additionally, it provides a mechanism that allows the user to interact with the world. Cortona is designed on the basis of the universal component technology that provides the opportunity to create a wide range of 3D applications: from graphical processors to complex multi-user Internet-services. Cortona VRML client supports: OpenGL; Direct3D; Java language; Splines and NURBS; External Authoring Interface (EAI). The client also has its own built-in VRML Automation Interface (Automation Interface for managing VRML scenes) based on the ActiveX Automation technology. Its hardware and system requirements are as below:

- Operating system: Microsoft Windows NT4/2000/XP/Vista;
- Internet Explorer 4.0 or a later version, or Netscape Navigator 4.0 or a later version;
- Computer: Pentium 90 MHz or faster processor;
- Random Access Memory (RAM): minimum of 16 MB;
- Free disk space: 6 MB of hard disk space for program files;
- Display: SVGA/256 colors. The 800x600 high color mode is recommended;
- Input devices: Mouse and keyboard, joystick optional;
- Sound card: optional.

Please first download it from www.parallelgraphics.com yourself, and install it onto your Windows computer or server. By running on its setup program (or clicking its setup icon on Windows), the automatic procedure will install it under the directory, e.g.,

C:\Program Files\Common Files\ParallelGraphics\Cortona\

After the successful installation on a computer or a server, you can simply click on an already generated *.WRL file, or open it from the Cortona window, and then follow the instructions to view, print or export the Thermo-Calc graphs in 3D. The detailed instructions are provided in the **Cortona User's Guide**, which can be obtained by simply positioning the mouse within the Cortona window and clicking the right button to choose Help on User's Guide to open the Cortona User's Guide window.

Figures 9-17 and 9-18 illustrate two examples of Thermo-Calc graphs in 3D (one as a triangular diagram, and the other as a tetrahedron diagram) viewed by the **Cortona VRML Client**.

In the distributed PDF file of this User's Guide (around this point), there is special bookmark named as “[Open the Cortona User's Guide - ParallelGraphics Software](#)”; you can click it to open the **Cortona User's Guide** on screen.

For Windows (NT4/2000/XP/Vista), Mac, Linux and various UNIX platforms, the **SIM VRMLView** (which is developed by SIM - Systems in Motion, and downloadable from www.sim.no) is also recommended. Please try to explore it yourselves.

Figures 9-19 and 9-20 show two examples of Thermo-Calc graphs in 3D (one as a triangular diagram, and the other as a tetrahedron diagram, which are using the same *.WRL files as in *Figures 9-17 and 9-18*) viewed by the **SIM VRMLView**.

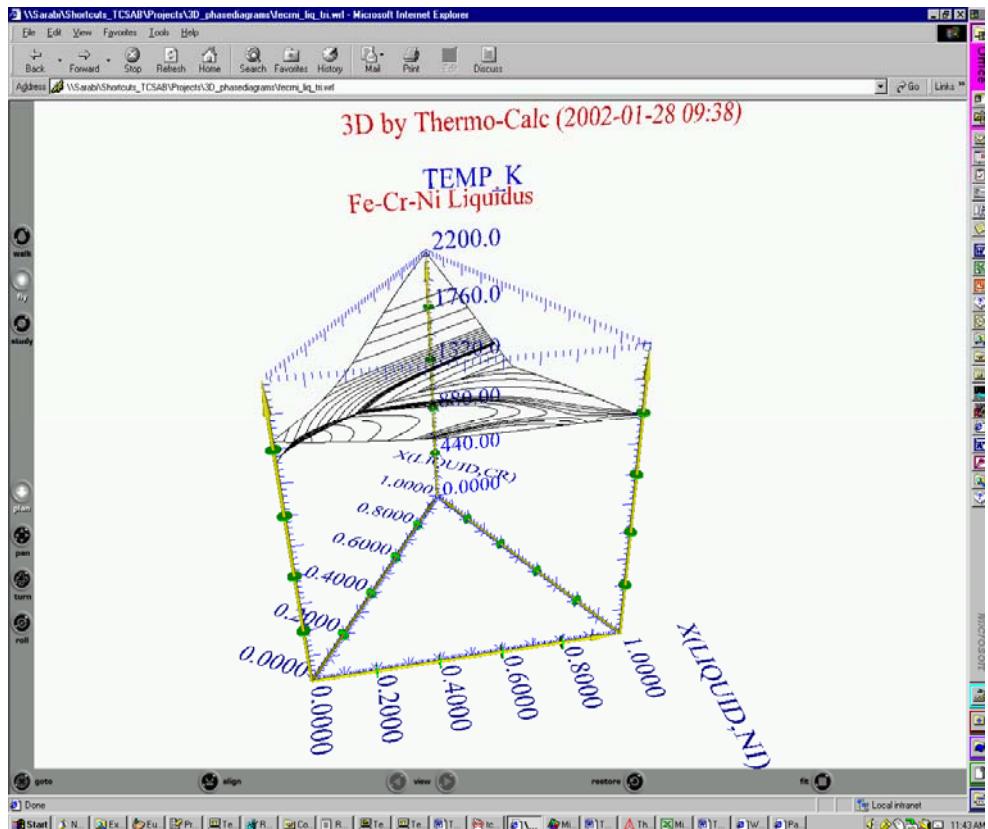


Figure 9-17. An Example of Triangular TC Diagram Viewed in 3D by Cortona VRML Client.

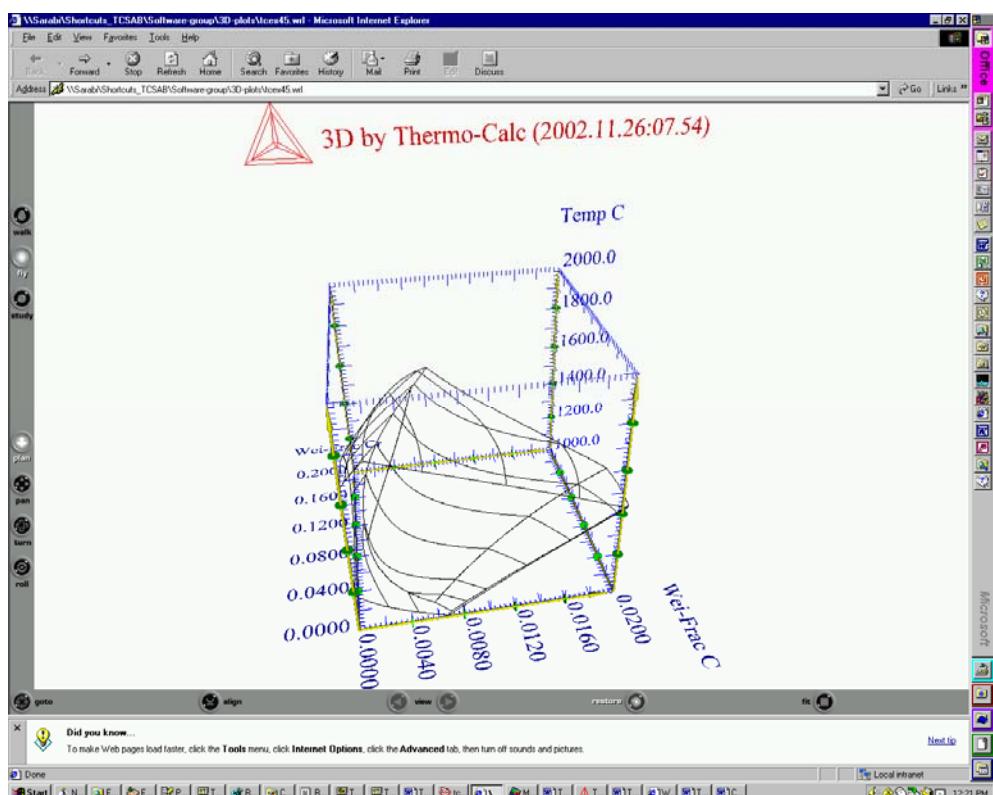


Figure 9-18. An Example of Tetrahedron TC Diagram Viewed in 3D by Cortona VRML Client.

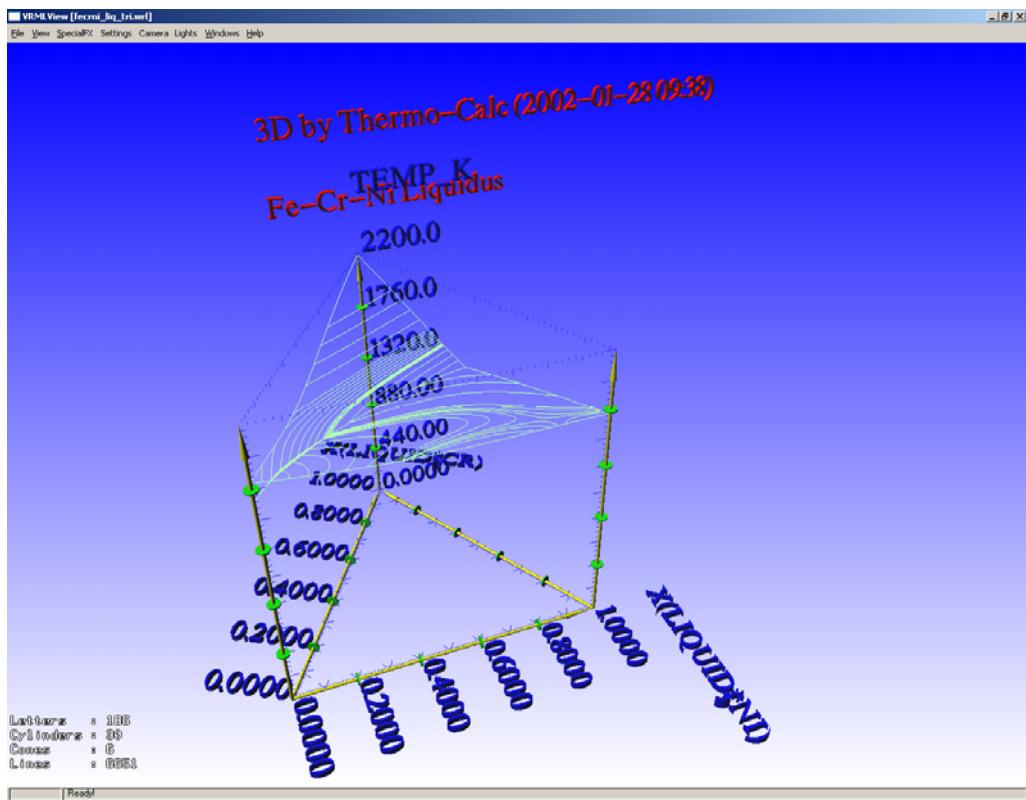


Figure 9-19. An Example of Triangular TC Diagram Viewed in 3D by SIM VRMLView.

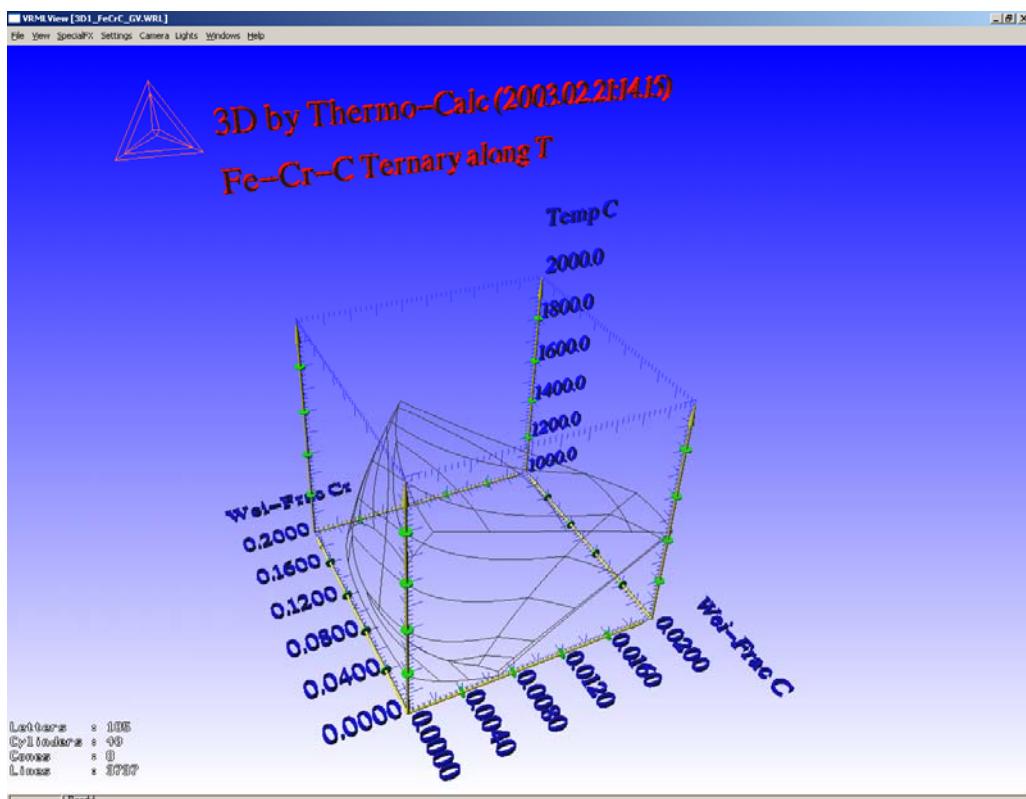


Figure 9-20. An Example of Tetrahedron TC Diagram Viewed in 3D by SIM VRMLView.

10 Some Special Modules

10.1 Introduction

As described in the document *Thermo-Calc Software System*, continuous improvements on user-friendly features have been parts of the development of the Thermo-Calc software/database/interface package. A good user interface must bring benefits to users in enhancing their efficiency of using the software/database/interface in various applications.

The Thermo-Calc Classic (TCC) software is an interactive system and the user controls it by giving commands from a menu. Great efforts have been continuously made to make the system's command-menu user interface more and more "user friendly". The improvement of the Thermo-Calc user interface has been focused on five levels, as summarized in *Table 4 in the document Thermo-Calc Software System*.

One of the levels is the development of some simplified modules, to allow the users to perform calculation and plotting of common types of phase or property diagrams (*e.g.*, binary, ternary, potential, Pourbaix, *etc.*) or simulation of materials processes (*e.g.*, solidification, multi-staged steady-state reactors, *etc.*). There are currently the following special modules available in the TCC software:

- | | |
|--------------|--|
| 1) BIN | - Binary Phase Diagram Calculation Module |
| 2) TERN | - Ternary Phase Diagram Calculation Module |
| 3) POTENTIAL | - Potential Property Diagram Calculation Module |
| 4) POURBAIX | - Pourbaix Diagram and Property Diagram Calculation Module |
| 5) SCHEIL | - Scheil-Gulliver Solidification Simulation Module |
| 6) REACTOR | - Multi-staged Steady-state Reactor Simulation Module |

Since TCCQ, the SCHEIL and REACTOR modules had been greatly modified and improved; and since TCCR, the TERN module had been further improved and enhanced. More important improvements for the TERN and SCHEIL modules have been made available since TCCS. For all details about such continuous improvements/modifications, please refer to *Sections 10.5, 10.8 and 10.9*.

In all such special modules, except for the REACTOR module, one just needs to answer some simple questions and the program automatically handles all the necessary steps for the calculation and post-processing; this is why they are sometimes called as easy-to-use modules. The REACTOR module has a command-menu user interface, which is similar to those in all basic modules (*e.g.*, SYS, TDB, GES, TAB, POLY, POST, PARROT and ED-EXP). However, a software/database system to be used for calculations and simulations cannot be constructed in the same way as a retrieval system for bibliographic data, and the user must thus know how to define his problem within the framework of the software/database system.

More such easy-to-use modules will be built up in future releases, so that some common problems with wider applications, or some difficult cases that require more comprehensive techniques for specifying the calculation/simulation conditions and stepping/mapping variables, can be handled in a standard way. For instance, a PREDOM module would be developed to calculate and plot the so-called predominant diagrams, and a HTCORR module could be implemented to simulate high-temperature materials corrosion process.

More options in some existing modules (such as POURBAIX and SCHEIL) will be eventually added, and some options will be further enhanced or extended, so that the module can treat more complex interaction systems and provide more sophisticated results and presentations.

This Special Modules Guide [⇒] will give some detailed descriptions on the currently available special modules and their applications.

[⇒] **Revision History of the Special Modules User's Guide:**

Oct 1993	First release on the REACTOR module (Edited by Bo Sundman)
Jun 2000	Second release on all the other special modules (Edited by Pingfang Shi)
Nov 2002	Third release on all special modules (Edited by Pingfang Shi)
Jun 2004	Fourth release on the SCHEIL and REACTOR modules (Edited by Pingfang Shi)
May 2006	Fifth release on the TERNARY and SCHEIL module (Edited by Pingfang Shi)
Apr 2008	Sixth release on the TERNARY and SCHEIL module (Edited by Pingfang Shi)

10.2 Files Generated or Used by Special Modules

10.2.1 POLY3 Files

Executing a special and easy-to-user module (*i.e.*, the BIN, TERN, POT, POURBAIX or SCHEIL modules), the Thermo-Calc program will automatically save all the information about the following definitions and settings onto a file in the current working directory:

- system definitions;
- thermodynamic data;
- component definitions;
- reference-state settings;
- status settings of phases, components and species;
- equilibrium condition specifications;
- stepping/mapping variable settings;
- initial equilibrium points;
- mapping or stepping calculation results;
- axis variable/text settings for the first diagram;
- various symbol definitions (constants, variables, functions and tables).

Such a file will be generated and saved in the POLY3-workspace structure, and thus is always associated with the file-name extension of “.POLY3” (under PC Windows Vista/XP/2000/NT4 environments) or “.poly3” (under PC Linux and various UNIX platforms).

In the automatic procedure, the specific module name always appears as the default name, *e.g.*, BINARY/GCURVE/PFCURVE.POLY3 (BIN module), ISOTERN/MONOVAR/PFCURVE.POLY3 (TERN module), POT.POLY3 (POTENTIAL module), POURBAIX.POLY3 (POURBAIX module), and SCHEIL.POLY3 (SCHEIL module). A user may wish to make a copy of such an *.POLY3 file under another name after the module calculation is finished, or rename it if another workspace (POLY3, GES5 or PAR) is loaded or if the program is exited.

All such *.POLY3 files can be read by the normal POLY module, and be used for many other purposes of calculations (in POLY module), graphical processing (in POST module), or data manipulations (in GES module). There are also options in the POURBAIX and SCHEIL modules to open previously saved *.POLY3 files (as either POURBAIX- or SCHEIL-type, respectively), in order to make another POURBAIX or SCHEIL module calculations for the same chemical system but different temperature-(pressure)-composition conditions, or to plot other property diagrams from the same calculated systems and conditions.

10.2.2 RCT Files

Executing special REACTOR module, the Thermo-Calc program will automatically save all the information related to the definitions of the reactor, distribution coefficients, in addition to those in the POLY3-workspace (see *Section 10.2.1*), onto a file in the current working directory. Such a file will be generated and saved in the REACTOR-workspace structure, and thus is always associated with the file-name extension of “*.RCT”.

10.2.3 GES5 Files

After executing a special module, one can go to the GES module to manipulate various data and descriptions, and to save all the information about system definitions, thermodynamic data and various symbols (constants, variables, functions and tables) onto an *.GES5 file in the current working directory. The specific module name may appear as the default name [*e.g.*, BINARY.GES5, TERNARY.GES5, POT.GES5, POURBAIX.GES5, SCHEIL.GES5, and REACTOR.GES5], or the program may ask for a user-specified name. A user may wish to make a copy of such a GES5 file under another name, or rename it if another workspace (POLY3, GES5 or PAR) is loaded or if the program is exited. All such *.GES5 files can be read by the normal GES module, and be used for other purposes.

10.2.4 MACRO Files

A MACRO file (with the default extension *.LOG or *.TCM) for a special module can be created, by using the SYS-module command SET-LOG-FILE, or by using a textual editor to write various command lines or answering lines.

One should remember that, in a MACRO file created for a special module (BIN, TERN, POTENTIAL, POURBAIX or SCHEIL), each line between the entering module command (*e.g.*, go BIN, go TERN, go POT, go POUR or go SCH) and SET-INTERACTIVE command (for a POURBAIX-module MACRO file) or an ordinary POST command (for a BIN-/TERN-/POT-/SCHEIL-module MACRO file) is a question-corresponding answer. Thus, an empty line is always a default answer to a corresponding question in the module calculation or graphical processing. Therefore, neither comment line [after the characters @@] nor MACRO-file level-controlling comment [MACRO-FILE-OPEN <another file name>] can be inserted in between such answering lines. The pause-controlling characters [@&] or semi-interactive input-controlling characters [@?] should not be inserted either. Otherwise, the modules will not be able to properly execute the MACRO file continuously.

However, a MACRO file for the REACTOR module does not have such behaviors, so that one can write all kinds of legal comment lines or characters at any position, just like in a normal MACRO file that going through various basic modules for a calculation and post-processing.

10.3 Interactions with Special Modules

From the SYS (System Utility) module and from all modules where the GOTO_MODULE command is available (*e.g.*, the TDB, GES, TAB, POLY or PARROT module), one can directly enter into such special modules. For an example, a command GO BIN will switch on the BIN module for calculating a phase diagram or Gibbs energy curve in a binary system.

As mentioned above, the command-line structure is not utilized in those five special and easy-to-use modules (*i.e.*, BIN, TERN, POT, POURBAIX and SCHEIL); instead, the user just needs to answer some simple questions to perform the automatic calculations and graphic generations.

If entered from the SYS module, four of those five special modules (*i.e.*, BIN, TERN, POTENTIAL and SCHEIL) normally end up at the POST module, so that the user can further make adjustments and modifications on the appearance of a plotted diagram (such as setting curve-labeling options, adding texts, alternating axis types or diagram types, changing axis scales to zoom a particular area, suspending uninteresting phases, imposing experimental data points, making experimental data files, generating hard copies, and so on). However, the BIN, POTENTIAL and SCHEIL module may also end up at the previous module from where such a special and easy-to-use module was called; under such circumstance, one can always go to the POST module (via the POLY module) to immediately make the necessary adjustments and modifications on the appearance of the plotted diagram, if the current workspace has not been changed.

The POURBAIX module always ends up at the SYS module, no matter where it was called. As a unique feature of the module, one can directly and easily make adjustments and modifications on the appearance of a plotted Pourbaix diagram or other types of diagrams inside the module. With one option, one can even communicate with the POST module for further graphical refinement without interrupting the POURBAIX module, so that one can make immediate changes on the diagram.

As the REACTOR module was established in the same style of the basic modules, one will need to interact with the program step by step via various commands, that are similar to all basic modules (SYS, TDB, GES, POLY, POST, PARROT and ED-EXP modules).

In the next six sections, the interactions between the each of these special modules and users will be extensively demonstrated. One may also refer to the MACRO (*.TCM) files related to these modules in the \TCEX\ area and in the **TCCS Examples Book**.

10.4 BIN Module

10.4.1 Descriptions of the BIN module

Binary phase diagrams are extensively used in various materials design and engineering, materials chemistry, chemistry, geochemistry, environmental chemistry, and so forth. A tremendous amount of experimental investigations of binary alloy systems have been conducted over the last century, and such binary diagrams are especially useful for alloy developments. The serial editions of the “*ASM Binary Alloy Diagrams*”, published by ASM International based on available experimental data and some topological methods, have been widely applied by the materials R&D society.

However, the advanced Computational Thermodynamics approach employed in the Thermo-Calc software/database/interface package provides a more sophisticated way to look insight the studies of phase diagrams and many other thermodynamic properties of steels and alloys, as well as a lot of industrial and natural materials (ceramics, inorganic materials, organic substances, polymers, minerals, nuclear materials, *etc.*). It can precisely calculate and plot all kinds of phase diagrams and property diagrams, by using some well-established thermodynamic models and databases, powerful calculation and simulation tools, and easy-to-use modules, as described in the document *Thermo-Calc Software System* and many recent publications.

Various types of phase diagrams and property diagrams for binary systems can easily be calculated and plotted in TCC as a traditional routine. This is done going through its command-line structured modules, *i.e.*, to define the binary system and get thermodynamic data in the TDB module, to (if necessary) manipulate phase descriptions and parameters in the GES module, to set calculation conditions and mapping variables in the POLY module, and to specify the graphical appearance in the POST module. In this way, one can also calculate and plot the phase diagram for a pseudo-binary system (or called quasi-binary system where one component in a ternary system has a fixed activity or chemical potential, *e.g.*, CaO-SiO₂ at a constant O₂ activity, Ca-Fe-O when liquid oxide FeO is in equilibrium with liquid iron Fe, and Fe-Cr-C at a fixed carbon activity). Of course, the use of a MACRO file for a binary phase diagram or property diagram calculation and graphical processing (for instance, the examples TCEX4.TCM, TCEX16.TCM and TCEX17.TCM in the \TCEX\ area and the **TCCS Examples Book**) will make such command-line calculation and graphical processing much simple and efficient.

To make the user interface much easier and more straightforward, a special and easy-to-use module, namely the **BIN Module**, has been available since the TCCL; as well as in the **Binary Phase Diagram Module** of the TCW software. The BIN module is constructed as a simple question-based line structure, and the user just needs to provide some appropriate answers. The module can directly and automatically calculate and plot the phase diagrams (composition *vs.* temperature) or Gibbs energy curves (Gibbs energies of various phases *vs.* temperature) or phase-fraction plots (fractions of various stable phases *vs.* temperature) for various binary alloys systems. Furthermore, utilizing the BIN-module calculation results, many other types of phase diagrams and property diagrams can be generated in the POST processor.

The procedure of the BIN module is very simple, with a few steps to complete a calculation and graphical processing of a phase diagram, or a Gibbs energy curve or phase-fraction plot, for a binary system, as described below:

Step 1: Select a suitable solution database

The module asks for a suitable solution database from where all thermodynamic data will be retrieved. Basically, the end-member properties, binary interaction parameters and all the assessed system information (at least for the binary system to be calculated and plotted) must be available in the database. Such a solution database can be based on an ordinary solution database (implying that ternary and high-order interaction data may also be available), but normally is specially designed for the BIN module (see more details in *Section 10.4.2*). The user can also switch to his/her own solution database(s) that meet the basic requirements for making the BIN-module calculations.

Step 2: Specify the two elements

The names of the first and second elements (in one single line with one space in between, or in two separate lines) that are available in the currently selected solution database must be given.

Step 3: Select a main option

The program asks (as three main options) if a phase diagram (**P**, as default), phase fraction plot (**F**), Gibbs energy curves (**G**) or activity graph (**A**) of defined binary system is to be calculated and plotted.

For the main option **G** or **A**, the program further asks for a specific temperature (in °C) for calculating the Gibbs energies curves [expressed by GMR (*)] of all possible phases, or activity curves [expressed by ACR (*)] of each of the binary components, over the entire composition range of the binary joint at this temperature.

For the main option **F**, a composition in term of the mole fraction of the second element in the binary joint must be specified so that the phase fractions (default as mass fractions) of all stable phases in the system at this composition can be calculated over a temperature range.

Step 4: Automatically calculate and plot the desired diagram

The module automatically execute the following tasks:

- retrieving all the thermodynamic descriptions and information from the selected database;
- defining the initial equilibrium conditions for the desired calculation;
- setting the mapping or stepping variables [for the main option **P**, always the mole-fraction of the second element as the first mapping variable ranging from 0.0 to 1.0, and the temperature condition (in K) as the second mapping variable changing over a reasonable low-high range; and for the main options **G** and **A**, always the mole-fraction of the second element as the stepping variable ranging from 0.0 to 1.0; and for the main option **F**, always the temperature condition (in K) as the stepping variable changing over a reasonable minimum-maximum range];
- calculating the start points;
- making the mapping or stepping calculations;
- plotting the first desired diagram on the screen. This is the default output for the chosen main option (**P** or **G** or **A** or **F**), and its settings of axis variables are always as below:
 - For the main option **P**: $X = X(2^{\text{nd}}\text{El})$ Mole fraction of the second element in the joint
 $Y = T-C$ Temperature (in °C)
 - For the main option **G**: $X = X(2^{\text{nd}}\text{El})$ Mole fraction of the second element in the joint
 $Y = \text{GMR}(\ast)$ Gibbs energies for all possible phases (in J/mol)
 - For the main option **A**: $X = X(2^{\text{nd}}\text{El})$ Mole fraction of the second element in the joint
 $Y = \text{ACR}(\ast)$ Activity for all component (dimensionless)
 - For the main option **F**: $X = T-C$ Temperature (in °C)
 $Y = \text{BPW}(\ast)$ Mass fractions for stable phases in the system

Default settings are also made on axis types, axis texts, axis scaling factors, curve-labeling option (N or E for the main option **P**, and F for the main options **G** and **F**), colors and thickness (for texts and axes, invariant equilibrium points, tie-lines, and other lines like for solubility lines), fonts and sizes (for labels and numbers), diagram title (composed by the switched database name and the element names for the binary joint), and so on.

Figure 10-1 illustrates examples of three BIN-module generated diagrams for the Fe-Cr binary system, each diagram corresponding to one of the three main options, **P** (phase-diagram), **G** (Gibbs energy curves), **A** (activity curves) and **F** (phase-fraction plot), respectively. Note these diagrams have been slightly refined through the next step (see *Step 5*).

At the same time, an POLY3 file will be automatically saved in the current working directory, with the default name **ISOTERM.POLY3** for the main option **P**, **GCURVE.POLY3** for the main options **G** and **A**, and **PFCURVE.POLY3** for the main option **F**. These POLY3 files can be re-opened in the normal POLY module, in order to conduct further graphical processing of other phase diagrams or property diagrams in the POST module, or to perform other types of mapping or stepping calculations directly in the POLY module. Note that such a file must be copied (if that specific

POLY3 workspace is still in process) or renamed (if that POLY3 workspace has been quitted) to another file before making another BIN-module calculation with the same main option.

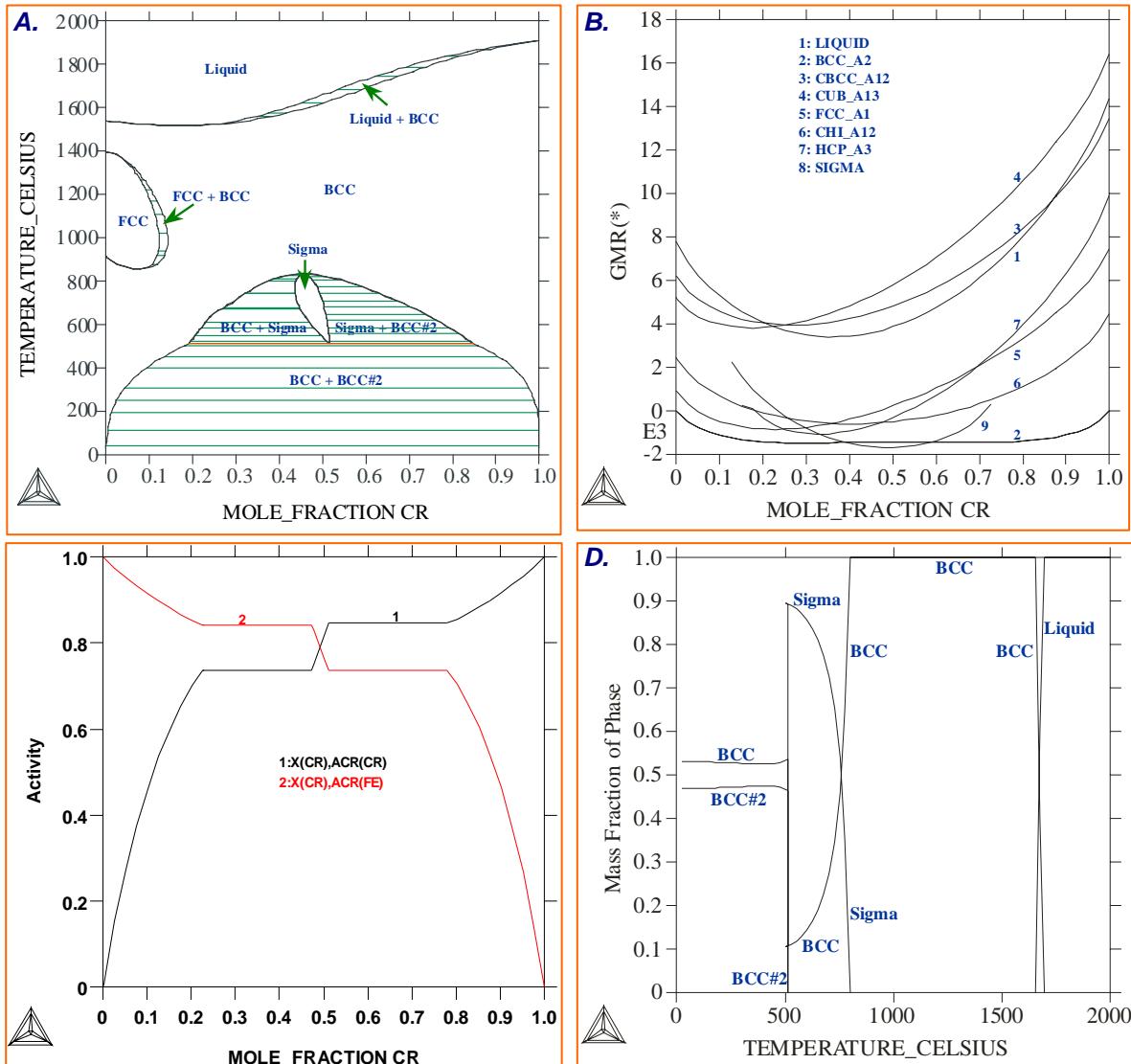


Figure 10-1.

Phase diagram, Gibbs energy curves and phase-fraction plot of the Fe-Cr binary system, automatically calculated by the BIN module (outputs corresponding to the three main options, P, G and F, respectively).

- A presents the Fe-Cr binary phase diagram [mole fraction of Cr vs. temperature (in °C)];
- B draws the Gibbs energy curves for all possible phases [*i.e.*, GMR (*) in J/mol] in the entire Fe-Cr binary joint at 600°C;
- C shows the activity curves for the two system components [*i.e.*, ACR (*)] in the entire Fe-Cr binary joint at 600°C;
- D plots the mass fractions of stable phases [*i.e.*, BPW (*)] along temperature (in °C) when the mole fraction of Cr in the binary joint is 0.55.

Step 5: Further refine the generated phase diagram or Gibbs energy curves or phase-fraction plot

Finally, the module always ends up in the normal POST module for further graphical processing and handling the automatically generated (first) diagram, as he/she desires. The diagrams presented in Figure 10-1 have been refined in this step.

The user can now apply all kinds of POST-module commands to refine the diagrams, by modifying and/or adding the settings for axis types, axis texts, axis scaling factors and manner, axis-tic status, tie-

line status, curve-labeling option, phase-region or property labeling texts, colors and thickness, diagram title and subtitle, axis length, diagram size, and so forth.

One can also re-set the reference states for the components, suspend and/or restore phase(s) from/onto a plotted diagram, impose experimental data points on a calculated diagram, change the plot options, and specify a different plot format.

After such further graphical processing, the user can choose to print the generated diagram on a connected printer, to save the diagram as a graphical file (using a preferable font and size), to dump the diagram to a graphical device (also saved as a graphical file), or to export the complete X-/Y-coordinates of the curves, lines and points, as well as some graphical settings, of the generated and modified diagram onto an *.EXP file. See *Chapter 9* for details of various POST commands.

Step 6: Further plot many other types of phase diagrams or property plots

If the user wants, he/she can generate many other types of phase diagrams or property diagrams in the POST-module monitor, using the same calculation results (saved in the currently running or re-opened BINARY.POLY3, GCURVE.PLOY3, PFCURVE.POLY3 file, or other copied BIN-module *.POLY3 files). The user may define the appropriate axis variables for the new diagrams, and can now apply all kinds of POST-module commands to re-define and refine the diagrams exactly as described in the *Step 6*.

One can set the axis variables, by selecting suitable equilibrium quantities which are either state variables (among internal variables, energetic variables, compositional and constitutional variables) or as derived variables. One may also enter some additional symbols (functions or variables or tables) which can be used as new axis variables or used for tabulations.

Normally, a variable corresponding to one of the mapping variables (if the calculation was made by the main option P), or the stepping variable (if by the main option G or A or F), must be chosen as the independent axis variable (the X- or Y-axis for diagrams corresponding to the main option P, and the X-axis for diagrams corresponding to the main option G or A or F). If a composition variable has been used as a mapping variable, the composition or activity (in Mole-Fraction, Mole-Percent, Weight-Fraction, Weight-percent or Activity) of any one of the two elements (components) in the binary joint can be set as the independent axis variable; if temperature has been used as a stepping variable, T (in K) or T-C (in °C) can be set as the independent axis variable.

For defining the second axis variables (the Y-axis, or the X-axis if the Y-axis is set as the independent axis variable using one of the mapping variables in the main option P calculation), one can choose any other suitable equilibrium quantity. One may even define a third axis variable (the Z-axis) in the similar way as defining the second one, if a triangular diagram is to be plotted in the POST module, or if a three-dimensional diagram is to be generated using proper graphical software.

These can be summarized as below:

- For the main option P: $X/Y = C/A(E1)$ Composition/Activity of one element in the joint
 or $T/T-C$ Temperature (in K or °C)
 $X/Y/Z =$ Any equilibrium quantity of the system or component(s) or phase(s); corresponding its variation along the phase boundaries (and tie-lines and invariant points)
- For the main option G/A: $X = C/A(E1)$ Composition/Activity of one element in the joint
 $Y/Z =$ Any equilibrium quantity of the system or component(s) or phase(s)
- For the main option F: $X = T/T-C$ Temperature (in K or °C)
 $Y/Z =$ Any equilibrium quantity of the system or component(s) or phase(s)

For more details, please refer to *Part 9 (the Post-Processor Module, POST)*.

Figure 10-2 illustrates some examples of such non-default diagrams plotted through graphical processing of the BIN-module saved POLY3 files for the Fe-Cr binary system, which are generated by the three main options, P (phase-diagram), G (Gibbs energy curves), A (activity curves) and F (phase-fraction plot), respectively.

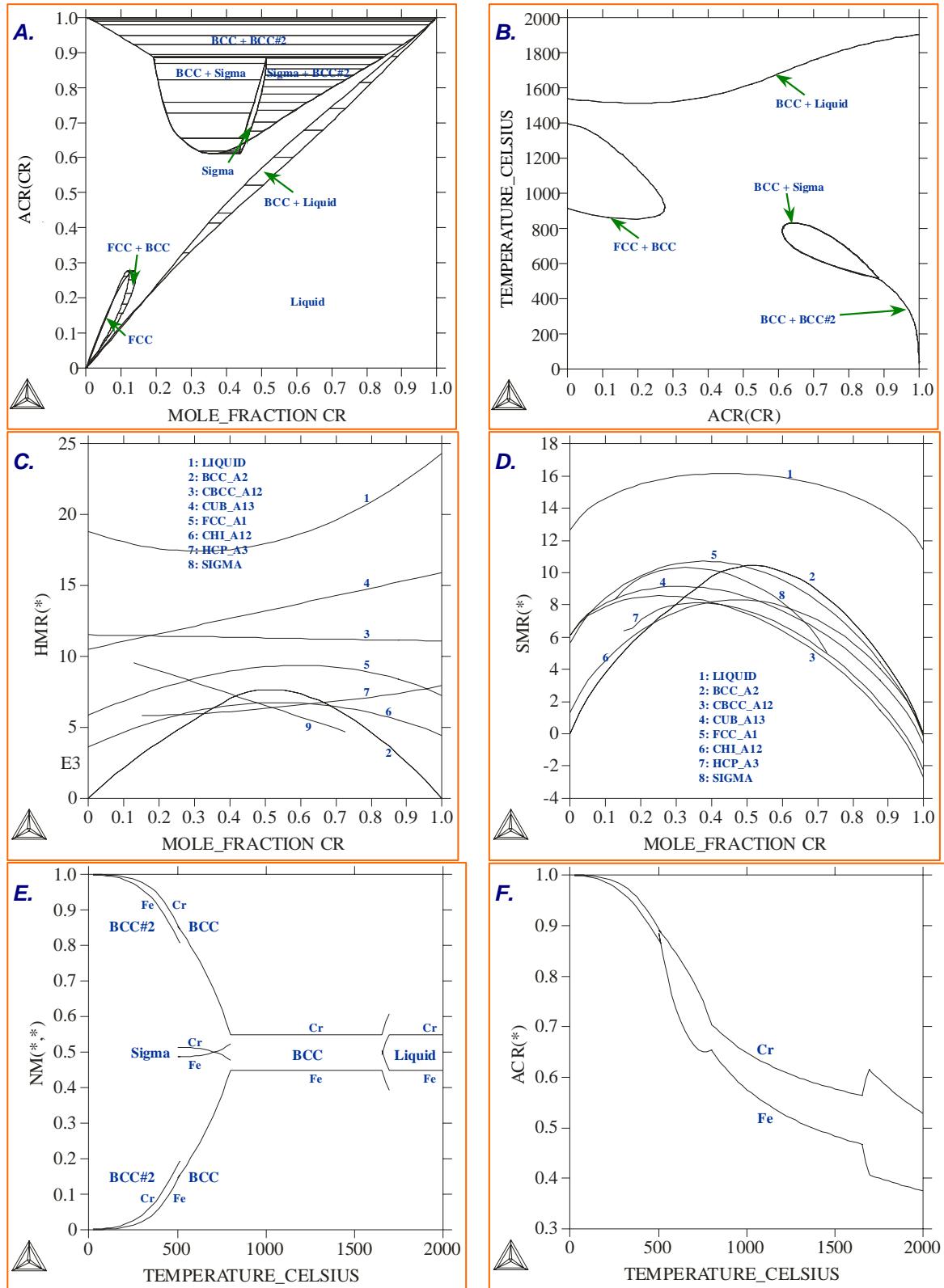


Figure 10-2. Other types of phase diagrams and property diagrams of the Fe-Cr binary system, generated by the POST module, based on calculation results from the main options (P, G and F).

A and B present the activity of the Cr component [ACR (Cr)] varied with either system composition [mole fraction of Cr] or temperature [in °C] along the phase boundaries; C and D draw the enthalpies and entropies for all possible phases [HMR (*) in J/mol and SMR (*) in J/mol/K] in the entire Fe-Cr binary joint at 600°C; E and F plot the mole fractions of Cr and Fe in stable phases [NR (*, *)] and activities for components Cr and Fe [ACR (*)] along temperature (in °C) when the mole fraction of Cr in the joint is 0.55.

Note that due to the main option P calculations (saved in default BINARY.POLY3 files) are performed with two independent mapping variables (temperature and composition of the binary joint), one can usually present the calculation results in various types of phase diagrams or property plots. A phase diagram for a binary system can be plotted with one axis variable as temperature (in °C or K), and the other as the system composition (mole-fraction M–F, or mole-percent M–P, or weight-fraction W–F, or weight-percent W–P of one of the elements e.g. components) or a composition-dependent state variable (such as activity, AC or ACR, of one component). If only one of such variables is set as one axis and another specific equilibrium property of the system or component(s) is set as the other axis, then a property diagram is plotted. In such a property diagram, the curves/lines/points will only present the variations of that specific equilibrium property along the phase boundaries (and tie-lines and invariant points).

Moreover, some thermodynamic properties and equilibrium quantities which are already defined (e.g., standard state variables or in a table) can be tabulated and plotted, along one mapping variable (the main option P) or the stepping variable (the main option G or A or F).

10.4.2 Structures of special BIN-module databases

As described above, the BIN Module in the TCC software (and Binary Phase Diagram Module in the TCW software) automatically retrieves all thermodynamic data from a switched suitable solution database, in which various binary interaction parameters and all the assessed system information (at least for the binary system to be calculated and plotted) must be available besides the standard thermodynamic properties of various phases. Normally, such a solution database is specially designed for the BIN module, but it can also be an ordinary solution database (implying that ternary and high-order interaction data may also be available) with an additional ASSESSED_SYSTEM information block (see below). The user can also switch to his/her own solution database(s) that meet the basic requirements for making the BIN-module calculations.

Previously, there were two specially designed binary alloy solution databases that can be directly used in the BIN Module of TCC (and Binary Phase Diagram Module of TCW). Since TCCR/TCW4, a new public database, TCBIN (TC Binary Alloy Solutions Database, version 1) is free-of-charge included along with the TCCR/TCW4 software; however, unlike other public databases, *TCBIN is always and can only be distributed in the encrypted database form, and can only be used inside the BIN Module (of the TCC software since TCCR) or Binary Phase Diagram Module (of the TCW software since TCW4)*.

Some detailed descriptions of these three special BIN-module databases can be found in their corresponding Database Description Forms (in the document *Thermo-Calc Database Guide*, as well as on our web site www.thermocalc.com/Products/Databases_TCC_TCW.html). They are also briefly described below:

➤ **PBIN** *TC Public Binary Alloy Solutions Database* (previously called BIN97)

- free-of-charge distributed with TCC and TCW, and with TCC_Demo/TC4A and TCW-Demo/TC4U
- covering 21 elements (Ag-Al-C-Co-Cr-Cu-Fe-Mn-Mo-N-Nb-Ni-O-Pb-S-Si-Sn-Ti-V-W-Zn)
- containing the following 35 binary systems (for liquids and solid solutions, and many stoichiometric phases)

Al-Cu	Al-Ni	Al-Zn	C-Co	C-Cr	C-Fe	C-Mn	C-Mo	C-Nb	C-Ni
C-Si	C-V	C-W	Co-Cr	Co-Fe	Co-Mn	Cr-N	Cr-Fe	Cr-Mo	Cr-W
Cu-Fe	Cu-Zn	Fe-Mn	Fe-Mo	Fe-N	Fe-Nb	Fe-O	Fe-S	Fe-V	Fe-W
Mo-Nb	Mo-W	N-Ti	N-V	Pb-Sn					

➤ **SBIN2** *SGTE Binary Alloy Solutions Database (A subset of SSOL2)* (previously called BIN)

- only distributed together with a formerly licensed SSOL2/SSOL4 database
- covering 83 elements (same as in the SGTE SSOL Solutions Database, version 2)
- containing the following 106 binary systems (for liquid and solid solutions, and many stoichiometric phases)

Ag-Au	Ag-Cu	Ag-Ge	Ag-Pb	Ag-Si	Ag-Sn	Al-As	Al-Bi	Al-Ca	Al-Fe
Al-Ga	Al-Ge	Al-In	Al-Mg	Al-Mn,	Al-Pb	Al-Si	Al-Sn	Al-Zn	As-Ga

As-Ge	As-In	Au-Bi	Au-Ge	Au-In	Au-Pb	Au-Sb	Au-Si	Au-Sn	Au-Tl
Bi-Cu	Bi-Ga	Bi-Ge	Bi-In	Bi-K	Bi-Sn	Bi-Tl	Bi-Zn	C-Co	C-Cr
C-Cu	C-Fe	C-Mn	C-Mo	C-Nb	C-Ni	C-Pb	C-Ti	C-V	C-W
Ca-Fe	Co-Cr	Co-Fe	Co-Mn	Co-Ni	Cr-Cu	Cr-Fe	Cr-Mo	Cr-N	Cr-Ni
Cr-V	Cr-W	Cs-K	Cs-Na	Cs-Rb	Cu-Fe	Cu-Mg	Cu-Ni	Cu-P	Cu-Pb
Cu-Si	Cu-Tl	Cu-Zn	Fe-Mn	Fe-Mo	Fe-N	Fe-Nb	Fe-Ni	Fe-Pb	Fe-S
Fe-Si	Fe-Ti	Fe-V	Fe-W	Ga-Ge	Ga-In	Ga-Pb	Ga-Sn	Ga-Zn	Ge-In
Ge-Pb	Ge-Sb	Ge-Sn	Ge-Tl	Ge-Zn	K-Rb	(Mg-Sb)	Mg-Si	(Mg-Sn)	Mo-N
Mo-Nb	Mo-Ni	Mo-W	N-Ni	N-Ti	Na-Rb	Ni-W	Pb-Si	Pb-Sn	Sb-Sn
Si-Mo	Si-Sn	Si-Ta	Si-Ti	Si-W	Si-Zn				

➤ **TCBIN** *TC Binary Alloy Solutions Database (only in encrypted form)*

- free-of-charge distributed together with the TCCR and TCW4 software
- *TCBIN is always and can only be distributed in the encrypted database form, and can only be used inside the BIN Module (of the TCC software since TCCR) or Binary Phase Diagram Module (of the TCW software since TCW4)*
- covering 67 elements
- containing the following 360 binary systems (for liquid and solid solutions, and many stoichiometric phases)

Ag-Al	Ag-Au	Ag-Bi	Ag-Cu	Ag-Ge	Ag-In	Ag-Ir	Ag-Mg	Ag-Os	Ag-Pb
Ag-Pd	Ag-Pt	Ag-Rh	Ag-Ru	Ag-Sb	Ag-Si	Ag-Sn	Ag-Ti	Ag-Tl	Ag-Zn
Ag-Zr	Al-As	Al-Au	Al-B	Al-Bi	Al-C	Al-Ca	Al-Ce	Al-Co	Al-Cr
Al-Cu	Al-Fe	Al-Ga	Al-Ge	Al-In	Al-Li	Al-Mg	Al-Mn	Al-Mo	Al-N
Al-Nb	Al-Nd	Al-Ni	Al-O	Al-P	Al-Pb	Al-Sb	Al-Si	Al-Sn	Al-Ta
Al-Ti	Al-V	Al-W	Al-Y	Al-Zn	Al-Zr	As-Au	As-Cu	As-Fe	As-Ga
As-Ge	As-In	As-P	As-Sb	Au-Bi	Au-C	Au-Cr	Au-Cu	Au-Ge	Au-In
Au-Pb	Au-Pd	Au-Rh	Au-Ru	Au-Sb	Au-Si	Au-Te	Au-Tl	B-C	B-Co
B-Cr	B-Fe	B-Hf	B-Mg	B-Mo	B-N	B-Nd	B-Ni	B-SC	B-Si
B-Ti	B-V	B-W	Ba-Cu	Ba-Eu	Ba-Sr	Ba-Y	Bi-Cu	Bi-Ga	Bi-Ge
Bi-In	Bi-K	Bi-Mg	Bi-O	Bi-Pb	Bi-Sb	Bi-Si	Bi-Sn	Bi-Tl	Bi-Zn
C-Co	C-Cr	C-Cu	C-Fe	C-Hf	C-Mn	C-Mo	C-Nb	C-Ni	C-Pb
C-Si	C-Ta	C-Ti	C-V	C-W	C-Y	C-Zr	Ca-Cu	Ca-Mg	Ca-Pb
Ca-Si	Ca-Zn	Cd-Ga	Cd-Ge	Cd-Hg	Cd-In	Cd-Pb	Cd-Sb	Cd-Sn	Cd-Te
Cd-Zn	Ce-Mg	Co-Cr	Co-Cu	Co-Dy	Co-Fe	Co-In	Co-Mn	Co-Mo	Co-N
Co-Nb	Co-Ni	Co-Pd	Co-Pt	Co-Si	Co-Ta	Co-Ti	Co-V	Co-W	Cr-Cu
Cr-Mg	Cr-Mn	Cr-Mo	Cr-N	Cr-Nb	Cr-Ni	Cr-P	Cr-Pd	Cr-Pt	Cr-Si
Cr-Sn	Cr-Ta	Cr-Ti	Cr-V	Cr-W	Cr-Zn	Cr-Zr	Cs-K	Cs-Na	Cs-Rb
Cu-Fe	Cu-Ge	Cu-In	Cu-Li	Cu-Mg	Cu-Mn	Cu-N	Cu-Nb	Cu-Ni	Cu-O
Cu-P	Cu-Pb	Cu-S	Cu-Sb	Cu-Si	Cu-Sn	Cu-Sr	Cu-Ti	Cu-Tl	Cu-V
Cu-Y	Cu-Zn	Cu-Zr	Dy-Er	Dy-Ho	Er-Ho	Er-Tb	Fe-Mg	Fe-Mn	Fe-Mo
Fe-N	Fe-Nb	Fe-Nd	Fe-Ni	Fe-O	Fe-P	Fe-Pb	Fe-Pd	Fe-Pr	Fe-Pt
Fe-S	Fe-Si	Fe-Sn	Fe-Ti	Fe-V	Fe-W	Fe-Zn	Fe-Zr	Ga-Ge	Ga-Hg
Ga-In	Ga-P	Ga-Pb	Ga-Sb	Ga-Sn	Ga-Te	Ga-Zn	Ge-In	Ge-Pb	Ge-Sb
Ge-Si	Ge-Sn	Ge-Te	Ge-Tl	Ge-Zn	H-Nb	H-Zr	Hf-Ta	Hf-Si	Hf-Ti
Ho-Tb	In-P	In-Pb	In-Sb	In-Si	In-Sn	In-Zn	Ir-Pd	K-Rb	La-Ni
Li-Mg	Li-Zr	Mg-Mn	Mg-Ni	Mg-O	Mg-SC	Mg-Si	Mg-Y	Mg-Zn	Mg-Zr
Mn-Mo	Mn-N	Mn-O	Mn-Pb	Mn-Si	Mn-Ti	Mn-V	Mn-Y	Mn-Zr	Mo-N
Mo-Nb	Mo-Ni	Mo-Si	Mo-Ta	Mo-Ti	Mo-W	N-Nb	N-Ni	N-Ta	N-Ti
N-V	N-W	Na-Rb	Nb-Ni	Nb-O	Nb-Ti	Nb-V	Nb-W	Nb-Zr	Nd-Pr
Nd-Sb	Ni-P	Ni-Pd	Ni-Si	Ni-Ta	Ni-Ti	Ni-V	Ni-W	Ni-Y	Ni-Zr
O-Pb	O-Sn	O-Sr	O-Ti	O-Y	O-Zr	P-Sb	P-Si	Pb-Pd	Pb-Sb
Pb-Si	Pb-Sn	Pb-Tl	Pb-Zn	Pd-Ru	Pd-Sn	Pr-Sb	Pt-Rh	Pt-Ru	Re-Ta
Re-W	Sb-Si	Sb-Sn	Sb-Zn	Se-Sn	Se-Te	Se-Tl	Si-Sn	Si-Ta	Si-Te
Si-Ti	Si-U	Si-V	Si-W	Si-Y	Si-Zn	Si-Zr	Sn-Ti	Sn-Zn	Sn-Zr
Ta-Ti	Ta-V	Ta-W	Ta-Zr	Te-Zn	Ti-V	Ti-W	Ti-Zr	V-Zr	Y-Zr

Exactly like in a normal solution database, a specially designed BIN-module solution database must contain all kinds of information about elements, species and phases, phase descriptions, constants, thermodynamic functions and parameters for elements, end-members (stoichiometric phases) and binary interaction pairs of solution phases, type definitions, default definitions, references, and so on.

Since there is no option to either reject or restore phases from a defined binary system, some unnecessary phases might need to be rejected (using the keyword sequence `DEFAULT_COMMAND REJECT_PHASE` in the setup file of the solution database) or simply deleted from the database.

Most importantly, a specific alloy solution database that can be directly used in the BIN Module of the TCC software (and in the Binary Phase Diagram Module of the TCCW software) must contain some descriptive information about all (or partially on some particularly suitable) binary systems (that have been critically assessed) in the entire chemical system of the database, and about the way that how to calculate binary phase diagrams for each binary system. Such information must be coded by the legal TDB keyword ASSESSED_SYSTEMS, and is normally given at the end of the setup file of the database. Moreover, the BIN module will only be able to calculate various types of phase diagrams (and thermodynamic function curves and phase-fraction plots) for the binary joints that are included in the ASSESSED_SYSTEM descriptive information block.

The keyword `ASSESSED_SYSTEMS` and the manners of coding the sequential descriptive information block are detailed in [Section 3.3.18 of the document Thermo-Calc Database Guide](#).

An example for giving the assessed binary system information in the Fe-Cr-Ni-C system is given below:

```
ASSESSED_SYSTEMS
C-FE(;P3*) C-NI(;P3*)
C-CR(;P3 TMM:273/6000 STP:.7/900/1/-1/2/-2 STP:.3/1500/2)
CR-FE(TDB -HCP ;G5 C_S:BCC/CR:VA ;P3 STP:.6/1200/1/-2/2) CR-NI(;P3*)
FE-NI(;P3 STP:.2/500/1/-1 STP:.1/1500/1/2/-2) !
```

In practice, any solution database that contains an appropriate description for the binary join we are looking for in the `ASSESSED_SYSTEM` descriptive information block can be used in the BIN module. Therefore, one can always code an `ASSESSED_SYSTEM` descriptive information block in an available ordinary solution database or in a user-specified solution database, and then use it in the BIN module for different purposes.

Such a solution database, together with the BIN module, is particularly interesting for systematic investigations of various types of phase diagrams and property diagrams (Gibbs energy curves, activity curves and phase-fraction plots) for binary systems which have been critically assessed, and is especially useful for teaching.

10.4.3 Demonstrative examples of the BIN-module calculations

There are two examples in the `\TCEX\` area (`TCEX1.TCM` and `TCEX13.TCM`) and in **TCCS Examples Book**, which illustrate that how various BIN-module calculations and graphical processing are performed:

Example 1	<code>TCEX1.TCM</code>	Demonstration of the main option <code>P</code> (phase diagram) calculation of the BIN module, as well as some further normal TCC command-line calculations and graphical processing of various types of binary phase diagrams and property diagrams; going through various on-line help features.
Example 13	<code>TCEX13.TCM</code>	Demonstration of the main option <code>P</code> (phase diagram) and <code>G</code> (Gibbs energy curves) calculations and graphical processing for binary systems.

10.5 TERN Module

10.5.1 Descriptions of the TERN module

Like binary phase diagrams, ternary phase diagrams are extensively used in various materials design and engineering, materials chemistry, chemistry, geochemistry, environmental chemistry, and so forth. A tremendous amount of experimental investigations of ternary alloy systems have been conducted over the last century, and such ternary diagrams are especially useful for alloy developments. The serial editions of the “*ASM Ternary Alloy Diagrams*”, published by ASM International based on available experimental data and some topological methods, have been widely applied by the materials R&D society. The Thermo-Calc software/database/interface package provides a more sophisticated way to investigate the phase diagrams and various thermodynamic properties of steels and alloys, as well as of many industrial and natural materials (ceramics, inorganic materials, organic substances, polymers, minerals, nuclear materials, etc.). It can precisely calculate and plot all kinds of phase diagrams and property diagrams, by using some well-established thermodynamic models and databases, powerful calculation and simulation tools, and easy-to-use modules, as described in the document *Thermo-Calc Software System* and many recent publications.

Using the traditional command-line routines in the TCC software, one can easily calculate and plot various types of phase diagrams (isothermal sections, monovariant lines or isothermal projection of liquidus surface, isopleths, vertical sections, *etc.*) in a ternary system (A–B–C), as summarized in *Table 8-1* in *Section 8.5.2*. This can be done going through its command-line structured modules, *i.e.*, to define the ternary system and get thermodynamic data (for all end-member properties, and binary and ternary interactions) in the TDB module, to (if necessary) manipulate phase descriptions and parameters in the GES module, to set calculation conditions and mapping variables in the POLY module, and to specify the graphical appearance in the POST module. In this way, one can also calculate and plot the phase diagram for a pseudo-ternary system (or called pseudo-ternary system, *e.g.*, CaO–MgO–SiO₂). Of course, the use of a MACRO file for a ternary phase diagram or property diagram calculation and graphical processing will make such command-line calculation and graphical processing much simple and efficient, for instance, the examples TCEX3.TCM, TCEX14.TCM, TCEX19.TCM, and TCEX37.TCM in the \TCEX\ area and the **TCCS Examples Book**.

To make the user interface much easier and more straightforward, a special and easy-to-use module, namely the TERN module, has been available since TCCM, for calculations of isothermal sections and monovariant lines (with respect to liquid phase) in ternary alloy systems (such as Fe-Cr-C and Al-Mg-Si). Since TCCR (and TCW4), this module has been further improved, by adding the main option L (Liquidus Surface) [in addition to the previously-existing main option P (Phase Diagram) and M (Monovariants)], in order to directly/automatically calculate and plot liquidus surface projections (monovariant lines, isothermal sections, as well as invariant reactions and their reaction types/temperatures; all with respect to liquid phase) of ternary alloy systems. Therefore, this module has become significantly powerful and efficient for calculations of ternary systems.

It is constructed as a simple question-based line structure, and the user just needs to provide some appropriate answers. Corresponding to its three main options (**P**, **M** and **L**), the module can directly and automatically calculate and plot three specific types of phase diagrams for a defined ternary alloys system:

- *Phase diagram:* utilizing the main option P (Phase Diagram) for an *isothermal section* of the entire ternary system at a specific temperature (in °C); automatically saving the workspace as the default file ISOTERN.POLY3; using two of the component compositions in the system as the X/Y-axes.
 - *Monovariant lines:* utilizing the main option M (Monovariants) for the *monovariant lines with regard to liquid phase* in the entire ternary system; automatically saving the workspace as the default file MONOVAR.POLY3; using two component compositions in the liquid phase as the X/Y-axes, and using liquidus temperature (in °C) as the projected Z-axis.
 - *Liquidus surface projections:* utilizing the main option L (Liquidus Surface) for the *liquidus surface projections (monovariant lines, isothermal sections, as well as invariant reactions and their reaction types/temperatures; all with regard to liquid phase)* in the entire ternary system;

- automatically saving the workspace as the default file MONOVAR.POLY3; in the meantime, those starting points for calculating isothermal sections involving liquid phase are saved into another POLY3 file LIQUIDUS.POLY3;
- automatically saving the compositions (in terms of mole-fractions of three system components) of all the phases involved at all the invariant reaction points (*i.e.*, LIQUID mixture plus three other phases), in the default file ISOTHER.EXP;
- using two component compositions in the liquid phase as the X/Y-axes, and using liquidus temperature (in °C) as the projected Z-axis.

Furthermore, using the TERN-module calculation results, many other types of phase diagrams and property diagrams can be generated, during the following graphical processing in the POST module (either immediately after the TERN-module calculation and default plotting are accomplished, or directly by re-opening a TERN-module POLY3 file). Yet, other normal POLY calculations and associated POST graphical processing of various property diagrams for a ternary system can be conducted using the saved or re-opened TERN-module POLY3 workspace (ISOTERN.POLY3, LIQUIDUS.POLY3, MONOVAR.POLY3 or other copied POLY3 files).

Note that for calculations of isopleths and vertical sections, the TERN module does not work; instead, one shall use the ordinary command-line structured modules (refer to *Section 8.5.2*).

Since TCCS, all the calculation routines for three TERN-Module options Phase Diagram (for calculating isothermal sections), Monovariants (for calculating monovariant lines involving liquid) and Liquid Surface (for calculating liquid surface projections) have been further modified and improved, so that the Global Minimization Technique can be used on a full scope.

The procedure of the TERN module is very simple, with a few steps to complete a calculation and graphical processing of an isothermal section, or monovariant lines (with regard to liquid), for a ternary system, as described below:

Step 1: Select a suitable solution database

The module asks for a suitable solution database from where all thermodynamic data will be retrieved. Basically, the end-member properties, binary and ternary interaction parameters, and all the assessed system information (at least for the ternary system to be calculated and plotted) must be available in the database. Such a solution database can be based on an ordinary solution database (implying that binary, ternary and high-order interaction data may be available), but normally is specially designed for the TERN module (*i.e.*, having appropriate descriptions for already-assessed ternary systems in the special ASSESSED_SYSTEM section; for more details see the next section, 10.5.2). The user can also switch to his/her own solution database(s) that meet the basic requirements for making the TERN-module calculations.

Since the TCCR/TCW4 version, the TERN module does not any long require a selected solution database necessarily having a description for a defined ternary system in the special ASSESSED_SYSTEM section (in such cases, a warning will be issued on screen, which the user should pay a special attention). This also means that: *the user should be crystal-clear about that the currently-chosen solution database must contain a completely-incorporated, critically-assessed and internally-consistent data set for the interested-in ternary system; otherwise, the calculated results may be entirely wrong or partially inappropriate.*

Step 2: Specify the ternary system

The names of the three elements (in one single line with one space in between, or in two or three separate lines) that are available in the currently-switched solution database must be given.

Step 3: Select a main option and sequentially its necessary sub-option(s)

The program asks (as two main options) if a phase diagram (**P**, as default; *for an isothermal section*), monovariant lines (**M**; *for monovariant lines involving liquid phase*), or liquidus surface projections (**L**; *for monovariant lines, isothermal sections, as well as invariant reactions and their reaction types, all involving liquid phase*) is to be calculated and plotted.

Sequentially, it may ask for some answers for some sub-options:

- For the main option P, the program asks for a specific temperature [in °C; for calculating the isothermal section of the ternary system at this temperature], *i.e.*,

Temperature (C) /1000/: <temperature for the isotherm>

- For the main option M, the program does not need to have any sub-option for preceding the automatic mapping calculations and graphical generation (see next step).

- For the main option L, the program requires 4 further inputs of sub-options [including the lower and upper temperatures (in °C), temperature interval for isotherms, and your decision on if wishing to utilize Global Minimization Technique for sequential calculations], *i.e.*,

Min temperature, C /25/: <lower temperature limit>

Max temperature, C /2500/: <upper temperature limit>

Temperature interval /100/: <temperature interval for isotherms>

Global minimization on: /N/: <Y or N>

Step 4: Automatically calculate and plot the desired diagram

The module automatically execute the following tasks:

- retrieving all the thermodynamic descriptions and information from the selected database;
- adding additional composition set(s) for solution phase(s) if the Global Minimization routine finds possible miscibility gap(s) for the phase(s), when the sub-option “Global minimization on” has been switched on (by answering Y) during the calculations of liquidus surface projections.
- defining the initial equilibrium conditions for the desired calculation [Note that for the main option P calculations, the specified temperature value is always set as a condition, and no phase should be set with a fixed phase status; while for the main option M calculations, it always set the phase status of the default liquid phase as fixed at one mole, and let the temperature vary along the monovariant lines; while for the main option L calculations, it always set the phase status of the default liquid phase as fixed at one mole, and have the temperature condition set as constant when calculating liquid-bearing isotherms];
- setting the mapping variables:
 - For all the main options P, M and L, always the mole-fractions of the second and third elements as the first and second mapping variables, respectively, and both axis variables range from 0.0 to 1.0; meanwhile, for the main options M and L, always set temperature condition as the third axis variable (using the aforementioned minimum/maximum temperatures and temperature interval as its settings) for mapping the monovariant lines on the liquidus surface projection.
- calculating the start points;
- making the mapping calculations:
 - For the main option P: The isothermal section is mapped.
 - For the main option M: The monovariant line(s) involving the liquid phase is mapped.
 - For the main option L: All isothermal sections possibly involving the liquid phase in the defined temperature range (under the specified temperature interval) are mapped, prior to the mapping of monovariant lines in the entire ternary system. At the end of mapping procedure, very useful information on all possible invariant reaction points on the liquidus surface projection (including their reaction temperatures, involved phases, and their reaction types) in the calculated ternary system are shown upon on screen, for instance:

INVARIANT REACTIONS:

```

U 1: 1571.35 C: LIQUID + M3C2 -> GRAPHITE + M7C3
U 2: 1295.20 C: LIQUID + M23C6 -> BCC_A2 + M7C3
E 1: 1284.49 C: LIQUID -> BCC_A2 + FCC_A1 + M7C3
P 1: 1210.67 C: LIQUID + GRAPHITE + M7C3 -> CEMENTIT
U 3: 1177.98 C: LIQUID + M7C3 -> CEMENTIT + FCC_A1
E 2: 1150.67 C: LIQUID -> CEMENTIT + FCC_A1 + GRAPHITE
  
```

- plot the first desired triangular diagram on the screen. This is the default output for the chosen main option (P or M or L), and its settings of axis variables are always as below:

- For the main option P:

$$\begin{aligned} X &= X(2^{\text{nd}}\text{El}) \\ Y &= X(3^{\text{rd}}\text{El}) \end{aligned}$$

Mole fraction of the second element
Mole fraction of the third element

- For the main option M or L:

$$\begin{aligned} X &= X(\text{LIQ}, 2^{\text{nd}}\text{El}) \\ Y &= X(\text{LIQ}, 3^{\text{rd}}\text{El}) \\ Z &= T-C \end{aligned}$$

Mole fraction of the second element in LIQUID phase
Mole fraction of the third element in LIQUID phase
Temperature (in °C), but expressed as a correlation between
the tic-mark scaled Z-axis values and projected liquidus
temperatures over the entire ternary composition space,
such as $Z\text{-axis} = 400 + 50*Z$

Default settings are also enforced on axis types, axis texts, axis scaling factors (always from 0.0 to 1.0 for all three axes), curve-labeling option (E or N for the main option P, F for the main option M, and N for the main option L), colors and thickness (for texts and axes, invariant equilibrium points, tie-lines, and other lines like for solubility lines), fonts and sizes (for labels and numbers), diagram title and subtitle, tic-marks (available for the main options M and L), and so on. Information on all possible invariant reaction points involving liquid phase (including their reaction temperatures, involved phases, and their reaction types) in the calculated ternary system are also given on the default plotted diagram for the liquidus surface projections.

The default diagram title for the main option P calculations is composed by the ternary system name and the temperature value for the calculation, appeared as “A-B-C at $T = t$ ”, where A, B, and C are element names arranged in alphabetic order and t is the temperature (in °C). The default diagram title for the main option M calculations is always as “Monovariant lines in A-B-C”, where A, B, and C are element names arranged in alphabetic order. The default diagram title for the main option L calculations is always as “A-B-C (lt/ht/it) xx s”, where A, B, and C are element names arranged in alphabetic order, lt/ht/it are lower and upper temperature limits (in °C) and temperature interval for the calculated/plotted isotherms on the liquidus surface projections, and xx is the real computation time (in second) for the current calculations.

The default subtitle for the main option M or L calculations is set as like “ $Z\text{-axis} = a + b*Z$ ”, where expresses the linear correlation between the labeled Z tic-marks and Z-axis value (the liquidus temperatures, in °C), and thus the a and b values are correlation constant and coefficient, respectively.

Figure 10-3 illustrates examples of three TERN-module generated diagrams for the Al-Mg-Si ternary system, corresponding to the main option P (isothermal section phase-diagram), M (monovariant lines curves involving liquid phase), and L (liquidus surface projections), respectively.

At the same time, a POLY3 file will be automatically saved in the current working directory, with the default name of ISOTERN.POLY3 for the main option P, and of MONOVAR.POLY3 for the main option M or L. For the main option L, another POLY3 file LIQUIDUS.POLY3 is also saved in the same area for only storing the workspace related to those starting points for calculating isothermal sections involving liquid phase (while this file is not useful). These POLY3 files can be re-opened in the normal POLY module, in order to conduct further graphical processing of other types of phase diagrams or property diagrams in the POST module, or to perform other types of mapping or stepping calculations directly in the POLY module. Note that such a file must be copied (if that specific POLY3 workspace is still in process) or renamed (if that POLY3 workspace has been quitted) to another file before making another TERN-module calculation with the same main option.

Furthermore, an EXP file with the default name of ISOTHER.EXP is automatically saved in the current working directory, which lists out (in the DATAPLOT Language format) various details about all possible invariant reaction points on the liquidus surface projection (including their reaction temperatures, reaction types involved phases and their compositions).

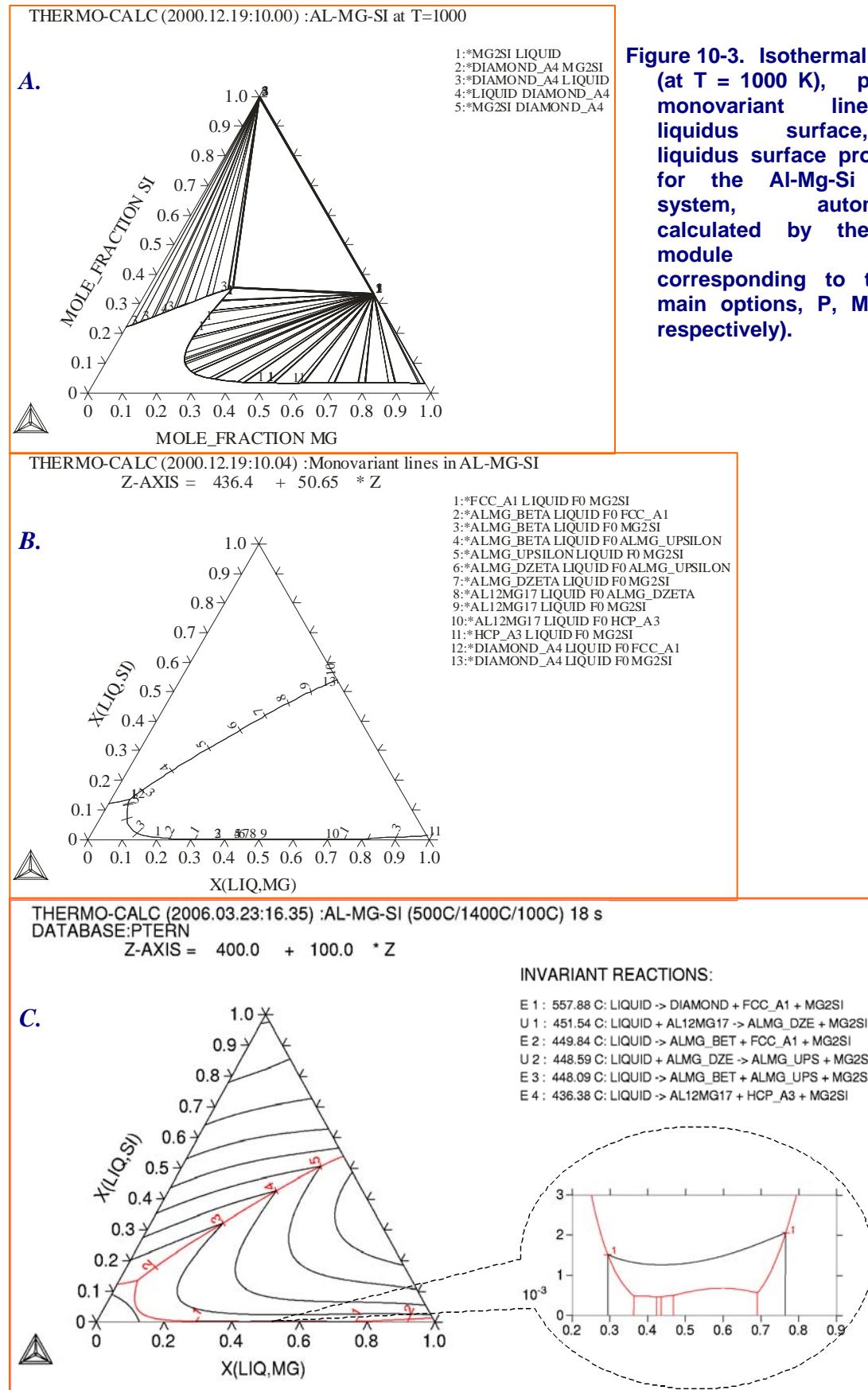


Figure 10-3. Isothermal section (at T = 1000 K), projected monovariant lines on liquidus surface, and liquidus surface projections for the Al-Mg-Si ternary system, automatically calculated by the TERN module (outputs corresponding to the two main options, P, M and L, respectively).

Step 5: Further refine the generated phase diagram or Gibbs energy curves or phase-fraction plot

Finally, the module always ends up in the normal POST module for further graphical processing and handling the automatically generated (first) diagram, as he/she desires.

The user can now apply all kinds of POST-module commands to refine the diagrams, by alternating diagram type (triangular or perpendicular), modifying and/or adding the settings for axis types, axis texts, axis scaling factors and manner, axis-tic status, tie-line status, curve-labeling option, phase-region or property labeling texts, colors and thickness, diagram title and subtitle, axis length, diagram size, and so forth.

One can also re-set the reference states for the components, suspend and/or restore phase(s) from/onto a plotted diagram, impose experimental data points on a calculated diagram, alternate the plot options, and specify a proper plot format.

After such further graphical processing, the user can choose to print the generated diagram on a connected printer, to save the diagram as a graphical file (using a preferable font and size), to dump the diagram to a graphical device (also saved as a graphical file), or to export the complete X-/Y-coordinates of the curves, lines and points, as well as some graphical settings, of the generated and modified diagram onto an EXP file. See *Chapter 9* for details of various POST commands.

Step 6: Further plot many other types of phase diagrams or property plots

If the user wants, he/she can generate many other types of phase diagrams or property diagrams in the POST-module monitor, using the same calculation results (saved in the currently running or re-opened ISOTHERN.POLY3, MONOVAR.PLOY3 or other copied TERN-module *.POLY3 files). The user shall define the appropriate axis variables for the new diagrams, and can now apply all kinds of POST-module commands to re-define and refine the diagrams exactly as described in the above step.

One can set the axis variables, by selecting suitable equilibrium quantities which are either state variables (among internal variables, energetic variables, compositional and constitutional variables) or as derived variables. One may also enter some additional symbols (functions or variables or tables) which can be used as new axis variables or used for tabulations.

Normally, a system composition variable (*e.g.*, moles N, mass B, mole-fraction M–F, mole-percent M–P, weight-fraction W–F, weight-percent W–P) or activity (AC or ACR) of any one of the three elements (components) in the entire ternary system can be set as the independent axis variable (the X- or Y-axis for diagrams). A phase composition variable (*e.g.*, moles N, mass B, mole-fraction X, weigh-fraction W) or activity (AC or ACR) of any one of the three elements (components) in a stable phase (for the main option M, normally only for the liquid phase) can also be used as the independent axis variable. However, if the main option M has been previously used in the calculation, the temperature condition, T (in K) or T–C (in °C), can also be set as the independent axis variable.

For defining the second axis variables, one can choose any other suitable equilibrium quantity. One may even define a third axis variable (the Z-axis) in the similar way as defining the second one, if a triangular diagram with the Z-axis projected is to be plotted in the POST module, or if a three-dimensional diagram is to be generated using proper graphical software.

The possibility of such additional diagrams can be summarized as below:

- After the main option P:

X/Y =	C/A(E1)	Composition/Activity of one element (component) in the system
	or C(ph,E1)	Composition of one element (component) in a stable phase
	or AC(sp,ph)	Activity of one element (species) in a stable phase
X/Y/Z =		Any equilibrium quantity of the system or component(s) or phase(s); corresponding its variation along the phase boundaries (and tie-lines, monovariant lines and invariant points)

- After the main option M or L:

X/Y	=	C/A(E1)	Composition/Activity of one element (component) in the system
	or	C(liq,E1)	Composition of one element (component) in the liquid phase
	or	AC(sp,liq)	Activity of one element (species) in the liquid phase
	or	T/T-C	Temperature (in K or °C)
X/Y/Z	=		Any equilibrium quantity of the system or component(s) or phase(s); corresponding its variation along the monovariant lines (with regard to the liquid phase).

Figure 10-4 illustrates some examples of such non-default diagrams plotted through graphical processing of the TERN-module saved *.POLY3 files for the Al-Mg-Si ternary system, which are generated by the three main options, P (phase diagram), M (monovariant lines) and L (liquidus surface projections), respectively.

Due to that in the main option P (saved in default ISOTERN.POLY3 files) and main option M or L (saved in default MONOVAR.POLY3 files), the mapping calculations are performed with the mole-fractions of last two components in the ternary system [*i.e.*, if the ternary system has been specified as in the A-B-C order, the X(B) and X(C) conditions will be chosen] as independent mapping variables (actually, in the main option M or L, the temperature condition is additionally chosen as the third mapping axis variable), one can usually to present the calculation results in various types of phase diagrams. A phase diagram for a ternary system can be plotted as isothermal triangular or perpendicular section with the X-/Y-axes presenting the system composition (mole-fraction M-F, or mole-percent M-P, or weight-fraction W-F, or weight-percent W-P of two elements *e.g.* components) or composition-dependent state variables (such as activities, AC or ACR, of two components), along the phase boundaries (and tie-lines, monovariant lines and invariant points) at the defined temperature for the main option P, or along the monovariant lines (*i.e.*, where the liquid phase is stable together with two solid phases) on the liquidus surface projection at varied temperatures (which can be plotted as the Z-axis) for the main option M or L.

If only one of such variables is set as one axis and another specific equilibrium property of the system or component(s) is set as the other axis, then a property diagram is plotted. In such a property diagram, the curves/lines/points will only present the variations of that specific equilibrium property along the phase boundaries (and tie-lines, monovariant lines and invariant points) at the defined temperature for the main option P, or along the liquidus monovariant lines where the liquid phase is stable together with two solid phases at varied temperatures for the main option M or L.

Moreover, some thermodynamic properties and equilibrium quantities which are already defined (*e.g.*, standard state variables or in a table) can be tabulated and plotted, along one mapping variable (the main option P or M or L).

10.5.2 Structures of special TERN-module databases

As described above, the TERN module automatically retrieves all thermodynamic data from a suitable solution database, in which *a completely-incorporated, critically-assessed and internally-consistent data set (at least for the currently interested-in ternary system to be calculated and plotted; including data for standard thermodynamic properties of various unary/binary/ternary phases and binary/ternary interaction parameters) must be available; otherwise, the calculated results may be entirely wrong or partially inappropriate.*

Previously (in older versions before TCCR/TCW4), this is ensured through enforcedly applying the appropriate information coded in the ASSESSED_SYSTEM section in the chosen solution database. Normally, such a solution database is specially designed for the TERN module, but it can also be an ordinary solution database (implying that binary, ternary and high-order interaction data exist) with an additional ASSESSED_SYSTEM information block (see below). The user can also switch to his/her own solution database(s) that meet the basic requirements for making the TERN-module calculations.

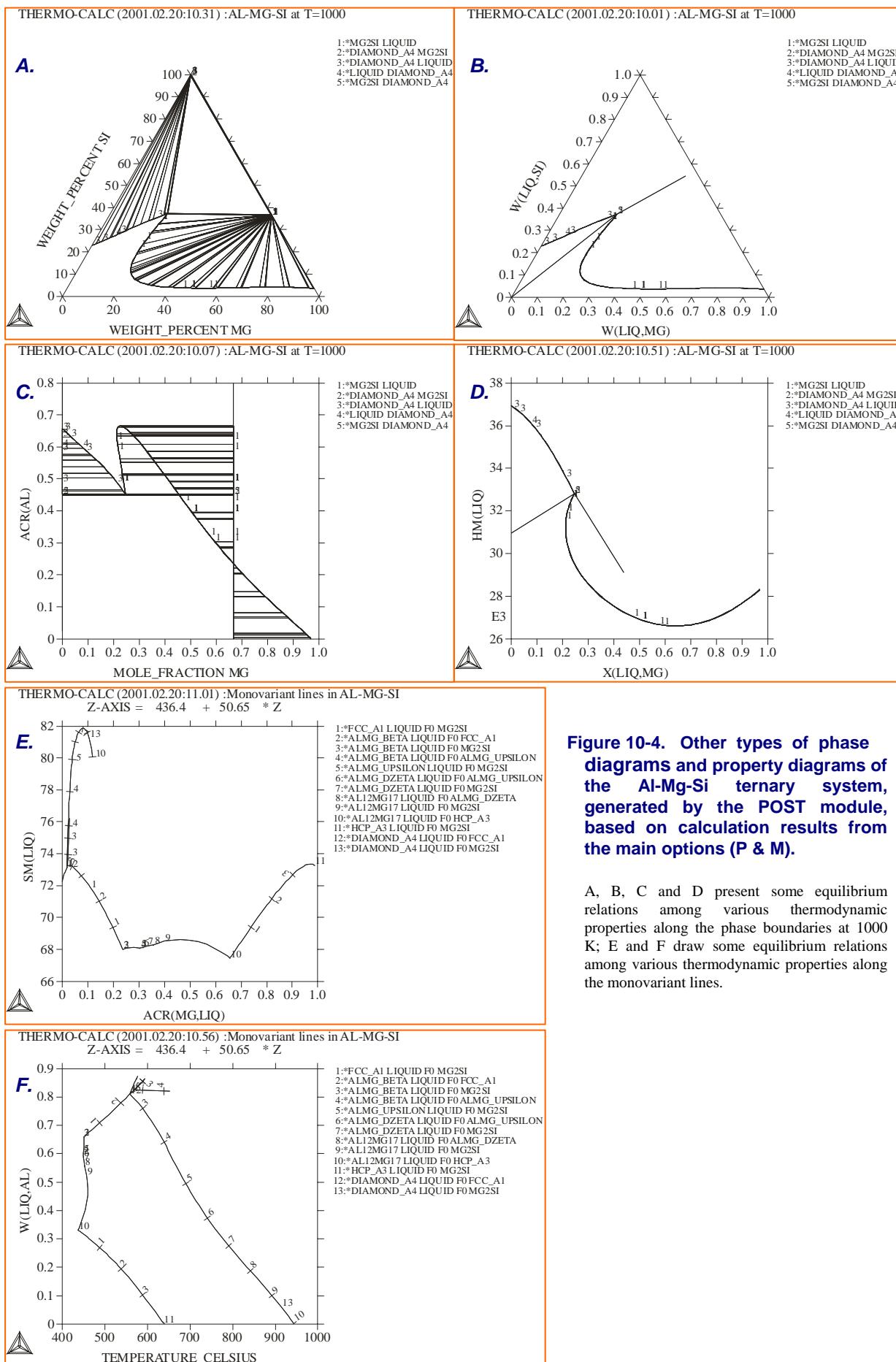


Figure 10-4. Other types of phase diagrams and property diagrams of the Al-Mg-Si ternary system, generated by the POST module, based on calculation results from the main options (P & M).

A, B, C and D present some equilibrium relations among various thermodynamic properties along the phase boundaries at 1000 K; E and F draw some equilibrium relations among various thermodynamic properties along the monovariant lines.

At present moment, there is one specially designed ternary alloy solution database for direct use in the TERN module, as detailed below:

➤ **PTERN TC Public Ternary Alloy Solutions Database** (previously called TER98)

- freely distributed with TCC and TCW, and with TCC_Demo/TC4A and TCW-Demo/TC4U
- covering 7 elements (Al-Cr-Fe-Mg-Si-V)
- containing the following 3 ternary systems (for liquids and solid solutions, and many stoichiometric phases)

Al-Mg-Si,	C-Cr-Fe,	C-Fe-V
-----------	----------	--------

Exactly like in a normal solution database, a specially designed TERN-module solution database must contain all kinds of information about elements, species and phases, phase descriptions, constants, thermodynamic functions and parameters for elements, end-members (stoichiometric phases), binary and ternary interaction joints of solution phases, type definitions, default definitions, references, and so on. Since there is no option to either reject or restore phases from a defined ternary system, some unnecessary phases might need to be rejected (using the keyword sequence `DEFAULT_COMMAND REJECT_PHASE` in the `setup` file of the solution database) or simply deleted from the database.

Importantly, an alloy solution database that can be directly used in the TERNARY module shall contain some descriptive information about all the ternary systems (that have been critically assessed) in the entire chemical system of the database, and about the way that how to calculate ternary phase diagrams for each ternary system. Such information must be coded by the legal TDB keyword ASSESSED_SYSTEMS, and is normally given at the end of the setup file of the database. Moreover, the TERNARY module will only be able to calculate various types of phase diagrams (and monovariant lines) for those ternary systems that are included in the ASSESSED_SYSTEM descriptive information block.

The keyword `ASSESSED_SYSTEMS` and the manners of coding the sequential descriptive information block are detailed in [Section 3.3.18 of the document Thermo-Calc Database Guide](#). An example for giving the assessed ternary system information in the Al-Cr-Fe-Mg-Si system is given below:

`ASSESSED_SYSTEM AL-MG-SI(;P3*) C-CR-FE(;P3*) !`

In practice, any solution database that contains an appropriate description for assessed ternary alloy systems we are looking for in the `ASSESSED_SYSTEM` descriptive information block can be used in the TERN module. Therefore, one can always code an `ASSESSED_SYSTEM` descriptive information block in an available ordinary solution database or in a user-specified solution database, and then use it in the TERN module for different purposes.

Such a solution database, together with the TERN module, is particularly interesting for systematical investigations of various types of phase diagrams (and liquidus surface projection, plus monovariant lines and invariant reaction points inside the liquid surface projection) for ternary systems which have been critically assessed, and is especially useful for teaching. Please note that the PTERN database is only an exemplary database with some published data, primarily aimed at demonstrating the module. We highly recommend the users to build their own databases/datasets for the TERN-module calculations.

However, since TCCR/TCW4, this requirement is somewhat ignored, i.e., the TERN module does not any long require a selected solution database necessarily having a description for a defined ternary system in the special ASSESSED_SYSTEM section (in such case a warning will be issued on screen, which the user should pay a special attention). This also means that: the user should be crystal-clear about that the currently-chosen solution database must contain a completely-incorporated, critically-assessed and internally-consistent data set for the interested-in ternary system; otherwise, the calculated results may be entirely wrong or partially inappropriate.

10.5.3 Demonstrative examples of the TERN-module calculations

There is one example in the `\TCEX\` area (`TCEX3.TCM`) and in **TCCS Examples Book**, which illustrates that how various TERN-module calculations and graphical processing are performed:

Example 3 `TCEX3.TCM`

Demonstration of the TERN-module main option `P` (phase diagram) calculation for the C-Cr-Fe ternary system, as well as some further normal TCC command-line calculations and graphical processing of various types of ternary phase diagrams and property diagrams.

10.6 POTENTIAL Module

10.6.1 Descriptions of the POTENTIAL module

In a metal-gas interaction system, the so-called potential diagrams (plotted with fugacities of two major gaseous species as the X/Y-axes) at certain temperature and pressure conditions are useful for investigations of the formations of various forms of metal-involving oxides, sulfides, sulfates, carbonates, silicates, nitrides, nitrates, etc., and their relations with the interacting gaseous mixture phase. Using the traditional command-line routines in the TCC software, one can easily calculate and plot various types of potential diagrams in a metal-gas interaction system, as described in *Chapter 8*.

Since TCCL, a special and easy-to-use module, namely the POTENTIAL module, can automatically calculate and plot such potential diagrams at specified temperatures and a pressure of 1 bar. It is constructed as a simple question-based line structure, and the user just needs to provide some appropriate answers.

It uses the chemical potentials or activities of two major species in the gaseous mixture phase as the mapping variables, and then plot the potential diagram with the activities (*i.e.*, fugacities at 1 bar) of the two gaseous species (in natural logarithm) as the X/Y-axes. The phase relations, between the gaseous mixture and various metal forms, metal-oxides, metal-sulfides, or other metal-bearing solids, are represented by different fields controlled by these two potential quantities.

Furthermore, many other types of potential diagrams and property diagrams (for a defined metal-gas interaction system at the specified temperature and 1 bar) can be generated using the POT-module calculation results. This is done during the following graphical processing in the POST module (either immediately after the POTENTIAL-module calculation and default plotting are accomplished, or directly by re-opening a POTENTIAL-module *.POLY3 file). Yet, other normal POLY calculations and associated POST graphical processing of various potential diagrams and property diagrams can be conducted using the saved or re-opened POT-module POLY3 workspace (POT.POLY3 or other copied *.POLY3 files). This is particular useful when a potential diagram at a pressure condition rather than 1 bar is calculated: the user can go to the POLY module, set the pressure condition (*e.g.*, s-c P=1e8 for 1 kbar), calculate an initial equilibrium and add it in the default direction, perform the mapping calculations, and then go back to the POST to plot the potential diagram and other property plots (see Step 6).

The procedure of the POTENTIAL module is very simple, with a few steps to complete a calculation and graphical processing of the default potential diagram and other plots, as described below:

Step 1: Select a suitable substance or solution database

The module asks for a suitable substance or solution database from where all thermodynamic data will be retrieved. Any substance or solution database with a gaseous mixture phase, metals (or alloys), metal-oxides/sulfides/sulfates/carbonates/silicates/nitrides/nitrates/... solids (stoichiometric or solution phases) can be selected in the POTENTIAL module, such as the SSUB, TCMP, and TCES databases. The specially designed SPOT database is a subset of the SSUB Substances Database, and another special database, PSUB (TC Public substance database), are developed for the purposes of demonstrations of the POTENTIAL-module calculations and graphical processing (see more details in the next section, 10.6.2). The user can also switch to his/her own substance or solution database(s) that meet the basic requirements for making the POTENTIAL-module calculations.

Step 2: Specify the metal-gas interaction system

The names of the matrix element (metal) and two gaseous species (in one single line with one space in between, or in three separate lines) that are available in the currently selected substance or solution database must be given. The names of the two gaseous species as given here must be exactly the same as in the definitions of such species and of the constituents of the gaseous mixture phase in the database.

Step 3: Select a temperature

The program asks for a specific temperature (in K) for calculating the potential diagram for the defined metal-gas interaction system at this temperature and a pressure of 1 bar.

Step 4: Automatically calculate and plot the desired diagram

The module automatically execute the following tasks:

- retrieving all the thermodynamic descriptions and information from the selected database;
- defining the initial equilibrium conditions for the calculation;
- setting the mapping variables [as the LNACR (SP1, GAS) and LNACR (SP2, GAS)];
- calculating the start points;
- making the mapping calculations;
- plot the first desired potential diagram, LNACR (SP1, GAS) *verse* LNACR (SP2, GAS), on the screen.

Default settings are also made on axis types, axis texts, axis scaling factors (always from -100.0 to 0.0 for both axes), curve-labeling option (normally as B), colors and thickness (for texts and axes, invariant equilibrium points, tie-lines, and other lines like for solubility lines), fonts and sizes (for labels and numbers), diagram title, tic-marks, and so on.

The default diagram title for the main option P calculations is composed by the ternary system name and the temperature value for the calculation, appeared as “Me SP1 SP2; Database: ABCD”, where Me, SP1, and SP2 are the names of the specified metal and two major gaseous species, and ABCD is the name of the selected database.

Figure 10-5 illustrates an examples of the POTENTIAL-module generated diagrams for the interaction between Fe and SO₂/O₂-bearing gaseous mixture phase at 1000 K and 1 bar.

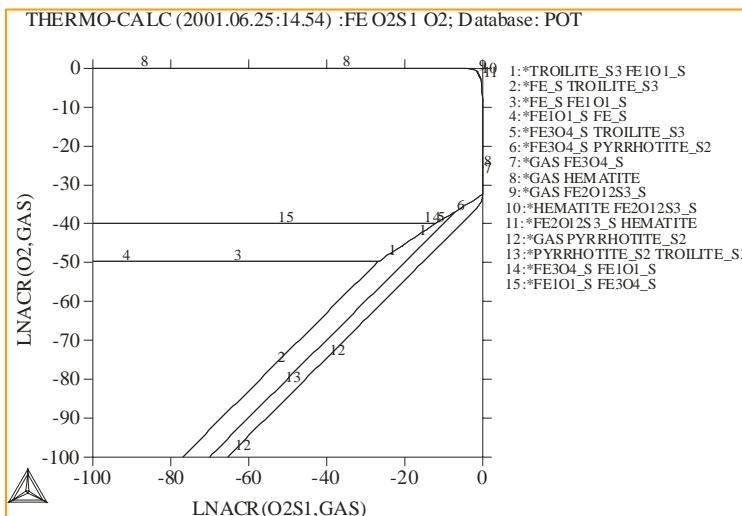


Figure 10-5. Potential diagram of the interaction between Fe and SO₂/O₂-bearing gas at 1000 K and 1 bar, that is automatically calculated and plotted by the POT module.

At the same time, an *.POLY3 file will be automatically saved in the current working directory, with the default name POT.POLY3. This file can be re-opened in the normal POLY module, in order to conduct further graphical processing of other potential diagrams or property plots in the POST module, or to perform other types of mapping calculations directly in the POLY module (for instance, change the pressure and temperature conditions to make a potential diagram; see Step 6). Note that such a file must be copied (if that specific POLY3 workspace is still in process) or renamed (if that POLY3 workspace has been quitted) to another file before making another POTENTIAL-module calculation.

Step 5: Further refine the generated potential diagram

Finally, the module always ends up in the normal POST module for further graphical processing and handling the automatically generated (first) diagram, as he/she desires.

The user can now apply all kinds of POST-module commands to refine the diagrams, by alternating diagram type (triangular or perpendicular), modifying and/or adding the settings for axis types, axis texts, axis scaling factors and manner, axis-tic status, tie-line status, curve-labeling option, phase-region or property labeling texts, colors and thickness, diagram title and subtitle, axis length, diagram size, and so forth.

One can also re-set the reference states for the components, suspend and/or restore phase(s) from/onto a plotted diagram, impose experimental data points on a calculated diagram, alternate the plot options, and specify a proper plot format.

After such further graphical processing, the user can choose to print the generated diagram on a connected printer, to save the diagram as a graphical file (using a preferable font and size), to dump the diagram to a graphical device (also saved as a graphical file), or to export the complete X-/Y-coordinates of the curves, lines and points, as well as some graphical settings, of the generated and modified diagram onto an *.EXP file. See *Part 9* for details of various POST commands.

Step 6: Further plot many other types of potential diagrams or property plots

If the user wants, he/she can generate many other types of potential diagrams or property plots in the POST-module monitor, using the same calculation results (saved in the currently running or re-opened POT.POLY3 file). The user shall define the appropriate axis variables for the new diagrams, and can now apply all kinds of POST-module commands to re-define and refine the diagrams exactly as described in the above step.

One can set the axis variables, by selecting suitable equilibrium quantities which are either state variables (among internal variables, energetic variables, compositional and constitutional variables) or as derived variables. One may also enter some additional symbols (functions or variables or tables) which can be used as new axis variables or used for tabulations.

Normally, a chemical potential (*e.g.*, MU or MUR, or in natural logarithm) or activity (AC or ACR, or in natural logarithm) of any one of the two major gaseous species (SP1 or SP2) in the interaction system can be set as the independent axis variable (the X- or Y-axis for diagrams). For defining the second axis variables, one can choose any other suitable thermodynamic property or equilibrium quantity.

A potential diagram at a pressure condition rather than 1 bar, and/or at another temperature condition, can be calculated and plotted in this step, as described here:

```

POST: back (to the POLY module; if not, use the GO POLY command sequence)
POLY_3: read POT.POLY3
        (By reading an existing POT-module POLY3 file, it makes sure that LNAC(SP1,GAS)
         and LNAC(SP2,GAS) are still as two of the conditions, and as the two mapping
         variables)
POLY_3: s-c P=1e8
POLY_3: s-c T=800
POLY_3: c-e
POLY_3: add default
POLY_3: map
POLY_3: post
POST: plot

```

Figure 10-6 illustrates some examples of such non-default diagrams plotted through graphical processing of the POTENTIAL-module saved *.POLY3 files for the Fe-S-O system.

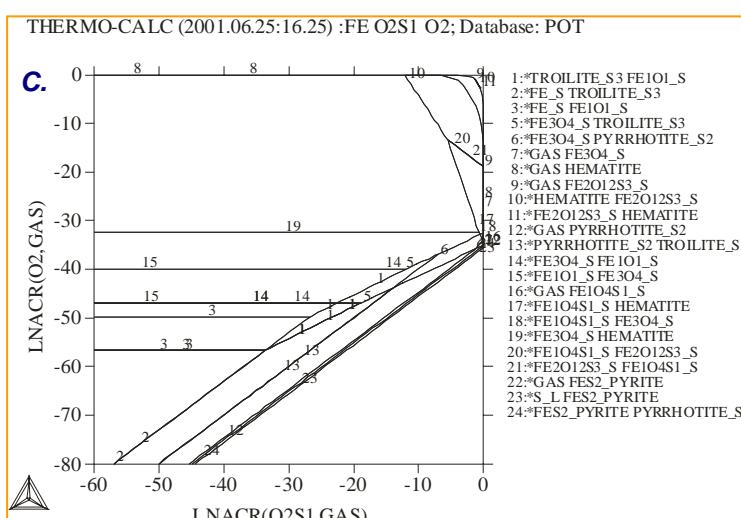
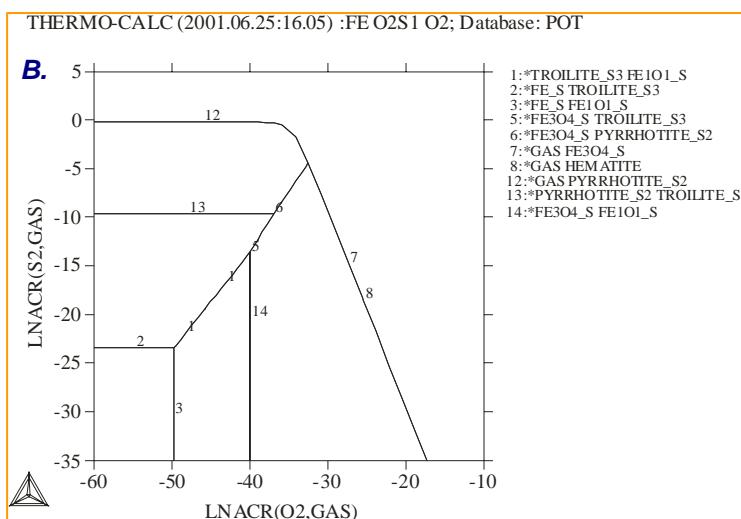
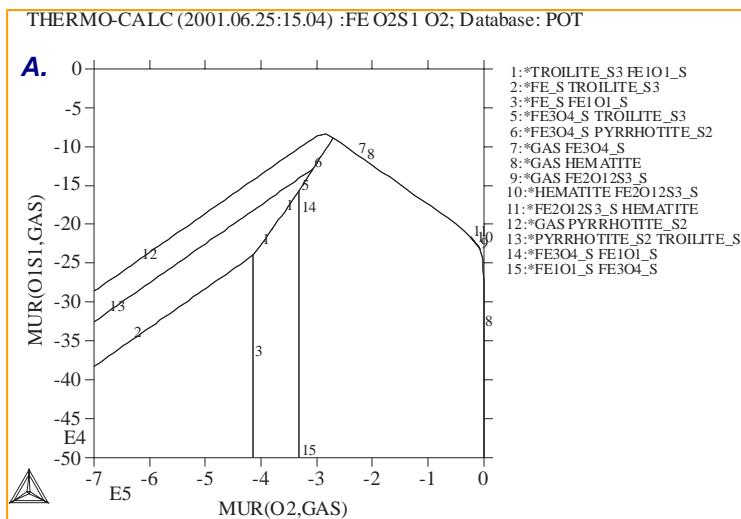


Figure 10-6. Other types of potential diagrams and property plots of the Fe-S-O system, generated by the POST module (directly after the POT-module calculation), or recalculated by the POLY module at different pressures and/or temperatures.

A and D present some other types of potential diagram generated by the POST module, based on the POTENTIAL-module calculation results at 1000 K and 1 bar; C draws a potential diagram calculated by the POLY module (based on an existing POT.POLY3 file) but with modified temperature- pressure conditions (800 K and 1 kbar).

10.6.2 Structures of the databases used by the POTENTIAL-module

As mentioned above, the POTENTIAL module uses any substance or solution database which meets the basic requirement that there must be a gaseous mixture phase, metals (or alloys), metal-oxides/sulfides/sulfates/carbonates/silicates/nitrides/nitrates/... solids (stoichiometric or solution phases). In this regard, it is different from the BIN, TERN and POURBAIX modules. Therefore, the SSUB, TCMP and TCES databases can be generally selected for potential diagram calculations.

The specially designed SPOT database is a subset of the SSUB Substances Database, and another special database, PSUB (TC Public Substances Database), are developed for the purposes of demonstrations of the POT-module calculations and graphical processing. They are briefly described below:

➤ **PSUB** *TC Public Substances Database*

- freely distributed with TCC and TCW, and with TCC_Demo/TC4A and TCW-Demo/TC4U
- covering 6 elements (Fe-Cu-H-O-S-N)

➤ **SPOT3** *A subset of the SGTE SSUB Substances Database v3* (previously called POT)

- only distributed together with a formerly licensed SSUB database
- covering 101 elements (same as in the *SGTE SSUB Substance Database, version 3*)

The user can also switch to his/her own substance or solution database(s) that meet the basic requirements for making the POTENTIAL-module calculations.

10.6.3 Demonstrative examples of the POTENTIAL-module calculations

There is one example in the \TCEX\ area (TCEX35.TCM) and in **TCCS Examples Book**, which illustrate that how the POTENTIAL-module calculations and graphical processing are performed:

Example 35 TCEX35.TCM Demonstration of the POTENTIAL-module calculation for the Fe-O-S system at 1000 K and 1 bar.

10.7 POURBAIX Module

10.7.1 Pourbaix diagram

In a heterogeneous interaction system involving an aqueous solution, the speciation and partition in aqueous solution and various interacting phases (gas mixture, stoichiometric solids and solid solutions) depend on the acidity (pH) and electronic potential (Eh), besides other controlling factors (bulk composition, temperature, pressure and so on) in the system. To present the speciation and partition in various phases as functions of pH and/or Eh conditions in an equilibrium state or dynamic process is very interesting to R&D in aqueous chemistry, electrolyte solution chemistry, materials corrosion, chemical engineering, geochemistry, environmental chemistry, etc.

Pourbaix (1966) attempted to apply thermodynamics to predict materials corrosion resistance, by determining the phase stability relations in terms of varied pH and Eh values in an interaction system between metal and pure water or dilute aqueous solution, and thus presenting the stability regions of metal and secondary phases (metal-oxides) on a pH-Eh diagram, the so-called Pourbaix diagram. Pourbaix labeled a diagram as three regions:

- ◆ **Immunity** - No metal dissolution;
- ◆ **Corrosion** - Active metal dissolution;
- ◆ **Passivation** - Formation of a protective metal-oxide film prevents further metal dissolution.

So a Pourbaix diagram is a kind of phase diagram, showing the stability boundaries for a metal-aqueous system, by plotting the phase boundaries as a function of pH and Eh. An aqueous solution phase is always present; however, at a given pH and Eh a metal may either be in equilibrium with the aqueous solution (insoluble/immune) or in equilibrium with an oxide or sulfide or other formed secondary solid-phase film which may provide passivation. On the other hand, a pre-dominance or speciation-percentage diagram is, however, a property diagram, showing the dominant aqueous species or aqueous speciation percentage in the heterogeneous equilibrium system. Such diagrams can also be projected onto the Pourbaix diagram if required, to see how aqueous speciation is varied along the phase boundaries and within phase stability fields.

Figure 10-7 gives an example of the Pourbaix diagram of Fe, in a system with 0.001 mole of Fe in 1 kg of pure water at 25°C and 1 bar, where gas phase is not considered in the calculation. The formation of magnetite (Mt, Fe_3O_4) and hematite (Hm, Fe_2O_3), which are coexisting with the Fe-containing dilute aqueous solution, represent the passivation of iron in the upper-right pH-Eh field. When pH value is low and Eh relatively high, Fe will complete dissolve into the aqueous solution phase, implying the corrosion of Fe. At low Eh conditions, Fe remains in its stable solid state (the BCC phase), and it neither dissolves into water nor alternates to Fe-oxides, standing for the immunity of iron.

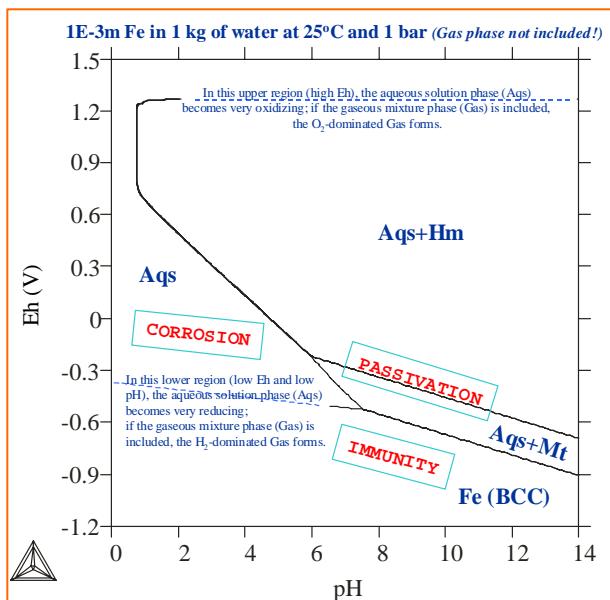
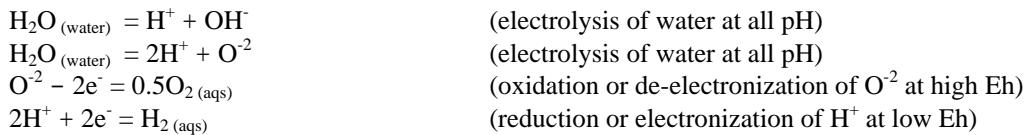


Figure 10-7. Pourbaix diagram of Fe (with 0.001 mole Fe in 1 kg of pure water at 25°C and 1 bar; gas excluded in the calculation).

Note that the gas mixture phase is not included in the calculated system. In the upper region (high Eh) and lower region (low Eh and low pH), the aqueous solution phase becomes very oxidizing and reducing, respectively.

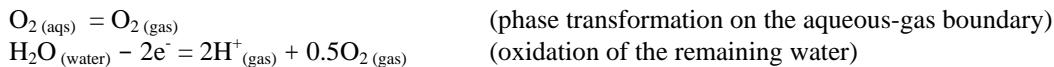
In reality, accordingly, from a global equilibrium point of view, an O_2 - or H_2 -dominated gas mixture phase would form in these two regions (See *Figure 10-8* for a complete pH-Eh diagram).

There is one important respect regarding a realistic Pourbaix diagram for a truly equilibrated system, as indicated on *Figure 10-7*, and described below. In reality, the water component is electrolyzed to H^+ and O^{2-} at all pH conditions, but the electrolysis degree depends on the pH value in the aqueous solution phase. As Eh increases and gets high enough, the O^{2-} ion will be oxidized to $O_2(aq)$. On the contrary, as Eh decreases and falls low enough, the H^+ ion will be reduced to $H_2(aq)$, through the major electrolysis and redox reactions:

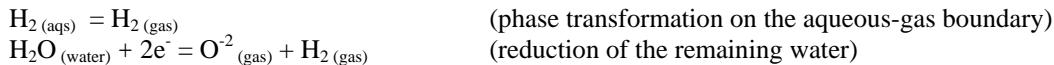


Accordingly, if one does not consider a gaseous mixture phase in calculating a Pourbaix diagram, the aqueous solution phase may end up with an extremely high $O_2(aq)$ concentration at high Eh condition, or an extremely high $H_2(aq)$ concentration at low Eh condition, where the concept of "aqueous solution phase" is no longer valid and consequently no proper aqueous solution model can actually applied. Therefore, from a restrict thermodynamic equilibrium point of view, one has to include a gaseous mixture phase in the system for the Pourbaix diagram calculation.

At a critically high value under a given pH condition, the aqueous solution phase with a high enough O_2 activity becomes less stable than an O_2 -dominated gas mixture phase, and thus the gas phase replaces the aqueous solution phase in the system, through the following major-species reactions along all other reactions involving minor oxidized and neutral species:



Similarly, at a critically low value under a given pH condition, the aqueous solution phase with a high enough H_2 activity becomes less stable than an H_2 -dominated gas mixture phase, and thus the gas phase replaces the aqueous solution phase in the system, through the following major-species reactions along all other reactions involving minor reduced and neutral species:



Of course, the phase transformation from an aqueous solution phase to an O_2 - or H_2 -dominated gas mixture phase depends upon the total molar Gibbs energies of the phases which are complicated functions of the phase constitutions, temperature and pressure. The Gibbs energy minimization technique in the Thermo-Calc software ensures the correct determination of such phase transformations.

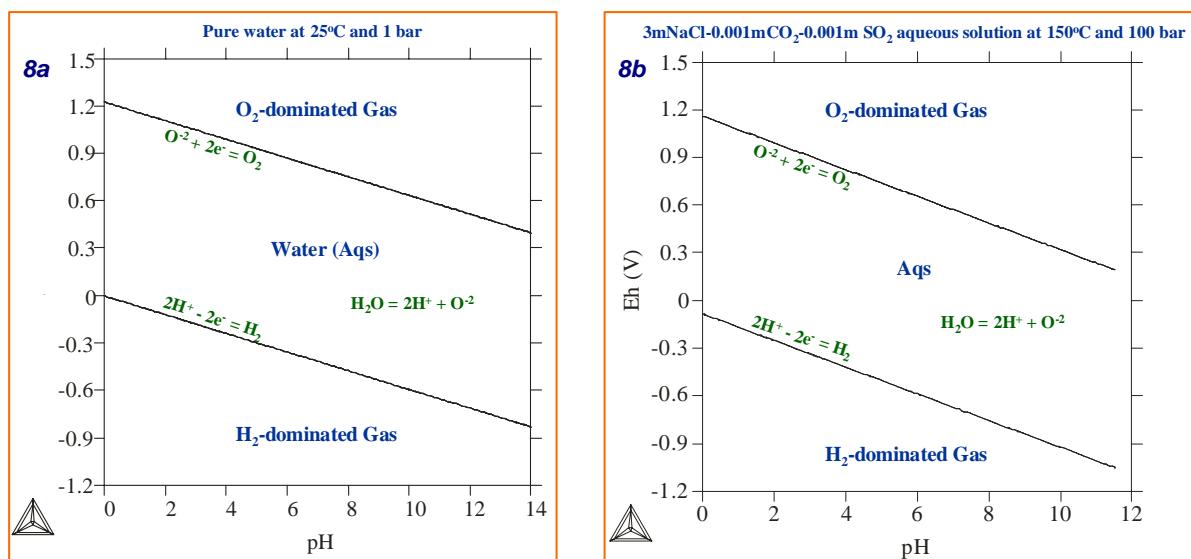


Figure 10-8. The pH-Eh diagrams for pure water at 25°C and 1 bar (in 8a), and for a 3mNaCl-0.001mCO₂-0.001mSO₂ aqueous solution at 150°C and 100 bar (in 8b). Note that gas is included in the calculation.

In Figure 10-8, the gaseous mixture phase is included in the calculated system. In the upper region (high Eh) and lower region (low Eh), the aqueous solution phase (water) is oxidized and reduced, respectively, to O₂⁻ and H₂-dominated gas phase. Comparing the two diagrams, one can see that the upper and lower boundaries between the aqueous phase (water) and gas phase (either dominated by O₂ or H₂) may shift, when some solutes dissolve in the aqueous solution and when the temperature-pressure conditions change.

Therefore, with such redox reactions considered in the equilibrium system, *i.e.*, the gaseous mixture phase is included in the calculation, the Pourbaix diagram of Fe in a system with 0.001 mole of Fe in 1 kg of pure water at 25°C and 1 bar (Figure 10-9) appears differently from that where the gas phase is excluded (Figure 10-7). It represents the complete Pourbaix diagram for the system, in a sense of restrictively thermodynamic equilibrium.

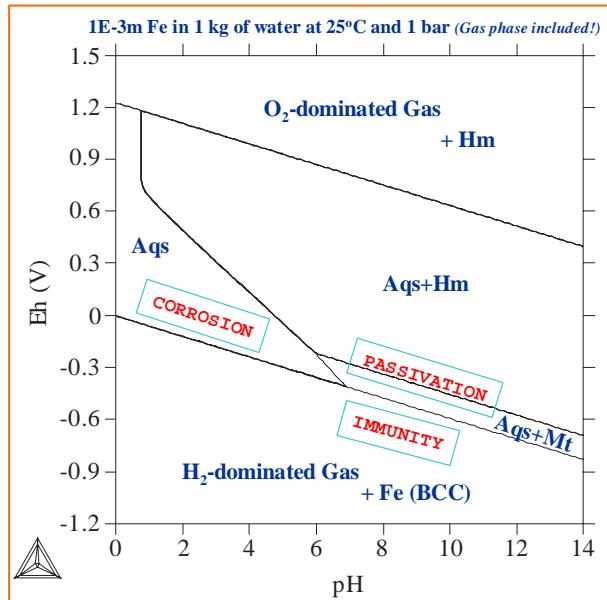


Figure 10-9. Pourbaix diagram of Fe (with 0.001 mole Fe in 1 kg of pure water at 25°C and 1 bar; gas included in the calculation).

Note that the gas mixture phase is included in the calculated system. In the upper region (high Eh) and lower region (low Eh), the aqueous solution phase is oxidized and reduced, respectively, to O₂⁻ and H₂-dominated gas phase. Hematite also coexists with the O₂-dominated gas, while Fe-BCC is stable together with the H₂-dominated gas. The stability field of magnetite may extend slightly towards in the H₂-dominated gas region (where it coexists with gaseous mixture, instead of aqueous solution), but its boundary with Fe (BCC) can not be drawn, due to the absence of aqueous solution phase.

Note that in both the O₂- and H₂-dominated gas phase stability region where aqueous solution phase is absent, it is impossible to calculate the pH value, due to its definition $pH = -\log_{10} [ACR(H+1)]$ and to that the reference state of the components H+1 (always as H⁺ in aqueous solution phase). Therefore, on a normal Pourbaix diagram, no line is drawn for any possible phase boundary between the gas phase and metal or secondary phases (*i.e.*, metal-oxides).

Furthermore, depending on the bulk chemical composition in a heterogeneous interaction system, the secondary phases (that formed along the interaction process) can not only be oxides, but also hydroxides, sulfides, sulfates, carbonates, halides, *etc.*, which may or may not bear metal(s). In this sense, one may be interested in calculating the Pourbaix diagram for an interaction system which consists of a multicomponent alloy (or other materials), a complex aqueous solution (such H₂O-NaCl-KF-CaSO₄-MgCO₃-...) and an associated gaseous mixture phase, and thus the formations of various secondary phases (*e.g.*, oxides, hydroxides, sulfides, sulfates, nitrates, silicates, carbonates, halides, *etc.*, which possibly form during the hydrolysis, oxidation, reduction and other reaction processes) in different pH-Eh regions will be valuable information to understand the passivation behaviors in the corrosion processes of the materials.

Another important concept regarding Pourbaix diagrams is that a pH-Eh plot is always related to an interaction system in which a certain amount of initial alloys (or other condensed materials) has been **actively** reacted with an aqueous solution initially comprising of normally 1 kg of water and specified solute concentrations at defined temperature and pressure conditions. By “actively”, it means that such an amount of condensed material is fully in equilibrium with the defined aqueous solution phase. When the bulk composition of the interaction system varies, the Pourbaix diagrams will be different at certain extents. For instance, in Figure 10-9 (and Figure 10-7), 0.001 m Fe is considered in the equilibrium calculation as the initial Fe composition that has actively reacted with 1 kg of pure water at 25°C and 1 bar, while in Figure 10-10 other initial Fe compositions are used, or the interacting aqueous solution composition is alternated, or the temperature and/or pressure are changed.

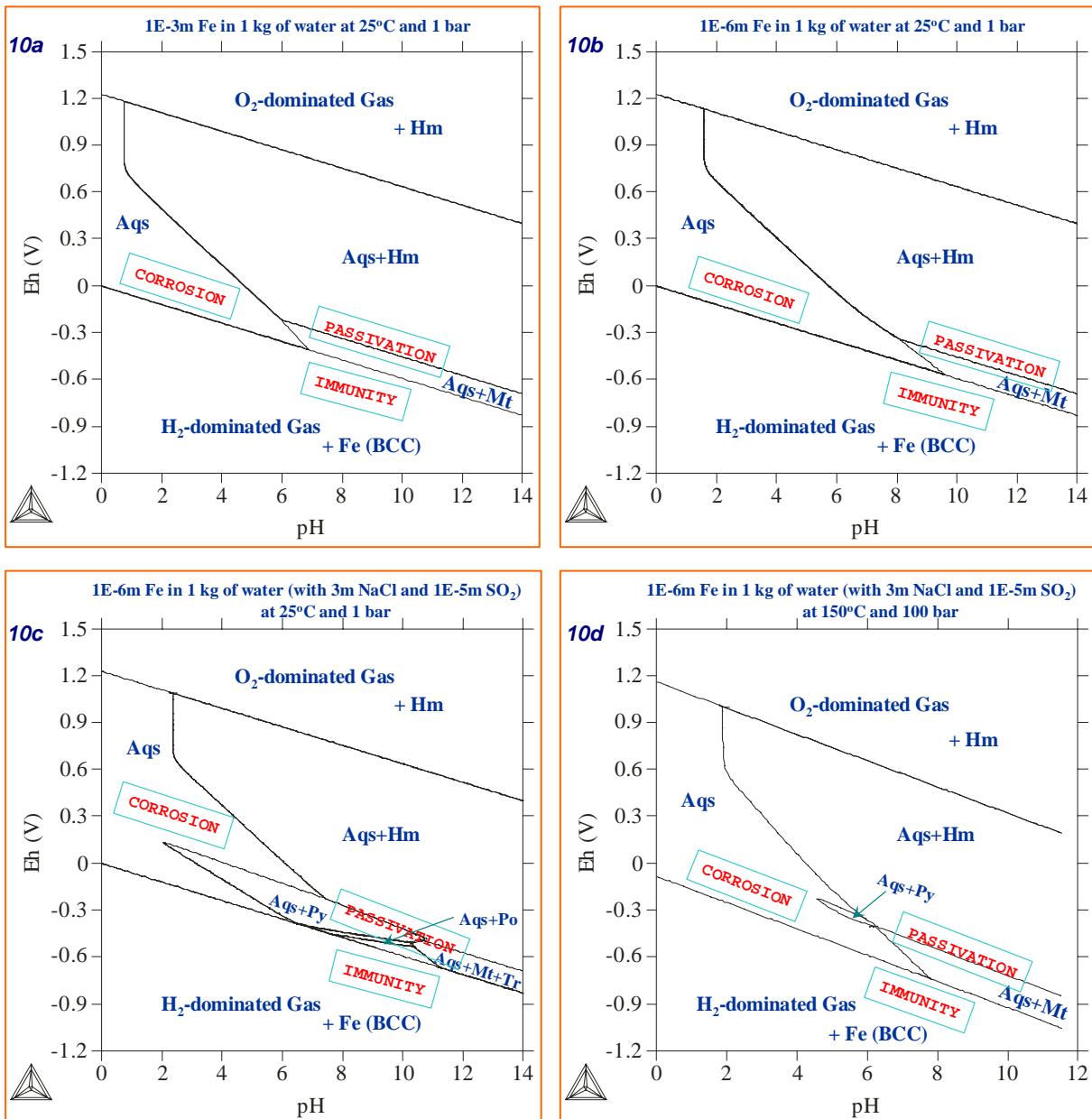


Figure 10-10. Pourbaix diagrams of Fe (for varied Fe actively interacted with pure water or aqueous solution at different temperature and pressure conditions; gas included in the calculations).

10a. 1E-3 m Fe actively reacted with 1 kg of pure water at 25°C and 1 bar (same as in Figure 10-9)

10b. 1E-6 m Fe actively reacted with 1 kg of pure water at 25°C and 1 bar

10c. 1E-6 m Fe actively reacted with 1 kg of water, 3m NaCl and 1E-5 m SO₂ at 25°C and 1 bar

10d. 1E-6 m Fe actively reacted with 1 kg of water, 3m NaCl and 1E-5 m SO₂ at 150°C and 100 bar

Note that the active metal corrosion region enlarges as the initial Fe amount decreases from 1E-3m (10a) to 1E-6m (10b). Introducing SO₂ into the interaction system causes the formation of various metal-sulfides (Py-pyrite, Po-pyrrhotite, Tr-troilite) and increases the passivation region (compared 10c with 10b). Changing the system temperature and/or pressure conditions affects the stability fields of various Fe-oxides/sulfides (compared 10d with 10c). The phase boundaries between the aqueous solution and O₂- or H₂-dominated gaseous mixture shift as the initial bulk compositions or temperature-pressure conditions in the system change.

Therefore, the shape of a Pourbaix diagram of an element (or alloy or condensed material) and the stability relations of various secondary phases (oxides, hydroxides, sulfides, sulfates, carbonates, nitrates, silicates, halides, or other forms) depends upon the following factors in the heterogeneous interaction system:

- a) Initial amount and composition of the condensed materials
- b) Initial amount and composition of the interacting aqueous solution phase
- c) Temperature and pressure conditions

10.7.2 Descriptions of the POURBAIX module

The pH and Eh values of the aqueous solution phase are normally very complicated functions of bulk composition, temperature and pressure conditions. For traditional treatments adapted by various communities as mentioned above, some thermodynamic approximations on the pH-Eh calculations are often used, and such approaches can only be applied to some simple systems. Still, a mechanistic model that will predict the aqueous corrosion resistance of a multicomponent alloy at a given physico-chemical condition is evasive.

However, with some comprehensive thermodynamic models and databases for various non-ideal solution phases implemented in the Thermo-Calc software system, one can accurately calculate the pH and Eh values and thus the complete Pourbaix diagrams, as well as many other types of property diagrams, in any complicated aqueous solution involved heterogeneous interaction system under a wide temperature-pressure-composition range.

Such calculations can be performed in a normal way, *i.e.*, going through various basic modules (TDB, GES, POLY and POST). In other words, one can calculate Pourbaix diagrams in the traditional way, *i.e.*, using various TCC commands; as illustrated in the next *TCC Example 53* that has been made since TCCS. However, this needs a very high skill in doing so. Due to the complexities in defining the heterogeneous interaction systems, reference states of some components, equilibrium conditions, aqueous solution properties, mapping variables, plotting variables, and so on, it is usually rather difficult to properly handle the equilibrium calculations and graphical presentations. Therefore, it is necessary and convenient to make use of some special and easy-to-use modules already implemented in the TCC software.

The so-called **POURBAIX** module was first implemented since TCCL (1996), and has been continuously modified and updated afterwards. It is developed to automatically construct Pourbaix diagrams and other types of property diagrams for various heterogeneous interaction systems involving aqueous solution phases. Some other special and easy-to-use modules, *e.g.*, PREDOMIN module for calculating predominant-area diagram, and PRECSP module for calculating speciation percentage diagram, will be further implemented into the coming versions.

Given in this section are some comprehensive descriptions of the POURBAIX module, including the development history of the module, basic requirements for definitions of the heterogeneous interaction systems, and main feature of the module (modular structures, question-line based interface, special techniques for calculating and plotting Pourbaix diagrams and property diagrams, *etc.*). The related thermodynamic models and databases for various phases (aqueous solution, gaseous mixture, solid solutions and stoichiometric solids) will be briefly reviewed, while for the details of such models and databases one shall refer to the relevant parts in this User's Guide. Furthermore, some details are given on the definitions of some aqueous solution properties (as saved in the memory space of Thermo-Calc software, or listed on an equilibrium calculation result in the POLY module, or shown on a plotted diagrams in the POURBAIX module). Finally, the potential applications and possible future improvement of the module are overviewed, and one typical example of the module calculation procedure is illustrated.

10.7.2.1 Development of the POURBAIX module

The development of the POURBAIX module is parallel to: (1) the implementation of some thermodynamic models in the GES module for complex aqueous solution phase which are effective and efficient over a wide temperature-pressure-composition range; (2) the development of aqueous solution databases in the TDB module which are internally consistent and compatible with other databases available in Thermo-Calc package for various solid, liquid and gas phases; and (3) the modification of output results for the aqueous solution phase in the POLY module.

The versions 1.0 and 2.0 of the module were available in previous releases of the TCC software (versions L and M) and TC4A (versions 1.0 and 2.0). These previous versions can only use a single database containing an aqueous solution phase, a gas phase and some pure solids and solid solutions, which is specially designed for the module (*i.e.*, PAQ, previously as AQ and PAQ). This limits its applications since these two small-sized free database cover only few elements, and an option for multiple-database usage in the module is highly demanded. There were many other limitations and defects in these previous versions, as outlined below:

- the module might fail in calculating all the stable phase boundaries for some complicated systems;
- the *.POLY3 files created by the module were difficult to use for any other purposes such as making stepping calculations in the POLY module, due to the complexity in appropriately defining various functions/variables/tables for aqueous solution properties and in defining status and reference states for aqueous and electron phases/constituents/components);
- the module had some difficulties to be interacted by a user, in making Pourbaix and property diagrams with better appearance and easier access for modifications;
- the module did not have complete on-line helpful descriptions at some questioning prompts;
- the module did not consider a charged gas phase, and all charged gaseous species will be ignored;
- the module did not calculate the high pH limits under which an aqueous solution phase could possibly bear under a certain temperature, pressure and bulk composition, so the user had some difficulties in choosing the scaling status of a pH axis.

The POURBAIX versions 3.0 and 4.0 were made available in the TCCN and TCCP respectively. Many options and automatic routines have been modified or added for a better control of the POURBAIX-module calculations and graphical processing. Consequently, the module can efficiently perform the calculations under various temperatures, pressures and bulk compositions, and can appropriately conduct the graphical processing of various types of diagrams for complex heterogeneous interaction systems. A user may even be allowed to directly interacting with the POLY-module calculations or the POST-module post-processing, within the POURBAIX-module framework. The major modifications and improvements are summarized below:

- Implementation of the Multiple-Database option, which allows a combination of an aqueous solution database and one or more appended databases for gaseous mixture, stoichiometric solid and solids solution phases.
- Multiple compatibility with various thermodynamic models and associated databases for aqueous solutions.
- More precise calculations for the H₂O EOS and thermodynamic properties.
- More stable solutions of equilibrium calculations and much higher security to calculate a complete pH-Eh diagram, through two easily-accessible options:
 - to add more starting points in more directions (coded as being dependent on the defined temperature and pressure),
 - to use smaller mapping steps (in terms of pH and Eh values),
- and other two invisible options (automatically operated by the module itself, if it finds necessary at some computing points):
 - to set higher numerical precision,
 - to interact with the POLY-3 module, so that one can better set conditions and easily get initial equilibria, and by typing a command “B” (BACK) can return to the POURBAIX module.
- More intelligent and automatic settings of some limitations and parameters for various calculations and graphical processing, such as for:
 - the calculation of the highest pH limit for a chemical system under a certain set of temperature-pressure-composition conditions;
 - the search of the highest temperature limit for a proper calculation once the system composition and pressure are specified;
 - the arrangements of suitable starting points for a mapping or stepping calculation when the temperature-pressure-composition conditions have been defined;
 - the assignments of proper mapping or stepping variables and their minimum-maximum values and step factors;

- the specifications of axis variables for a certain diagram, and their axis types, axis texts and units, scaling status (minimum-maximum values and true-/semi-manual or automatic scaling manner), and so on;
 - the definitions of diagram title, subtitle, curve-labeling options, region-labeling texts and their coordinates, fonts and sizes;
 - the creations of default or user-specified file names and extensions of various types during the calculation and graphical processing.
- More prompt and reliable interactions with the POLY or POST module, either optionally or forcedly, so that the user can conveniently and properly modify necessary definitions on some system components and reference states, or adjust some conditions for initial equilibrium calculations, or further refine a plotted diagram.
- More freedom and easier procedure of generating a Pourbaix diagram and various property diagrams, and of substantially modifying and improving the appearance of such diagrams, by providing more automatic and indirect interactions with various POST-module facilities, and even allowing some optional and direct interactions with the POST module.

All these major and other minor modifications and improvements will encourage the Thermo-Calc users to utilize the module together with many more available aqueous/steel/alloy/slag/mineral/... databases for a much wider range of industrial R&D applications. The description below is based on this new version.

10.7.2.2 Basic requirements for calculating Pourbaix diagrams

It is important to meet the following requirements in order to build a Pourbaix diagram in a thermochemical calculation program:

- 1) One must have a database containing an aqueous solution phase (*i.e.*, water as solvent, and aqueous ion/complex species as solutes). Such a database can be ranged from the following two catalogues:
 - Databases containing an aqueous solution phase as well as a gaseous phase and some solid phases, which can be used for the Single-Database option in the POURBAIX module. The *TC Public Aqueous Solution Database, PAQ* (freely distributed within the TCC and TCW, and TCC-Demo/TC4A and TCW-Demo/TC4W) belong to this catalogue (see more details below). A user may use his own specified database that must meet some special requirements for such a database.
 - Databases containing only an aqueous solution phase, but the POURBAIX module will also require to append from some other database(s) the gaseous mixture and solid phases. Such databases can be used for the Multiple-Database option in the module. The *TCAQ and AQS Aqueous Solution Databases* belong to this catalogue, which can be connected with the *SGTE Substances/Solutions Databases*, the *TCAB Steels/Alloys and Superalloy Databases*, the *TGG Minerals/Geochemical/Environmental Databases*, and so forth, as well with user's databases.
- 2) Due to the restrictions of aqueous solution models (SIT, HKF, Pitzer) used in the Thermo-Calc software, the aqueous solution database must be designed in the same format as in the PAQ, TCAQ or AQS database for aqueous solution phase (including the definitions on phase-constituent-species, default components, status of phase-constituent-species, reference-state for the O-H-ZE components). Among others, one should keep in mind the following regulations:
 - ELECTRON is defined as an special element (ZE), which is the only constituent of the reference electrode phase (REFERENCE_ELECTRODE), as well the charge-determining constituent of aqueous cations and anions but not of any neutral aqueous species;
 - The vacancy (VA) is unnecessary for the definition of aqueous solution phase, and thus must be avoided in phase constitution for the aqueous phase in a database;
 - All aqueous species including solvent and solutes are located in a single-sublattice sit, rather than two or multiple sublattices. Compared with a two-sublattice model for the solution, this leads the calculation, interface and post-processing much simpler, quicker and convenient.

- 3) Beside the AQUEOUS (aqueous solution) phase, there generally exists a GAS (gaseous mixture) phase, and for multicomponent systems there normally exist some solid phases (stoichiometric or solution). The gaseous mixture phase at least contains H₂O, O₂ and H₂, while any other neutral and charged gaseous species may also be its constituents. As described above in *Section 10.7.1*, a Pourbaix diagram can be built up for a heterogeneous interaction system with or without the gaseous mixture phase, but the complete Pourbaix diagram in the restrictive sense of thermodynamic equilibrium must have the gaseous mixture phase in the system. The thermodynamic models for either gaseous mixture phase or solid solution phases are usually independent of that of the aqueous solution phase; however, the EOS (equation of state) and thermodynamic properties for the major component H₂O in the aqueous (pure water), gaseous (pure steam) and solid (pure ice) states must be internally consistent.

As already mentioned in *Section 10.7.1*, a Pourbaix diagram normally presents heterogeneous interaction among an aqueous solution phase, a gaseous mixture phase and some solid (stoichiometric or solution phases; *e.g.*, for alloy system the alloy phases and secondary phases). However, one could calculate a Pourbaix diagram without considering a GAS mixture phase; in this case only interactions between the AQUEOUS solution phase and some solid phases (alloys and secondary phases) are calculated. One could also make a Pourbaix diagram calculation without taking into account the secondary phases; in this case only the dissolution of alloy components into the AQUEOUS solution (and GAS mixture) will be calculated. However, such calculations may not present the real heterogeneous interactions. Therefore, some secondary phases are usually needed in the calculations (such phases exist in the PAQ Public Aqueous Solution Database or can be appended from the SSUB/SSOL or other databases). Similarly to the PAQ Public Aqueous Solution Database, one can also include some necessary secondary phases (and a GAS mixture phase) in an alloy database, so that one just needs at once to append it onto the system in which the AQUEOUS solution phase is retrieved from the TCAQ or AQS Aqueous Solution Databases.

10.7.2.3 Main features of the POURBAIX module

Some main features are summarized below, which are also illustrated in *Sub-section 10.7.8* with the details of performing a typical Pourbaix-diagram calculation.

1) Clearly-defined modular structure and unique Main Options:

The module performs a new Pourbaix-diagram calculation in a straight way in which the selection of data, definition of system, and setting of temperature, pressure and composition are made step by step. It also provides the possibility of making many other property diagrams based on one same calculation (which is saved as an *.POLY3 file), of performing other calculations under different temperature-pressure-composition conditions but using same selected data and defined system which are saved in the *.POLY3 file, and of conducting normal stepping calculations.

These will be decided once the user has opened the module, by answering the question “Select option” with one of the following four options:

- 1 as Start a new Pourbaix-diagram calculation;
- 2 as Open old file and plot other property diagrams;
- 3 as Open old file and make another Pourbaix-diagram calculation; and
- 4 as Open old file and make a stepping calculation.

Note these four options are referred in the following text as the Main Option 1, 2, 3 and 4, respectively. For convenience, we call the Main Options 1 and 3 as the Pourbaix-diagram calculation options, the Main Option 4 as the stepping calculation option, and the Main Option 2 as the property-diagram processing option.

Table 10-1 gives a brief description of each of the Main Options. More details about these options will be given later on.

Table 10-1. Four Main Options in the POURBAIX Module.

<i>Option</i>	<i>Descriptions</i>
1	<p>Start a new Pourbaix-diagram calculation</p> <ul style="list-style-type: none"> • It makes a completely new Pourbaix-diagram calculation; • It defines system bulk composition, temperature and pressure; • It gets all thermodynamic data from either a single database or multiple databases. • It automatically sets up the phase status of some phases (AQUEOUS, GAS, REFERENCE_ELECTRODE), the reference states of some components (O, H, ZE), etc.; • It automatically define necessary symbols for species/system parameters beside state variables; • It automatically starts mapping the whole Eh-pH region, and save the results on an *.POLY3 file (default as POURBAIX.POLY3 or specified by the user); • It automatically plots the Eh-pH diagram as the first resulted plot; • The user can decide on whether to make various modifications on a diagram, to plot other types of diagrams, to save the X/Y coordinates on an EXP file (default as POURBAIX.EXP or specified by the user), and so on.
2	<p>Open an old file and plot other property diagrams</p> <ul style="list-style-type: none"> • It opens an already existing POURBAIX-module calculation file (e.g., the POURBAIX.POLY3 resulted from the Main Option 1 or 3 or 4 calculation), reads all the system definition, specified calculation conditions, and calculated results from a previous POURBAIX-module run; • It automatically shows the available variables which can be chosen as X/Y axis for plotting new diagrams; • It can directly plot an Eh-pH diagram or a corresponding property diagram, depending upon the types of the opened file (i.e., the Main Option 1/3 or 4 used in the previous run). It can also plot many other property diagrams (using various aqueous solution properties as X/Y axes). As in the Main Option 1, the user can decide on whether to make various modifications on a diagram, to plot other types of diagrams, to save the X/Y coordinates on an *.EXP file (default as POURBAIX.EXP or specified by the user), and so on.
3	<p>Open an old file and make another Pourbaix-diagram calculation</p> <ul style="list-style-type: none"> • It opens an already existing POURBAIX-module calculation file (e.g., the POURBAIX.POLY3 resulted from the Main Option 1 or 3 calculation), reads all the system definition and specified calculation conditions from a previous POURBAIX-module run; • It uses the same system definition (in terms of elements) as the previously obtained POURBAIX-module calculation (as performed in the Main Option 1 or another run of the Main Option 3), but it allows the user to modify the calculation conditions (such as bulk composition, temperature and pressure). • It saves all the settings and calculated results on the opened *.POLY3 file, and produces the Eh-pH diagram and other types of diagrams in the same way as in the Main Option 1 or 2.
4	<p>Open an old file and make a stepping calculation</p> <ul style="list-style-type: none"> • It opens an already existing POURBAIX-module calculation file (e.g., the POURBAIX.POLY3 resulted from the Main Option 1 or 3 or 4 calculation), reads all the system definition and specified calculation conditions from a previous POURBAIX-module run; • It uses the same system definition (in terms of elements) as the previously obtained POURBAIX-module calculation (as performed in the Main Option 1 or another run of the Main Option 3), but it allows the user to modify the calculation conditions (such as bulk composition, temperature and pressure), change some component definitions if necessary, and choose one condition as the stepping variable and specify the stepping range; • It overwrites all the settings and stepping results on the opened *.POLY3 file, and produces first a default property diagram and then other types of diagrams in the same way as in the Main Option 1 or 2 or 3.

2) Easy-to-use and question-line based interface:

- In order to make the interactive procedures, for certain thermodynamic calculations, between a user and the software much simpler and easier than those as normal Thermo-Calc options, the POURBAIX module was established in such a way that: *a user just needs to answer a few and very simple questions related to the choice of database(s) to be used, to the definition of interested chemical system, and to the setting of temperature, pressure and composition conditions, and consequently the program automatically sets the mapping or stepping variables, starts or repeats the calculations, and plots an Eh-pH diagram or a property diagram; in cases of plotting another property diagrams, or refining or exporting a plotted diagram, clear responses to some simple questions will be enough.*
- All questions occurring in the calculation procedure are designed in a very simple way. Normally, the questions are well self-explained, and a default answer has already been chosen for a certain question, which is shown in the block (/ . . . /).
- There are usually some helpful information available if a question mark “?” is given under the question prompt so that the module can really provide the user with some useful and appropriate comments and suggestions; after which the program usually repeats the question which the user shall answer. Under some sub-options (which are normally easy to understand), the module may not give a specific comment, but this will not cause any fatal interruption in proceeding the module calculation and processing.
- In some places, there exist some corresponding options for going back and forth to confirm or modify previous settings and definitions for calculations or graphic processing, so that the user could modify them as desired and preferred.
- The POURBAIX module has its entirely self-defined structure and question-line based interface, meaning all its external connections with various TCC modules (TDB, GES, POLY and POST) are intelligently coded in various subroutines and beyond the user interface. Through this structure and interface, the user does need to bother about the commands and technical details on how to retrieve the thermodynamic data, to define the system components and equilibrium conditions, to calculate the initial equilibrium point, to specify the mapping or stepping variables, to generate the starting points for mapping/stepping procedure, to choose the variables/types/lengths/texts for diagram-axes, to name diagram title and subtitle, to add curve-/region-labeling texts, to modify and improve the diagram appearance, to save and organize various types of files, and so on. However, the module also provide with some choices as options or forced at some steps for temporally interacting with the POLY or POST modules, that allow the user to appropriately modify and improve some settings and definitions for either equilibrium calculations or graphic processing. Unlike in the normal POLY or POST modules, error messages may not be displayed on screen during such optional or forced interaction. Moreover, only the B (BACK) command will allow the user to leave the temporally interacted POLY or POST module and to return to the POURBAIX module at the previous point exactly from where the POLY or POST module is initialized. In the below, such optional or forced interaction with the POLY or POST modules will be described.

3) Convenient option for choosing single database or multiple databases:

The POURBAIX module (versions 3.0 in TCCN, 4.0 in TCCP/TCCQ, 5.0 in TCCR) can also handle both the Single-Database and Multiple-Database options for thermodynamic data inputs, and for the Multiple-Database case it is unnecessary to use those databases specially designed for the module.

- If the Single-Database option is chosen, the module will ask for a specially-designed database for the module (*i.e.*, PAQ Public Aqueous Solution Database for Pourbaix Module Calculations, which covers only 11 elements). Such a database contains all necessary phases (aqueous solution phase, a gaseous mixture phase, and some pure solids and solid solutions) for calculating complete Pourbaix diagrams, but is normally limited to a relatively small number of elements and phases.
- If the Multiple-Database option is chosen, the module will first ask for an aqueous solution database (*e.g.*, the TCAQ Aqueous Solution Database that cover 83 elements which are the same as in the SSUB/SSOL databases, or the AQS Aqueous Solution Database that cover 46 elements which are the same as in the GCE database, or a user-specified aqueous database). Such an aqueous solution database

shall contain at least the AQUEOUS and REF_ELECTRODE phases. Then the user can append one or more databases for the gas, pure solid and solid solution or other phases from appropriate databases (for instance, SSUB/SSOL, TCFE/ SLAG/ION, TTNi/TTi/TTAl/TTMg, TCMP, TCES, GCE, etc.).

4) Arbitrary definition of chemical system, system components and bulk composition:

- Normally a Pourbaix diagram and related equilibrium properties in a heterogeneous interaction system are calculated for a certain system bulk composition that is usually 1 kg of water with specific amounts of dissolving solute substances. It implies that by default two chemical elements (hydrogen and oxygen) have been defined in the system which are unnecessary to be defined again, and only dissolving solutes are needed to be input for defining a complete system. At an equilibrium state, the solutes may either be completely dissolved into the aqueous solution (in the corrosion region of a Pourbaix diagram), or be partially dissolved and form simultaneously some solid phases (in the passivation region).
- A chemical system and its physico-chemical properties (bulk composition and temperature/pressure) are normally defined in the Main Option 1, while the bulk composition and temperature/pressure conditions can be modified in all three main options for calculations (1, 3 and 4). The chemical system, and bulk composition and temperature/pressure conditions saved in an existing or re-opened POURBAIX-type *.POLY3 file (usually standing for the last equilibrium point in the previous calculation), are always listed out in the Main Options 2, 3 and 4. If one wish to change the chemical system (in terms of elements), he/she must start with the Main Option 1 once again.
- The Pourbaix-diagram calculation options (Main Options 1 and 3) usually use a unique and unchangeable set of definitions for system components and their reference states. For the basic elements O and H, and for electron, the system components are always defined as H₂O, H+1 and ZE, respectively. Their reference states and status for the H₂O, H+1 and ZE components are *permanently* and *unchangeably* defined as follows:

Component	Status	Reference State	Ref-T(K)	Ref-P(Pa)
H ₂ O	ENTERED	AQUEOUS	*	100000
H+1	ENTERED	SER		
ZE	ENTERED	REFERENCE ELECTRODE	*	100000

Note that the reference state for the component H+1 set as in the AQUEOUS phase is identical to that set as at SER (standard element reference).

For other chemical elements, the system components are usually defined same as elements. Their reference states definitions are normally the standard element reference (SER, as defined in the AQUEOUS solution phase bearing database (*e.g.*, PAQ, TCAQ or AQS), but could be changed to others by the user when the POLY module is optionally or forcedly interacted within the POURBAIX module or called outside the POURBAIX module. Their statuses are always set as ENTERED in the Pourbaix-diagram calculations.

However, the stepping calculation option (Main Option 4) allows one to alternate the definitions of some dissolving components (*i.e.*, except for H₂O, H+1 and ZE), by changing from the default and pre-defined elements to some hypothetical salt forms, such as from Na or Cl to NaCl, from K to KOH, from Ca to CaSO₄. Such changes on components definitions (and on reference state of any component, or on status of any phase, or some conditions) can be made is the user optionally chooses to further modify system definitions and conditions in the internally-connected POLY module. Consequently, it will be easier and more straightforward to make a calculation with a stepping variable of the composition for such new components, for an example, m(NaCl).

Furthermore, in all the calculation options (Main Options 1, 3 or 4), the status of the AQUEOUS and REFERENCE_ELECTRODE phases are *permanently* and *unchangeably* set as ENTERED and SUSPEND, respectively. If necessary, the phase statuses for other various phases defined into the heterogeneous interaction system could be changed to other statuses (SUSPEND, FORCED or DORMANT), when the POLY module is optionally or forcedly interacted within the POURBAIX module or called outside the POURBAIX module.

- The chemical system and its bulk composition can be defined in terms of either elements (such as Fe, Ni, Na, Cl, C or S) or arbitrary complex species (such as NaCl, CO₂, CoCl₃, FeO·93S, NaSO₄·1, or H₂SO₄) and their individual initial concentrations in mole (*i.e.*, as molality since the interacting solvent is usually 1kg of water, see above). The program will, each time, ask for one solute and corresponding molality, which can be given either in two lines consequently or in one line at once (for an example, NaCl 2.5); then it will automatically split them into elements and their mole-fractions. Note that as inputting the bulk composition, the first letter of an element symbol must always be in UPPER case, while the second one (if existing) in lower case (see examples given above). If some wrong input (*e.g.*, absent element in databases, or incorrect character) have been given by any chance, the program will later on figure out as checking the databases, and automatically eliminate them from the system definition.

Note that it may sometimes have problems in counting or splitting up some negatively-charged ions if the negative sign is given immediately after the ion name, at least for the *first* such species [*e.g.*, Cl⁻, F⁻, I⁻, Br⁻, HCOO⁻, due to some unknown bugs in the current versions]. In such a case, one can avoid the problem by giving the number “1” before the negative sign [*e.g.*, Cl1⁻, F1⁻, I1⁻, Br1⁻, HCOO1⁻, *etc.*]. There is no such problem for negative ions/complexes if the stoichiometric coefficient for the last element in the name is not unity [*e.g.*, HCO2⁻, NaSO4⁻, SO₄⁻², HSO4⁻] and for all positive ions/complexes [*e.g.*, Cu⁺², CuNO3⁺, Cu(OH)⁺]. So it would be better to give the system bulk composition, if possible, in neutral species (hypothetical salts, or acids, or alkaline, such as NaCl, NaNO₃, HCl, Ca(OH)², *etc.*), for instance,

```
Ni .7
Cr .21
Mo .09
NaCl 3.5
KF 0.5
NaNO3 .5
KNO3 .3
K2SO4 .1
Ca 1e-3
Mg 1e-3
SiO2 0.1
```

instead of charged species (such as Na⁺, Cl1⁻, F⁻, *etc.*):

```
Ni .7
Cr .21
Mo .09
Na+ 4
K+ 1
Ca+2 1e-3
Mg+2 1e-3
Cl1- 3.5      (standing for Cl- 3.5; see above description)
F- 0.5
NO3- 0.8
SO4-2 .1
SiO2 0.1
```

The program will then split up them into compositions in terms of defined components (NA, K, CA, MG, SI, NI, CR, MO, CL, F, N and S). As mentioned above, for the component H₂O, the module always assign its initial composition as B(H₂O)=1000, and for the components H+1 and ZE, the module will evaluate their equilibrium composition at each point of the pH and Eh condition. Due to the electrolysis and redox reactions of the solvent water [H₂O = 2H⁺ + O⁻², H₂O = H⁺ + OH⁻, O⁻² - 2e⁻ = 0.5O₂(aq), 2H⁺ + 2e⁻ = H₂(aq)] and the electronic balance, both controlled by pH and Eh conditions and internal exchange reactions, the module will automatically calculate on how much O⁻² will be produced or consumed (to build up or break up the O⁻² bearing species such as NO₃⁻, SO₄⁻², CO₂, HCOO⁻, *etc.*), and on how much H⁺ and ZE will be added to or removed from the system.

- Like in other easy-to-use modules (*e.g.*, SCHEIL, BIN and TERN modules), when the program retrieves thermodynamic data from a chosen database (either as single or multiply appended), some options are available for deciding whether or not reject/restore any phase from the system already defined in previous steps; and they are repeated till the user is satisfactory. Such options are particularly useful in the following cases:
 - An incomplete Pourbaix diagram with a gaseous mixture phase is to be calculated;
 - There are too many phases found for the system;
 - Some phases are obviously unstable compared to others under the considered P-T-X conditions;
 - Some phases are same in different databases (if appending databases).
- Additionally, a user can specify the phases which may have miscibility gaps in the defined system.

5) Knowledge-based definition of calculation conditions for initial equilibrium point:

After defining the chemical system (and bulk composition) in the Main Option 1, or after opening previously-saved POURBAIX-module *.POLY3 file in the Main Option 2 or 3 or 4, the module will list out the default and pre-defined calculation conditions (in terms of temperature, pressure and composition) for the initial equilibrium point to be calculated.

If such conditions are not confirmed (in the Main Option 1 or 3 or 4) before the program conducts the mapping calculation of a Pourbaix diagram or stepping calculation of a property diagram, one can modify them step by step:

- Firstly, the system bulk composition (in molality) must be specified.

When specifying the system bulk composition for Pourbaix-diagram calculations (the Main Option 1 or 3), the initial molality of the dissolving components (*i.e.*, entered elements excluding O, H and ZE) will be prompted, with the pre-defined or previously defined values as default, for confirmation (pressing <RETURN>) or modification (specifying other values). The bulk composition of the basic solvent and electron components (*i.e.*, H₂O, H+1 and ZE) are not supposed to alternate, so they will not be prompted for further specification.

When specifying the system bulk composition for stepping-diagram calculations (the Main Option 4), the picture is almost the same, except for that the dissolving components might be either as the default entered elements (excluding O, H and ZE) or as the user-specified hypothetical salts (such as NaCl, KOH, CaSO₄, *etc.*), if the user has chosen to further modify system definitions and conditions in the internally-connected POLY module (see the point 4 above).

- Secondly, a system pressure (in bar) must be given.

The program will then automatically calculate the highest applicable temperature limit which is normally the so-called “boiling temperature” of the aqueous solution under given composition and pressure, *i.e.*, a gaseous mixture phase just starts to form at expense of aqueous solution phase which solves all or part of solutes (some solid phases might be stable as well). Therefore, when a pressure is given, the module can tell the user on the highest applicable temperature for the defined multicomponent system in a POURBAIX-module calculation; otherwise a user may fail in properly making the Pourbaix diagram by choosing a temperature under which the aqueous solution phase is unstable.

Note that the completion of calculating the highest applicable temperature limit might take some time in some cases, due to some computational difficulties for any possible reason such as:

- (a) too many phases taking part in calculation;
- (b) insufficient computation iterations;
- (c) ill bulk composition or pressure condition; and
- (d) ill status set for some of phases.

Consequently, the program will, at first, automatically alter some calculation methods (*e.g.*, to change the numerical limits in calculating phase equilibrium, or to change the status of some of phases in the defined system). Then, if not succeeded in finding the solution, the POURBAIX module will be temporally interconnected with the POLY module, and the user will have to perform some interactive POLY options as in a normal phase equilibrium calculation herewith. All

POLY features can be utilized for such purposes, while some restrictions (that relate to the system definition and being-calculated equilibrium) must be kept in mind, for instance,

- temperature should not be set as a condition;
- status for AQUEOUS phase must be as ENTERED;
- status for GAS phase must be as FIXED at zero amount (since here it is calculating the boiling temperature);
- $B(H_2O)=1000$ should not be changed;
- $N(H+1)$ and $N(ZE)$ as zero could be adjusted slightly.

Once the equilibrium (in which the AQUEOUS and GAS phases must be stable simultaneously) has been obtained, just type the BACK command to go back to the POURBAIX module to continue the calculation.

- Thirdly, a desired system temperature needs to be entered, which must be between 0°C and the highest limit calculated.

6) Automatic saving of definitions, conditions and mapping/stepping calculation results:

Prior to the mapping or stepping calculations, all the information related to system definitions, component-phase status, temperature-pressure-composition condition, mapping/stepping variables, and others, are saved in an *.POLY3 file, with a default name as POURBAIX.POLY3 in case of the Main Option 1, or the same as that entered in the initial step of the Main Option 3 or 4. Such a file will be overwritten immediately after the mapping or stepping calculation has succeeded.

Many symbols (constants, functions, derived variables and tables) for various properties of aqueous solution phase and aqueous species are automatically defined in the module for all four main options. They are temporally recorded in the POLY3 and GES5 workspaces, but can optionally saved into the POURBAIX-module *.POLY3 file.

Therefore, it is important to remember that, before making any further Main Option 2 or 3 or 4 calculation, the automatically saved POURBAIX.POLY3 file (always the default name after a Main Option 1 calculation) or the opened and overwritten POURBAIX-module *.POLY3 file has been copied to another *.POLY3 file for saving the previous results or making other POURBAIX-module calculations, if the user considers it necessary.

7) Direct opening of an existing POURBAIX-module file for another calculation or graphic processing:

As running the Main Option 2 or 3 or 4, the program will first ask for the name of a preserved POURBAIX module calculation file which must be in the current directory and must be readable by the POLY3 and GES5 workspaces. The default filename shown is as either POURBAIX.POLY3 if the module is opened for the first time, or another which has been entered previously in Thermo-Calc (meaning that the user has not left the main program) if the module has been opened with the Main Option 1 or 3 or 4. Then the program will list those default and pre-defined calculation conditions (*i.e.*, temperature, pressure and bulk composition) saved in that *.POLY3 file.

- In the Main Option 2, none of such conditions can be modified. The user is only allowed to open the *.POLY3 file with all kinds of definitions and calculated results, and to conduct the graphic setting, refining and other processing.
- In the Main Option 3, due to the practical purposes of this option, normally some of such conditions need to be modified as desired (see the point 5 above for the details on how to modify conditions), and all the mapping calculation results will be overwritten onto the opened *.POLY3 file. Then the user shall follow almost exactly the same consequent procedures as in the Main Option 1.
- In the Main Option 4, if desired, one can modify some of such conditions for the initial equilibrium point calculation, and can even change some of system component definitions and thus system conditions in internally-connected POLY module. Later on, the stepping variable (temperature, pressure, pH, Eh or molality of one component) and its minimum/maximum limits will be determined by the user. Then the user must follow the stepping calculation and graphic processing procedures.

8) Automatic mapping procedure in a Pourbaix-diagram calculation:

Before the mapping procedure in the Main Option 1 or 3, the program will in most cases try to find as many as eight starting points in the whole Eh-pH field. The program normally does so in an automatic procedure with some intelligent arrangements regarding possible shifts resulted by the variations in temperature, pressure and system bulk compositions. If this procedure is countered by any difficult, some additional options will have to be taken, either automatically by the program or interactively by the user (in internally-connected the POLY module). This is similar to the case where it fails in finding the highest applicable temperature limit (as described below). In case of interaction with POLY module, the definition of system components and their reference states should not be changed, but a temperature differing from the previously confirmed value can be set as a condition in the equilibrium calculation, and the new value will be saved as the equilibrium temperature.

The mapping variables, for building up the Pourbaix diagram and other related property diagrams within the oxidation/acidity states, are inside the program set permanently as MUR (ZE) (the chemical potential of hypothetical electron in the system) and LNACR (H+1) (the natural logarithm of activity of the ion H+1 with a reference state as its aqueous species), respectively. The minimum and maximum values and increment steps for both MUR (ZE) and LNACR (H+1) are automatically set by the program. When assigning the maximum value of the LNACR (H+1) condition in mapping, the program always automatically calculates the equilibrium constant in logarithmic ($\log_{10}K_a$) of the electrolysis reaction $H_2O_{(water)} = H^+ + OH^-$ for the real aqueous solution, which is a function of temperature, pressure and aqueous solution composition. Thus this constant value is counted as the highest applicable pH limit for the defined chemical system under the current temperature-pressure-composition conditions.

The automatic set up of starting points may take some time, especially for a complex multicomponent system under elevated temperature and pressure or high aqueous concentration. Afterwards, the program starts the automatic mapping calculations.

9) Automatic stepping procedure in a stepping calculation:

Before the stepping procedure in the Main Option 4, the program will ask the user to choose a suitable parameter as the stepping variable from the defined equilibrium conditions. Available parameters are as follows:

T	Temperature (°C)
P	Pressure (bar)
pH	Acidity
Eh	Electronic potential (V)
N (Comp)	Mole of one component

Note that “N (Comp)” is the mole number of one interacting component (*i.e.*, the dissolving components, such as Fe or Na or Cl or NaCl) in 1 kg of water. The mole numbers of key components (*i.e.*, the basic solvent and electron components H₂O, H+1 and ZE) should not be used for stepping variables, because that their compositional definition and variations are automatically determined by the AQUEOUS solution phase in the module. One can type the “N (Comp)” such as “N (Fe)”. If the user has just typed a “N” as the stepping variable, the program will list out all the components (with an index for each one) which are possibly suitable for being selected as the varying component, and the program is expecting an index input to complete the “N (Comp)” input.

Since the program normally sets the calculated initial equilibrium point as the starting point, the minimum value for the stepping variable must be specified as smaller or equal to the specific initial condition, and the maximum value as larger or equal to that condition. If pH is set as the stepping variable, the program will automatically calculate the equilibrium constant in logarithmic ($\log_{10}K_a$) of the electrolysis reaction $H_2O_{(water)} = H^+ + OH^-$ for the real aqueous solution under the current temperature-pressure-composition conditions, and thus this constant $\log_{10}K_a$ value is considered as the highest pH in specifying the maximum value along stepping. Subsequently, an increment factor with a default value should be confirmed or modified; for pressure and mole of a component, one can specify the step in logarithmic ($\log_{10}P$ or $\log_{10}N$).

Afterwards, the program starts the automatic stepping calculations.

10) Automatic generation of the first diagram after a Pourbaix-diagram or stepping calculation:

A complete Pourbaix-diagram calculation, either newly accomplished (*i.e.*, through the Main Option 1) or modified (*i.e.*, through the Main Option 3), will automatically produce a Pourbaix diagram with the pH value as X-axis and Eh as Y-axis. There are also possibilities for the user to choose an option to manually add potential starting points (if one doubts on that any phase boundary may be missing or incomplete on the generated Pourbaix diagram) and another option to manually change the mapping steps of pH and/or Eh axes to smaller values (if one thinks of the mapped pH-Eh curves are not smooth enough). However, in most cases, these two options are unnecessary.

On the other hand, a complete stepping calculation (*i.e.*, through the Main Option 4) will automatically generate a property diagram with the stepping variable as X-axis and the amount of stable phase (in mole/kg of water) as Y-axis.

Such an automatically generated diagram has always been polished up, by optimally setting the default scaling-statuses, lengths, types and texts of X-Y axes, the default diagram title and subtitle, the default curve-labeling option, and so on.

However, such a Eh-pH diagram or property diagram can be further modified in the sequential steps (by changing axis types, making zooms, changing the subtitle or axis texts, alternating the curve-labeling option, adding and modifying region/property-labeling texts, *etc.*), or can optionally be further refined in the internally-connected POST module. It can also be converted to a hard copy (either transferred to a graphic file in PostScript or other format, or sent to a specified printer), and saved the X-Y coordinates of the modified/refined diagram into a textual *.EXP file. See the point 12 below for details of such graphic processing and refinements.

In the Main Options 1, 3 and/or 4, once the Eh-pH diagram has been generated, and optionally been modified/converted/saved, the program will further ask if the user needs to make other kinds of diagram using the same calculation result. Consequently, the user can define, plot and modify many more diagrams. These sequential options are actually the same as in the Main Option 2 (see the point 11 below for details).

11) Straightforward plotting of many property diagrams based on one single calculation:

A POURBAIX-module calculation (either a Pourbaix-diagram or a stepping calculation) preserved in an *.POLY3 file, which exists in the current directory, can be used for making many useful property diagrams for the same defined system and conditions.

This applies not only to the Main Option 2 (which is essentially designed for such purposes), but also to all of the three main calculation options. In the Main Options 1 or 3, after a Pourbaix diagram is produced, more property diagrams can be plotted; and in the Main Option 4, after a default property diagram is generated, more property diagrams can be drawn.

For the Main Options 1 and 3, and for the Main Option 2 after a calculation of the Main Option 1 or 3, the following common “selection table” for setting plotting parameters as X/Y axis variables is designed:

```
-----
I                                         I
I   The following properties are available as axis variables I
I       for making other diagrams from the same calculation: I
I                                         I
I       pH --- Acidity                               I
I       Eh --- Electronic Potential (V)           I
I       Ah --- Electronic Affinity (kJ)          I
I       pe --- Electronic Activity (log10ACRe) I
I       IS --- Ionic Strength                      I
I       TM --- Total Concentration                 I
I       Aw --- Activity of Water                  I
I       Oc --- Osmotic Coefficient                I
I       MF (AQsp) --- Mole Fractions of Aqueous Species I
I       ML (AQsp) --- Molalities of Aqueous Species I
I       AI (AQsp) --- Activities of Aqueous Species I
I       RC (AQsp) --- Activity Coefficients of Aqueous Species I
I                                         I
```

I where "AQsp" is the name of an individual aqueous species, I
I and "*" can be used as a wild character for all "AQsp"; I
I It is unnecessary to give a complete aqueous species name, I
I however, the entered AQsp name must be unique. I
I If neither "AQsp" nor "*" given, species will be searched. I
I

List of All the AQsp (Aqueous Species) in the Defined System:

CL-1	CL2	CLO-1	CLO2	CLO2-1
CLO3-1	CLO4-1	FE+2	FE+3	FE2O2H2+4
FECL+2	FEO3H3-1	FEOH+1	FEOH+2	H+1
H2	H2O	H2O2	HClO	HClO2
HO2-1	NA+1	O2	O3	OH-1

Meanwhile, for the Main Option 4, and for the Main Option 2 after a calculation of the Main Option 4, the common “selection table” is designed as below:

I
I The following properties are available as axis variables I
I for making other diagrams from the same calculation: I
I
I TC --- Temperature (C) I
I TK --- Temperature (K) I
I P --- Pressure (Pa) I
I NP --- Stable Phase (mole/kg of water) I
I BP --- Stable Phase (gram/kg of water) I
I N(Comp) --- Composition of chosen component (mole/kg) I
I B(comp) --- Composition of chosen component (gram/kg) I
I pH --- Acidity I
I Eh --- Electronic Potential (V) I
I Ah --- Electronic Affinity (kJ) I
I pe --- Electronic Activity (log10ACRe) I
I IS --- Ionic Strength I
I TM --- Total Concentration I
I Aw --- Activity of Water I
I Oc --- Osmotic Coefficient I
I MF (AQsp) --- Mole Fractions of Aqueous Species I
I ML (AQsp) --- Molalities of Aqueous Species I
I AI (AQsp) --- Activities of Aqueous Species I
I RC (AQsp) --- Activity Coefficients of Aqueous Species I
I
I where "AQsp" is the name of an individual aqueous species, I
I and "*" can be used as a wild character for all "AQsp"; I
I It is unnecessary to give a complete aqueous species name, I
I however, the entered AQsp name must be unique. I
I If neither "AQsp" nor "*" given, species will be searched. I
I

List of All the AQsp (Aqueous Species) in the Defined System:

CL-1	CL2	CLO-1	CLO2	CLO2-1
CLO3-1	CLO4-1	FE+2	FE+3	FE2O2H2+4
FECL+2	FEO3H3-1	FEOH+1	FEOH+2	H+1
H2	H2O	H2O2	HClO	HClO2
HO2-1	NA+1	O2	O3	OH-1

In both tables, the upper part lists all possible parameters that can be selected as X and/or Y variables; while in the second type of table, some additional parameters (depending on what kind of stepping variable has been chosen in the Main Option 4 calculation) are also listed at the top of the upper part. The lower part, which is the same for all four options for an identical system, lists all possible aqueous

species that can be selected into the definitions of those properties (*i.e.*, MF, ML, AI and RC) chosen as X-Y axis variables.

In the upper part, the method for entering a certain property (MF, ML, AI or RC) as X-Y variables are also given, *i.e.*, such a property could be specified in one of the following three ways (with an example shown in “ ” for each case):

- For a single aqueous species, just enter the property with aqueous species name in a format, “ML (Fe+2)”; the species name is unnecessary to be complete, but it must be unique for the same system.
- For all aqueous species, just enter the wild character *, *e.g.*, “ML (*)”.
- For several (or a single one) aqueous species, only enter the property name at first, “ML”. Subsequently, the program will list out all available aqueous species (with a corresponding index for each one) for such a property in the system. Then the species index number(s) should be entered in one single line, “2 4 25 8 10 12 15”.

When selecting “N (Comp)”, the composition of the component which is already chosen for a stepping calculation, the component name must be clearly specified, *e.g.*, “N (Fe)”.

Moreover, as selecting X-Y variables, both UPPER-case and lower-case are allowed for property and species names; for instance, ML (FE+2) or ml (Fe+2) or ml (fe+2) are identical.

For the definitions of various parameters in the selection tables, please refer to *Section 10.7.2.4*:

12) Easy graphic processing and refinement of a plotted diagram:

Once the Pourbaix-diagram or stepping calculation is finished (in the Main Option 1 or 3 or 4) or both the X and Y axes have been assigned with properties (in the Main Option 2), the program will immediately plot a diagram on screen. Normally, such a diagram needs to be modified and refined in some ways, such as changing X-Y axis types, and making zoom(s) in the diagram (*i.e.*, altering X-Y axis scaling factors), alternating the curve-labeling option, adding and modifying region/property-labeling texts, changing the subtitle or axis texts, *etc.*), or can optionally be further refined in the internally-connected POST module. The modified diagram will be illustrated on screen in conjunction with one or more proper options. If the diagram is satisfactory, the user can go on to decide *if* and *how* to produce its hard copies (either transferred to a graphic file in PostScript or other format, or sent to a specified printer) and *if* and *where* to save the X-Y coordinates of the modified/refined diagram into a textual EXP file. After these performances, there are still more chances to polish the diagram, and this is particularly useful if one wants to generate several versions of the same diagram, either in different appearances or in varied graphic formats.

- When modifying a diagram, the program first asks for if needing to change any axis type and/or to alter scaling factor for both X and Y axes. Then the program asks for if wishing to use any other curve-labeling option, to add and subsequently to modify any region/property-labeling text for phase region(s), to change the default or previously-modified diagram subtitle, and to modify any axis text. After each such modification option, the program will further ask for how to accomplish that option. It may take some time to go through all such options, because the modified diagram will be plotted on screen at some steps in order to review the changes. Finally, the program allows the user to further refine the diagram, if desired, in the internally-connected POST module.
 - The module automatically pre-defines the Axis-Type (LIN/LOG/INV), depending upon the property chosen as the axis. For changing X-Y axis types, there are three choices to set the axis as either linear (LIN) or logarithm (LOG) or reversed (REV); once these changes have been made, the program will save these settings for the next diagram (that is useful if one produces many diagrams with X-Y axes of same type of properties but for different aqueous species).
 - The module automatically pre-defines the Axis-Scaling parameters (Min/Max) depending upon the property chosen as the axis, and pre-sets the scaling manner as Automatic. For altering X-Y axis scaling factors, the minimum and maximum values should be given (the program always gives some suggested values), and then the module will automatically decide a scaling manner (True- or Semi-manual). As desired, this option

and that for changing X-Y axis types (see above) can be repeated, if some modifications are not suitable or incorrect. When pH is chosen as either the X or Y axis in generating a pH-Eh or property diagram, the module will always automatically take into account of the highest applicable pH limit. As zooming such a diagram, the module will reminder the user this limit.

- The default option for a Pourbaix diagram is B, and for a property diagram is D, with the meanings of labeling listed beside the diagram. For modifying the Curve-Labeling Option, one of the following options should be entered:

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

After change the option, the diagram is plotted on screen, so that a user can re-set the curve labeling to his/her preference. If the user wants to label phase regions or curve properties directly inside the diagram by adding some region-labeling texts (see below), the curve-labeling option is better to be set as N.

- The module allows to add and/or modify Phase-Region/Property Labeling Texts in all kinds of diagrams. For adding some region/property-labeling texts (normally under the curve-labeling option N, see above), the user can write many desired labeling texts for either stable phase regions (on a Pourbaix diagram) or properties (on a property diagram). The X- and Y-coordinates for starting the labeling texts must be specified first. A subsequent option for “Automatic label text” gives the possibility of an automatic execution for finding the stable phases in the specified point. Note that it will normally fail in doing so in an pH-Eh diagram due to that both pH and Eh are just derived variables but not intensive state variables; however, it is kept in the module because it may work successfully for some property diagrams where the X and Y axes are defined as intensive quantities. It is recommended that a user always choose an “N” (No) answer here, and subsequently gives the phase/component/species names for composing labeling texts can be input in full or abbreviated, and as separate or in a combination, totally as the user's preference. A suggestion is always given automatically by the module for specifying the coordinates and labeling texts, but the user should be aware of that he/she might need to have a more appropriate input on the plotted diagram. Immediately after this action, another option is available to allow further labeling.

Then the user can decide if any of the added labels should be further modified. In doing so, the user needs to first identify which label to be modified, and then change the X/Y-coordinates and (if required) the label text which have already been specified in the above option or in a previous round of diagram refinement. Note that the length of the new text must not exceed the previously added or modified text.

Immediately after adding or modifying each labeling text, the diagram will be plotted on screen once again. To assist the user make correct phase-region/property labeling texts, the module gives an opportunity to go forth and back to the Modify Curve-Labeling Option, if the user considers it necessary. So the user can more easily figure out the phase assemblages by looking through some curve-labeling options (normally “B” or “D”), and eventually re-set the curve-labeling option to “N”.

In order to easily and appropriately identify phase assemblages in various regions on a Pourbaix diagram during the phase-region labeling, one can copy the automatically saved POURBAIX.POLY3 (or the opened and updated file) to another *.POLY3 file. Then open the new file in another TCC window and go to the POLY module to analysis the details of equilibrated phase assemblages in each region by calculating individual equilibria at varied pH-Eh conditions through the following relations:

$$\begin{aligned} \text{LNACR (H+1)} &= -[\text{pH}] * 2.302585 \\ \text{MUR (ZE)} &= [\text{Eh (V)}] * 96485.309 \end{aligned}$$

- All diagrams produced from the same POURBAIX-module calculation will have the same default subtitle which comprises of temperature, pressure and system bulk composition, not matter if these diagrams are or not generated subsequently in a single or separate calculation and processing procedure (*i.e.*, through the Main Option 1 or 2 or 3 or 4). However, such a subtitle can be changed upon confirming a related option. In doing so, the default or previously-modified name will be show up as a reference, and the user can type in the new preferred subtitle under a single line. One should avoid writing a very long subtitle, even the system bulk composition is rather complicated; otherwise, there might be no enough space above the diagram to code the subtitle. A modified subtitle will be saved systematically and remain the same for all the sequential diagrams, until it is alternated again or the same POURBAIX-type *.POLY3 file is re-opened or another *.POLY3 file is opened.
- The module automatically pre-defines the Axis-Text for both X and Y axes, according to the properties chosen as the axis variables. For modifying any of the axis text, the default or previously-modified name will be show up as a reference, and the user can type in the preferred axis name under a single line or press <RETURN> to accept the old one. One should avoid write a very long axis name; otherwise, there might be no enough space beside the diagram to place it.
- After all these modifications and adjustments in the above-described steps, the user can even go, through an additional option if he/she really wants to further improve the appearance of the generated diagram, to the internally-connected POST module to make further refinements on the plot. Under such an interactive mode, all kinds of POST commands can be called. This is particular useful when one wants to append experimental data onto a diagram, or to suspend and restore some phases, set plot size, set axis-tic type, set tie-line status, set axis plot status, set axis length, set diagram title, make a quick plot of another previously plotted diagram, *etc.* If the user prefers to make all kinds of refinements using the ordinary POST commands rather than going through the above-described POURBAIX-module question-lines, this feature is also necessary. However, as mentioned in the point 2 above, unlike in the normal POST module, error messages may not be displayed on screen during such optional interaction; and after the user has made all the desired refinements, only the “B” (BACK) command can lead the user leaving the temporally interacted POST module and returning to the POURBAIX module for further graphic processing.
- When making hard copy of a diagram, the module will ask for the options for setting plot format (*i.e.*, unit for printer, PostScript or other graphical outputs), for setting font number and font size, and for specifying plot file name or plot unit name. Once one hard copy has been made, the program will provide more chances for producing other hard copies (for instance, once can make a printed paper and save as a Postscript or other graphic formats for the same diagram). Beside the normal screen illustration, one can also open or view or print or copy a diagram on another window using a proper graphical editor or system command, after it has been saved as *e.g.* a PostScript file.
- When converting X-Y coordinates of curves on one diagram, it needs only entering a file name, which normally has a default extension “.EXP”. One can accept the default file name set by the program (usually as POURBAIX.EXP or as the opened POURBAIX-type *.POLY3 file name but with the EXP extension) or type another file name (if an extension is not given, the written file will have the standard EXP extension). This kind of textual file, which uses the DATAPLOT format (see Part 15 in this User's Guide), contains all the X-Y coordinates of curves, as well as some of the graphical processing and refinement information which have been evaluated and modified in the entire above-described POURBAIX-/POST-module processing procedures. Such a file can be further edited by any simple textual editor, and can be utilized for further plotting diagrams combined some experimental and/or editorial information using the POST module later on, or re-plotted using other graphical software. This also makes it possible to merge two or more diagrams of the same type from separate calculations, or to impose experimental data onto a calculated diagram.
- By choosing the “Modifying the diagram” option, one can further modify the defined diagram repeatedly, till the user is fully satisfied. Depending on what steps the user has chosen to

go through previously, the module will prompt some necessary options for the user to make such further refinements. All adjustments made in the previous steps will be properly saved in the memory, so the user will not need to repeat those changes at all. Very often happened, a user may wish to create more than one version of a specific diagram for varied presentation purposes. This particular option can be switched on for such cases.

13) Early chance for enhancing the calculation accuracy:

Thanks to the complexity of aqueous solution models (especially the HKF and PITZ models) and the aqueous involved heterogeneous equilibrium calculations in the POLY and GES modules, and to the requirements of special treatments of aqueous solution concentrations and activities (especially for minor species), it may encounter some difficulties in precisely presenting the equilibrium states. Furthermore, the quality of the diagrams (in terms of curve smoothness, continuity and entirety) is sometimes not good enough as desired. For these reasons, one could try to manually change the numerical limits and calculation accuracy for the POLY (and GES) calculations, just after entering a calculation option (the Main Option 1 or 3 or 4) in the POURBAIX module. One can also set these numerical limits in the POLY module before entering the POURBAIX module, or in the optionally or forcedly interconnected POLY module during a POURBAIX-module run.

Three types of numerical limits (*i.e.*, maximum number of iterations in a single search for an equilibrium in the heterogeneous interaction system, required accuracy for a stable mathematical solution, and smallest site fraction for constitutional calculations) and one method of calculating driving forces for metastable phases) can be amended, as listed below:

Maximum number of iterations /500/: <a new high limit for searching a single equilibrium point>
 Required accuracy /1E-06/: <a new accuracy limit for reaching a stable mathematical solution>
 Smallest fraction /1E-30/: <a new low limit for accounting the site fractions in a phase>
 Approximate driving force for metastable phases /Y/: <Yes or No>

Such limits are the criteria for convergence and calculation speed in the POLY (and GES) calculation. Please refer to *Chapter 8* for the amendment methods.

14) Multiple compatibility of aqueous solution models and databases:

The POURBAIX module is compatible with various kinds of aqueous solution models implemented in the Thermo-Calc software, *e.g.*, SIT (Specific Ionic Interaction Theory), HKF (Complete Revised Helgeson-Kirkham-Flowers Model) and PITZ (Generalized Pitzer Formalism). For the details on such aqueous solution models, please refer to *Section 10.7.2.5* and *Chapter 11*.

Moreover, it is compatible with various available aqueous bearing databases (PAQ, TCAQ or AQS). The PAQ database contains aqueous solution, gaseous mixture, stoichiometric solids and solid solution phases, are specially designed for uses with the Single-Database option in the POURBAIX module. The TCAQ and AQS are developed to only the aqueous solutions, and can be used with the Multiple-Database option in the POURBAIX module, where various other necessary phases (gaseous/fluid mixtures, steels, alloys, melts, slags, salts, ceramics, minerals, nuclear materials, *etc.*) shall be appended from some available databases (*e.g.*, SSUB, SSOL, TCFE, TCNI, SLAG, SLAT, ION, TTAl/Ti/Ni/Mg, TCPM, TCES, GCE, *etc.*). For the details on such aqueous solution databases, please refer to *Section 10.7.2.5* and the document *Thermo-Calc Database Guide*.

15) Direct use of an existing POURBAIX-module file for various purposes in other normal TCC modules:

As described above, a POURBAIX-module generated *.POLY3 file contain all kinds of information on system definitions, thermodynamic data for various phases, calculation condition settings, and calculated heterogeneous equilibria along the mapping or stepping steps.

The POURBAIX module also asks the user for the necessity of saving all the POURBAIX-module defined symbols (functions/variables/tables) related to various aqueous solution properties. This option will generate an *.POLY3 file containing all the appropriate definitions of various functions/variables/tables for aqueous solution properties and in defining status and reference states for aqueous and electron related phases/constituents/components), which is much bigger than a normal POURBAIX.POLY3 created without the option.

Such a file can be utilized not only for various calculations of Pourbaix diagrams and many property diagrams, but also for any simulation of complex multicomponent heterogeneous interaction processes involving aqueous solutions. It is frequently demanded on that a POURBAIX-module generated *.POLY3 file could be directly opened and easily used for various purposes in other normal TCC modules, such as:

- Making other mapping and stepping calculations in the POLY module;
- Conducting graphical processing of other diagrams in the POST module;
- Calculating thermodynamic properties for substances, phases and reactions in the TAB module;
- Saving GES5 or TDB or DAT files, and modifying system and phase definitions and thermodynamic data in the GES module;
- Creating SETUP or PAR or POP files, and performing data assessments based on experimental information in the PARROT module.

16) Default handing of charged gaseous species:

The module is able to handle a charged gas phase, by default defining “/-” into a system when retrieving data from a single or multiple (appended) databases. This additionally assigned “/-” is regarded as a special element for the GAS mixture and some liquid mixture or solid solution phases, such as defined in the SSUB, SSOL and ION databases. However, it is not considered as a system component in the POLY, GES and POURBAIX modules; for this reason, the GES/POLY module can not calculate its chemical potential, nor its participation coefficients in equilibrated gaseous species.

However, one should notice that in the AQUEOUS solution phase the “ZE” is also defined as a special element, but for which the chemical potentials and equilibrium participation coefficients are properly calculated in GES/POLY and of course in POURBAIX module. Therefore, the electron potential and H⁺ potential in a stable AQUEOUS solution phase can always be appropriately calculated, but not in a GAS mixture phase (or other phases). This is why for that phase boundaries in a Pourbaix diagram calculated in the POURBAIX module can not be extended to aqueous-absent regions, where a charged gaseous mixture phase might be in equilibrium with solid phases while no aqueous solution is stable, or where other possible charged phases (for instance, charged solid constituents in surface reactions) may present.

10.7.2.4 Descriptions of various properties for the aqueous solution phase

For calculating aqueous bearing heterogeneous interaction systems, many additional derived variables for the aqueous solution phase are required. In *Section 8.13*, a summary has been given on various possible derived variables and their definitions.

In the following, the definitions and internal relationships of some derived variables, which are related to the selection of various properties of the AQUEOUS solution phase or aqueous species (solvent or solute) as axis variables in the POURBAIX module, are again briefly described:

- Composition quantities of the AQUEOUS phase or an aqueous species in the AQUEOUS phase:
 - ❖ NP (AQ) Mole Number of the AQUEOUS phase (mole)
 - ❖ NS (AQ, sp) Mole Number for an aqueous species (mole)
 - ❖ MF (AQsp) = X (AQ, sp) Mole Fraction of an aqueous species
 - ❖ Y (AQ, sp) Site Fraction of an aqueous species
 - ❖ ML (AQsp) = ML (AQ, sp) Molal Concentration of an aqueous species (mol/kg of water)
- $$\text{NS (AQ, sp)} = \text{Y (AQ, sp)} * \text{NP (AQ)}$$
- $$\text{Y (AQ, sp)} = \text{X (AQ, sp)}$$
- $$\text{ML (AQ, sp)} = \text{Y (AQ, sp)} * 55.5084 / \text{Y (AQ, H}_2\text{O)}$$
-
- Activity properties of an aqueous species in the AQUEOUS phase:
 - ❖ AI (AQsp) = AI (sp, AQ) Activity of an aqueous species
 - ❖ RC (AQsp) = RC (sp, AQ) Activity of an aqueous species

❖ Log10Act (AQ, sp) Common logarithm of Activity of an aqueous species

$$\begin{aligned} \text{AI}(\text{H}_2\text{O}, \text{AQ}) &= \text{ACR}(\text{H}_2\text{O}, \text{AQ}) \\ \text{RC}(\text{H}_2\text{O}, \text{AQ}) &= \text{ACR}(\text{H}_2\text{O}, \text{AQ}) / Y(\text{AQ}, \text{H}_2\text{O}) \\ \log_{10}\text{Act}((\text{H}_2\text{O}, \text{AQ})) &= \log_{10}(\text{AI}(\text{H}_2\text{O}, \text{AQ})) \\ \text{AI}(\text{sp}, \text{AQ}) &= \text{ACR}(\text{sp}, \text{AQ}) * 55.5084 \\ \text{RC}(\text{sp}, \text{AQ}) &= \text{ACR}(\text{sp}, \text{AQ}) / Y(\text{AQ}, \text{sp}) * Y(\text{AQ}, \text{H}_2\text{O}) \\ \log_{10}\text{Act}((\text{sp}, \text{AQ})) &= \log_{10}(\text{AI}(\text{sp}, \text{AQ})) \end{aligned}$$

➤ Solution properties of the entire AQUEOUS phase:

- ❖ pH Acidity of the aqueous solution phase
- ❖ Eh Hypothetical electrical potential of the aqueous solution phase
- ❖ Ah Thermodynamic affinity per electron of a redox couple with respect to the standard hydrogen electrode, *i.e.*, the chemical potential of the hypothetical electron
- ❖ pe Logarithm of the hypothetical electron activity
- ❖ IS = I Ionic strength of the aqueous solution phase
- ❖ TM = m* Total molality (in equivalent molality) of all solute species (including free ions and complexes) in the aqueous solution phase
- ❖ Aw Water activity in the aqueous solution phase
- ❖ OC = OS Osmotic coefficient of the aqueous solution phase
- ❖ AT Titration alkalinity of the aqueous solution phase, generally defined as the *equivalent molality* of carbonate and bicarbonate (in some cases plus sulfide) at the methyl orange endpoint (*i.e.*, pH=4.5)

$$\begin{aligned} \text{pH} &= -\log_{10}(\text{AC}(\text{H}^+, \text{AQ})) \\ \text{Eh} &= u(\text{ZE}) / 96485.309 \\ \text{Ah} &= u(\text{ZE}) \\ \text{pe} &= u(\text{ZE}) / (2.303\text{RT}) \\ \text{I} &= 1/2 * \text{sum}(\text{ML}(\text{AQ}, \text{sp}) * Z(\text{AQ}, \text{sp}) ^ {**2}) \\ \text{m}^* &= \text{sum}[\text{ML}(\text{AQ}, \text{ions})] + \text{sum}[\text{ML}(\text{AQ}, \text{complexes})] \\ \text{Aw} &= \text{ACR}(\text{H}_2\text{O}, \text{AQ}) \\ \text{Os} &= -55.5084 / \text{m}^* \ln \text{Aw} \\ \text{At1} &= \text{m}(\text{HCO}_3^-) + \text{m}(\text{CaHCO}_3) + \dots + 2\text{m}(\text{CO}_3^{2-}) + 2\text{m}(\text{CaCO}_3) + \dots \\ \text{At2} &= \text{m}(\text{HCO}_3^-) + \text{m}(\text{CaHCO}_3) + \dots + 2\text{m}(\text{CO}_3^{2-}) + 2\text{m}(\text{CaCO}_3) + \dots \\ &\quad + \text{m}(\text{OH}^-) + \text{m}(\text{CaOH}) + \dots + \text{m}(\text{HS}^-) + 2\text{m}(\text{S}^{2-}) + \dots \end{aligned}$$

10.7.2.5 Thermodynamic models and databases for Pourbaix-module calculations

As mentioned in *Section 10.7.2.2*, all thermodynamic data for calculating a Pourbaix or stepping diagram in the POURBAIX module should be either retrieved from one single database that consists of aqueous solution, gaseous mixture and solid (stoichiometric and solution) phases, or appended from several individual databases that separately contain varied phases. Some databases are already set as the default databases in the module. In the Single-Database case, the PAQ (*TCS Public Aqueous Solution Database for POURBAIX diagram calculations*) is the default database; *since TCCS/TCW5, the PAQS (*TGG Public Aqueous Solution Database for POURBAIX diagram calculations*) is also freely included*. In the Multiple-Database case, the TCAQ or AQS (*Aqueous Solution Databases* using the SIT or the complete HKF Models, respectively) is default for the major database (for aqueous solution), and SSUB (*SGTE Substances Database* for stoichiometric solid phases and gaseous mixture) and SSOL (*SGTE Solutions Database* for solid solution phases) are default for the first and second appended databases. A user can also choose other available databases or his/her own specific databases in various cases. However, one should be well aware of that a certain aqueous bearing database must be associated with appropriate thermodynamic model.

For detailed descriptions of thermodynamic models and databases for various phases/materials, please refer to the related parts in this Users' Guide. For the convenience of users, the available thermodynamic models and databases for aqueous solutions are briefly reviewed in the following:

1) Thermodynamic models for aqueous solutions:

In the previous three Thermo-Calc releases (in TCC versions L, M and N), the aqueous solutions have been treated as relatively simple, using the SIT (Specific Ionic Interaction Theory) Model. The EOS (Equation of

State) and thermodynamic properties of H₂O itself are applicable for calculations from room temperature/pressure up to a little bit below its critical temperature and pressure. The model is not so good in the H₂O critical region and high concentration. For aqueous solutes (cation/anion and inorganic/organic complexes species), the available data are mainly for the room temperature and pressure region, but for many species there are C_p data so that they can be applied over 0–350°C. In conjunction with the SIT model, three different databases have been developed (see more details below) and made available in the Thermo-Calc package.

- TCAQ 84 elements, incl. only the aqueous solution phase. (current version TCAQ2)
The ordinary TC-AQ aqueous solution database associated with the SIT model.
- PAQ 11 elements, incl. aqueous, solid and gas phases. (current version PAQ2)
The specially-designed public database for Pourbaix-module calculations in TCC and TCW, as TCC-Demo/TC4A and TCW-Demo/TC4U.

Since TCCP, the complete Revised HKF (Helgeson-Kirkham-Flowers) Model is used to handle complex aqueous solutions. Simultaneously, a comprehensive database, namely the AQS (TGG Aqueous Solution Database, with the current version as AQS2; see more details below), has been developed. Compared with the SIT model, this model is much more comprehensive and has much wider applications, thanks to the following factors:

- ❖ The EOS and thermodynamic and transport properties for pure H₂O can be precisely calculated using the Johnson-Norton Model in a very wide temperature-pressure region covering the entire subcritical, supercritical and critical regions, *i.e.*, 0–1000°C and 1–5000 bar.
- ❖ The EOS and standard thermodynamic properties for aqueous solute species (free cation/anion ions, inorganic/organic complexes) are described by the Complete Revised HKF Model and some modifications. There are always some assessed or estimated C_p (T,P,X) data for various aqueous species. Therefore the calculations for aqueous solution can be made in the same temperature-pressure region as for pure H₂O, *i.e.*, 0–1000°C and 1–5000 bar.
- ❖ The thermodynamic properties for mixing behaviors in aqueous solutions are described by the Complete Revised HKF Model and some modifications. There are many available binary and ternary interaction data for complex aqueous solutions. So the calculations can be made for concentrated aqueous solutions, *i.e.*, up to 6 m (eq. NaCl) at room temperature-pressure and more than 10 m at high temperature-pressure.

In future releases, the Generalized PITZ (Pitzer Formalism) Model will also be implemented to handle extremely concentrated and specific aqueous solutions (such as brines). Due to the weakness of the Pitzer Formalism in dealing with EOS and standard thermodynamic properties of pure H₂O and aqueous solution species at evaluated temperatures and pressures, parts of the treatments used for the Complete Revised HKF Model will be adopted to the PITZ model, including the Johnson-Norton model for pure H₂O, the standard thermodynamic properties for solvent and solutes, and the DHLT term of the HKF model. With this combination and in conjunction with the AQS database and some extensions regarding high-order interaction parameters which are required by the Pitzer model, it will give a much better description than any previously modified Pitzer formalism, for the EOS and thermodynamic properties of solvent and solutes, and the entire aqueous solution phase.

2) Thermodynamic databases for aqueous solutions:

As for the demonstration purposes, PAQ (*TC Public Aqueous Solution Database*) is freely distributed in the full versions of TCC and TCW, and in their demo versions (TCC-Demo/TC4A and TCW-Demo/TC4U). It was specially designed for the Single-Database case in POURBAIX module calculations, which satisfies with varied requirements for calculating Pourbaix diagrams and various property diagrams as described above. Beside the aqueous solution phase, they also contain some related gaseous mixture, pure solid and solid solution phases. However, this small-sized database only contains few chemical elements: 11 (H-O-C-N-S-Cl-Na-Fe-Co-Ni-Cr). Therefore, they are considered as rather basic and demonstrating databases, and can only be applied to some limited chemical or hydro-metallurgical or alloy corrosion problems. For other specially-designed databases (with some limited elements) to be used with the Single-Database option in the POURBAIX module, one could contact with Thermo-Calc Software.

Intended for much wider applications, two large-scale aqueous solution databases covering very big chemical frameworks have been developed and gradually updated. The first one is the TCAQ Aqueous Solution Database, and the second one is the AQS Aqueous Solution Database. Both aqueous solution databases can be used in the POURBAIX module via the Multiple-Database option, by appending various phases (gaseous/fluid mixtures, steels, alloys, melts, slags, salts, ceramics, minerals, nuclear materials, *etc.*) from many other available databases (*e.g.*, SSUB, SSOL, TCFE, TCNI, SLAG, SLAT, ION, TTNi/Ti/Al/Mg, TCMP, TCES, GCE, NUOX, *etc.*). Moreover, both can also be used in other basic-module calculations of complex heterogeneous interaction systems involving aqueous solutions. Therefore, they can be widely applied in materials design and engineering, chemical engineering, chemistry, geochemistry, environmental assessments, and so on.

The TCAQ database (current version as TCAQ2) is a comprehensive aqueous solution database covering 76 elements (which are in a similar framework as in the SSUB and SSOL databases) as listed below:

Ag	Al	Ar	As	Au	B	Ba	Be	Br	C	Ca	Cd	Ce	Cl	Co	Cr	Cs	Cu	Dy	Er
Eu	F	Fe	Ga	Gd	H	He	Hg	Ho	I	In	K	Kr	La	Li	Lu	Mg	Mn	Mo	N
Na	Nd	Ne	Ni	O	Os	P	Pb	Pd	Pr	Pt	Ra	Rb	Re	Ru	S	Sb	Sc	Se	Si
Sm	Sn	Sr	Tb	Te	Th	Tl	Tm	U	V	W	Xe	Y	Yb	Zn	Zr				

It is associated with the SIT Model. Various aqueous species (totally about 350, mainly as complex species) within this element structure have been gradually added and updated since it was first released in early 1998.

The AQS database (current version as AQS2) covers 83 elements and about 1600 inorganic/organic species.

Ag	Al	Ar	As	Au	B	Ba	Be	Bi	Br	C	Ca	Cd	Ce	Cl	Co	Cr	Cs	Cu	Dy
Er	Eu	F	Fe	Fr	Ga	Gd	H	He	Hf	Hg	Ho	I	In	K	Kr	La	Li	Lu	Mg
Mn	Mo	N	Na	Nb	Nd	Ne	Ni	O	P	Pb	Pd	Pm	Pr	Pt	Ra	Rb	Re	Rh	Rn
Ru	S	Sb	Sc	Se	Si	Sm	Sn	Sr	Tb	Tc	Th	Ti	Tl	Tm	U	V	W	Xe	Y
Yb	Zn	Zr																	

This database is associated with the Complete Revised HKF Model. It will incorporate more critically-assessed data (for more complex species of these existing elements, and for more other elements and their complex species) in the near future.

Along with the implementation of the Generalized PITZ Model, a suitable aqueous solution database or some small aqueous datasets for using the Pitzer formalism will be developed in the near future.

10.7.3 Applications of the POURBAIX module

The POURBAIX module is unique in multiple functionality, and well designed in user-friendly interface. Various options at different levels make it very simple and easy to perform calculations and graphical processing of Pourbaix diagrams and many property diagrams for varied heterogeneous multicomponent systems over a very wide temperature-pressure-composition range. As already mentioned in the module descriptions (*Section 10.7.2*), a POURBAIX-module generated *.POLY3 file can be utilized not only for various calculations of Pourbaix diagrams and many property diagrams, but also for any simulation of complex heterogeneous interaction processes involving aqueous solutions.

The module and the associated models and databases can be extensively used in R&D of aqueous chemistry, materials chemistry, materials corrosion, materials engineering and processing, chemical engineering, geochemistry, Earth materials exploitation, environmental engineering, energy transfer, nuclear power production, nuclear waster repository assessment, recycling, and so on.

In the following paragraphs, some reviews will be given on some existing applications but with emphases on materials corrosion.

The aqueous corrosion resistance of stainless steels is attributed to the formation of a passive surface film. As early as in 1836, Faraday noticed the passivation of iron in nitric acid solution, and observed that the corrosion rate in dilute HNO₃ solution initially increased with acid concentration, and that at a critical concentration the rate fell to negligible values despite the enhanced thermodynamic driving force, due to the formation of a submicron thin metal oxide film. It was further confirmed later that chromium also passivates, however, at a much lower current density (<1 µA/cm² in 1 m H₂SO₄) than iron (7 µA/cm² in 1 m H₂SO₄). Therefore, it has

been advantageous to add approximately 12 wt% Cr to Fe to enhance the formation of a passive film. To achieve the aqueous corrosion resistance provided by stainless alloys with lower strengths and impact resistance, the stability of the passivating film formed must be maintained. This has been the basis of the stainless steel industry.

The POURBAIX module can be used for either pure alloy (*e.g.*, Fe, Cr, Ni, Al) or rather complex alloys (*e.g.*, Fe-Cr-Co-Ni-C-N-O-S-...), interacting with various specific aqueous solutions at certain temperature and pressure conditions.

It has been demonstrated, in *Figure 10-10* in the *Section 10.7.1*, that the shape of a Pourbaix diagram of pure iron and the stability relations of various Fe-bearing secondary phases depends upon the initial amount of Fe, initial amount and composition of the interacting aqueous solution phase, and temperature and pressure conditions in the heterogeneous interaction system. *Figure 10-11* further shows that the Pourbaix diagram of two Fe-based alloys also depends on its initial alloying composition. In the figure, other property plots that are generated from the POURBAIX module are also presented, *e.g.*, molality of aqueous species verse pH, ionic strength verse pH.

However, for a complicated multicomponent alloy system, there might be highly loaded curves on the diagrams which look overlapped by lines corresponding to individual alloying elements' diagrams in similar aqueous solutions. So if the alloy system in an interaction system is relatively simpler, it will be more meaningful for the calculated curves, and be much easy to understand to plotted pH-Eh diagram and property diagrams.

When the alloy composition is simple while the aqueous solution compositions are rather complicated, the Pourbaix diagrams will also look very crowded as there are so many elements in the systems; and sometimes it would be rather difficult to map the whole pH-Eh range for the interaction systems. Therefore, it might be more interesting to make simple equilibrium calculations at each points (with the specific bulk composition and pressure and temperature), or to make the stepping calculations by utilizing the Main Option 4 [varying with either T, X(Fe) or X(Cr) in the alloy, or m(NaCl), or m(SO₄²⁻), or other conditions], rather than mapping a wide pH-Eh range.

An equilibrium point calculation for a heterogeneous interaction system involving aqueous solution can be made in the POLY module, most conveniently by reading a POURBAIX-module *.POLY3 file (with various symbol definition saved; see *Point 15 in Section 10.7.2.3*). Once can modify temperature, pressure, bulk compositions (of alloys or aqueous solution), pH and Eh conditions, in order to calculate the heterogeneous equilibrium. The results can be easily listed out by using the POLY command LIST_EQUILIBRIUM.

By using the Main Option 4 of the POURBAIX module or the normal POLY module, an existing POURBAIX-module *.POLY3 file can be opened, and a stepping calculation can be made. Then, many different property diagrams can be plotted inside the POURBAIX module or POST module.

Figure 10-12 presents some property diagrams where stepping calculations (with temperature as the stepping variable) are made for the stable phase assemblages magnetite+linnaeite+FCC(Ni-Co-Fe)+aqueous and magnetite+hematite+pyrite+linnaeite+millerite+aqueous in the heterogeneous interaction between an Fe-based alloy (1E-3 m Fe, 5E-5 m Co and 5E-5 m Ni) and an aqueous solution (1 kg of water, with 1E-6 m S₂ or open to S₂) at a pressure of 1 kbar. So various equilibrium quantities (such as molality of aqueous species, pH, Eh, input amount of S₂) are plotted against temperature.

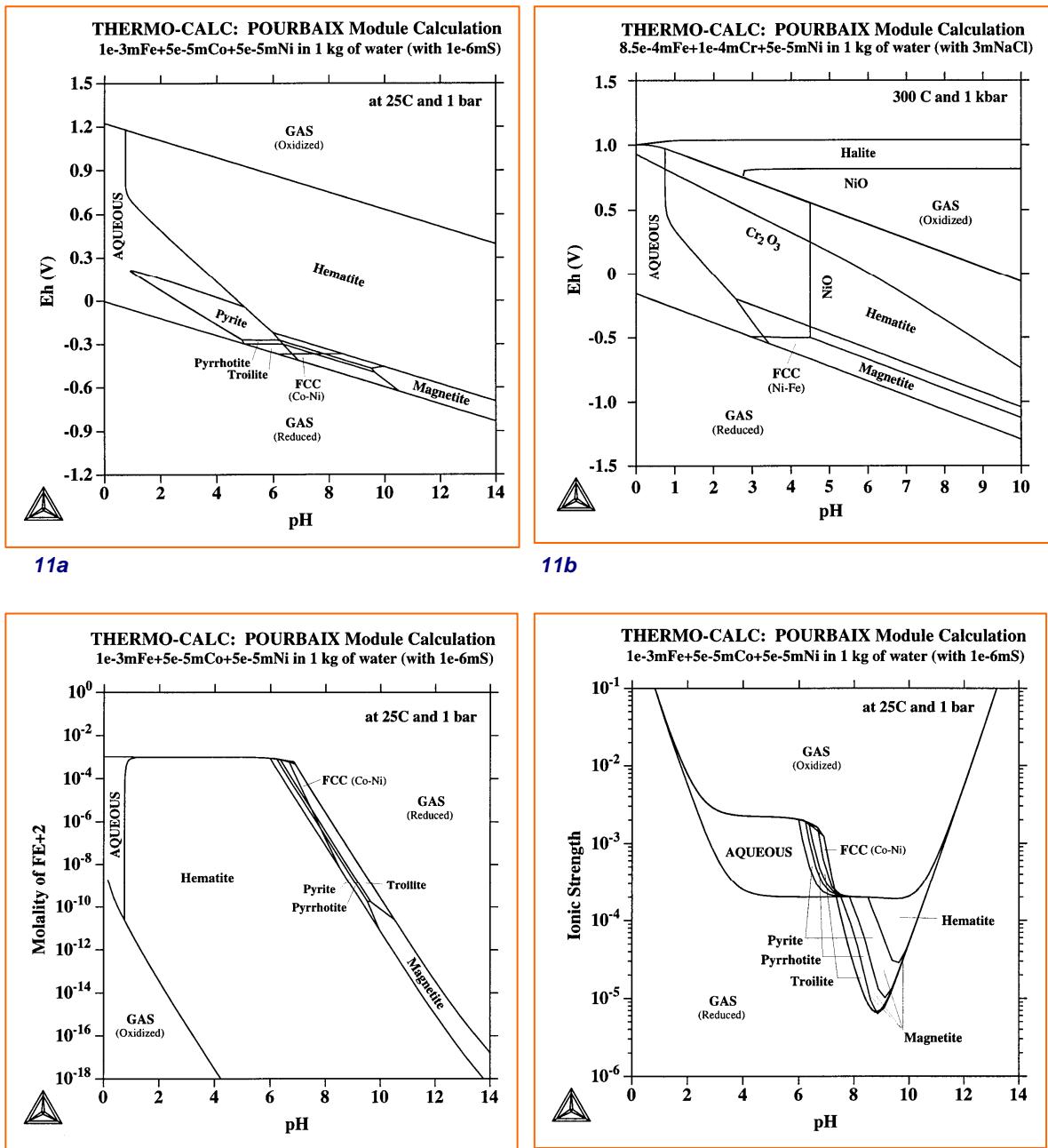
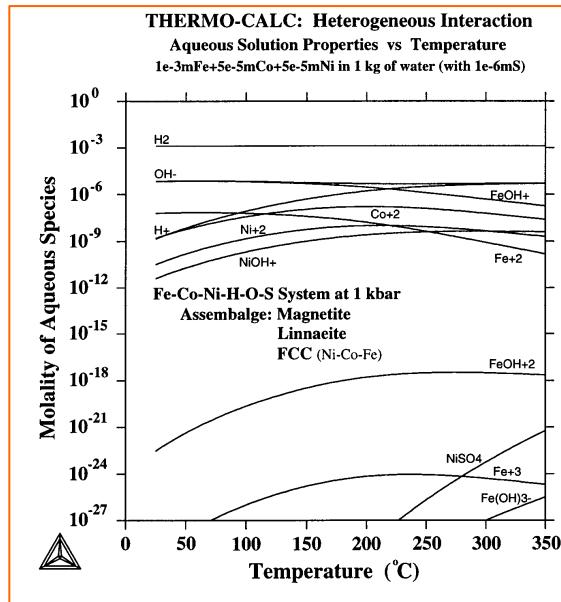
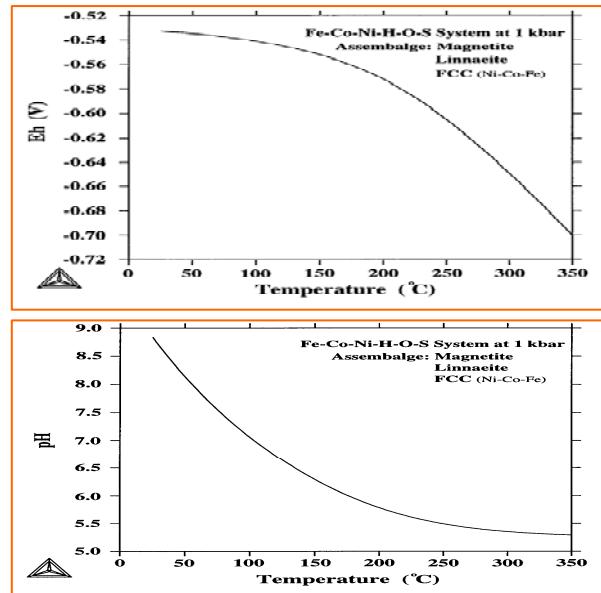


Figure 10-11. Pourbaix diagrams and property plots of two Fe-based alloys, actively interacted with an aqueous solution (1 kg of water with 3 m NaCl or 1E-5 m S₂) at constant temperature and pressure conditions.

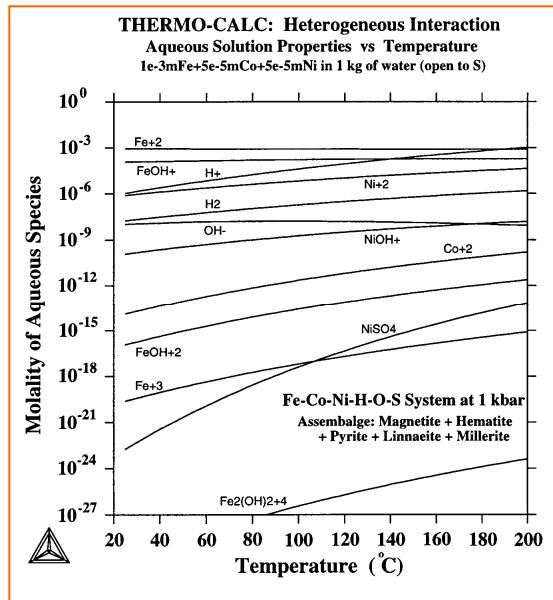
- I1a.** Pourbaix diagram for the Fe-based alloy (with 1E-3 m Fe, 5E-5 m Co, 5E-5 m Ni), actively interacted with an aqueous solution (1 kg of water with 1E-5 m S₂) at 25°C and 1 bar.
- I1b.** Pourbaix diagram for the Fe-based alloy (with 8.5E-4 m Fe, 1E-4 m Cr, 5E-5 m Ni), actively interacted with an aqueous solution (1 kg of water with 3 m NaCl) at 300°C and 1 kbar.
- I1c.** Molality of Fe⁺² verse pH plot for the Fe-based alloy (with 1E-3 m Fe, 5E-5 m Co, 5E-5 m Ni), actively interacted with an aqueous solution (1 kg of water with 1E-5 m S₂) at 25°C and 1 bar.
- I1d.** Ionic strength verse pH plot for the Fe-based alloy (with 1E-3 m Fe, 5E-5 m Co, 5E-5 m Ni), actively interacted with an aqueous solution (1 kg of water with 1E-5 m S₂) at 25°C and 1 bar.



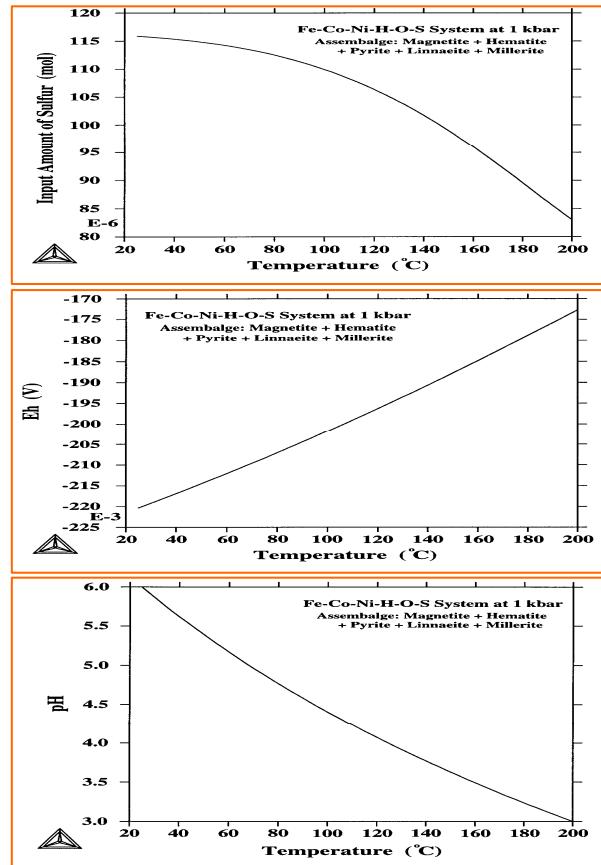
12a



12b, 12c



12d



12e, 12f, 12g

Figure 10-12. Property diagrams for the stable phase assemblages (aqueous solution involving magnetite+linnaeite+FCC(Ni-Co-Fe) and magnetite+hematite+pyrite+limnaeite+millerite in the heterogeneous interaction between an Fe-based alloy (1E-3 m Fe, 5E-5 m Co and 5E-5 m Ni) and an aqueous solution (1 kg of water, with 1E-6 m S₂ or open to S₂) at a pressure of 1 kbar.

12a,b. Variations of molality of aqueous species, pH and Eh along with temperature, for the assemblage magnetite+linnaeite+FCC(Ni-Co-Fe)+aqueous in the heterogeneous interaction between an Fe-based alloy (1E-3 m Fe, 5E-5 m Co and 5E-5 m Ni) and an aqueous solution (1 kg of water, with 1E-6 m S₂).

12d,e,f,g. Variations of molality of aqueous species, pH, Eh and input amount of S₂ along with temperature, for the assemblage magnetite+hematite+pyrite+linnaeite+millerite+aqueous in the heterogeneous interaction between an Fe-based alloy (1E-3 m Fe, 5E-5 m Co and 5E-5 m Ni) and an aqueous solution (1 kg of water, open to S₂).

10.7.4 Demonstrative examples of the POURBAIX-module calculations

The **Example 40** in the /TCEX/ area (TCEX40.TCM) and in **TCCS Examples Book** gives a very comprehensive demonstration on how various POURBAIX-module calculations and graphical processing are performed. This example consists of five separate parts:

- | | |
|--------------------------------|--|
| Example 40a TCEX40a.TCM | Demonstration of the Main Option 1 calculation (with the Single-Database choice) and graphical processing of a Pourbaix diagram for a simple application; going through various on-line help features. |
| Example 40b TCEX40b.TCM | Demonstration of the Main Option 2 graphical processing (by plotting more diagrams, and with various options for refining a defined diagram (including directly interacting with the POST module). |
| Example 40c TCEX40c.TCM | Demonstration of the Main Option 3 calculation (by modifying T-P-X conditions) and graphical processing of a Pourbaix diagram. |
| Example 40d TCEX40d.TCM | Demonstration of the Main Option 4 calculation (by utilizing various stepping variables) and graphical processing. |
| Example 40e TCEX40e.TCM | Demonstration of the Main Option 1 calculation (with the Multiple-Database choice) and graphical processing a Pourbaix diagram for a more complex application. |

Since TCCS, the new **TCC Example 53** in the /TCEX/ area (TCEX53.TCM) and in **TCCS Examples Book** demonstrates calculations of Pourbaix diagrams using various TCC commands, i.e., being performed in an ordinary way of going through various basic modules (TDB, GES, POLY and POST), rather than via the POURBAIX Module. This examples shows all the basis and techniques for calculating Pourbaix diagrams for complex heterogeneous interaction system, behind the POURBAIX module as well.

10.7.5 References on the POURBAIX-module calculations

- Bard A.J., Parsons R., and Jordan J. (Eds.) (1985) Standard Potentials in Aqueous Solutions, IUPAC.
- Campbell C.E. (1997) *Systems Design of High Performance Stainless Steels*. Ph.D. Dissertation, Northwestern Univ., 243 p.
- Craig B.D. (1991) *Fundamental Aspects of Corrosion Films in Corrosion Science*, Plenum Press, NY.
- Kirchheim R., et al. (1989) The passivity of iron-chromium alloys. *Corr. Soc.*, **29**, 899-917.
- Pourbaix M. (1966) *Atlas of Electrochemical Equilibria in Aqueous Solution*. Pergamon, NY.
- Shi P., and Sundman B. (1997) Aqueous solution models, databases and modules implemented in Thermo-Calc. *CALPHAD XXVI Technical Program & Abstract* (Palm Coast, FL, May 11-16, 1997), 5e.
- Shi P. (1998) Heterogeneous equilibrium calculations involving aqueous solutions using Thermo-Calc. *CALPHAD XXVII Technical Program & Abstracts* (Beijing, May 17-22, 1998), 65.
- Shi P. (1999) Applications of Thermo-Calc aqueous solution databases. *CALPHAD XXVIII Programme & Abstracts* (Grenoble, May 2-7, 1999), 134.
- Shi P., and Sundman B. (2000) New aqueous solutions model and database implemented in the Thermo-Calc package. *TOFA-2000 Technical Program & Abstracts* (Stockholm, May 8-11, 2000).
- Sundman B., and Shi P. (1997) Applications of Thermo-Calc in high temperature materials chemistry. In: Spear K.E. (Ed.) *High Temperature Materials Chemistry IX* (State College, PA, May 19-23, 1997), *Electrochemical Society Proceedings Volume 97-39*: 52-59.
- Yokokawa H., et al. (1990) Generalized electrochemical potential diagrams for complex aqueous (M-X-H-O-e⁻) system. *J. Electrochem. Soc.*, **137**, 388-398.

10.8 SCHEIL Module

10.8.1 Solidification processes

The solidification of multicomponent steels and alloys often involves various types of non-equilibrium interactions, linear or non-linear dynamic processes and random fluctuation in the system (diffusion, mold filling, eutectic and peritectic precipitation, segregation, solute-drag partitionless transformation, nucleation, particle growth/dissolution, ion-exchange reaction, surface reaction, fluid flow, advection, convection, infiltration, heat flow, curvature-controlled transformation, mechanically-enhanced transformation, chemical oscillation, self-organization, *etc.*).

An ambitious process simulation shall take into account the mass transfers, energy transports and momentum transmissions contributed by various processes. However, treat all these in one simulation software is extremely complicated, and hence various approximations and simplifications are always needed. There are two groups of approaches using different methods, one with the emphasis on geometric aspects and the other on thermodynamics and kinetics. Both techniques have been applied successfully to some practical cases. Recently, more and more simulations with emphasis on bridging two techniques become available, thanks to the applications of programming interfaces of some powerful simulation software and to the uses of modern computers.

In a geometrical simulation, approximate data on thermodynamic properties (like latent heat, heat capacity, solubility, *etc.*) and kinetic quantities (like diffusion coefficient, reaction constant, fluid flux, heat flux, *etc.*) are usually used. There is no ambition that the thermodynamic data can reproduce any equilibrium properties of the system (for instance, the phase diagrams), and the kinetic data can fit any interaction quantities (for example, the compositional profiles). Often happened is that the thermodynamic and kinetic values are just treated as adjustable parameters in order to fit some experimental results. The simulation is normally made by dividing the mold into a mesh with more than 1000 grid-points and in each grid-point the mass transfers and heat transport are solved using a finite element method under a certain type of boundary condition and initial condition.

On the other hand, in a thermodynamic simulation (*e.g.*, Thermo-Calc), precise thermodynamic descriptions on a system is utilized, while it is very important to know the final state the system is trying to reach. A good set of thermodynamic models and data allows reliable extrapolation into non-equilibrium states (treated as metastable states in local or partial equilibria), and can provide the driving forces for chemical reactions (precipitation or dissolution) and the thermodynamic factors for diffusion. This can greatly simplify the kinetic and other factors.

Moreover, in a kinetic simulation (*e.g.*, DICTRA), it is possible to use precise thermodynamic descriptions together with some kinetic models and data for describing mass and heat fluxes in the solidification processes under some kinds of restricted and simple geometry (*e.g.*, planar -- an infinitely wide plate of a certain thickness; cylindrical -- an infinitely long cylinder of a certain radius; tube -- an infinitely long tube of a certain radius; and spherical -- a sphere with a certain radius), which are often treated as one-dimensional.

With the applications of programming interfaces (*e.g.*, TQ and TCAPI interfaces of the Thermo-Calc engine, occasionally with the DICTRA extension), a combination of comprehensive thermodynamic and kinetic treatments with more general geometries in two- or three-dimensions has become possible nowadays. Although it is a somewhat difficult task as a full integration requires efficient and reliable solutions of both thermodynamic equilibrium and kinetic scheme at each grid-points in the mesh, such efforts have been enforced in some simulation exercises (such as the AMSS approach and applications by ACCESS e.V.; for more details, please refer to the forth-coming pages).

In the following part, some examples will be given on the thermodynamic and kinetic simulations on solidification processes performed by the Thermo-Calc and DICTRA software. As well, one example will be illustrated on how one can write his/her own application program(s), using the TQ and TCAPI application programming interfaces to integrating with the Thermo-Calc (and DICTRA) engines.

10.8.2 Scheil-Gulliver Solidification Model (under thermodynamic approximation)

As illustrated in *Figure 10-13*, one qualitative approximation in simulating such complex solidification processes is to use the Scheil-Gulliver Model (Scheil, 1942; Gulliver, 1913), by assuming that the diffusion coefficients of components in the liquid phase are infinitive fast whereas in solidified phases are zero. Along each step in the cooling process, local equilibria establish at the solidification interface where the composition of the alloy may be significantly different from the overall composition, and the compositions of liquid and solid at the interface are given by the system's phase diagram. The solid phase formed will retain its composition from when it was formed whereas the liquid composition is always homogeneous.

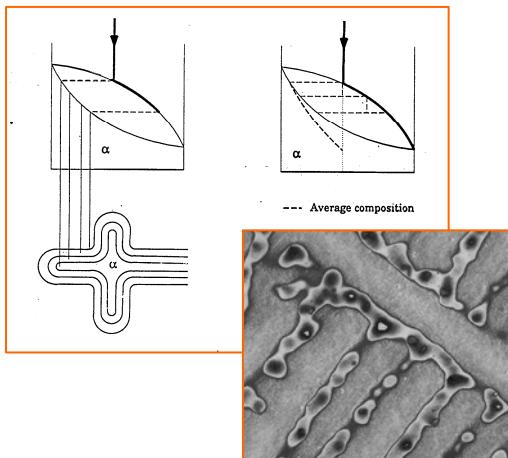


Figure 10-13. Scheil-Gulliver Model for simulations of alloy solidification process.

The Scheil-Gulliver model has been implemented as the so-called **SCHEIL Module** inside the Thermo-Calc (TCC and TCW) software for many years, which can be applied to any high-order multicomponent system. For all the details of the SCHEIL Module in the TCC Software, please refer to [Section 10.8.4](#) (below); and of the SCHEIL module in the TCW Software, please refer to the [TCW User's Guide and Examples Book](#).

10.8.3 Moving Boundary Model (under kinetic approximation)

By applying the Moving Boundary Model incorporated with accurate atomic mobility data in the DICTRA software package, one can more precisely predict the solidification paths by considering the complex diffusion mechanism.

Figure 10-14 illustrates an example of simulated solidification path for an alloy, resulted from the SCHEIL module in the TCC software and from the DICTRA software. For more details, refer to the **DICTRA Information Package** and the DICTRA manuals.

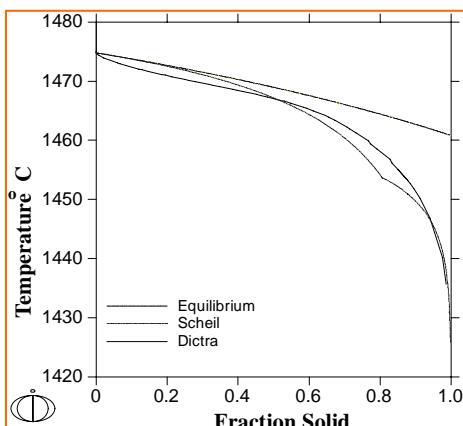


Figure 10-14. Solidification path simulation by both TCC (the SCHEIL module) and DICTRA, for an Fe-18%Cr-8%Ni alloy. The cooling rate in the DICTRA simulation is 1 K/s and the size of the system is 100 μm . The Scheil-Gulliver simulation in TCC assumes that there is no diffusion in solidified alloy while the diffusion in melt is infinitely fast.

10.8.4 Descriptions of the SCHEIL Module

Based on the widely-used Scheil-Gulliver model, a special and easy-to-use module, *i.e.*, the so-called **SCHEIL Module** has been implemented inside the Thermo-Calc (TCC and TCW) software. The module can be applied to any high-order multicomponent system. A simulation is made step by step along the cooling procedure (for instance, with decreasing temperature or extracted heat), and after each step the new liquid composition is used as the “local overall” composition at the next step.

The general procedure of the traditional Scheil-Gulliver model simulation can be illustrated by *Figure 10.15* and is briefly described as follows:

- 1) Start with system that is on the liquidus line at temperature T_1 and overall composition x_1 .
- 2) The temperature condition is decreased to T_2 and the equilibrium is calculated. This gives a certain amount of solid phase(s) formed and a new liquid composition x_2 . The system with the overall composition equal to x_2 would be completely liquid at this temperature.
- 3) The overall composition is set to x_2 . This effectively means that the program “forgets” the amount of solid phase(s) formed previously, and that the solid phase(s) will remain at the composition at which it was formed.
- 4) The simulation is continued from the step 2), and repeated until the lowest temperature where liquid can exist (either all liquid disappear or a certain fraction of liquid remain in the system) is found.

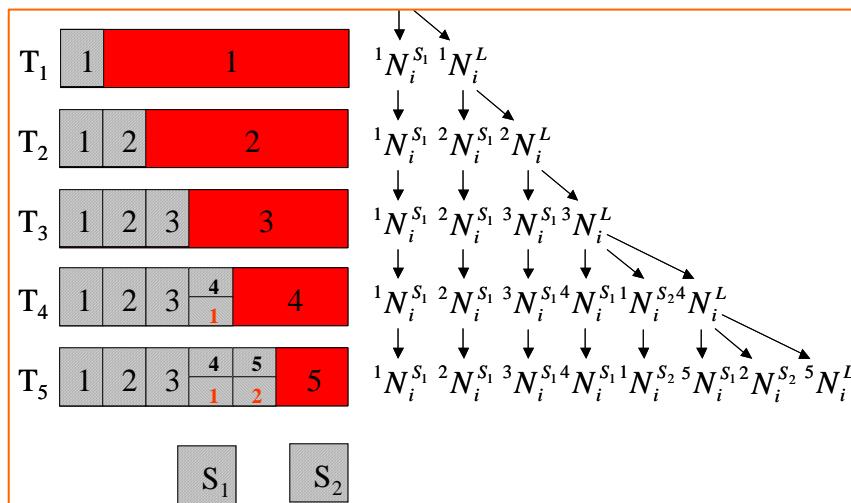


Figure 10-15. Algorithm in Traditional Scheil-Gulliver Model for simulations of alloy solidification process.

The simulation can easily handle cases where two or more solid phases precipitate at the same time. If it is a eutectic precipitation, all phases will form at each step and the liquid composition will follow the “eutectic valley”. If it is a peritectic transformation, the phase that is no longer stable will disappear from the local equilibrium at the liquid interface. The amount of that phase already formed will not transform as no account is taken of solid phases formed at earlier steps.

Thermo-Calc allows the users to enter functions using state variables and other functions. It is even possible to have a function referring to itself and in such a case the old value of the function will be used when calculating the new value. This unique feature is useful to keep track of the total amount of remaining liquid. Before the first step, the initial fraction of liquid is unity. After each step and equilibrium calculation, the fraction of liquid is given by the “lever rule”, and can be obtained by the state variable NP(LIQUID). If the total fraction of liquid before the step is N_L , it after the step becomes $N_L = N_L * NP(LIQUID)$. If one denotes $NP(LIQUID)$ at step k as f_k^L , the remaining fraction of liquid at step n is the product:

$$f_n^L = \prod_{k=1}^n f_k^L$$

After each step, the conditions for the overall composition X (A) must be changes accordingly, and this can be simply done by setting it to the new liquid composition X (LIQUID, A); however, for the reason that the syntax X (A)=X (LIQUID, A) is illegal in Thermo-Calc, this must be conducted using an intermediate variable like XA, in the way as shown below:

```
ENTER_SYMBOL VARIABLE XA=X(LIQUID,A)
SET_CONDITION X(A)=XA
```

Using these features in Thermo-Calc, it is possible to make a Scheil-Gulliver solidification simulation using the STEP EVALUATE command-sequence. It is of great interest that one may use not only the temperature as stepping variable but one may use the enthalpy, *i.e.*, the heat extracted. Depending on the cooling rate and if the amount of latent heat varies significantly during the solidification process, the use of enthalpy as stepping variable may give more realistic results.

For some time, it has been possible to utilize such an approach in the TCC and TCW software to simulate multicomponent non-equilibrium solidification processes of steels or alloys using the Scheil-Gulliver model (Sundman *et al.*, 1993), and it was found that the agreement between the predicted and experimental results was extremely good (Saunders, 1996; Jansson and Sundman, 1997).

However, due to the advanced commands and special techniques involved in the simulations (as described above), beginners and occasional users found it hard to follow the procedure and get the correct results. Therefore, one can write a simple application program to handle this type of simulation by using the TQ interface. Some examples of application programs connected with the Thermo-Calc Engine via the TQ interface for Scheil-Gulliver solidification simulation are presented in the document **TQ Thermodynamic Calculation Interface: Programmer's Guide and Examples**.

In addition, the special and easy-to-use module, namely the **SCHEIL** or **SCHEIL_SIMULATION** module implemented in the TCC and TCW software, can be directly used for the solidification simulations. The module was developed for the purpose of simplifying the user interface for performing a Scheil-Gulliver simulation, and generating various diagrams for the solidification process. The module is very user-friendly, and just asks the users to answer some simple questions on database name, major element, alloying elements and their amounts; the module automatically finds the exact liquidus point, starts the simulation procedure (with temperature as the stepping variable), and finally presents a Temperature ($^{\circ}\text{C}$) *verse* Fraction of Solid diagram. Then, one can repeatedly chose different variables to plot other diagrams of one's own interests (*e.g.*, fraction of remaining liquid against temperature, fraction of each solid phase or the total solid phases against temperature, microsegregation in each solid phase, latent heat evolution against temperature, *etc.*). With the SCHEIL module, one can either start a new simulation, open an old file but plot results again, or open an old file but make other simulations with different bulk compositions in the same system or perform a non-equilibrium calculation after an equilibrium stepping in the POLY module.

Figure 10-16 presents the simulation results (as various property diagrams) for the solidification process of an Al-based alloy (with 4 wt.% Mg and 2 wt.% Si) under a cooling temperature step (1°C), by using the so-called **Traditional-SCHEIL Module** (where no back-diffusion is considered) in the TCC software.

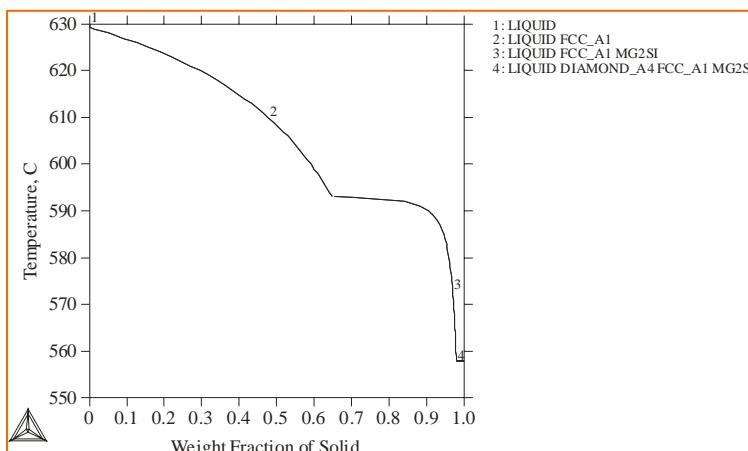
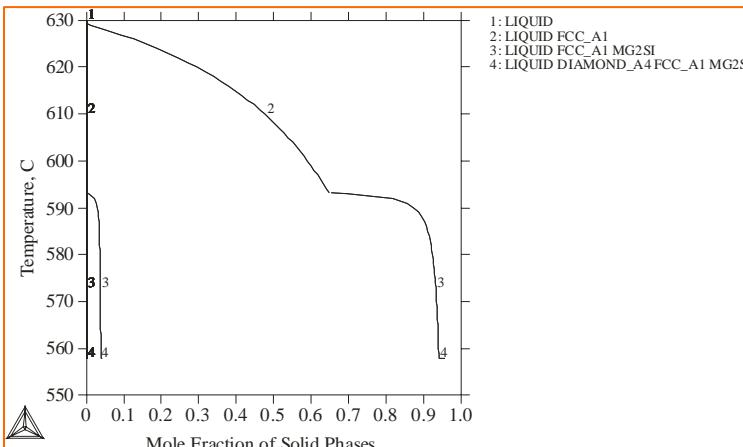
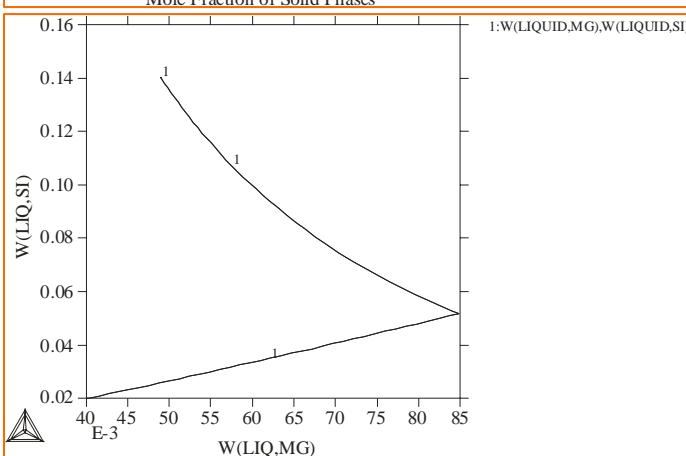


Figure 10-16. Some SCHEIL-module simulation results for the solidification process of the Al-4Mg-2Si alloy.

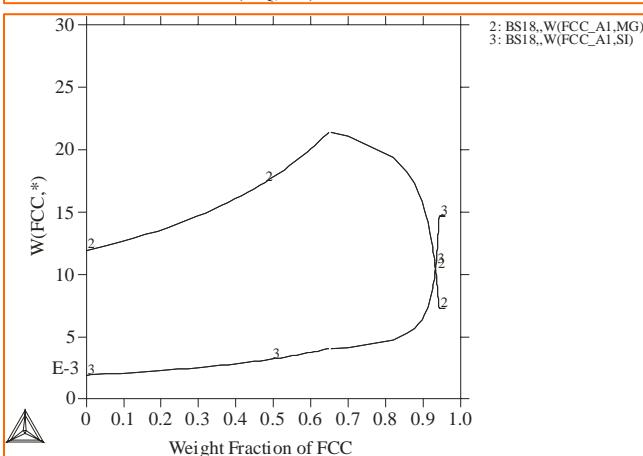
16a. Mole fraction of overall solid phases varied along temperature



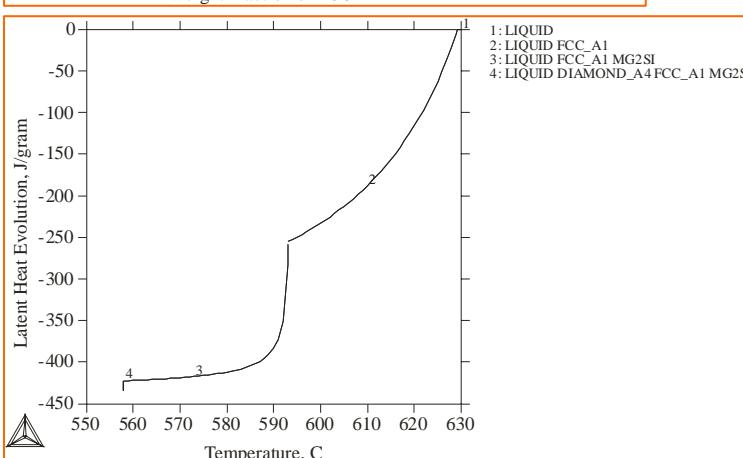
16b. Mole fractions of individual solid phases varied along temperature



16c. The $W(\text{Liq},\text{Si})$ vs $W(\text{liq},\text{Mg})$ plot (the compositional relation between Si and Mg in the remaining LIQUID phase)



16d. The $W(\text{Fcc},*)$ vs $\text{bs}(\text{fcc})$ plot (the weight fractions of all elements in the FCC phase varied as functions of the weight fraction of the phase)



16e. The latent heat evolution along temperature

Since TCCQ and TCW3, the SCHEIL module has one new option for users to define one or more fast diffusing components (usually interstitials such as C, N, O, S, etc.) so that their back diffusion in solid phases (which is significant for steels/Fe-alloys, Ni-superalloys and other alloys, while is probably less important sometimes for e.g. Al-/Mg-alloys) can be automatically taken into account during the Scheil-Gulliver simulation. For steels with primary ferrite, an additional option (allowing BCC \rightarrow FCC transition) has been given to consider its possible transformation to austenite. Therefore, by using this so-called **Modified-SCHEIL Module**, solidification of various alloys can be simulated by using either the original Scheil-Gulliver model or the modified one that allows equilibrium back diffusion of interstitial element(s) in solid phases.

A much better prediction can now be obtained by using the modified Scheil-Gulliver model in TCCQ and TCW3 software (Chen & Sundman, Materials Transaction, JIM, vol. 43, pp. 551-559, 2002), where interstitials are assumed to diffuse fast enough in both liquid and solid phases that the chemical potential of each of them is uniform everywhere in the whole simulation system. *Figure 10-17* gives an illustrative example, and more details can be found in **TCCR Example 48**.

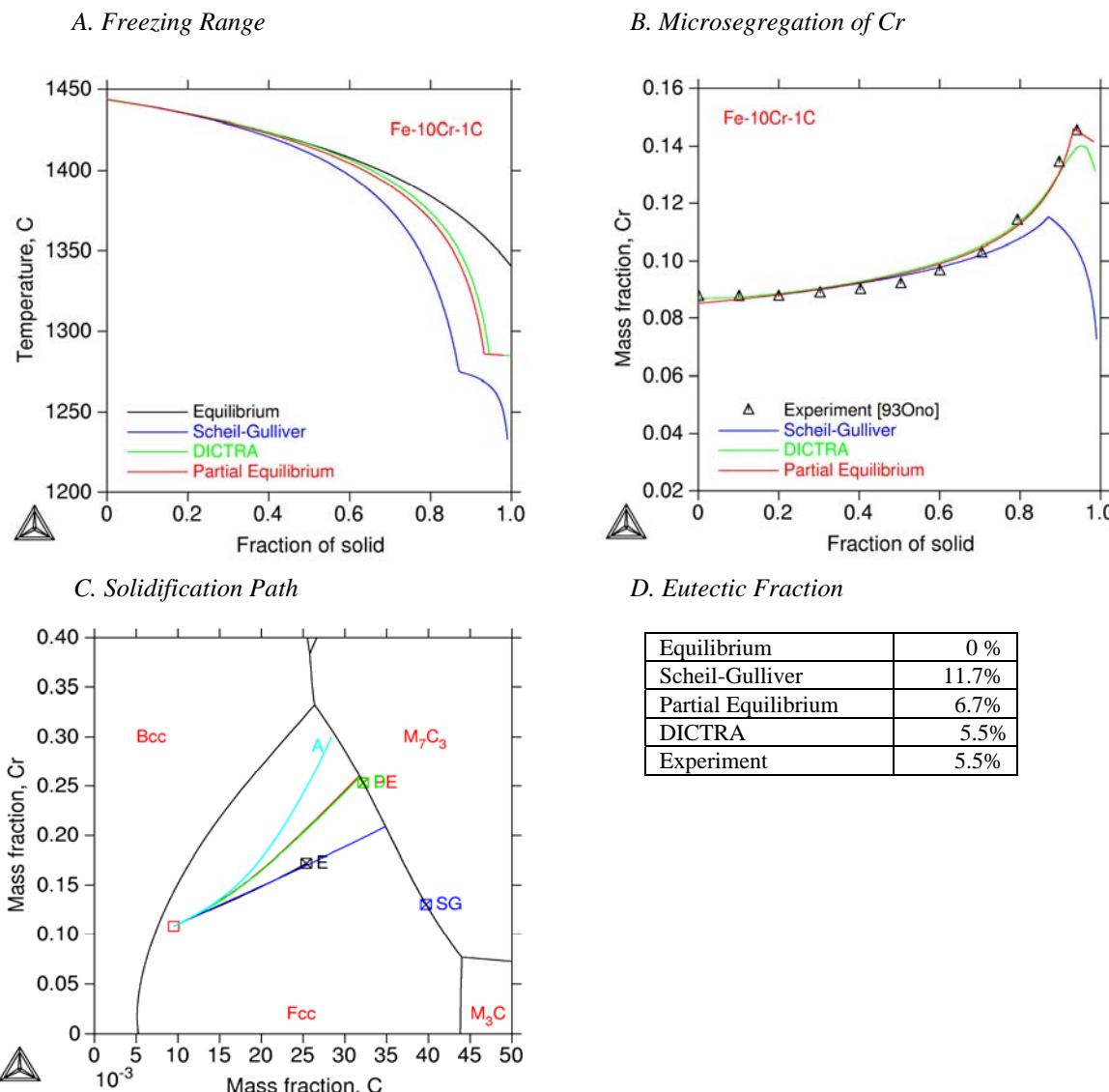


Figure 10-17. Some comparisons among calculated results on solidification path of an Fe-10Cr-C alloy, using various approaches: red lines for advanced Scheil-Gulliver simulation with carbon back diffusion (partial equilibrium), blue lines for normal Scheil-Gulliver simulation, green lines for DICTRA simulation (moving boundary model), black line for equilibrium calculation, and triangular symbol for experimental results on microsegregation measurements.

The *Modified SCHEIL Module* will guide the user through the simulation and plotting of results by prompting questions in successive steps. First it asks the user to choose the database and define the material, *i.e.*, choose major element and enter alloy elements and their compositions. Then the user can supply an initial temperature above the liquidus and select the desired phases in the alloy system. After the actual liquidus temperature is calculated, the user is then asked for solidification simulation conditions [such as cooling temperature step, termination criteria, fast diffusing component(s), *etc.*].

With the given alloy and simulation conditions, the program runs automatically and plots a typical solidification diagram with temperature versus fraction of solid. A user can then choose different quantities from a given list to plot many other interesting property diagrams. There are options in the module to choose either opening a previously saved SCHEIL-type POLY file to make another SCHEIL-module simulation (but with different composition settings), or using such a file to plot more property diagrams.

Interstitial elements diffuse usually much faster than substitutional ones in solid, and due to this fact the Scheil-Gulliver solidification simulation has not been very useful for steels and other interstitial containing alloys.

With the special BCC→FCC option on, the frozen primary ferrite will be allowed to transform to austenite during the solidification process. Such a transformation will change the chemical potentials of interstitials and thus influence their back diffusion in solid. Hopefully, this can approximate more or less the effect of peritectic reaction. It should be mentioned that for the solidification of primary ferrite the equilibrium calculation usually gives better results than both the Scheil-Gulliver model and the modified one with back diffusion of interstitials in solid.

Any database containing a liquid mixture phase (except for ionic liquid mixture phase), solid solution phases and stoichiometric solid phases can be used in this module, such as SSOL, TCFE, TCNI, TCMP, TTNi/Ti/Al/Mg/Zr, NSLD, USLD, SMEC, SNOB, *etc.* When there is a need to combine thermodynamic data for various phases retrieved from two or more compatible databases (such as TCFE3+SSOL2), one can first go through the normal TCC module-routine (*i.e.*, TDB/GES5→POLY3) and save a POLY3 file (with all equilibrium conditions defined, and with one initial equilibrium point calculated), and then in the SCHEIL module open the saved POLY3 file (*i.e.*, the *SCHEIL Main Option 3*; see below) for the simulation.

Please note that since TCCS/TCW5, the following further improvements/modifications have been implemented into the SCHEIL Module:

- ☞ When a database (such as TCFE6/TCFE5/TCFE4) containing volume data (*i.e.*, molar volume, thermal expansivity, and compressibility) has been used for a multicomponent alloy systems, the further-revised/improved SCHEIL-Module can now plot diagrams using one of four additional properties varied during the solidification process [*i.e.*, CP/BCP – Apparent heat capacity per mole/gram, NV/NV (ph) – Molar volume of the system or a phase, DS/DS (ph) – Average density of the system or a phase, and BT – Apparent volumetric TEC of the system] as an axis variable.
- ☞ The further improved SCHEIL-Module can now also handle back diffusion of C in cases where graphite appears during solidification as in cast irons. For previous versions, the simulations always stop where graphite comes out; that was due to that the activity of C in graphite is always unity, then one cannot make the chemical potential of C the same in every phase. In the new treatment, the graphite is simply frozen at its amount, and does not take part in the redistribution of C among various phases; of course, the mass conservation of C is still kept.

The procedure of the SCHEIL module inside the TCC software is very simple, with a few steps to complete a calculation and graphical processing of a phase diagram, or a Gibbs energy curve or phase-fraction plot, for a binary system, as described below:

Step 1: Select a desired SCHEIL Main Option

Upon opening the SCHEIL-module (by the GO SCHEIL command-sequence), the module first show up the following message:

SCHEIL GULLIVER SIMULATION MODULE VERSION 3.0

- 1. Start new simulation
 - 2. Open old file and plot diagram
 - 3. Open old file and make another simulation

and then asks the following question:

Select option /1/: <a Main Option>

where the user shall type a desired SCHEIL Main Option to proceed the current SCHEIL-module simulations and/or graphical handling:

- 1 as Start a new SCHEIL-module simulation;
 - 2 as Open an old POLY3 file (saved by the SCHEIL-module) and plot other diagrams; and
 - 3 as Open an old POLY3 file (saved by the SCHEIL or another module) and make another SCHEIL-module simulation.

Note these three options are referred in the following text as the Main Option 1, 2 and 3, respectively. The Main Option 1 will first get the thermodynamic data for all necessary phases retrieved from a single database, define the bulk compositions of an alloy system and decide the way of performing SCHEIL-module simulation, then start the simulation, and finally plot on screen a default diagram which is temperature ($^{\circ}\text{C}$) versus mole-fraction of solidified solids during the cooling processes.

The Main Option 3 can first open a POLY3 file [that was either saved previously by the SCHEIL module or saved through a normal TCC module-routine (*i.e.*, TDB/GES→POLY); the later one makes it possible to use a GES5/POLY3 workspace that has combined thermodynamic data for various phases which have been retrieved from two or more compatible databases (such as TCFE3+SSOL2)], and modify the previously-defined alloy bulk compositions and modify the way of performing the simulation, and then conduct the simulation and plot the default diagram on screen.

Both the Main Options 1 and 3 also provide the opportunity for plotting other types of diagrams from the SCHEIL-module simulation (similar to the Main Option 2), as well as for refining the plotted diagram and printing/saving the plotted diagram.

Step 2: Choose a database or open an existing POLY3 file

The Main Option 1 allows the user to answer the following question:

Database /TCFE3/: <a database name>

for choosing an existing Thermo-Calc database (such as SSOL, TCFE, TCNI, TCMP, TTNi/Ti/Al/Mg/Zr, NSLD, USLD, SMEC, SNOB, etc.) that may have been available along the TCC installation or has been specified by the user, and that is suitable for the SCHEIL-module simulation for the considered alloy system.

Both the Main Options 2 and 3 require the user to specify the name of an existing POLY3 file that was either saved previously by the SCHEIL module or saved through a normal TCC module-routine (*i.e.*, TDB/GES→POLY); the later one makes it possible to use a GES5/POLY3 workspace that has combined thermodynamic data for various phases which have been retrieved from two or more compatible databases (such as TCFE3+SSOL2).

Under PC-Linux and UNIX environments, it prompts the following question:

File name /scheil/: <a POLY3 file name>

where the user should specify an existing POLY3 file that is located under the current working area.

Under PC Windows Vista/XP/2000/NT4 environments, a **Open file** window will pop up on 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 10-18*. Note that the file type should be always POLY3 and no other file type is legal here. By pressing the **Open** button, the program proceeds with opening the POLY3/GES5 workspaces from the specified, previously-saved POLY3 file. The user may also Cancel such an **Open file** window session, and thus no previously-saved POLY3/GES5 workspaces will be opened. Please note that such an **Open file** window session can be avoided completely, if the name of the existing POLY3 (with its appropriate path if not located in the current work directory) is specified immediately after inputting the Main Option 2 or 3 in the previous step (Step 1), for instance, “3 MyScheil.POLY3”.

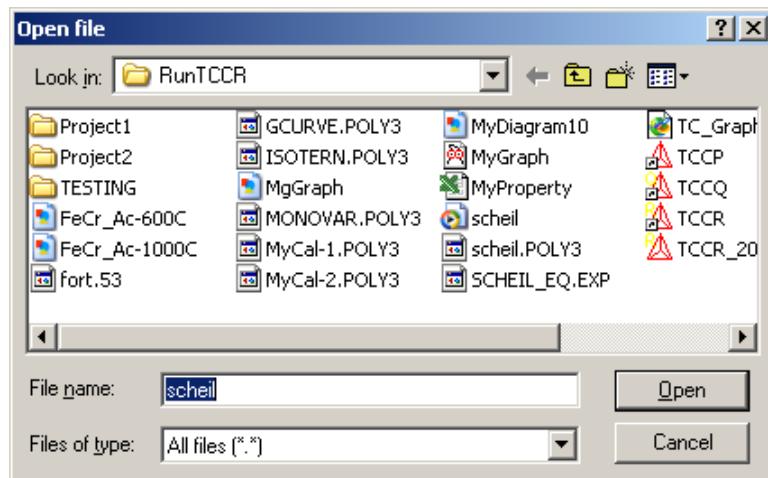


Figure 10-18. The “Open file” window: Reading the POLY3/GES5 workspaces from an *.POLY3 file.

Step 3: Define the alloy and its bulk compositions

The Main Option 1 requires the user to answer the following questions:

Major element of alloy: <major element name>

Specify the major element in the to-be-simulated alloy system (e.g., Fe for steels/Fe-alloy, Al for Al-based alloys, or Ni for Ni-based superalloys).

Composition input in mass (weight) percent? /Y/: <Y or N>

Decide on how to input the bulk compositions (of alloying elements) for the alloy system: Y (Yes) for in weight-percentage, while N (no) for in mole-percentage.

1st alloying element: <element name>

Give the 1st alloying element name (e.g., “Cr”), which may be followed by its composition on the same line (e.g., “Cr 5”).

Mass (weight) percent /1/: <weight-percentage of alloying element> or
Mole percent /1/: <mole-percentage of alloying element>

Specify the composition for the above-given alloying element (in either weight-percentage or mole-percentage, depending upon the previous answer on composition unit).

2nd alloying element: <element name or RETURN>

Next alloying element: <element name or RETURN>

Give the name for the 2nd, 3rd, 4th and so forth alloying element name, which may be followed by its composition. By simply entering “RETURN” key, the module will immediately end the definition of bulk composition for the alloy system. Actually, all the alloying elements (from the first to the last

one) and their corresponding compositions (same unit; see above) can be defined on the same line, e.g., “Cr 5 Ni 1 Mo 0.5 C 0.01 N 0.02”.

Temperature (C) /2000/: <a temperature point in the liquid field>

Specify a temperature condition that should be high enough to start the solidification simulation from the liquid single-phase region for the defined alloy system.

The Main Option 3 allows the user to further modify the alloy bulk compositions for all alloying elements and the starting temperature condition for the SCHEIL-module simulation, by answering:

```
Mass (weight) percent of EL1 /3/: <new composition of element EL1>
Mass (weight) percent of EL2 /1/: <new composition of element EL2>
....
```

```
Mole percent of EL1 /3/: <new composition of element EL1>
```

```
Mole percent of EL2 /1/: <new composition of element EL2>
```

```
....
```

Specify the new compositions for the previously-defined alloying elements (in either weight-percentage or mole-percentage, depending upon the previous decision on choosing inputting-composition unit saved in the opened POLY3 file (if it was a POLY3 file that contains combined data from two or more database, the composition for the major element must have not been defined as a condition and that should be dependent on the system-size condition definition).

Temperature (C) /1850/: <a new temperature point in the liquid field>

Specify a new temperature condition that should be high enough to start the solidification simulation from the liquid single-phase region for the modified alloy bulk compositions.

Step 4: Decide on how to include necessary phases (and possible miscibility gap if not defined)

The Main Option 1 requires the user to decide the details on how to include various necessary phases for the solidification simulations of the defined alloy system, by answering the following questions:

Reject phase(s) /NONE/: <phase name(s) or RETURN>

Some unnecessary phases in the currently-defined alloy system can be rejected from the simulations.

Restore phase(s) /NONE/: <phase name(s) or RETURN>

Some already-rejected phases (by the user within the “Reject phase” sub-option, or by default in the used database which are shown before the “Reject phase” sub-option) in the currently-defined alloy system can be restored into the simulations.

OK? /Y/: <Y or N>

By answering N (No), the module allows the user to re-consider the phase inclusion (through rejecting/restoring phases);

By answering Y (Yes), the module starts retrieving and getting thermodynamic data from the used database for the currently-defined alloy system (the program may then list out all the references that are documented in the used database for the currently-defined alloy system).

Should any phase have a miscibility gap check? /N/: <N or Y>

By answer N (No), the module will progress to the next step (Step 5);

By answering Y (Yes), the module allows the user to consider one or more solution phase(s) possibly having miscibility gap(s) (if such a possibility has not been considered in the default definitions of the used database), for which the program will further ask the user to specify the solution phase name(s) and their major constituents on each of their sublattice sites, as in the following prompts:

Phase with miscibility gap: <phase name>

Major constituent(s) for sublattice 1: /CR FE/: <major constituent(s)>

Major constituent(s) for sublattice 2: /VA/: <major constituent(s)>

Phase with miscibility gap: <phase name or RETURN>

Step 5: Decide on the SCHEIL simulation settings

Before proceeding in this step, the module will first calculate the real liquidus temperature for the defined alloy system (under the specified alloy bulk composition), and shows on screen a message similar to the following:

```
Calculated liquidus temperature is 1530.00 (C)
Please enter simulation conditions !
```

Both the Main Options 1 and 3 requires the user to decide the details on how to proceed the solidification simulations for the defined alloy system, by answering the following questions:

Temperature step (C) /1/:

Specify the *relative cooling rate* in terms of decreasing temperature step. The default value of 1°C is normally realistic for most simulations, but the user can use another better step he/she can think of, based on his/her knowledge of the kinetics of solidification process and of real experimental settings.

Default stop point? /Y/:

By answering Y (Yes), the module will use the default setting for terminating the SCHEIL-module simulation of the solidification process (that is when the mole fraction of the remaining liquid phase reaches roughly only 0.01) and start the real Scheil-Gulliver model simulation of the defined alloy system;

By answering N (No), the module allows the user to change the way the SCHEIL-module to end the simulations, for which the program will further ask the user the following question:

Fraction of liquid or Temperature? /F/: <F or T>

By answer F (Fraction of liquid), the user should decide on the *approximate* mole fraction of the remaining liquid phase in the considered system at which the Scheil-Gulliver model (traditional or modified) simulation will be terminated, in the following question:

Fraction of liquid /.01/: <a preferred liquid fraction to end>

Specify the user-desired *approximate* mole fraction of the remaining liquid phase (e.g., 0.5) for ending the Scheil-Gulliver model simulation.

By answering T (Temperature), the user should decide on the *approximate* ending-temperature in at which the Scheil-Gulliver model (traditional or modified) simulation will be terminated, in the following question:

End temperature (C): <a preferred temperature to end>

Specify the user-desired *approximate* temperature (e.g., 400; in Celsius) for ending the Scheil-Gulliver model simulation.

Step 6: Decide on the SCHEIL simulation methods

Since TCC version Q, the SCHEIL Module had been further modified and improved, in order to consider back diffusions of fast-diffusing components and possible BCC→FCC phase transformation phenomena during solidification processes, which is particularly important for solidification processes of various steels/Fe-alloys, Ni-based superalloy and other types of alloys (but maybe not that essential for Al-/Mg-based alloys). The questions need to be answered for deciding such simulation method details are:

Fast diffusing components: /NONE/:

Since TCCQ, the *Modified-SCHEIL Module* allows the user to specify the name(s) of fast-diffusing components (normally but not always necessarily the interstitial elements, such as C and N). For such specified components, back-diffusion phenomena (*i.e.*, their re-distributing within already-solidified solid phases through diffusion will be taken in account in the solidification simulation. Note that there are more than one fast-diffusing components their names should be given on the same line, such as “C N”.

In case that the default answer NONE is simply accepted by <RETURN> [meaning no back-diffusion element(s) are given under this prompt], the *Traditional-SCHEIL Module* will be applied in the current simulation, and the next question on possible BCC→FCC phase transformation will be skipped.

Allow BCC -> FCC ?: /N/ :

Since TCCQ, the *Modified-SCHEIL Module* allows the user to consider possible phase transformation of BCC→FCC in the already-solidified solids (which is particularly important for solidification processes of various steels/Fe-alloys).

Step 7: Perform the Scheil-Gulliver model simulation and then plot the default diagram

Before proceeding in this step in the Main Options 2 and 3, the module requires the user to specify the name of an POLY3 file into which the currently SCHEIL-module simulation results will be saved.

Under PC-Linux and UNIX environments, it prompts the following question:

Buffer-saving file name /scheil/: <a preferred POLY3 file name>

where the user should specify a desired POLY3 file that will be saved under the current working directory.

Under PC Windows Vista/XP/2000/NT4 environments, a *Save As* window will pop up on screen, 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 10-19*. Note that the file type should be always POLY3 and no other file type is legal here. By pressing the **Save** button, the program proceeds with saving the POLY3/GES5 workspaces into the specified POLY3 file. The user may also Cancel such an **Save As** window session, and thus no POLY3/GES5 workspaces will be saved. However, if there is already a POLY3 file with the same name under the directory (as indicated by the **Save in** box), a warming message will pop up on the screen, as shown in *Figure 10-20*. If clicking on the **No** button, the program returns to the **Save As** window, so that the user may choose another **Save in** path or a different **File name**. If clicking on the **Yes** button, the program may ask the question “Overwrite current file content /N/ : ” or “Proceed with SAVE? /N/ : ”, so that the user can decide if overwriting the current POLY3/GES5 workspaces on the existing POLY3 file. Please note that such a **Save As** window session can be avoided only when running a MACRO file in which a POLY3 file name has specifically given in the line following the previous step.

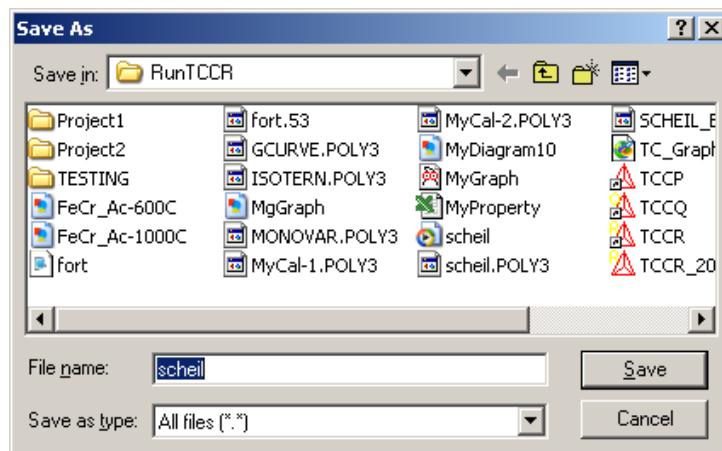


Figure 10-19. The “Save As” window: Saving the SCHEIL simulation results into a POLY3 file.

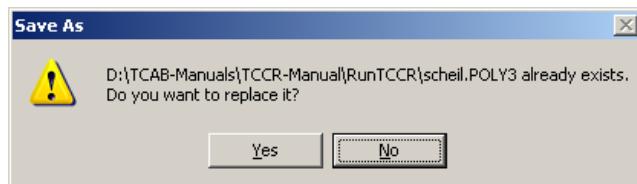


Figure 10-20. The warming message: If saving the POLY3 workspace on an existing POLY3 file.

Then the SCHEIL-module will first conduct a simple stepping calculation using the full-equilibrium approach (*Note this automatic-calculation part has been added since TCCR*), and then perform the advanced stepping calculation of the Scheil-Gulliver (traditional or modified) simulation of the solidification process of the defined alloy system. The stepping procedures of both the simple full-equilibrium calculation and advanced Scheil-Gulliver (traditional or modified) simulation will be gradually shown on screen, such as below:

Phase Region from 1797.15 LIQUID Calculated 4 equilibria Phase Region from 1796.55 LIQUID BCC_A2#1 Calculated 12 equilibria Phase Region from 1787.70 BCC_A2#1 *** Buffer saved on file: D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCH.POLY3 <i>Full-equilibrium calculation part:</i> for: Phase Region from 1797.15 LIQUID Calculated 4 equilibria Phase Region from 1796.55 LIQUID BCC_A2#1 Calculated 39 equilibria Phase Region from 1760.39 LIQUID BCC_A2#1 FCC_A1#1 Calculated 3 equilibria Phase Region from 1760.39 LIQUID FCC_A1#1 *** Buffer saved on file: D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCHT.POLY3 Calculated 21 equilibria *** Buffer saved on file: D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCHT.POLY3
 <i>Traditional-SCHEIL simulation part:</i> for: Phase Region from 1797.15 LIQUID Calculated 4 equilibria Phase Region from 1796.55 LIQUID BCC_A2#1 Calculated 39 equilibria Phase Region from 1760.39 LIQUID BCC_A2#1 FCC_A1#1 Calculated 3 equilibria Phase Region from 1760.39 LIQUID FCC_A1#1 *** Buffer saved on file: D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCHT.POLY3 Calculated 21 equilibria *** Buffer saved on file: D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCHT.POLY3
 <i>Modified-SCHEIL simulation part:</i> Solidification starts at 1797.15 K Phase Region from 1798.15 LIQUID Phase Region from 1796.53 LIQUID BCC_A2#1 1796.5250 0.9905 0.0095 0.0000 1796.4625 0.9690 0.0310 -267.6193 1796.4000 0.9481 0.0519 -526.5327 1779.8375 0.0100 0.9900 -12059.9772 1779.7750 0.0098 0.9902 -12062.0671 *** Buffer saved on file D:\TCAB-Manuals\TCCR-Manual\RunTCCR\Steell-SCHM.POLY3

Immediately after the completion of the whole simulation procedure within the Main Option 1and/or 3, or after opening a previously-generated SCHEIL-module POLY3 file within the Main Option 2 has been switched on, the default diagram (*i.e.*, the *Temperature (°C) versus Mole Fraction of Solid* plot) will be first plotted in the ***Thermo-Calc Graph window*** on screen, as shown in *Figure 10-21a* for the Traditional-SCHEIL Module Simulation (where the full-equilibrium solidification path is plotted in black solid curves, and the partial-equilibrium solidification path in colorized curve), and illustrated in *Figure 20-21b* for the Modified-SCHEIL Module Simulation (where the full-equilibrium solidification path is plotted in black dashed curves, and the partial-equilibrium solidification path in colorized curve):

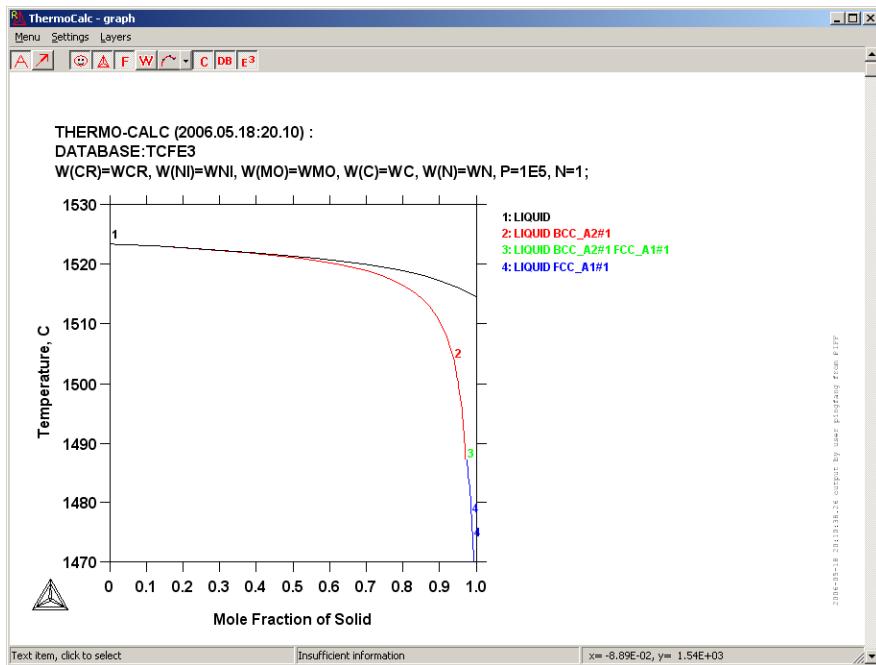


Figure 10-21a. The Default Diagram Plotted from the Traditional SCHEIL-Module Simulation.

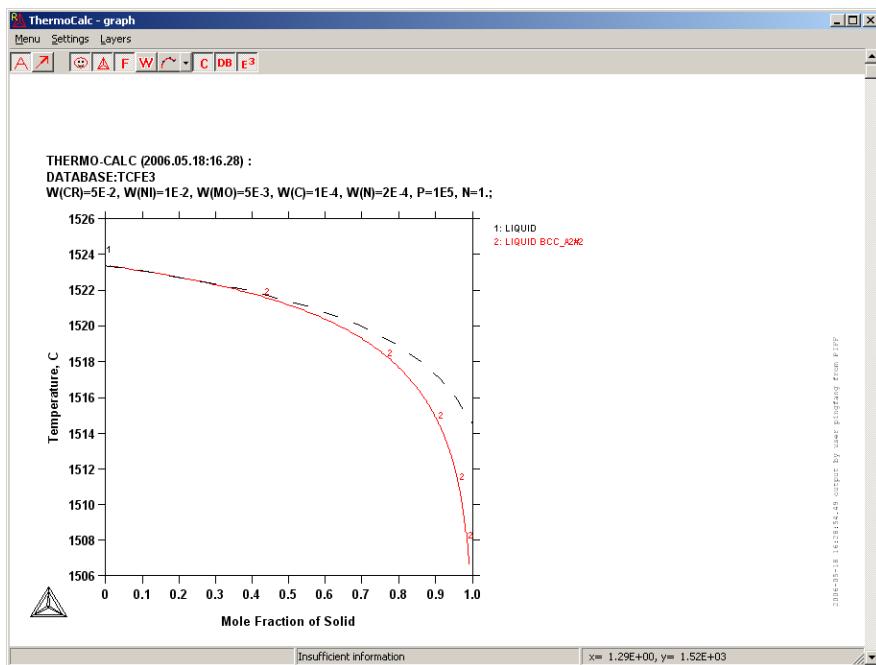


Figure 10-21b. The Default Diagram Plotted from the Modified SCHEIL-Module Simulation.

Step 8: Edit/Export/Save the plotted diagram

After the diagram has been plotted on screen from the SCHEIL Module simulation, the user can immediately edit/save/print the plotted graph on the *Thermo-Calc Graph window* (see more details in *Section 9.3.8* on the *Thermo-Calc Graph Window*).

The user can also choose (in the following sub-options; see below) to export the plotted graph as to hard copies (in various graphical formats; see details in *Section 9.3.6* on the *SET_PLOT_FORMAT* command), save the X-/Y-coordinates of the plotted graph onto simple textual file (*i.e.*, EXP experimental data file; see details in *Section 9.4.2* on the *MAKE_EXPERIMENTAL_DATAFILE* command), and decide to plot more diagrams from the same SCHEIL Module simulation.

Hard copy of the diagram? /N/: <N or Y>

By answering N (No), no hard copy (as any graphical file) of the currently-plotted diagram will be exported/saved;

By answering Y (Yes), a hard copy (as a graphical file) of the currently-plotted diagram will be exported/save, for which the program will further ask the user the following question:

Plot format /6/: <a graphical format such as 5>

Specify the desired graphical format. Afterwards, the graphical device change will be informed on screen:

CURRENT DEVICE: MS-Windows

NEW DEVICE: Postscript portrait mode

USE POSTSCRIPT FONTS /YES/: <Y or N>

By answering N (No), no PostScript font is needed to be specified;

By answering Y (Yes), a specific PostScript font should be further specifies, for which the user shall answer some further related questions:

SELECT FONTPHYSICAL /9/: <a PostScript font number, e.g. 27>

NEW FONT: Times-Bold

FONT SIZE /.3583299909/: <a desired font number, e.g..32>

TITLE : <a desired title such as Steel1-SCHT-a>

OUTPUT TO SCREEN OR FILE /SCREEN/: <a desired file name, e.g. Steel1-SCHT-a>

CURRENT DEVICE: Postscript portrait mode

NEW DEVICE: MS-Windows

USE HARD COPY FONTS /YES/: <Y or N>

By answering N (No), no hard copy font is needed to be specified;

By answering Y (Yes), a specific PostScript font should be further specifies, for which the user shall answer some further related questions:

SELECT FONTPHYSICAL /1/: <1 for Windows; 9 for Linux/UNIX>

NEW FONT: Arial Bold

FONT SIZE /.3409499914/: <the default size is normally OK>

Save coordinates of curve on text file? /N/: y

By answering N (No), no textual file (as any EXP file) of the currently-plotted diagram is saved;

By answering Y (Yes), a textual file (as a EXP file) of the currently-plotted diagram will be save, for which the program will further ask the user the following question:

File name /scheil/: < a desired file name, e.g. Steel1-SCHT-a>

Any more diagrams? /Y/:

By answering N (No), no other graph from the current simulation will be further plotted, and the current SCHEIL Module run will be ended and the TCC program will return to the point where the module was called (by the GO SCHEIL command-sequence);

By answering Y (Yes), the SCHEIL Module will allow the user to continue in plotting other graphs from the same simulation (which is the same as in the Main Option 2; see details in Step 8).

Step 9: Plot other diagram

By answering Y (Yes) to the question “Any more diagrams?” within any Main Option (1, 2 or 3), the SCHEIL module provide the chance to the user to define and plot many other diagrams from the same SCHEIL-module simulation (saved in the current workspace or in an opened previously-saved SCHEIL-module POLY3 file):

The following axis variables are available

T	--- Temperature in Celsius
NL/BL	--- Mole/mass fraction of liquid
NS/BS	--- Mole/mass fraction of all solid phases
NS (ph) /BS (ph)	--- Mole/mass fraction of a solid phase
W(ph,el)	--- Weight fraction of a element in a phase
X(ph,el)	--- Mole fraction of a element in a phase
NH/BH	--- Latent heat evolution per mole/gram

"el" and "ph" are name of element and phase, respectively
"*" can be used as a wild character for "el" and "ph"

Please note that since TCCS/TCW5, when a database (such as TCFE6/TCFE5/TCFE4) containing volume data (*i.e.*, molar volume, thermal expansivity, and compressibility) has been used for a multicomponent alloy systems, the further-revised/improved SCHEIL-Module can now plot diagrams using one of four additional properties varied during the solidification process as an axis variable, *i.e.*:

CP/BCP	--- Apparent heat capacity per mole/gram
NV/NV(ph)	--- Molar volume of the system or a phase
DS/DS(ph)	--- Average density of the system or a phase
BT	--- Apparent volumetric TEC of the system

X-axis Variable: <a property for the X-axis; *e.g.* T>

Y-axis Variable: <a property for the Y-axis; *e.g.* BL>

Immediately after these X/Y-axes settings, the desired diagram (*e.g.*, the *Temperature (°C) versus Mass Fraction of Liquid* plot) will first plotted in the **Thermo-Calc Graph window** on screen. Similar to the default diagram, the user can immediately edit/save/print the plotted graph on the **Thermo-Calc Graph window** (see more details in *Section 9.3.8*).

The user can also choose (in the following sub-options; see below) to zoom the plotted diagram, as well as to export the plotted graph as to hard copies (in various graphical formats; see details in *Section 9.3.6* on the **SET_PLOT_FORMAT** command), save the X/Y-coordinates of the plotted graph onto simple textual file (*i.e.*, EXP experimental data file; see details in *Section 9.4.2* on the **MAKE_EXPERIMENTAL_DATAFILE** command), and decide to plot more diagrams from the same SCHEIL Module simulation.

```

Zoom in? /N/: <N or Y>
Change scaling of X-axis? /Y/: <N or Y>
Minimum /0/: <e.g. 1505>
Maximum /1/: <e.g. 1525>
Change scaling of Y-axis? /Y/: <N or Y>
Minimum /0/: <e.g. 0>
Maximum /1/: <e.g. 1.0>
Zoom in? /N/: <N or Y>
Hard copy of the diagram? /N/: <N or Y>
Save coordinates of curve on text file? /N/: <N or Y>
Any more diagrams? /Y/: <N or Y>

```

In this way, many other diagrams from the same SCHEIL-module simulation (saved in the current workspace or in an opened previously-saved SCHEIL-module POLY3 file); some such non-default diagrams have already been illustrated in *Figure 10.16*.

10.8.5 Demonstrative examples of the SCHEIL-module simulations

There are two examples in the \TCEX\ area (TCEX15.TCM and TCEX30.TCM) and in **TCCR Examples Book** that demonstrate that how various SCHEIL-module simulations and graphical processing are performed:

- Example 15** TCEX15.TCM Demonstration of the SCHEIL-module simulation for the solidification process of the Cr-10Ni alloy (alloying element composition in wt.%).
- Example 30** TCEX30.TCM Demonstration of the SCHEIL-module simulation for the solidification process of the Al-4Mg-2Si-2Cu alloy (alloying element composition in wt.%).
- Example 48** TCEX48.TCM Demonstration of the SCHEIL-module simulation for the solidification process of the Fe-10Cr-1C alloy (alloying element composition in wt.%), allowing the interstitial component C having back diffusion in solid phases.

10.8.6 References on the Scheil-Gulliver Simulations

- Chen Q., and Sundman B. (1999) *TQ-I Thermodynamic Calculation Interface: Programmers' Guide and Examples*. Thermo-Calc AB.
- Jansson Å., and Sundman B. (1997) Simulation of solidification of light alloys using a thermodynamic database. In Jansson Å. *Phase Diagram Evaluation and Applications in Light Metal Alloys*, KTH Ph.D. Dissertation.
- Gulliver C.H. (1913) *J. Inst. Metals*, **9**, 120.
- Saunders N (1996) Phase diagram calculations for commercial Al-alloys. *5th Intl Conf. On Al-Alloys*, Grenoble, France.
- Sundman B., Jansson B., and Schalin M. (1993) Thermodynamic calculations made easy. *J. Phase Equilibria*, 573-562.
- Scheil E. (1942) *E. Metallkde*, **34**, 70.

10.9 REACTOR Module

10.9.1 Descriptions of the REACTOR module

Chemical kinetics for homogeneous and heterogeneous interactions is always a very important part of various materials processes. To model such processes coupled with equilibrium, chemical kinetics, and transport (diffusion, fluid flow and other mechanisms) is a complex task for some currently available software packages in materials R&D fields, for instance, PHOENIX, ASPEN, and FACSIMILE. However, it has been extremely difficult to use reliable chemical kinetic parameters in such software packages, especially when the kinetic rates are functions of temperature, pressure, composition, reactive surface area, and so on.

One well-known simplification of chemical kinetic problems is utilizing the multi-staged steady-state reaction assumption. Eriksson and Johansson (1978) developed a method for simulating such simplified process. This method was adopted in the TCC software many years ago, and was developed as a special module, namely the **REACTOR module**. This module enables dynamic simulations to be performed for multi-staged steady-state reactors with con- or counter-current materials flow and exchange of energy between different stages of the reactor, or relative to ambient conditions. Processes take place in such continuously working reactors (vertical or horizontal) into which raw materials and energy can be supplied at any level. Various chemical reactions may occur locally at rates that depend on temperature (and pressure), as well as phases formed in one volume segment flow for further reaction in other segments or form part of the products leaving the reactor. Since TCCQ, the REACTOR module has been upgraded. It can simulate not only steady-state processes but also dynamic ones that change with time, for example decarburization of liquid steel.

Some reactor concepts are further documents in *Section 10.9.4*. Such multi-staged steady-state reactions are characteristic for many metallurgic processes (*e.g.*, carbothermic silica reductions, rotary kilns, blast furnaces, electrothermal furnaces, reverberatory furnaces, gradual solidification, *etc.*).

To simulate such a dynamic process, the REACTOR module is conceptually divided into several sequential stages, between which gaseous and condensed phases flow in opposite or parallel directions. According to the direction, gaseous and condensed products leave a specific stage to react in neighboring stages, or leave the reactor at either the same end (concurrent flow) or opposite ends (counter-current flow). Thus, temperature and composition profiles inside the reactor can be predicted for various values of charge composition and energy (enthalpy or entropy) input. Optimal conditions for the process can be found, which ensure maximum product yield at minimum energy consumption; such information are particularly useful to develop special engineering techniques without having to resort costly trial-and-error experiments.

As mentioned in the *Section 10.3*, the REACTOR module has a command-line user interface that is similar to all basic modules (*e.g.*, TDB, GES, POLY, POST, PARROT and ED-EXP modules), rather than a question-line interface (as coded for other special modules, *e.g.*, BIN, TERN, POT, POURBAIX and SCHEIL).

The REACTOR module is internally connected with the TDB module for retrieving thermodynamic data of a defined system, the GES module for modeling and data manipulations of thermodynamic properties of each phase, and with POLY module for calculating local and partial equilibria in the multi-staged steady-state reactor and for listing simulation results in each stage box.

With this advance module, simulations of multi-staged steady-state reactors or other dynamical processes can be performed. Proven application examples is the silica arc formation in different parts in a furnace (*TCCS Example 24*) and decarburization of liquid steel (*TCCS Example 51*).

The following commands are available in the REACTOR module:

REACTOR:?		
AMEND_RECORD	EXECUTE_POLY3_COMMAND	LIST_RECORDS
BACK	EXIT	PATCH_WORKSPACE
CREATE_DIVIDERS	GOTO_MODULE	READ_WORKSPACE
CREATE_PIPES	HELP	SAVE_WORKSPACE
CREATE_PROBE	INFORMATION	SHOW_PROBE
CREATE_STAGE_BOXES	LIST_DESIGN	START_SIMULATION
REACTOR:		

10.9.2 Some concepts in multi-staged steady-state reactor simulations

Technically, the basic assumption that chemical reactions occurring in a reactor proceed to completion is applied in the module. Each stage is considered as a single one until a steady-state condition is reached, that is characterized by a total balance in material and energy (enthalpy or entropy) inputs and outputs. Two types of stages may possibly be classified in a reactor:

- ◆ *Energy-regulated stage:* Constant reaction temperature is determined by the chemical and thermal equilibria. At the reaction temperature, the total enthalpy or entropy change within the stage counter-balances the enthalpy or entropy supplied to the stage from outside the system (*e.g.*, electrical heating) or evolved from the stage as a result of heat losses.
- ◆ *Temperature-regulated stage:* When the process has a known temperature profile, the reaction temperature is fixed as a condition, and thus the total energy (enthalpy or entropy) of the reactions can be calculated as a local equilibrium.

All phase produced in a stage may move to the adjacent stages according to the type of a selected reactor (con- or counter-current). However, when there is incomplete reactions in the reactor, it is assumed that the gaseous phase and condensed intermediary reaction products leaving a certain stage may bypass the adjacent stages without being cooled or heated, or are distributed over several stages before reactions occurring. The amounts of reaction products affected are thus estimated for the purpose of the simulation, or can be based on actual practical experience. The amount converted to fraction values are called the “distribution coefficients”. For instance, 80% of the gas leaving a specific stage may react in the next stage, 15% may bypass one stage, and 5% may bypass two stages before reaction, and so on. Different sets of distribution coefficients may be used for the gaseous mixture phase and for each of the condensed phases (stoichiometric or solution). The top and bottom stages mean that the products formed in these particular stages may leave the reactor. Distribution coefficients in a reactor can be schematically illustrated in *Figure 10-18*.

The assumption of local chemical and thermal equilibrium in each stage combined with the given distribution coefficients provide the compositions and temperatures of the gaseous and condensed flows at each stage boundary. Such flows over stage boundary are not in chemical equilibrium, and the departure from the equilibrium state is proportional to the physical length of the stages. If there is only one stage that implies no flow distribution occurred, only the exit flow will have a composition and temperature corresponding to the chemical and thermal equilibrium inside the stage. If the number of stage in a reactor increases to infinity, chemical and thermal equilibrium will be attained in every point of the reactor. Therefore, the effects of chemical kinetics of reactions on mass and energy transfer processes can be simulated to some extent by the choices of stage numbers and distribution coefficients.

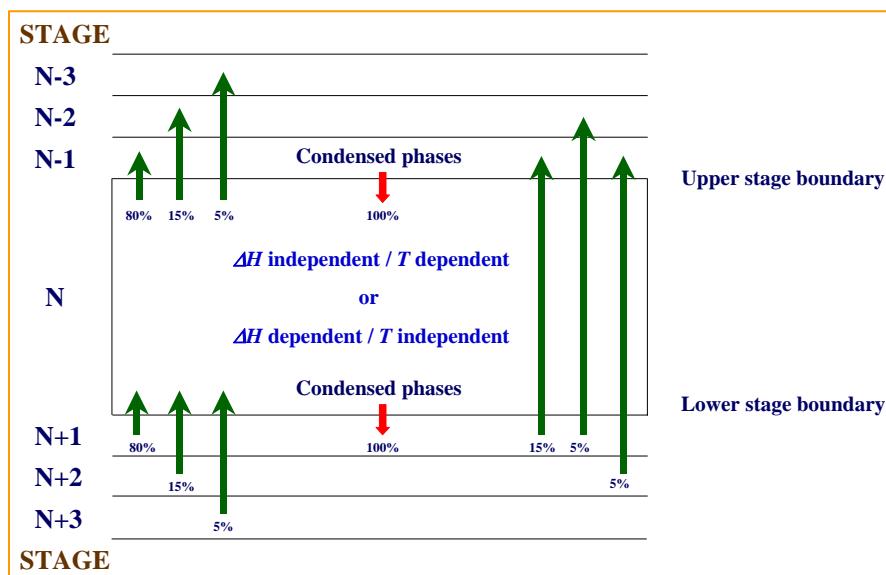


Figure 10-18. Schematic Illustration of Multi-Staged Steady-State Reactor Concepts

10.9.3 Commands in the REACTOR module

10.9.3.1 AMEND_RECORD

Description: This command is to amend some of the already-created record, but has not been implemented yet.

Synopsis: AMEND_RECORD

10.9.3.2 BACK

Description: This command switches control back to the most recent module. See also GOTO_MODULE.

Synopsis: BACK

10.9.3.3 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.

Synopsis: CREATE_DIVIDERS

Ensuing 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.

10.9.3.4 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).

Synopsis: CREATE_PIPES

Ensuing 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.

10.9.3.5 CREATE_PROBE

Description: This command is to create probe, but has not been implemented yet.

Synopsis: CREATE_PROBE

10.9.3.6 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.

Synopsis: CREATE_STAGE_BOXES

Ensuing 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 reactor 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:

First, as input amount of heat (enthalpy),

H=Value

Second, 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 ensuing 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:

First, as initial amount of heat (enthalpy),

H=Value

Second, 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 ensuing 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.

10.9.3.7 EXECUTE_POLY3_COMMAND

Description: This command provides the possibility of directly utilizing all kinds of POLY commands inside the REACTOR module.

Synopsis: EXECUTE_POLY3_COMMAND <a valid POLY command>
Any POLY command can be used here (see Part 8).

10.9.3.8 EXIT

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

Synopsis: EXIT

10.9.3.9 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 (also see Section 5.4.11).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing Prompt: MODULE NAME:
NO SUCH MODULE, USE ANY OF THESE:
SYSTEM UTILITIES
GIBBS ENERGY SYSTEM
TABULATION REACTION
POLY_3
BINARY DIAGRAM EASY
DATABASE RETRIEVAL
FUNC OPT PLOT
REACTOR SIMULATOR_3
PARROT
POTENTIAL DIAGRAM
SCHEIL SIMULATION
POURBAIX DIAGRAM
TERNARY DIAGRAM
MODULE NAME: <module name>

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

10.9.3.10 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing 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.

10.9.3.11 INFORMATION

Description: Basic information about various REACTOR (and POLY) subjects can be obtained with this command for a number of subjects, as they are described in different parts of this chapter.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: <subject name>

The name of the subject (or its abbreviation as long as it is unique, *e.g.*, PLOY, PRIV, PURP, MACR, MASS, MULT, *etc.*) must be given. Extensive information is available for various subjects (that are important to the use of the REACTOR Module) as listed below (this list can be seen if typing a question mark "?"):

PURPOSE (Introducing the REACTOR Module)	
MULTI-STAGED STEADY-STATE REACTOR	
STAGE TYPES	DISTRIBUTION COEFFICIENTS
LOCAL EQUILIBRIA	MASS AND ENERGY TRANSFER
USER INTERFACE	MACRO FACILITY
POLY-COMMAND EXECUTION	PRIVATE FILES
DEMONSTRATIVE EXAMPLES	REFERENCES
HELP (How to get on-line help in the REACTOR Module)	

10.9.3.12 LIST_DESIGN

Description: This command is to list the design details of the creator, but has not been implemented yet.

Synopsis: LIST_DESIGN

10.9.3.13 LIST_RECORDS

Description: This command lists on the screen all the already-created records for the reactor (including stage boxes, dividers, pipes, *etc.*)

Synopsis: LIST_RECORDS

10.9.3.14 PATCH_WORKSPACE

Description: This command is only for those who think they know what they are doing!

Synopsis: PATCH_WORKSPACE

10.9.3.15 READ_WORKSPACE

Description: The REACTOR workspace (along with the POLY3 and GES5 workspaces) and the simulated results from the SIMULATE command can be READ from a file where they must have been saved previously with a SAVE_WORKSPACE command. This *.RCT file is not printable.

Synopsis 1: READ_WORKSPACE <file name>

Options: file name -- The name of a previously saved RCT-file where the REACTOR and POLY3/GES5 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.

Synopsis 2: READ_WORKSPACE

Ensuing Prompt: File name /ABCDEF/: <file name>

The program shows up the "ABCDEF" as the most lately specified RCT-file name or as "RESULT" if there is no workspace already saved in the run. By pressing <RETURN> or typing a specific file name, one can read the REACTOR (and POLY3/GES5) workspaces from the file.

Notes: Under Windows Vista/XP/2000/NT4 environments, a *Open file* 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, as shown in *Figure 10-19*. The file type (*i.e.*, RCT, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with opening the REACTOR (and POLY3/GES5) workspaces from the specified, previously-saved *.RCT file. The user may also cancel such an **Open file** window session, and thus no previously-saved REACTOR (and POLY3/GES5) workspaces will be opened.

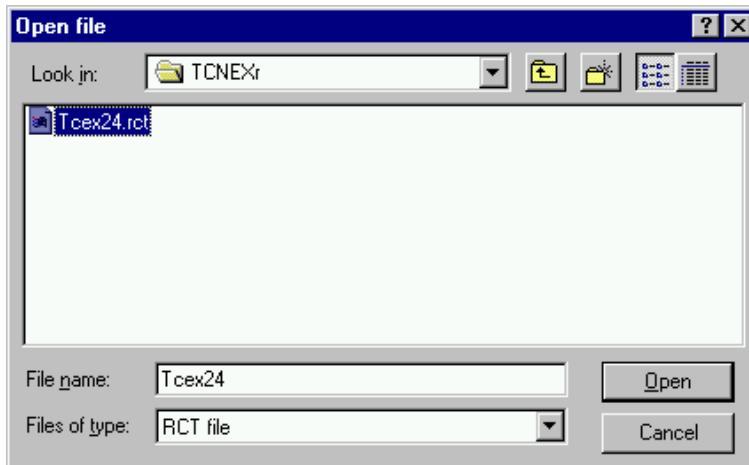


Figure 10-19. The “Open file” window: Reading the REACTOR workspace from an *.RCT file.

10.9.3.16 SAVE_WORKSPACE

Description: The REACTOR (and POLY3/GES5) workspaces are saved on a file with this command. In the GES5 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 reactor and its distribution coefficients are saved, together with the GES5 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/GES5) 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. Do not SAVE after a START_SIMULATE command!

Synopsis 1: `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.

Synopsis 2: `SAVE_WORKSPACE`

Ensuing Prompt: `File name /RESULT/: <file name>`

By pressing <RETURN>, one can save the REACTOR (and POLY3/GES5) workspaces onto a file under the default name RESULT and with a default extension of “.RCT”.

Or a user can specify a desired `file name`, for which the default extension is “.RCT”, while the user can also have any other extension as wished.

`Overwrite current file content /N/: <Y or N>`

(under Windows Vista/XP/2000/NT4 environments)

or

`Proceed with save /N/: <Y or N>`

(under PC Linux and various UNIX platforms)

This question is asked only if there is already a file with the same name, and the following message appears on screen:

This file contains results from a previous START_SIMULATE command.
The SAVE command will save the current status of the program but destroy
the results from the previous START_SIMULATE commands.

If you answer Y, the previous content will be overwritten. Note that results from the START_SIMULATE command are destroyed by SAVE. 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.

If you answer N, nothing will be saved, and the previous content will not be overwritten. You may later use the SAVE command with an unspecified name to save the REACTOR (and POLY3/GES5) workspaces.

Notes:

Under Windows Vista/XP/2000/NT4 environments, a **Save As** 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 **Save as type** box) can not be changed. By pressing the **Save** button, the program proceeds with saving the REACTOR (and POLY3/GES5) workspaces in the specified *.RCT file. The user may also cancel such a **Save As** window session, and thus the current REACTOR (and POLY3/GES5) workspaces will not be saved.

However, if there is already a file with the same name under the directory (as indicated by the **Save in** box), a warming message will pop up on the screen, as shown in *Figure 10-21*. If clicking on the **No** button, the program returns to the **Save As** window, so that the user may choose another **Save in** path or a different **File name**. If clicking on the **Yes** button, the program may ask the question “Overwrite current file content /N/ :”, so that the user can decide if overwriting the current REACTOR (and POLY3/GES5) workspaces on the existing *.RCT file (see above).



Figure 10-20. The “Save As” window: Saving the REACTOR workspace as an *.RCT file.

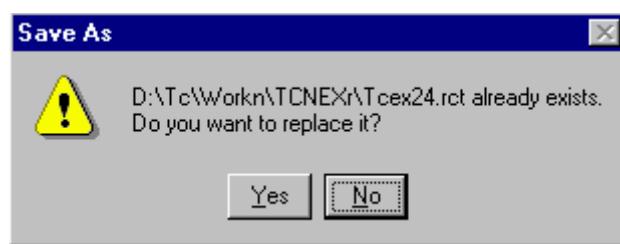


Figure 10-21. The warming message: If saving the REACTOR workspace on an existing *.RCT file.

10.9.3.17 SHOW_PROBE

Description: This command is to show the details of the created probe, but has not been implemented yet.

Synopsis: SHOW_PROBE

10.9.3.18 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.

Synopsis: START_SIMULATION

Ensuing 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

```

10.9.4 Demonstrative examples of the REACTOR-module simulations

There is one example in the \TCEX\ area (TCEX24.TCM) and in **TCCS Examples Book** that demonstrate that how the REACTOR-module simulation and graphical processing are performed:

Example 24 TCEX24.TCM Demonstration of the REACTOR-module simulation for the multi-staged steady-state reactions occurred in silicon arc furnace with output of gaseous phase at the top and output of condensed phases at the bottom. The gaseous mixture phase from one segment will flow to higher segments, 80 % will react in the first above, 15% in the second above and 5 % in the third above. The condensed phases will flow downwards and all of it will go to the next lowest segment.

Another example would be added in the \TCEX\ area (TCEX51.TCM) and in **TCCS Examples Book** that demonstrate that how the REACTOR-module simulates the dynamic process of decarburization of liquid steel.

Example 51 TCEX51.TCM Demonstration of the REACTOR-module simulation for a dynamic process of decarburization of liquid steel.

10.9.5 References on the REACTOR-module simulations

Eriksson G., and Johansson T. (1978) *Scandinavia Journal of Metals*, **7**, 264-285.
Johansson T., and Eriksson G. (1984) *J. Electrochem. Soc.*, **131**, 365-702.

(This page is intended to be empty)

11 Gibbs Energy System Module (GES)

11.1 Introduction

The **Gibbs Energy System (GES)** [⇒] is an implementation of the standard software interface for thermochemical calculations defined by the Scientific Group Thermodata Europe (SGTE). Being a basic module of the Thermo-Calc software (**TCC** and **TCW**), GES consists of a comprehensive subroutine package for various thermochemical calculations. The module is interactively connected with all basic and advanced modules in Thermo-Calc and DICTRA. Most users will never use the GES module unless they perform assessments.

The objective of the GES module is to provide a unified set of subroutines to be used in any application program that needs thermochemical data. All kinds of thermodynamic models for various types of substances are implemented in the GES module. However, most model-dependent features are hidden inside the module, and the application programmer can use a standardized set of subroutines, *i.e.*, as packaged into the Thermo-Calc programming-interfaces (namely **TQ**, **TCAPI** and **TC-MATLAB Toolbox**), to calculate the integral Gibbs energy or any partial derivative thereof for each phase at any composition, temperature and pressure.

There are subroutines to analytically calculate the first and second partial derivatives of integral Gibbs energy with respect to any variables, which is a unique feature of the Thermo-Calc software and its application programming interfaces.

The GES module also provides the users of application programs with a user interface with a set of simple and general commands for data manipulations. Through such a user interface, one can interactively enter and modify phase descriptions, model connections, basic thermodynamic parameters, and so on. The aim of this User's Guide is to describe this user interface.

POLY and all other basic and special modules, as well as the DICTRA software, interactively call the subroutines within the GES module to calculate any thermodynamic quantities.

Its current edition is the GES5, which is always referred in the Thermo-Calc software for storing all kinds of thermodynamic information and for performing various Gibbs energy calculations in the so-called GES5 workspaces.

The following commands are available in the GES module:

GES : ?		
ADD_COMMENT	ENTER_PHASE	LIST_PHASE_DATA
AMEND_ELEMENT_DATA	ENTER_SPECIES	LIST_STATUS
AMEND_PARAMETER	ENTER_SYMBOL	LIST_SYMBOLS
AMEND_PHASE_DESCRIPTION	EXIT	PATCH_WORKSPACE
AMEND_SYMBOL	GOTO_MODULE	READ_GES_WORKSPACE
BACK	HELP	REINITIATE
CHANGE_STATUS	INFORMATION	SAVE_GES_WORKSPACE
DELETE	LIST_CONSTITUITION	SET_INTERACTIVE
ENTER_ELEMENT	LIST_DATA	SET_R_AND_P_NORM
ENTER_PARAMETER	LIST_PARAMETER	
GES :		

Note that the `ADD_COMMENT` command is new since the TCCQ version.

⇒ Revision History of the GES Module User's Guide:

Feb 1984	First release (Edited by Bo Sundman)
Oct 1993	Second revised release (Edited by Bo Sundman)
Mar 1996	Third revised release (Edited by Bo Sundman)
Jun 2000	Fourth revised and extended release (Edited by Pingfang Shi)
Nov 2002	Fifth revised and extended release (Edited by Pingfang Shi)
June 2004	Sixth revised and extended release (Edited by Pingfang Shi)
June 2006	Seventh revised and extended release (Edited by Pingfang Shi)
Mar 2008	Eighth revised and extended release (Edited by Pingfang Shi)

11.2 Thermochemistry

The Gibbs energy of a thermochemical system is a function of the composition, temperature and pressure. The stable state of a system for constant values of these variables is represented by a minimum in the Gibbs energy. GES does not provide any routines to find the stable state but a method to do so with the use of GES is described in literature.

11.2.1 Definitions of some terms

In order to describe the construction of GES a few terms must be defined:

- **Elements** are those of the periodic chart and in addition vacancies (denoted Va) and electrons (denoted $/-$ or ZE) are used. Vacancies are used, *e.g.*, to describe a deviation from stoichiometry in a phase with sublattices. A user may also enter fictitious elements.
- **Species** are used to describe a molecular like aggregate which can be charged. All elements are also species.
- **Constituents** are the species which constitute a phase. The constituents of a phase are a subset of the species.
- **Phases** are a homogeneous part of the system with a uniform composition and structure. In GES all thermochemical data are referred to the phases and their constituents.
- **Composition** is the amount of each element in the system.
- **Constitution** of a phase is the amount of each constituent of the phase.
- **Sublattices** are used to describe a deviation from Random Mixing in a phase, *e.g.*, when interstitials or constituents with different sizes or electronegativity prefer different lattice sites in a phase.
- **Constituent array** is an array of constituents, one for each of sublattice(s) of the phase.
- **Site-fraction** is the fraction of a constituent in a specific sublattice. The sum of site-fractions in each sublattice is equal to unity.

11.2.2 Elements and species

The **elements** for a system in the GES module are stored in alphabetical order in a list. The alphabetical order has two exceptions. The electron is always the first and vacancy is the second element. In the element list each element is assigned a sequential number and the numbering starts from -1 for the electron, 0 for vacancy and then from 1 and upwards for the other elements. The element is identified by its name (maximum 2 characters), which can be one or two letters (the only exception is the electron which is denoted $/-$ or ZE). The only chemical and physical data associated with an element are its atomic mass, the phase name of a *Selected Element Reference State* (SER), the enthalpy difference for the element in the SER state at 298.15 K and 0 K, and the absolute entropy value for the element in the SER state at 298.15 K.

The **species** list is mainly for internal use in the program, but adds a flexibility to the system because a species has both a stoichiometric formula and a name which can be different from the formula. As constituents of a phase a species is referred to by its name (maximum 24 characters) and this makes it easy to distinguish species with the same stoichiometry but different thermochemical properties, *e.g.*, C2H2C12_cis and C2H2C12_trans with the same stoichiometry C2H2C12. The name of a species can include letters, numbers, parenthesis and the character (underscore). It must start with a letter. Note that the species in the species list have no reference to any phase and it is only as constituents of a phase that there are any thermochemical properties associated with the species.

11.2.3 Upper and lower case mode

The GES module can be used in **UPPER** or **lower** case mode. The user may select the mode by the GES command REINITIATE.

In the lower mode, it is mandatory to give the first letter of an element name in upper case and the second (if any) in lower case. In the UPPER case mode, lower and upper case letters are treated as identical. The stoichiometry of a species can in some cases be written more simple in the lower case mode, *e.g.*, CO can be

used for carbon monoxide whereas in upper case mode the same species must be written C1O because CO would mean cobalt.

11.2.4 Phases

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 GES 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*** (maximum 24 characters);
- 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.

A ***phase-type*** is specified by the so-called “***type code***”, when the phase is not an ordinary solution phases. The following type codes can be defined:

- G for a gaseous mixture phase;
- 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 Two-Sublattice 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. In a defined system, there may only exist one gaseous mixture phase.

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 phases, which always require that the solution phase must have 4 sublattices for substitutional ordering and can additionally have an interstitial sublattice.

In the descriptions of the ***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 next paragraph. Identical parameters (in terms of parameter-names) are stored only once in the GES workspaces.

The ***composition-dependence*** of the Gibbs energy is described in the GES 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.

11.2.5 Functions of temperature and pressure

The temperature-/pressure-dependence of parameters in the GES5 workspaces is described as a FORTRAN-like expression called a **TP-Function**. The rules for entering such a function are rather strict.

The basic entity of a TP-Function is called a “**simple-term**”. A simple-term has the *order of the factors*:

```
<a real number>*<a symbol name>**<exponent>*T**<exponent>*P**<exponent>
```

The text within <> describes the *factor* which is expected. The other quantities must be given literally and have their usual meaning, i.e., the sign * is for multiplication and ** for exponentiation. The exponent can only be an integer. Negative exponents must be surrounded by parenthesis. Redundant parts of a simple-term can be omitted. Examples of simple-terms are:

```
+14000
+1.15*T
+V1#
+1E-12*P**2
-456754.65*T**(-1)*P
+R*T
+10*V2#*T
```

In order to include the logarithm (always in natural logarithm, equally expressed by either LN or LOG) and exponential (expressed by EXP) in a function, it is allowed to multiply a single simple-term with the logarithm or exponential of another single simple-term. Instead of a logarithm or exponential, it is also possible to have just another symbol. This more generalized entity is called a “**term**”, and examples of such terms are:

```
+1.15*T*LN(T)
+1E-6*LN(-32000*T**(-1))
-5*V3#*EXP(V4#*P)
+0.078*V1#**(-1)*T**2*LN(V5#)
+0.078*V1#**(-1)*V2#*T**2
+R*T*LN(1E-5*P)
+RT#*LN(1E-5*P)
```

Simple-terms and terms are then written after each other (separated by the plus sign “+” or minus sign “-“) in order to form an **expression of function**. An expression can be continued on more than one line. Normally, a function is terminated with a semicolon “;” or an empty line; and a function can have several temperature ranges and a different expression (also ended with the semicolon sign “;”) for each region. Examples of functions are:

```
+10000-10*T+1.15*T*LN(T)+1E-6*P+134567*T**(-1)-1E-12*T*P**2;
+F1#+2.4*R*T*LN(T)+R*T*LN(P);
+F2#+3.5*RT#*LNT#+RTLNP#;
+F3#+4.6*RT#*LN(V5#*T);
+F4#+5.7*RT#*LN(2*V6#*T**(-1)*P);
```

Note that the symbols (such as V1, V2,, F1, F2,, LNT, etc. in the aforementioned function terms) can denote a numeric value or another TP-Function. The symbols R [that stands for the *Molar Gas Constant*, being equal to 8.31441 J/mol/K (in SIT Unit)] and RTLNP [that is the *volume contribution to the standard Gibbs energy of an ideal gaseous species*] are pre-defined functions by default in the GES5 workspaces, and thus they can be directly used in any function term. However, any other symbol must have already been defined (either as it exists in the used database being retrieved through the TDB module, or before it is referred as a part of an additionally/directly entered TP-Function in the GES module); otherwise, the TDB module will issue an error message (complaining the missing symbols when retrieving data from a database), or the GES module will assign such a symbol with a value of zero when it is referred as a part as an entered TP-Function. A

symbol is often suffixed with the # sign (in a database file or in GES workspace); however, since TCCN/TCW1 such a suffix can be optionally ignored.

Some of the restrictions on a function are:

- The *order of the factors* in a simple-term must be followed;
- No parenthesis is allowed (except for around an exponential, logarithm or negative exponent);
- No space is allowed between a sign and a numeric value;
- No division is allowed (but negative powers is allowed);
- Only integers can be used as exponents;
- More than one already-entered symbol can be used in a normal simple-term, but if a simple-term is a part of a logarithm or exponential term then only one symbol should be used in a simple-term;
- Unlike functions used in the POLY3 workspaces [where many different types of unary-functions, such as LOG for natural logarithm, LOG10 for 10-base logarithm, EXP for exponential function, SIN for sine function, COS for cosine function, and ERF for error function, can be entered], only natural logarithm (expressed in either LN or LOG) and exponential (expressed in EXP) are allowed in the GES5 workspaces.

Some of these restrictions are due to simplifications in the parsing of the functions and some are due to the requirement that it must be possible to calculate quickly the value of the function and also first and second derivatives with respect to T and P.

Due to such restrictions, terms such as $T^{**0.5}$ or $SQRT(T)$, or $T^{**(-0.5)}$, or $T^{**(1.5)}$ are illegal functions for GES5 workspaces (and of course within TDB files); for the purposes of defining mathematic terms of *e.g.* $T^{0.5}$, $T^{-0.5}$ and $T^{1.5}$, one should first enter a function named as LNT with the value of $LN(T)$ and then enter functions as $EXP(0.5*LNT)$, $EXP(-0.5*LNT)$ and $EXP(1.5*LNT)$. For a term in 10-base logarithm *e.g.* $\log_{10}T$, it must be entered as either $2.302585093*LN(T)$ or $2.302585093*LNT$.

It is possible to have pressure ranges instead of temperature ranges. This is indicated by giving the lower temperature limit as -100000 (meaning the lower pressure limit as 1E5 Pascal, *i.e.* 1 bar). The breakpoints will then be interpreted as pressures instead. It is essential that the functions are smooth, both first and second derivatives continuous, around a break point.

For more information on how such functions and parameters are stored in a database, please also see *Sections 3.3.14 (Function) and 3.3.15 (Parameter)* in the *Chapter 3 (Database Module (TDB) – Manager’s Guide)* in the document *Thermo-Calc Database Guide*.

The GES command LIST_SYMBOL can list all and/or any specified symbols (numeric constants, numeric variables, functions or tables) that are already defined in the current GES5 workspace.

Addition to the predefined functions retrieved from database(s), one can further use the GES command ENTER_SYMBOL command to define other necessary functions. Furthermore, any of the already-defined symbols can be modified by using the GES command AMEND_SYMBOL.

11.2.6 Symbols

Symbols can be used for numeric values when entering a TP-Function. As the thermodynamic parameters are TP-Functions, this makes it possible to easily use these already-defined symbols when further defining parameter functions as well.

All symbols are stored in a symbol list in the GES5 workspace. Actually, a symbol in this list can represent a numeric constant, a numeric variable, a function or a table:

- A *numeric constant* may not have its value changed; by default, the gas constant in SI units is stored with the symbol R.
- A *numeric variable* can have its value amended anytime.
- A *function* in the symbol list can actually be several TP-Functions because it is possible to define lower and upper temperature limits for each TP-Function in its specific temperature range. These breakpoints in temperature are constants (*i.e.*, they cannot vary with the pressure).

- A *table* can be used when it is difficult to express how a quantity varies with temperature in any other way. In the table the value of the quantity is given for constant temperature increments. A linear interpolation is used between the increments.

However, it is also possible to have pressure ranges instead of temperature ranges. This is indicated by giving the lower temperature limit as *e.g.* -10000 (meaning the lower pressure limit as 1E4 Pascal, *i.e.* 0.1 bar). The breakpoints will then be interpreted as pressures instead. It is essential that the functions are smooth, both first and second derivatives continuous, around a break point.

An already-defined symbol can be directly used as a part of the definition of another symbol, of a specific TP-Function or parameter. The symbols R [that stands for the *Molar Gas Constant*, being equal to 8.31441 J/mol/K (in SIT Unit)] and RTLN_P [that is the *volume contribution to the standard Gibbs energy of an ideal gaseous species*] are pre-defined functions by default in the GES5 workspaces, and thus they can be directly used in any function term. However, any other symbol must have already been defined (either as it exists in the used database being retrieved through the TDB module, or before it is referred as a part of an additionally/directly entered TP-Function in the GES module); otherwise, the TDB module will issue an error message (complaining the missing symbols when retrieving data from a database), or the GES module will assign such a symbol with a value of zero when it is referred as a part as an entered TP-Function. A symbol is often suffixed with the # sign (in a database file or in GES5 workspace); however, since TCCN/TCW1 such a suffix can be optionally ignored.

The symbol feature makes it possible to create very complicated expressions with the rather simple means provided by the TP-Function syntax. As a symbol can be a single numeric value, this makes it also possible to manipulate simultaneously individual coefficients of all TP-Functions, which refer to the symbol by modifying the value of the symbol. In the PARROT module (for assessment of thermochemical data), this feature is used to adjust the description of the phases in order to obtain the best fit to experimental data. All symbols that can be used for entering TP-Functions are stored in a symbol list in the GES5 workspaces.

The GES command LIST_SYMBOL can list all and/or any specified symbols (numeric constants, numeric variables, functions or tables) that are already defined in the current GES5 workspace.

Addition to the predefined symbols retrieved from database(s), one can use the GES command ENTER_SYMBOL to define other necessary symbols. Furthermore, any of the already-defined symbols can be modified by using the GES command AMEND_SYMBOL.

11.2.7 Miscibility gaps

In an equilibrium state, a solution phase may exist with two or more different compositions simultaneously. This is usually called a *miscibility gap*, but of course it can also be due to an ordering reaction. The same set of thermodynamic parameters shall describe the two (or more) phases, but the GES module must be informed in advance so that there can be two or more composition sets (which are defined either in the GES module or in the POLY module). See the AMEND_PHASE_DESCRIPTION command for more information; also see the ADVANCED_OPTION command in the POLY module.

Since TCCR/TCW4, the implemented so-called *Global Minimization Technique*, used in the POLY module for various stable/meta-stable equilibrium calculations (of either single-points, or of stepped property diagrams, or of mapped phase diagrams), can automatically detect and add any possibly-needed additional composition sets for complex solution phases with miscibility gaps. For more details, please refer to *Chapter 8 - Equilibrium Calculation Module (POLY)*.

11.3 Thermodynamic Models

The integral Gibbs energy for each phase depends on its constitution, temperature and pressure, and this is described by a thermodynamic model and expressed in a mathematical equation in the GES module. The dependence on the constitution is usually the most complicated to find a good model for. However, the dependence on temperature and pressure can be also very complicated for some phases (such as aqueous solutions).

The GES module is able to use a large number of thermodynamic models for various substances in different states (as listed in *Table 11-1*). More details are given below.

Table 11-1. List of a wide spectrum of thermodynamic models implemented in the GES module

Model Name in GES	Full Model Name	Applicable Phases
Ideal	Ideal Substitutional Model	Ideal gases
Regular	Regular Solution Model	Binary alloys
Redlich-Kister	Redlich-Kister (<i>Subregular</i>) Model	Binary alloys
Polynom	Simple Polynomial Model	Binary phase with no sublattice
Legendre	Legendre Polynomial Model	Binary phase with no sublattice
Redlich-Kister_Muggiana	Muggiana Ternary Excess Model (based on R-K binary parameters)	Ternary or higher order alloys
Redlich-Kister_Kohler	Kohler Ternary Excess Model (based on R-K binary parameters)	Ternary or higher order alloys
Muggiana_Restor	Muggiana Ternary Extrapolation Model	Ternary or higher order alloys
Kohler-All	Kohler Ternary Extrapolation Model	Ternary or higher order alloys
Toop-Kohler	Toop-Kohler Ternary Extrapolation Model	Ternary or higher order alloys
CEF	Compound-Energy-Formulism (Sublattice Model)	Alloys, Liquids, Gases, Oxides, ...
ITSLM	Ionic Two-Sublattice Liquid Model	Liquid slag
Quasichem_Ionic	Quasichemical Model of ionic liquid	Liquid slag
Quasichem_FACT00	F*A*C*T Quasichemical Model for liquid	Liquid slag
Quasichem_IRSID	IRSID (Kapoor-Frohberg-Gaye Cell) Model for liquid	Liquid slag
AM	Associate Model for liquid	Liquid slag
CAM	Central Atom Model (<i>not fully implemented yet</i>)	Liquid slag
SEHM	Sigworth-Elliott_Hillert Model for dilute liquid	Dilute Fe-rich liquid
G2SLM	Generalized Two-State Liquid Model	Amorphous phases
MO	Inden Magnetic Ordering Model	Alloys
CVM	Chemical Ordering via CVM Approach	Alloys
Murnaghan	Murnaghan Model	High-PT minerals/alloys
Birch-Murnaghan	Birch-Murnaghan Model	Extra-high-PT minerals
GPVT	Generalized PVT Model	Alloys, Liquids, Minerals, etc.
SUPERFLUID	SUPERFLUID Model	Real gases & gaseous mixture
DHLL	Debye-Hückel Limiting Law	Dilute aqueous solutions
SIT	Specific Ionic Interaction Theory	Dilute aqueous solutions
PITZ	Generalized Pitzer's Formalism	Concentrated aqueous solutions
HKF	Revised Helgeson-Kirkham-Flowers Model	Concentrated aqueous solutions
FHP	Flory-Huggins Polymer Model	Polymers

An advanced user can select a specific model for each included phase in a defined system. For a normal user, the database will select the appropriate model for each phase. In any model, the integral Gibbs energy consists of a number of parameters which depend on T and P and are multiplied with fractions of the constituents.

11.3.1 Standardized Gibbs energy

The simplest type of phase has only a single element as constituent, and the only parameter is the “standardized” Gibbs energy of the phase. The standardized Gibbs energy is the difference between

- the Gibbs energy for the phase at current temperature and pressure, and
- the enthalpy at 298 K and 1 atm in the SER state of the element.

For phases with more than one constituent, it is necessary to give the standardized Gibbs energy for each constituent (or each constituent array if the phase has sublattices). The constituents can also be species which are combinations of several elements, but the standardized Gibbs energy will always refer to the SER state of the element anyway.

The reason to use standardized Gibbs energies, rather than the “Lattice Stabilities” that are frequently used in phase diagram calculations for alloys, is that it makes it possible to calculate heat capacities and absolute enthalpy differences.

11.3.2 Ideal Substitutional Model

For a mixture phase with variable constitution, the simplest model is the ***Ideal Substitutional Model***. In this model, the constituents (being either neutral or charged cation/anion species) are treated in a way that is identical to that for the elements, and the parameters are the standardized Gibbs energies of the constituents in the mixture phase. These parameters are then multiplied with the fraction of the constituents, and an ideal entropy of mixing is added in order to calculate the integral Gibbs energy.

It has been applied to model gaseous mixtures as the so-called GAS phase in *e.g.* SSUB, SSOL, TCFE, TCNF, ION, SLAG, TCMP, TCES and other databases.

11.3.3 Regular/Subregular Solution Model

To realistically describe the Gibbs energy of a solution phase in a defined system, the ideal substitutional model is usually not sufficient, and there are many possibilities open for the choice of a reliable solution model.

In the ***Regular Solution Model***, a so-called “binary interaction parameter” for each pair of constituents of the phase is introduced. This parameter can also be given some physical significance by comparing with the difference in bond energies between *like* and *unlike* atoms. The binary interaction parameter is multiplied with the fractions of the two constituents, and thus it gives no contribution when any of the fractions is zero.

A single binary interaction parameter is in many cases not sufficient, and there are a number of so-called “**Excess Energy Models**” that use various power series in the fractions in order to describe how a binary interaction parameter may depend on the composition. All these are equivalent when the phase has only two constituents, because they can be transformed into each other by using the fact that the sum of their fractions is unity. Thus, the GES module has only been programmed to handle the power series that is usually known as the ***Redlich-Kister Model*** (or ***Subregular Solution Model***). This is generally written as a polynomial:

$$X_a * X_b * (L_0 + L_1 * (X_a - X_b) + L_2 * (X_a - X_b)^2 + L_3 * (X_a - X_b)^3 + \dots)$$

Where the “binary composition quantities” X_a and X_b are the fractions of the constituents a and b , and the so-called “binary L parameters” L_0, L_1, L_2, L_3 etc. are only as functions of temperature and pressure. Therefore, the terms with L_0, L_1, L_2, L_3 etc. describe how the binary interaction parameter depends on the composition. Obviously, the terms with L_0, L_2, L_4 and so forth are symmetric, while the terms with L_1, L_3 and so forth are asymmetric and they thus depend upon how the constituents a and b are referred to. Note that when there is only one regular binary interaction parameter L_0 , then it becomes the so-called ***Bragg-William Random Model***.

In order to extrapolate binary systems into a ternary system ($a-b-c$), it is natural to use the fractions of the constituents from the ternary system in the expression above (*i.e.*, in the combinations of X_a-X_b , X_a-X_c and X_b-X_c , respectively); similar assignments should be applied when extrapolating from binary systems to quaternary and higher-order systems. This is called the ***Muggianu Ternary Extrapolation Model***; when the

used binary interaction parameters are of the Redlich-Kister type, then it is called the ***Redlich-Kister_Muggianu Model***, which normally is the default ternary excess energy model for complex alloy solution phases in most of the currently-available Thermo-Calc databases.

However, another choice of power series for the binary interaction parameters would now give different results as the sum of the binary fractions are not unity. This difference has caused some discussions on the possibilities to describe asymmetrical features of a system by the choice of *power series* and *extrapolation* model. As there must be a physical reason for this asymmetry, GES employs the much more general ***Sublattice Model***. Only to make it possible to check calculations made with other programs, GES allows the use of the so-called ***Muggianu Ternary Extrapolation Model***, ***Kohler Ternary Extrapolation Model*** and ***Toop-Kohler Ternary Extrapolation Model***.

The GES module is not restricted to binary interaction parameters, and ternary, quaternary and higher-order interaction parameters can also be used if there is enough information to evaluate these from experimental data. Ternary interaction parameters are not infrequent and in some cases there has been suggested that they can be composition-dependent. In the GES module, an accessed (rather than extrapolated) ternary interaction contribution should always have the following formula (following the suggestion by Prof. Mats Hillert, 1980):

$$X_a \cdot X_b \cdot X_c \cdot (V_1 \cdot L_1 + V_2 \cdot L_2 + V_3 \cdot L_3)$$

where the “ternary composition quantities” X_a , X_b and X_c are fractions of constituents a, b and c, respectively, and

$$\begin{aligned} V_1 &= X_a + (1 - X_a - X_b - X_c) / 3 \\ V_2 &= X_b + (1 - X_a - X_b - X_c) / 3 \\ V_3 &= X_c + (1 - X_a - X_b - X_c) / 3 \end{aligned}$$

The ternary interaction parameters L_1 , L_2 and L_3 are only as functions of temperature and pressure.

In a ternary system a-b-c, V_j and X_i are identical, and the ternary interaction contribution part becomes a terms of:

$$X_a \cdot X_b \cdot X_c \cdot (X_a \cdot L_1 + X_b \cdot L_2 + X_c \cdot L_3)$$

However, in extrapolations to quaternary and higher-order systems, the sum of V_j is always unity while it is not true for the sum of X_i .

The L_1 , L_2 and L_3 parameters are identified by the degrees of the so-called “ternary L parameter” (*i.e.*, 0, 1 and 2, respectively) in the GES command `ENTER_PARAMETER`. If L_1 , L_2 and L_3 are equal, one just needs a single interaction parameter with a degree of zero.

If there is a ternary composition-dependent interaction parameter, one must enter all three parameters L_1 , L_2 and L_3 . This is to ensure that the user knows what he is doing. If such a ternary interaction parameter should be zero, then assign it a function which is zero or give it a very small value. If L_1 , L_2 and L_3 are the same value in a ternary system, then that is identical to having a composition-independent ternary parameter; and due to the fact that $[X_a + (1 - X_a - X_b - X_c) / 3] + [X_b + (1 - X_a - X_b - X_c) / 3] + [X_c + (1 - X_a - X_b - X_c) / 3] = 1$, such a composition-independent ternary term becomes redundant.

In a system with the a, b and c constituents in a substitutional phase with no sublattice (such as the LIQUID solution phase), the ternary interaction parameters (L_1 , L_2 and L_3) for the phase would be:

- $L(\text{LIQUID}, a, b, c; 0)$ to be multiplied with V_1
- $L(\text{LIQUID}, a, b, c; 1)$ to be multiplied with V_2
- $L(\text{LIQUID}, a, b, c; 2)$ to be multiplied with V_3

If a phase (such as the FCC_A1 solution phase) has both a substitutional site (in which the a, b and c constituents locate) and an interstitial site (where some interstitial species *e.g.* C and N coexist with vacancy), the ternary interaction parameters for the substitutional site would be:

- $L(\text{FCC_A1}, a, b, c; v; 0)$ to be multiplied with V_1 (that is corresponding to the substitutional site)
- $L(\text{FCC_A1}, a, b, c; v; 1)$ to be multiplied with V_2 (that is corresponding to the substitutional site)
- $L(\text{FCC_A1}, a, b, c; v; 2)$ to be multiplied with V_3 (that is corresponding to the substitutional site)

Important Notes: As a solution phase may be described as a substitutional phase with no sublattice which is occupied by two or more constituents (such as LIQUID, SLAG, GAS, AQUEOUS, DIAMOND_A4 and other phases in many databases), or as having two or more sublattice sites each of them may have two or more constituents (such as IONIC_LIQUID, FCC_A1, BCC_A2, HCP_A3, SIGMA, CEMENTITE, HALITE, among many other phases), the aforementioned “fractions of constituents” are always referred in the GES module as the composition quantities on a specific sublattice site, that is described by the so-called “site-fractions of constituents on a given sublattice site”, *i.e.*, the “Y” quantities. In case of a substitutional phase with no sublattice, the “Y” quantity is the same of the “X” quantity if considered as the “mole-fraction of the species in the phase”. However, if a phase has two or more sublattice sites, the **constituent array** concept has to be introduced, and one should always clearly distinguish the “Y” quantity (for a specific constituent on a given sublattice site in the phase) from the “mole-fraction of a certain species in the phase”; for more details, please refer to the *Sublattice Model (Compound-Energy-Formulism)* and *Ionic Two-Sublattice Liquid Model*.

11.3.4 Use of constituents other than elements

In the gaseous mixture phase, there are usually molecules formed, and the number of constituents of the gas phase is often much larger than the number of elements. This is also true for aqueous solutions and sometimes also for other liquid phases. Even some crystalline solids are formed by molecules rather than the individual elements. The standardized Gibbs energies that are required for a constituent that is not an element is the Gibbs energy of the constituent relative to the elements that form the species in their SER state. In a solution phase with non-element species, it is possible to introduce *interaction parameters* between various constituents of the phase, as described in the following models (the *Sublattice Model*, *Ionic Two-Sublattice Liquid Model*, *Quasichemical Model*, *Associate Model*, *Flory-Hussins Model*, *SIT/PITZ/HKF Models*,, etc.).

11.3.5 Sublattice Model -- Compound-Energy-Formulism

If the atoms are sufficiently different in size, electronegativity or charge, they may prefer different types of sites in the lattice of *crystalline solids*. In some cases, a solute atom may even occupy interstitial sites between the normal lattice sites. All such phenomena are treated in GES by the **sublattice** concept.

The entropy of mixing for a solution phase with sublattices is calculated as a sum of the ideal entropy of mixing among the constituents on each sublattice [multiplied with the number of sites (*i.e.*, the stoichiometric coefficient) of the specific sublattice site]. If there is only a single constituent on a specific sublattice in a solution phase, there is thus no entropy of mixing on that particular sublattice site.

By introducing sublattices, many traditional concepts in thermodynamics become complicated for the general case. For example, it is not possible to give an explicit formula for the partial Gibbs energies for the individual elements of a solution phase. In order to describe the thermochemistry of a solution phase with sublattices, it is necessary to introduce the **constituent array** concept. A constituent array is an array specifying one constituent in each sublattice. The surface of reference for the Gibbs energy of a solution phase with sublattices is constructed by the standardized Gibbs energies of all possible constituent arrays. The interaction parameters on each constituent array can then be expressed in much the same way as for a substitutional phase with no sublattice.

For a comprehensive description on the *Sublattice Model*, or the more generally-referred *Compound-Energy-Formulism*, please refer to Hillert (1998), Sundman and Ågren (1985).

This **Compound-Energy-Formulism (Sublattice Model)** has proved to be the most reliable model for many different types of solution phases, and thus it is, *by default* within the Thermo-Calc software, applied to various solid solution phases (such as metallic alloying solutions, intermediate compound solutions, simple carbide/nitride solutions, simple oxide/sulfide/silicate/carbonate/... solutions, and so on) within most of the currently-available Thermo-Calc databases (*e.g.*, TCFE, TCNI, TCNF, SLAG, TCMP, TCES; SSOL, SNOB, STBC, SALT, SEMC; TTNi, TTTi, TTTA, TTAI, TTMg, TTZr; NOX, NSLD; COST2, USLD; NUOX/SNUX; GCE). Unless specifically amended in the databases, such solution phases also have, *be default*, the excess Gibbs energy terms modeled by the *Redlich-Kister-Muggianu Model*.

Moreover, the **Compound-Energy-Formulism (Sublattice Model)** can also take ionic constraints (charged cation/anion species) into account, for complex solid solution phases with ionic constituents. Therefore, it has also been applied to model some complex carbide/nitride solutions, complex oxide/sulfide solutions and complex silicate solutions [such as HALITE, SPINEL, CORRUDUM, BETA_SIALON, OPRIM, MULLITE, OLIVINE, among others] in *e.g.* ION, SALT, STBC, TCFE and TCMP databases.

Examples of phase-constitution definitions using the **Compound-Energy-Formulism (Sublattice Model)**:

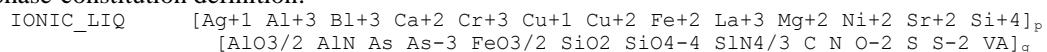
FCC_A1	$[Al\ Co\ Cr\ Cu\ Fe\ Mg\ Ni\ Si\ Ti]_1[C\ N\ O\ VA]_1$
BCC_A2	$[Al\ Co\ Cr\ Cu\ Fe\ Mg\ Ni\ Si\ Ti]_1[C\ N\ O\ VA]_3$
HCP_A3	$[Al\ Co\ Cr\ Cu\ Fe\ Mg\ Ni\ Si\ Ti]_1[C\ N\ O\ VA]_{0.5}$
CEMENTITE	$[Co\ Cr\ Fe\ Mn\ Mo\ Nb\ Ni\ V\ W]_3[C\ N]_1$
SIGMA	$[Al\ Co\ Fe\ Mn\ Ni]_8[Cr\ Mo\ Nb\ Ti\ V\ W]_4[Al\ Co\ Cr\ Fe\ Mn\ Mo\ Nb\ Ni\ Si\ Ti\ V\ W]_{18}$
CR3SI	$[Cr\ Fe\ Mo\ Nb\ Si]_3[Al\ Cr\ Nb\ Si]_1$
SPINEL	$[Al+3\ Cr+2\ Cr+3\ Fe+2\ Fe+3]_1[Al+3\ Cr+2\ Cr+3\ Fe+2\ Fe+3\ VA]_2[Fe+2\ VA]_2[N-3\ O-2]_4$
BETA_SIALON	$[Al+3\ Si+4]_3[N-3\ O-2]_3[N-3]_1$
OLIVINE	$[Ca+2\ Fe+2\ Mn+2]_1[Ca+2\ Fe+2\ Mg+2]_1[Si+4]_1[O-2]_4$

11.3.6 Ionic Two-Sublattice Liquid Model, for liquids with a tendency for ordering

Ionic liquids have usually very low entropy of mixing at certain compositions that coincide with neutral combinations of the ions.

According to a suggestion by Temkin, this can be modeled by assuming that the cations mix only with each other and anions with each other. Mathematically, this is equivalent with the Sublattice Model, already described for crystalline solids. The complication for an ionic liquid is that the numbers of sites (*i.e.*, the stoichiometric coefficients) on the cation sublattice and anion sublattice may vary with the composition.

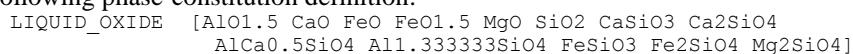
This problem is discussed in details by Hillert (1985), and a general formula, namely the **Ionic Two-Sublattice Liquid Model**, has been implemented in GES. This model has been applied to the so-called IONIC_LIQ solution phase (within *e.g.* ION, TCFE, SALT, SNUX/NUOX databases), which can have *e.g.* the following phase-constitution definition:



11.3.7 Associate Model

In some cases, the existence of molecules as constituents has been postulated, in order to describe experimental data for the liquid phase. The life-time of such a molecule may be so short that it cannot be found as independent entity but the *associate* has been introduced in order to describe tendencies for ordering around certain compositions. The existence of associates is supported if there is a tendency for covalent bonding between the elements.

The **Associate Model** implemented in the Thermo-Calc package treats this type of phase like a “gaseous mixture” phase with molecules, and one may have interaction terms between the molecules. This model has been used for the so-called LIQUID_OXIDE solution phase (within the NOX database), which can have *e.g.* the following phase-constitution definition:



11.3.8 Quasichemical Models

Three different **Quasichemical Models** have been implemented now:

- ☞ The *Quasichem-FACT00 Model* is based on the modified Quasichemical Model developed by Kongoli *et al.* (1998) in the F*A*C*T group in Montreal. It was implemented (into the GES module in 2000) as an associate model with a special mixing entropy expression.
- ☞ The *Quasichem-IRSID Model* is developed for the so-called SLAG solution phase (within the SLAG2 and TCMP2 databases), based on the IRSID Slag Model (*i.e.*, Kapoor-Frohberg-Gaye Quasichemical Cell Model).

- ☞ The *Quasichem-Ionic Model* is developed for the ionic liquid phase (the so-called QUAS_IONIC solution phase) for which the entropy can be calculated by the Quasichemical Model. For more details, please refer to the paper by Prof. Mats Hillert (2001; J of Alloys and Compounds, 320, 161-176). But it has not been applied to any real material system so far, and it may not improve things very much.

11.3.9 Non-chemical contributions to Gibbs energy (e.g., ferromagnetism)

In some cases, there are important physical phenomena that depend on the composition in a special way, the magnetic ordering is an example of this. In order to treat such phenomena, it is possible to introduce quantities that depend on the constitution of the phase by parameters multiplied with fractions in the same way as the parameters described above for the “chemical” Gibbs energy. An example of such a quantity is the Curie temperature and Bohr magneton number for magnetic ordering contributions (ferromagnetism or anti-ferromagnetism).

The values of the Curie temperature and Bohr magneton number at a certain composition is then used in a model to calculate the contribution to Gibbs energy due to magnetic ordering. The **Inden Magnetic Ordering Model** (Inden, 1976) is implemented in the GES module, in which two composition-dependent quantities [*i.e.*, TC for the Curie temperature for magnetic ordering, and BMAGN (or BM) for the Bohr magneton number] are considered for calculating the magnetic ordering contribution.

As will be mentioned in *Sections 11.3.13 and 11.2.14*, another example for non-chemical contributions to Gibbs energy is the hypothetical electrostatic effect in an aqueous solution phase. Various Born functions (X, Y, Z, ω , *etc.*) are used to describe the contributions from electrostatic properties, and the so-called Debye-Hückel Limiting Law term, ionic solvation, ionic association, *etc.*

11.3.10 Phases with order-disorder transformations

Some solution phases have chemical order-disorder transformations: for an example, the disordered FCC_A1 phase (FCC structure with the structural designation A1) can transform to the ordered L1₂ or L1₀ structure (FCC_L12 or FCC_L10) where the atoms of different kinds occupy different sublattices. In some cases, it may be advantageous and easy to describe such ordered superstructures as completely different phases; but in some other cases, like the B2 ordered BCC structure in Al-Fe alloys, the disordered and ordered structures (BCC_A2 and BCC_B2, respectively) must be described as one single phase because the transformation is of second order along a line in temperature and composition.

The ordering can easily be described by the Sublattice Model. For instance, an ordered FCC phase with only L1₂ or L1₀ ordering can be efficiently handled by the so-called **Two Substitutional-Sublattice Ordering Model**, while an ordered FCC phase with both the L1₂ and L1₀ ordering must have 4 substitutional sublattices using the so-called **Four Substitutional-Sublattice Ordering Model**; please note that, in both such models, all the substitutional constituents must enter into these substitutional sublattices. In addition, one may also have one sublattice for interstitials; in some cases, one may even describe ordering of the interstitials in some cases, like for the M2C carbides (in an ordered HCP structure).

Examples of phase-constitution definitions for the corresponding disordered and ordered parts for some phases with order-disorder transformations:

```

FCC_A1      [Al Co Cr Fe Mg Ni Si Ti]1[C N VA]1
FCC_L12     [Al Co Cr Fe Mg Ni Si Ti]0.75[Al Co Cr Fe Mg Ni Si Ti]0.25[C N VA]1
FCC_A1      [Ag Au Cu]1[VA]1
FCC_L102    [Ag Au Cu]0.25[Ag Au Cu]0.25[Ag Au Cu]0.25[Ag Au Cu]0.25[VA]1
BCC_A2      [Al Co Cr Fe Mg Ni Si Ti]1[C N VA]3
BCC_B2      [Al Co Cr Fe Mg Ni Si Ti]0.5[Al Co Cr Fe Mg Ni Si Ti]0.5[C N VA]3
BCC_A2      [Al Co Cr Ni]1[C N VA]3
BCC_B2      [Al Co Cr Ni]0.25[Al Co Cr Ni]0.25[Al Co Cr Ni]0.25[Al Co Cr Ni]0.25[C N VA]3

```

In the GES module, it is possible to enter solution phases with sublattices. However, the Sublattice Model description does not take short-range ordering into account.

If a solution phase with a chemical ordering/disordering transformation form a two-phase region, one must have two or more composition sets of that phase. New composition sets for solution phases can be created in the GES module (by the `AMEND_PHASE_DESCRIPTION` command) or in the POLY module (by the `ADVANCED_OPTIONS` command). However, since TCCR/TCW4, new composition sets for solution phases (whenever it is really necessary) can also be automatically generated thanks to the recently-implemented ***Global Minimization Technique***.

The GES module supports a feature of *splitting the parameters of a chemically ordered phase onto two different phase descriptions*. This has become very popular in current assessment work. In this way, one can have one phase for all parameters describing the disordered state (the ***configuration independent part***) and in the other phase only those parameters needed to describe the remaining part (the ***configuration dependent part***). This has been implemented in such a way that the contribution from the configuration dependent part is zero when the solution phase is disordered. In higher order systems, this is particularly useful as an ordered phase can dissolve several elements that have no particular contribution to the actual ordering. The GES module automatically adds these together after the command has been given:

```
AMEND_PHASE_DESCRIPTION <ordered phase> DISORDERED_PART <disordered phase>
```

It is often necessary to have two or more composition sets in solution phases which have order/disorder transformations, but the users have often forgotten to create composition sets for both the ordered and disordered phases, and this leads to program crashes. The command `AMEND_PHASE_DESCRIPTION` in the GES module, or the command `ADVANCED_OPTION_SET_MISCIBILITY` in the POLY module, can desirably creates new composition sets also for disordered phase when it is executed for the ordered phase. Once again, since TCCR/TCW4, new composition sets for solution phases (whenever it is really necessary) can also be automatically generated due to the ***Global Minimization Technique***.

11.3.11 CVM Approach: for chemical ordering/disordering phenomenon

The CVM (cluster-variation-method) Approach to model chemical ordering/disordering phenomenon has been implemented in the GES module, but at present moment it is only available for binary systems (Sundman and Mohri, 1990). See *Example 31* in the `\TCEX\` area (`TCEX31.TCM`) and in ***Thermo-Calc Examples Book***.

11.3.12 Birch-Murnaghan Model: for high-PT contributions

Under high pressure (and high temperature) conditions, the EOS (equation of state) of metals, minerals and other types of materials can be handled by either the Murnaghan or Birch-Murnaghahn models, to describe their *P-V-T* relations and thermodynamic properties (Saxena *et al.*, 1992).

The ***Murnaghan Model*** can be used from 1 bar to pressure up to 20 GPa, and the Gibbs energy of a stoichiometric phase can be implicitly expressed as a function of temperature and pressure.

The ***Birch-Murnaghan Model*** can be utilized over an even wide pressure range (from 1 bar to above 20 GPa), while the Gibbs energy of a stoichiometric phase can only be converted from the Helmholtz energy that is implicitly expressed as a function of temperature and pressure.

By default in the GES module, the composition dependence of a solution phase under high pressures, is normally handled by the general *Compound-Energy-Formulism (Sublattice Model)* with the *Redlich-Kister_Muggianu Model* for calculating the excess Gibbs energies.

11.3.13 Generalized PVT Model: for high-PT contributions

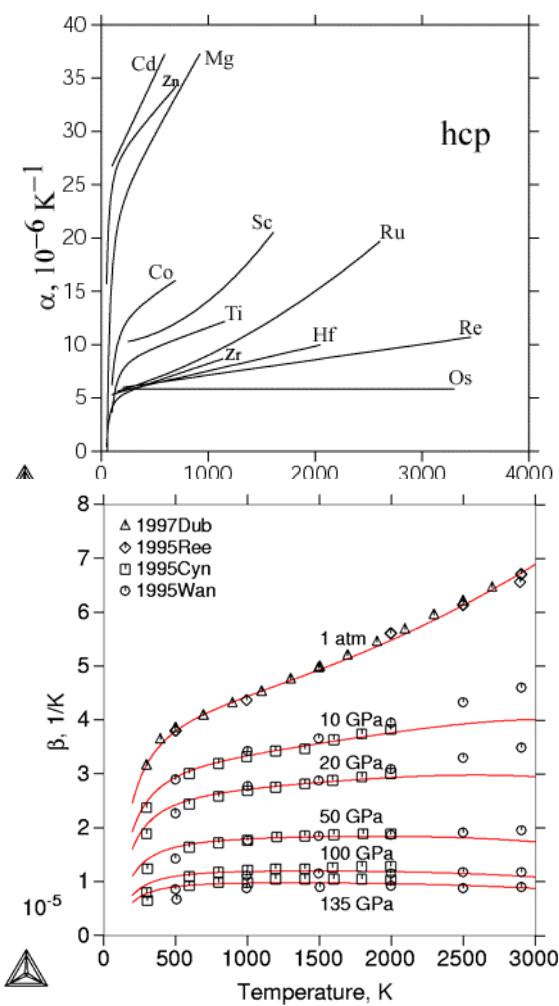
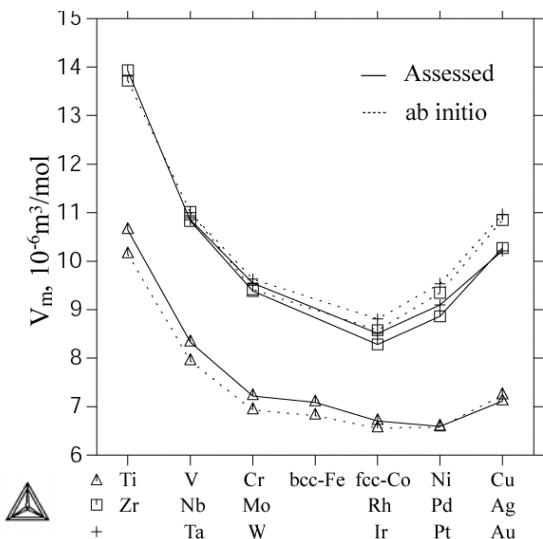
In the previous versions (up to TCCQ/TCW2) of Thermo-Calc software, there are several thermodynamic models to handle pressure-volume-temperature EOS (equation of state) for metallic alloys and other varied forms of solids/minerals (carbides, nitrides, hydrides, oxides, sulfides, hydroxides, borides, phosphites, halites, silicates, carbonates, sulphates, nitrates, phosphates, etc.), as well as for liquids (melts/slag), aqueous solutions and gaseous mixtures. Particularly for alloys and minerals, there are Murnaghan Model, Birch-Murnaghan

Model and Modified Birch-Murnaghan Model that have been implemented into the GES (Gibbs Energy System) of the Thermo-Calc software; and associated with such models several databases bears some assessed data (*e.g.*, GCE2/PGE0 for minerals, while far from complete for alloys), on molar volume, thermal expansion, bulk modulus and compressibility, as well as their dependencies on temperature and pressure.

Since TCCR/TCW4 (and corresponding programming interfaces), a new model, called **Generalized PVT Model (GPVT)**, will be made available, and a special database/data-set using this model will be released. This makes it possible to calculate volume and related thermo-physical properties (as a function of temperature, pressure and composition) whenever precise information of such properties is required.

The new developments (within the TCFE database framework) aim at: 1) generating a molar volume and bulk modulus database, *i.e.*, including $V_m(T, P, X_i)$ and $K(T, P, X_i)$ data for systems involving Al, B, C, Co, Cr, Cu, Fe, Mg, Mn, Mo, N, Nb, Ni, O, P, S, Si, Ti, V, W, etc., in various structured phases *e.g.* fcc, bcc, hcp, carbides, nitrides, oxides, liquids and so on; 2) creating theoretical methods to calculate thermo-physical properties, especially for metastable phases; 3) establishing and implementing a new robust model for pressure dependence of Gibbs free energy.

Three comprehensive approaches have been used during these new developments: a) performing the CALPHAD-type critical-assessments (based on various available experimental data) using the PARROT module in the Thermo-Calc software; b) conducting some necessary calculations using first-principles calculations plus Debye-Grüneisen modelling; c) building thermal EOS on fundamental physics. Such approaches has resulted in a high-quality volume dataset (within the TCFE steels/Fe-alloys database, namely TCFE3/TCFE4/TCFE5/TCFE6) that uses the new model specifically for steels and many other types of alloys and their liquids, which had been released since TCCR/TCW4.



Proven successful applications include: high-pressure phase diagrams, dilatometric curves during temperature changing, dilatometric analysis of isothermal decompositions, lattice misfit, among many others.

11.3.14 Ideal gas model vs. non-ideal gas/gaseous mixture models

As an approximation, the ideal gas model may handle a pure gas phase or a gaseous mixture phase under low pressures (and low temperatures). This implies that the P - V - T relations and thermodynamic properties of pure gaseous species are calculated as for ideal gas, and that there is interaction between gaseous species in the mixture. In most cases of currently available substance and solution databases (such as SSUB, SLAG, TCMP, TCES, TCFE, etc.), the GAS mixture phase is treated in this way.

However, under high pressure and high temperature conditions, the EOS (equation of state) of pure gas species and real gaseous mixtures are different from the ideal behaviors. The P - V - T relations and thermodynamic properties of either the pure gas species or gaseous mixture phases must then be described by some non-ideal gas and gaseous mixture models that are reliable under both subcritical and supercritical temperature-pressure conditions. There are some comprehensive models that are extensively applied in the fields of physical chemistry, chemical engineering, geochemistry/geophysics, etc.

The **SUPERFLUID Model** (Belenoshko, Shi and Saxena, 1992), as coded in the **SUPERFLUID program** (published in *Computers and Geosciences*, 18(1992):1267-1269) will be implemented in the next TCC version. This model was developed for studying P - V - T relations and thermodynamic properties of the C-H-O-S-N-Ar gaseous/fluid mixtures (with 13 species H_2O - CO_2 - CO - CH_4 - O_2 - H_2 - S_2 - SO_2 - COS - H_2S - N_2 - NH_3 -Ar) under sub-/super-critical conditions. It incorporates non-ideal models that are applicable over a very wide temperature-pressure-composition range, for both various pure gaseous species [Johnson and Norton, 1991; Shi and Saxena, 1992; Belonoshko and Saxena, 1991, 1992; Saxena and Fei, 1987, 1988; Saul and Wanger, 1987, 1989; Haar *et al.*, 1984; Levelt Senger *et al.*, 1983; Jacobs and Kerrick, 1981; Kerrick and Jacobs, 1981; among others] and gaseous mixtures [Shi and Saxena, 1992; Belonoshko and Saxena, 1992; Saxena and Fei, 1988]. The SUPERFLUID program itself can be run independently, under Windows Vista/XP/2000/NT4, PC Linux and various UNIX platforms.

The SUPERFLUID model implemented in the GES module of the next TCC version will be possibly switched on for investigating complex heterogeneous interactions among such non-ideal gaseous/fluid mixtures, metals/alloys/slag, minerals, aqueous solutions, etc.

11.3.15 DHLL and SIT Models: for dilute aqueous solutions

In dilute aqueous solutions, the activity coefficients of the solvent (H_2O) and solute species may be described by the **DHLL (Debye-Hückel Limiting Law) Model** that ignores the effects of ionic solvation and only uses some simple terms for the electrostatic contributions (Atkins, 1982). With the aqueous solution concentration increases, the **SIT (Specific Ionic Interaction Theory) Model** can be applied, which takes accounts of not only the DHLL term but also some simple binary, ternary and higher-order ionic interaction terms (Ciavatta, 1990).

In both cases, the P - V - T relations and standard thermodynamic properties of the solvent H_2O (and solute species) may be calculated by some simple models.

Applying the SIT model, the **TCAQ Aqueous Solution Database** (or its public edition **PAQ** for demonstrations of easy calculations of the so-called Pourbaix diagrams) can be used to investigate various heterogeneous interaction systems involving aqueous solutions, gaseous/fluid mixtures, metals/alloys/slag, minerals (oxides/hydroxides/silicates/carbonates/sulfides/sulphates/nitrides/nitrates/halites), etc., under relatively-lower temperature-pressure-composition conditions (if compared with the complete HKF Model with AQS Database).

11.3.16 HKF and PITZ Models: for concentrated aqueous solutions

At elevated temperature and pressure condition, and at higher concentrations, the activity coefficients of the solvent (H_2O) and solute species should be described by the **HKF (Complete Revised_Helgeson-Kirkham-Flowers) Model** that uses the most comprehensive approach to study the electrostatic contributions, effects of ionic solvations and ionic interactions (binary, ternary and higher-order). For some specific aqueous solution systems that have been experimentally investigated at extremely high concentrations, the **PITZ (Generalized Pitzer's Formalism) Model** should be applied.

The complete HKF model (implemented since TCCP/TCW2) will contain two highly non-ideal parts for aqueous solution. The first part is the P - V - T relations and standard thermodynamic properties of the solvent H_2O and solute species, that are calculated by the **Johnson-Norton Model for H_2O** (Johnson and Norton, 1991) and the Revised_Helgeson-Kirkham-Flowers model for aqueous solutes (Helgeson and Kirkham, 1976; Helgeson *et al.*, 1981; Shock and Helgeson, 1988, 1990; Shock *et al.*, 1989; Shi *et al.*, 1993), as coded in the **SUPCRT92 program** (Johnson *et al.*, 1992) and **SUPERFLUID program** (Belenoshko *et al.*, 1992; see Section 11.3.13). The second part is the non-ideal mixing properties of cations, anions and complexes in the aqueous solution phase, that are calculated by the complete Revised_Helgeson-Kirkham-Flowers model (Helgeson *et al.*, 1981; Shi *et al.*, 1993), as coded in the **AQSMIX program** (Shi and Saxena, 1994). The contributions of Born-functions of solvent H_2O and solute species in the standard and mixing states are systematically calculated. The complete Revised-HKF model can be effectively and reliably applied to complex aqueous solution phase from room temperature and pressure to 1000°C and 5 kbar.

Applying the complete HKF model, one can use the **AQS Aqueous Solution Database** (or its public edition **PAQS** for demonstrations of easy calculations of the so-called Pourbaix diagrams) to investigate various heterogeneous interaction systems involving aqueous solutions, gaseous/fluid mixtures, metals/alloys/slag, minerals (oxides/hydroxides/silicates/carbonates/sulfides/sulphates/nitrides/nitrates/halites), etc., over a wide range of temperature, pressure and composition.

The generalized PITZ model (Pitzer, 1973, 1987, 1991; Pitzer *et al.*, 1984; Spencer *et al.*, 1990; Sterner *et al.*, 1992) will be implemented in the Thermo-Calc package in the near future.

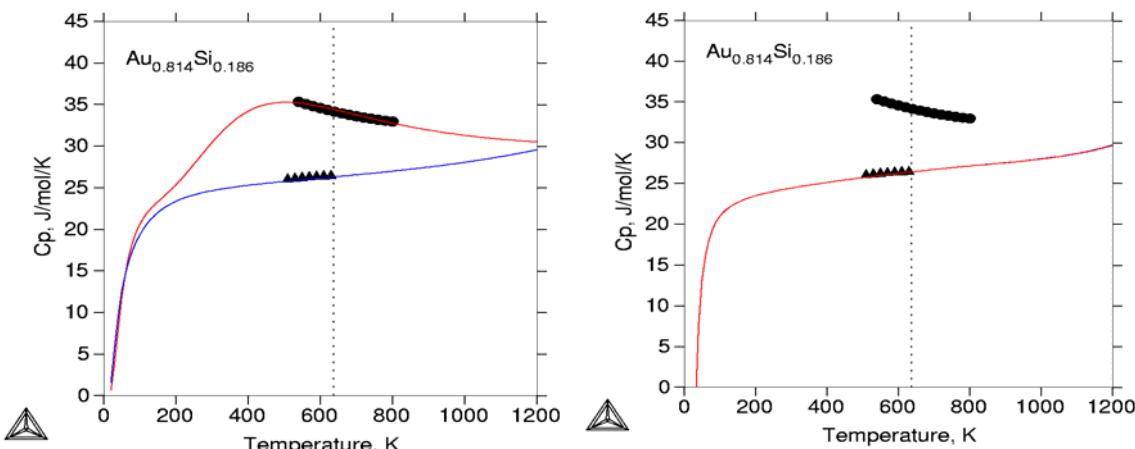
11.3.17 Generalized Two-State Model

The new model G2SLA (**Generalized Two-State Model for Liquid Amorphous Phases**) provides an unified way to describe thermodynamic properties of liquid amorphous phases smoothly and more reasonably. In the next version of Thermo-Calc, a special parameter related to this model can be entered and hence the contribution to the Gibbs free energy from the two-state excitation can be easily added together with contributions due to other physical effects, such as magnetic ordering.

The essence of the generalized two-state model is the assumption that the atoms in a liquid amorphous phase can be in either the amorphous-like state or the liquid-like state and the excitation from the amorphous-like state to the liquid-like state contributes to the total Gibbs free energy of the liquid-amorphous phase.

Advantages of this model:

- ☞ Very general. Parameters can be easily fitted;
- ☞ Reasonable extrapolation of heat capacity of liquid phase below the melting point;
- ☞ Accurate estimation of enthalpy of crystallization of amorphous phase; Provides accurate thermodynamic data for kinetic simulation of amorphous to crystal transformation.



Example: Au-Si

Comparison between the calculated and experimental heat capacity of the eutectic liquid-amorphous phase and solid mixture in the Au-Si system. (a) new modelling using two-state model. (b) results calculated using the SSOL database, which give the same heat capacities for liquid and solid. The dashed lines indicate the melting of this alloy.

11.3.18 Flory-Huggins Model: for polymers

This is a model for liquid polymers where the constituents can be only different in SIZE and located in only one set of sites. First, the new composition variable is introduced as below:

$$\Phi_i = \frac{v_i y_i}{\sum_j v_j y_j}$$

where v_i is related to the size of the constituent. It is a constant independent of composition, temperature and pressure. For each constituent, the value of v_i is set with the GES command

```
ENTER_PARAMETER PLYV(POLYMER, I) 298.15 vi; 6000 N
```

Due to the standard syntax, one must specify a low and high temperature limits, but one must not give anything else than a constant vi . $PLYV$ is a special identifier for the v_i parameter.

The Gibbs energy for this model is

$$G_m = \sum_i y_i G_i^o + RT \sum_i y_i \ln\left(\frac{\Phi_i}{y_i}\right) + G_m^E$$

$$G_m^E = \sum_i \sum_{j>i} \Phi_i \Phi_j \left(y_i + \frac{v_j}{v_i} y_j\right) \cdot L_{ij}$$

The L_{ij} parameter is the GES excess energy parameter and related to the standard X_{ij} parameter of the Flory-Huggins Model, which is defined as

$$L_{ij} = RT v_j X_{ij}$$

Note that j is the second (alphabetically) constituent.

11.4 Thermodynamic Parameters

From the above discussions, the thermochemical parameters of a phase are functions of temperature and pressure, and they are multiplied with various fractions of the constituents of the phase before adding its value to the Gibbs energy. The parameter is thus identified by the phase and the set of its constituents; the fractions of the constituents are multiplied with the parameter. In some cases, the interaction parameters are composition-dependent (see *Section 11.3.3* about the Regular/Subregular Solution Model), and the various terms of this composition-dependence are denoted by an *index number* (*i.e.*, “*degree of the composition-dependence*”). The interpretation of this index value is model-dependent.

A number of other quantities may also be composition-dependent (*e.g.*, the Curie temperature and Bohr magneton number of magnetic ordering, and various volume-related parameters), and they can also be expressed as a function of parameters multiplied with fractions. Some other quantities may be complex functions of temperature, pressure and composition (*e.g.*, the Born functions of hypothetical electrostatic effects).

Such parameters for magnetic ordering, volume changes or electrostatic contributions are distinguished from those describing the “chemical” Gibbs energy by special identifiers:

- For “chemical” parameters, the identifiers are G (for standardized Gibbs energy) or L (for excess Gibbs energy by interactions).
- For “volume” contributions (to either standard Gibbs energy or excess Gibbs energy), the identifiers are V0 (for molar volume at 298.15 K and 1 bar), VA (for integrated thermal expansivity), VB (for bulk modulus at 1 bar), VC (for isothermal compressibility) and VK (for high-pressure fitting parameter).
- For “magnetic ordering” contributions (to either standard Gibbs energy or excess Gibbs energy), two composition-dependent quantities are needed, and the identifiers used for them are:
 - TC for the Curie temperature,
 - BMAGN (or BM) for the Bohr magneton number.
- For “electrostatic” contributions in aqueous solutions (using the complete Revised HKF Model), the Born functions (X, Y, Z, ω , *etc.*) for the solvent (H_2O) are stored as data blocks in some specially-implemented subroutines in the GES module, while the Born functions $\omega_{Pr,Tr}$ for solute species are stored in the normal GES5 workspaces using the identifier BM. (Note: this identifier similar to that for Bohr magneton number is just for temporal use since TCCP/TCW2, and it will be changed to WB in the future versions).

The general form of a parameter is:

```
<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.

Note that other thermodynamic quantities such as H (enthalpy), S (entropy), V (Volume), F (Helmholtz energy), etc. can be calculated from the integral Gibbs energy when necessary, but they are never stored as an explicit expression.

A parameter is always associated with one or several specific end-members in a certain phase. The phase name, and its constituent(s) (for a substitutional phase with no sublattice) or constituent-array (for a phase with sublattices) which correspond to the specific end-member(s), must be supplied. The phase name can be abbreviated.

A parameter is identified by the constituents, of which the stoichiometric coefficients are multiplied with the parameter. The names of the constituents can be abbreviated. Note that it is the NAME of the constituent, not its stoichiometric formula, that is needed in the parameter. If the phase has only one constituent in each sublattice, these can be omitted.

For phases with several sublattices, the GES program will ask for at least one constituent in each sublattice. The excess energy parameters, e.g., the regular/subregular (binary) parameters or ternary parameters, are multiplied with two or more fractions of the constituents in the same sublattice. These additional constituents must be given as interacting constituents. Note that phases with sublattices may have interacting constituents in each sublattice.

An interaction parameter, which is used to describe the excess term of a quantity, must have two or more constituents that “interact each other”. 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 GES 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.

For each parameter, an expression to describe the mathematical functions must be given after the parameter name. Sometimes, an expression can be used for several parameters (for example for reference energy for the elements). In this case, it is recommended that the expression is entered as a function symbol, and this function symbol can then be used when further defining the parameters.

A thermodynamic parameter can be a simple TP-Function or a pre-defined symbol (that is presented by a numeric constant, a numeric variable, a function or a table). An already-defined symbol can be directly used as a part of the definition of a parameter. The symbols R [that stands for the *Molar Gas Constant*, being equal to 8.31441 J/mol/K (in SIT Unit)] and RTLNP [that is the *volume contribution to the standard Gibbs energy of an ideal gaseous species*] are pre-defined functions by default in the GES5 workspaces, and thus they can be directly used in any function term. However, any other symbol must have already been defined (either as it exists in the used database being retrieved through the TDB module, or before it is referred as a part of an additionally/directly entered TP-Function in the GES module); otherwise, the TDB module will issue an error message (complaining the missing symbols when retrieving data from a database), or the GES module will assign such a symbol with a value of zero when it is referred as a part as an entered TP-Function. A symbol is often suffixed with the # sign (in a database file or in GES5 workspace); however, since TCCN/TCW1 such a suffix can be optionally ignored.

A parameter always starts with a low temperature limit of its applicability, followed by one or more expressions that are coded as mathematical relations of constants, entered functions (normally with a # suffix, e.g. +3*GSERAL#), and TP-Functions. Each expression should end with a semicolon “;”, and be followed by its upper applicable temperature limit and a continuation indicator (Y for continuing with the next expression, or N for ending the parameter’s expression). If there is no continuation after a specific expression, the reference index can optionally be given.

However, it is also possible to have pressure ranges instead of temperature ranges. This is indicated by giving the lower temperature limit as e.g. -10000 (meaning the lower pressure limit as 1E4 Pascal, i.e. 0.1 bar). The breakpoints will then be interpreted as pressures instead. It is essential that the functions are smooth, both first and second derivatives continuous, around a break point.

A complete parameter entry can be written in several continuation lines if the GES parameter name and parameter’s expressions are too long or if there are more than one applicable expression, as the maximum length of each line is 78 characters.

Examples of various parameter names:

```

G(GAS,AL1CL2H1O1) 298.15 +F243T#+R*T*LN(1E-5*P); 6000 N 151 !
G(STEAM,H2O1) -1E4 +F1001P#+RTLNP#; 1E5 Y +F1002P#+RTLNP#; 1E7 Y
+F1003P#+RTLNP#; 1E8 Y +F1004P#+RTLNP#; 1E11 Y
+F1005P#+RTLNP#; 1E15 N R152 !
G(CD_L,CD) 298.15 +F5462T#+6192.3-10.42123864*T; 6000 N 2609 !
G(LIQUID,CA) 298.15 +5844.846+62.4838*T-16.3138*T*LN(T)
-.01110455*T**2-133574*T**(-1); 500 Y
+7838.856+18.2979*T-8.9874787*T*LN(T)-.02266537*T**2
+3.338303E-06*T**3-230193*T**(-1); 1115 Y
-2654.938 +188.9223*T-35*T*LN(T); 3000 N R283 !
G(BCC_A2,FE:VA;0) 298.15 +GHSERFE#; 6000 N 91DIN !
G(BCC_A2,FE:C;0) 298.15 +GHSERFE#+3*GHSERC#+322050+75.667*T; 6000 N R186 !
G(M23C6,V:MN:C) 298.15 0.869565*GVM23C6#+0.130435*GMN23C6#; 6000 N REF275 !
G(QUARTZ_S2,O2SI1) 298.15 F13149T#+732.2-.8644628099*T; 6000 N R5934 !
G(AQUEOUS,FE+3) 298.15 -48534.4+315.892*T+ZAD; 350 N RPAQ2 !
G(AQUEOUS,S2O3-2;0) 298.15 +Z0064GTP#+ZAD#; 1600 N REF1 !

L(FCC_A1,CR,NI:VA;0) 298.15 +8030-12.8801*T; 6000 N R322 !
L(FCC_A1,FE,MO,NI:VA;0) 298.15 -204791+163.93*T; 6000 N R132 !
L(FCC_A1,FE,MO,NI:VA;1) 298.15 11555-55.81*T; 6000 N R132 !
L(FCC_A1,FE,MO,NI:VA;2) 298.15 77975; 6000 N R132 !
L(M23C6,CR,FE:CR:C;0) 298.15 -205342+141.6667*T; 6000 N R322 !
L(M23C6,CR,FE,NI:FE:C;0) 298.15 -460000; 6000 N R322 !
L(M23C6,FE,MN:CR,MN:C;0) 298.15 -100000; 6000 N R326 !
L(LIQUID,CR,FE;0) 298.15 -14550+6.65*T; 6000 N R107 !
L(LIQUID,CR,FE,NI;0) 298.15 14510; 6000 N R322 !
L(LIQUID,CR,FE,NI;1) 298.15 11977; 6000 N R322 !
L(LIQUID,CR,FE,NI;2) 298.15 5147; 6000 N R322 !
L(AQUEOUS,AL+3,CL-1;0) 298.15 -0.481; 1273.15 N R1 !
L(AQUEOUS,CUCL2,NA+1,CLO4-1;0) 298.15 +0.27; 1273.15 N R5 !

TC(CO_S2,CO) 298.15 1396; 1768 Y 1E-05; 6000 N ,!
TC(FCC_A1,FE:VA) 298.15 -201.00; 6000 N R281 !
TC(BCC_A2,CR,MN:VA;0) 298.15 -1325; 6000 N R326 !
TC(BCC_A2,CR,MN:VA;2) 298.15 -1133; 6000 N R326 !
TC(BCC_A2,CR,MN:VA;4) 298.15 -10294; 6000 N R326 !
TC(BCC_A2,CR,MN:VA;6) 298.15 26706; 6000 N R326 !
TC(BCC_A2,CR,MN:VA;8) 298.15 -28117; 6000 N R326 !

BMAGN(CO_S2,CO) 298.15 1.35; 1768 Y 1E-05; 6000 N ,!
BMAGN(FCC_A1,FE:VA) 298.15 -2.10; 6000 N R281 !
BMAGN(BCC_A2,CR,FE:N;0) 298.15 -.85; 6000 N R126 !
BMAGN(BCC_A2,CR,MN:VA;0) 298.15 .48643; 6000 N R326 !
BMAGN(BCC_A2,CR,MN:VA;2) 298.15 -.72035; 6000 N R326 !
BMAGN(BCC_A2,CR,MN:VA;4) 298.15 -1.93265; 6000 N R326 !

BM(AQUEOUS,FE+3) 298.15 +Z0127PW0#; 1600 N R127 !
BM(AQUEOUS,U3O6C6O18-6) 298.15 +Z0270PW0#; 1600 N R270 !

V0(BCC_A2,FE:VA;0) 298.15 +V0FEBCC#; 6000 N REF05LU !
VA(BCC_A2,FE:VA;0) 298.15 +VAFEBCC#; 6000 N REF05LU !
V0(BCC_A2,CR,FE:VA;0) 298.15 +ZERO#; 6000 N REF06V !
V0(BCC_A2,CR,FE:VA;1) 298.15 -1.10524097E-7; 6000 N REF06V !
V0(BCC_A2,CR,FE:VA;2) 298.15 +1.40024130E-7; 6000 N REF06V !
VA(BCC_A2,CR,FE:VA;0) 298.15 -6.49444634E-6*DELTAT#; 6000 N REF06V !
VA(BCC_A2,CR,FE:VA;1) 298.15 +2.91269321E-5*DELTAT#; 6000 N REF06V !

```

The GES command `LIST_SYMBOL` can list all and/or any specified symbols (numeric constants, numeric variables, functions or tables) that are already defined in the current GES5 workspace; the `LIST_PARAMETER` command can list an already-defined parameter and the `LIST_PHASE_DATA` command can list various parameters of a specified phase (if the phase parameter have been retrieved from an unencrypted database through the TDB module, or directly entered in the GES module).

Addition to the predefined parameters retrieved from database(s), one can further use the GES commands `ENTER_PARAMETER` or `ENTER_SYMBOL` to define other necessary parameters. Furthermore, any of the already-defined parameters can be modified by using the GES commands `AMEND_PARAMETER` or `AMEND_SYMBOL`.

11.5 Data Structures

It is possible in the GES module to combine all the models described in the previous sections, and a phase can thus have sublattices, associates, interaction parameters, magnetic contributions and so on. Each phase is also treated completely independent in the GES5 workspaces.

The variety of models that can be used in GES requires that the data storage must be developed beyond the *static arrays* that are provided by FORTRAN. The most general data structure, which is used in modern computer languages like LISP and Pascal, is “*lists*”. These languages have this facility built into the language itself. In FORTRAN, it is necessary to write a subroutine package for this type of data structures. A very general package has been developed in both the Thermo-Calc and DICTRA software. In a *linked structure*, the actual data is stored in records that may be scattered around in the memory of the computer. This part of the memory is usually called a heap. The records can only be accessed through pointers and a record may contain pointers to other records. Thus very flexible and general lists can be created. The disadvantage of the list structures is that they may require longer computational times for simple problems compared to a program which has been tailored for that problem.

In the GES module, the workspace routines are used to create a linked structure where the data structure in itself defines the mathematical expression for the Gibbs energy. This structure can be used to calculate both the integral Gibbs energy and any partial derivative analytically.

11.5.1 Constitution

A phase has one list for each sublattice(s) where the constituents of the sublattice and their stoichiometric coefficients are stored.

Sublattices are used to describe a deviation from Random Mixing in a phase, *e.g.*, when interstitials or constituents with different sizes or electronegativity prefer different lattice sites in a phase.

In *Constituent Array* (of a stoichiometric or solution phase with sublattices) in phase Constituents and phase Parameters, various sublattices are separated by the “:” signs, and within a specific sublattice various constituents are separated by the “,” signs. For more details, see [Section 11.4](#) and [Chapter 3 \(Database Module \(TDB\) – Manager’s Guide\)](#) in the document *Thermo-Calc Database Guide*.

11.5.2 Surface of reference for the Gibbs energy

The standardized Gibbs energy parameters for each phase define a surface of reference for the phase. These parameters are stored in a linear list starting from the phase record. Each record in this list has pointers to the constituent records where the fractions are stored, which shall be multiplied with the value of the parameter. The parameter itself is a TP-Function and is stored in another list.

11.5.3 Excess Gibbs energy

The parameters used to describe the excess Gibbs energy are stored in a *tree structure* where the root is from a record of a standardized Gibbs energy. For each level in this tree, there is a list of parameters that are multiplied with the same number of fractions. On the next level, the parameters are multiplied with one more fraction. Excess energy parameters that are zero or not determined are simply missing in this tree. This saves calculation time, because the number of possible parameters for a phase with many constituents is much larger than the actual number.

11.5.4 Stored private files

In GES the definition of the system and all thermochemical data which are stored in the memory of the computer can be written on an unformatted file (with a default extension “GES5”). A file written in this way can also be read back into the program. If there is no databank connected to the application program, this file storage is the only possibility to save data that have been entered interactively or have been transferred thermochemical data between various thermochemical software. Note that the unformatted files written with the

“GES5” extension cannot be used for transferring data between different computers. The content of these files cannot be printed to a printer either.

Such stored *.GES5 files have the same format on PC Windows Vista/XP/2000/NT4, PC Linux and various UNIX platforms, and they are essential for running various Thermo-Calc application programming interfaces (*e.g.*, TQ, TC-API and/or TC-Toolbox in MATLAB).

Parts of the stored information on a defined system and retrieved thermodynamic data can be written as simple texts on a formatted file (with a default extension “DAT” or “TDB”). Such files could be opened and edited outside of the Thermo-Calc software, using some textual editors.

*Note that for encrypted databases since TCCP/TCW2, one may not be able to use this command to list the any thermodynamic data for a defined system or a specific phase in the system, on screen or in such *.DAT/*.TDB files.*

11.5.5 Encrypted and unencrypted databases

Since TCCP (as well as TCW2 and DICTRA22), all commercial Thermo-Calc and DICTRA databases are distributed in an encrypted format in order to avoid illegal copying, while for all free/public databases the unencrypted format is still used. Such an encrypted database can be used only together with a specific database license key that is obtained from TCS or its agent.

Such database encryption has caused some problems in TCCP/TCW2/DICTRA22 in the simultaneous use of databases of different types. From TCCQ/TCW/DICTRA23, significant efforts have been done to remove those inconveniences and it is now possible to combine encrypted databases with each other or with public databases.

An encrypted database always consists of binary data files with extension *.TDC (both for Thermo-Calc thermodynamic databases and for DICTRA mobility databases), while an unencrypted database may contain various types of textual data files (unblocked, blocked or sorted) with different extensions (*e.g.*, *.TDB, *.DAT, *.REF, *etc.*).

An *.TDC data file cannot be viewed or edited by any editor, and is only usable inside the TCC/TCW/DICTRA software, as well as in the TQ/TC-API programming interfaces and TC-MATLAB Toolbox, with an appropriate database key (*i.e.*, a database feature in a License File for an already-licensed Designated Installation), which is solely made by TCS for a specific computer/server based on the unique identification number(s) obtained from the end-user(s).

A blocked or sorted data file (*e.g.*, *.TDB, *.DAT, *.REF, *etc.*) can be viewed by a simple textual editor, but a user should never try to edit it due to its restricted structure (otherwise, it can not be used by the TCC/TCW/DICTRA software anymore). An unblocked data file can be viewed and edited with a great carefulness by a simple textual editor. A user-specified database is normally built as unblocked data file(s).

Previously, the database files are occasionally CPU-dependent, implying the files for UNIX/Linux system and for PC Windows OS are incompatible, due to different data structures. Since TCCP, TCW2.1 and DICTRA22, a standard routine for automatically converting database files has been implemented, making it possible to read database files from both systems. The routine gives a warning message if conversion is necessary.

The retrieval of thermodynamic data in the TDB module, and the storage of data in the GES5 workspace, from such encrypted databases remain the same as for unencrypted ones. But thermodynamic parameters and related functions retrieved from an encrypted commercial database (except for two SGTE-owned databases, namely SSUB3 and SSOL2/SSOL4), for any defined system or any phase in the system, cannot be shown/listed up on screen or into textual files (*.TDB or *.DAT) in the GES module, while the basic definitions of phases and phase-constitutions can be shown using the GES command LIST_CONSTITUENT or TDB command-sequence LIST_SYSTEM_CONSTITUENT; and since TCCS/TCW5 (for all encrypted databases), the GES commands LIST_PHASE_DATA or LIST_DATA can also list definitions of phases/phase-constitutions and references (but not thermodynamic parameters) for a defined system. However, for some (but not all) commercial Thermo-Calc or DICTRA databases, upon a special request from a client, a simple-textual version (*i.e.*, without data encryption) of a database could be received and installed, on the necessary and restrict condition that a specific “**TCS Database License Agreement**” would have been bilaterally signed between the client and TCS; if you are interested in such a special service, please contact info@thermocalc.com for the related details.

11.6 Application Programs for the GES Module

The GES module is a subroutine package, and although it is possible to use it for direct tabulation of thermochemical quantities, the normal use of GES is together with an application program, like the POLY module and DICTRA software, which needs these thermochemical quantities in order to perform some calculations. Examples of such thermodynamic computations are calculations of chemical reactions, equilibrium calculations, and calculations of driving forces for diffusion-controlled phase transformations.

Normally, the GES module is connected to databases with thermochemical parameters that is needed to make any calculation. However, it is also possible to enter data interactively in the GES module, and to use a file to store and retrieve such data.

The GES module currently used in the Thermo-Calc software is internally connected with the TDB module (for thermochemical database retrieval), the POLY module (for heterogeneous equilibrium calculations), the TAB module (for thermodynamic property tabulations), and the PARROT module (for assessing experimental data). GES is also connected with the DICTRA software.

Another level of application programs is for the user-written programs using the Thermo-Calc application programming interfaces, TQ, TC-API or TC-Toolbox in MATLAB software. Under such circumstances, the user needs first to retrieve all the necessary thermochemical data in the TDB module and to generate an *.GES file in the GES module of the Thermo-Calc software for a particularly defined system, and then to open such a file within the application programs for thermodynamic calculations and kinetic simulations.

11.7 User Interface

The GES module uses a command structure inherited from **SINTRAN**, the operating system on the *Norwegian NORD Computer Family*. Like in other modules in Thermo-Calc and DICTRA, the commands in GES are often a full sentence and the words are separated by hyphens “-” or underscore “_”. The hyphen and underscore characters are treated as identical. The commands can always be abbreviated until they become ambiguous, and each part between hyphens can be shortened. Trailing parts of a command can be excluded completely if they are unnecessary.

For each command, the program usually needs one or more additional values, and if the user presses the <RETURN> key after the command the program will ask for these values. Usually, the program also suggests a default value as the answer between slashes, *e.g.*, /Y/ or /.50/, and if the user is satisfied with this value he can simply press the <RETURN> key.

An advanced user, who knows the questions that the program will ask, can type the values that should be taken as the answers on the same line as the command. The values must then be separated by spaces. In order to obtain the default value for a question he can type a comma “,” for each value where the default should be used.

For each command the questions that may be asked after the command is also explained. Usually this is the same information as you will receive on-line by typing a question mark, “?”.

There are two commands for help in general. However, the most important help facility is the possibility to type a question mark, “?”, anytime you are asked a question you do not fully understand; usually, the information you will receive on-line by typing a question mark is the same as documented here for each command and sub-prompt. You should then be given a concise description of what the program wants you to give as input. In some cases a more extensive description can be obtained by typing two question marks, “??”.

11.7.1 Modularity and Interactivity

As mentioned above, the GES module is a subroutine package, and it is thus used inside an application program, *e.g.*, the Thermo-Calc software. Normally, the application program will consist of several modules, and there are a standardized set of commands to switch between various modules. Each module of an application program normally has a specific task, and thus several modules may have to be used in order to obtain the desired results.

The GES module will indicate that it is prepared to receive input by issuing the prompt:

GES :

The user must then type a GES command, and after the command has been executed the GES module will write its prompt again.

11.7.2 Use of Control Characters

During the proceeding of a command that generates an output on the screen, it is possible to stop the output by CTRL-S and later to restart it with CTRL-Q. When you receive on-line help on the screen, it may in some cases exceed a full screen. CTRL-S and CTRL-Q can then be used to make it possible to read the output information. In order to interrupt the execution of the program, you can press CTRL-C twice.

11.8 Commands for Help and Information

There are two commands for help in general. However, the most important help facility is the possibility to type a question mark, “?”, anytime you are asked a question you do not fully understand; usually, the information you will receive on-line by typing a question mark is the same as documented here for each command and sub-prompt. You should then be given a concise description of what the program wants you to give as input. In some cases a more extensive description can be obtained by typing two question marks, “??”.

11.8.1 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

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

Specifying a unique command will print an explanation of that command on the screen (usually the same text as found in this Users' 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.

11.8.2 INFORMATION

Description: Basic information about various GES subjects (concepts and models) can be obtained with this command for a number of subjects, as they are described in different parts of this chapter.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: <subject name>

Specify a subject (or its abbreviation as long as it is unique, *e.g.*, SPE, STA, STO, SUB, SUR, SYM, *etc.*) on which information should be given, from the following subjects that are important to the use of the GES module (this list can be seen if typing a question mark “?”):

PURPOSE	HELP	APPLICATION PROGRAMS
USER INTERFACE	MODULARITY	CONTROL CHARACTERS
DATA STRUCTURE	CONSTITUTION	SURFACE OF REFERENCE
EXCESS GIBBS ENERGY	STORED PRIVATE FILES	ENCRYPTED DATABASES
THERMOCHEMISTRY	DEFINING SOME TERMS	UPPER AND LOWER CASE
ELEMENTS	SPECIES	PHASES
FUNCTIONS	SYMBOLS	PARAMETERS
SUBLATTICES	CONSTITUENT ARRAY	MISCIBILITY GAPS
MODELS	STANDARDISED GIBBS ENERGY	IDEAL MODEL
REGULAR MODEL	USE OF CONSTITUENTS	COMPOUND-ENERGY MODEL
IONIC LIQUID MODEL	ASSOCIATE MODEL	QUASICHEMICAL MODEL
NON-CHEMICAL TERMS	FERROMAGNETISM	BORN FUNCTIONS
ORDER-DISORDER	CVM APPROACH	TWO-STATE MODEL
MURNAGHAN MODEL	BIRCH-MURNAGHAN MODEL	GENERALIZED PVT MODEL
AQUEOUS SOLUTIONS	DHLL AND SIT MODELS	HKF AND PITZ MODELS
FLORY-HUGGINS MODEL	GAS EOS AND MIXING	

11.9 Commands to Change Module and to Terminate the Program

11.9.1 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 (also see *Section 5.4.11*).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing Prompt: MODULE NAME:

```
NO SUCH MODULE, USE ANY OF THESE:  
SYSTEM_UTILITIES  
GIBBS_ENERGY_SYSTEM  
TABULATIONREACTION  
POLY_3  
BINARY_DIAGRAM_EASY  
DATABASE_RETRIEVAL  
REACTOR_SIMULATOR_3  
PARROT  
POTENTIAL_DIAGRAM  
SCHEIL_SIMULATION  
POURBAIX_DIAGRAM  
TERNARY_DIAGRAM  
MODULE NAME: <module name>
```

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

11.9.2 BACK

Description: This command switches control back to the most recent module. See also GOTO_MODULE.

Synopsis: BACK

11.9.3 EXIT

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

Synopsis: EXIT

11.10 Commands to Enter Data

Normally, the GES module is used together with a database, and the necessary data are fetched from the database. An advanced user may need to add data to a defined system, and GES thus provides commands to enter the basic entities: *elements, species and phases*. There are also commands to enter auxiliary quantities such as symbols, which can be used in entering thermochemical parameters.

11.10.1 ENTER_ELEMENT

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

- the mass in g/mol,
- 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 (see Section 11.12.6).

If an aqueous solution phase is involved, one should in the GES module enter a special aqueous electron called, ZE. This is specially designed for appropriately calculating the standard electric potential in the aqueous solution system.

Synopsis 1: ENTER_ELEMENT <element name>

Synopsis 2: ENTER_ELEMENT

Ensuing 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 (see Section 11.13.1). Note that the REINITIATE command removes all data, and thus should be executed before any other command.

11.10.2 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.

Synopsis 1: ENTER_SPECIES <species name> <chemical formula>

Synopsis 2: ENTER_SPECIES

Ensuing 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 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_CC12.

STOICHIOMETRY /species name/: <chemical formula>

The stoichiometry 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.

11.10.3 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 TDB database 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 GES 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 GES5 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 GES commands ENTER_PARAMETER and AMEND_PHASE_DESCRIPTION should be used sequentially.

Synopsis: ENTER_PHASE

Ensuing 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 Two-Sublattice 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 GES 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 Substitutional-Sublattice Ordering Model*, the binary interaction between A and B atoms is thus independent of the constituents on the other sublattices, where there are many other parameters to model the composition-dependence of the Gibbs energy (both in the ordered and disordered parts of the phase). The model for these ordered phases are always partitioned in a disordered part (with a single substitutional sublattice) and an ordered part (with 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 B2-ordering 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.*,

$$\begin{aligned} &L(\text{phase}, \text{A}, \text{B}: \text{C}, \text{D}: * : * ; 0 \dots 9) \\ &L(\text{phase}, \text{A}, \text{B}: * : \text{C}, \text{D}: * ; 0 \dots 9). \end{aligned}$$

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:

<u>L</u> 12 or <u>L</u> 10 for ordered FCC,	or <u>A</u> 1 for disordered FCC;
<u>B</u> 2, <u>B</u> 32, <u>D</u> 03 or <u>L</u> 21 for ordered BCC,	or <u>A</u> 2 for disordered BCC;
<u>D</u> 019 and <u>B</u> 19 for ordered HCP,	or <u>A</u> 3 for disordered HCP.

<i>Solution Phase</i>	<i>Suffix for Disordered Phase</i>	<i>Suffix for Ordered Phase</i>
FCC PHASE	PHASE_A1	PHASE_L12
		PHASE_L10
BCC PHASE	PHASE_A2	PHASE_B2
		PHASE_B32
		PHASE_D03
		PHASE_L21
		PHASE_D019
HCP PHASE	PHASE_A3	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 (denoted 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 show 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 (see *Section 11.10.5*). The default answer is 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> (NO), no list will show up.

11.10.4 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 GES module. The symbols entered in the GES module can also be listed and used in the PARROT module (for data optimization) where one may use the GES-entered symbols (variables, functions or parameters) to define parameters that shall be optimized. But symbols entered in the GES 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 (see *Section 11.10.5*). 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.

Synopsis: ENTER_SYMBOL

Ensuing 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. For further information, see *Section 11.2.5 (Functions of Temperature and Pressure)*. 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 (PT-function) 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 (TP-Function) 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 (TP-Function) 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 xxx /yy/: <table value>

The value of the table at the specified temperature (xxx) 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

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 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).*

11.10.5 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 GES workspaces.

The composition-dependence of the Gibbs energy is described in the GES 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 (see *Section 11.12.4*).

As described in *Section 11.4 (Thermodynamic Parameters)*, 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 (see more details as described in *Section 11.4 – Thermodynamic Parameters*):

```
<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 new types of identifiers are available since TCCN/TCW1 and are further expanded since TCCR/TCW4:

- ☞ 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.

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 UPPER-cases is recommended; however, if one would prefer to write it as a mixture of UPPER-case and lower-case, it will then automatically convert all lower-cases to UPPER-cases, as the GES module only recognizes UPPER-case 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, 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 - $\sum y_i$ (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)

$L(\text{phase}, A, *)$ is multiplied with $X(A) * (1-X(A))$
 $L(\text{phase}, B, *)$ 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.

Synopsis: ENTER_PARAMETER

Ensuing 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 UPPER-cases is recommended; however, if one would prefer to write it as a mixture of UPPER-case and lower-case, it will then automatically convert all lower-cases to UPPER-cases, as the GES module only recognizes UPPER-case 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, 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 $[1 - \sum y_{\text{specified constituents}}]$, 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 GES 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 composition-dependent 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 TP-Function. *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.

11.11 Commands to List Data

11.11.1 LIST_DATA

Description: All data for the current system that have been defined (retrieved from database in the TDB module, or directly entered in the GES module) within the current GES5 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 (except for two SGTE-owned databases, namely SSUB3/SSUB4 and SSOL2/SSOL4), since TCCP/TCW2, one may not be able to use this command to list any retrieved data. However, since TCCS/TCW5, 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.

Synopsis: LIST_DATA

Ensuing 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.

11.11.2 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 databases (except for two SGTE-owned databases, namely SSUB3/SSUB4 and SSOL2/SSOL4), since TCCP/TCW2, one may not be able to use this command to list any phase data. However, since TCCS/TCW5, 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.

Synopsis 1: LIST_PHASE_DATA <phase name>

Synopsis 2: LIST_PHASE_DATA

Ensuing Prompt: PHASE NAME: <phase name>

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

11.11.3 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 is (see more details as described in Section 11.4):

<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).

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

- ☞ 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.

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 $[1 - \sum y(\text{specified constituents})]$, which implies that in an A-B binary system the following three L parameters are identical (but in higher-order systems, they are different):

- $L(\text{phase}, A, B)$ is multiplied with $X(A) * X(B)$
- $L(\text{phase}, A, *)$ is multiplied with $X(A) * (1-X(A))$
- $L(\text{phase}, B, *)$ 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 (except for two SGTE-owned databases, namely SSUB3/SSUB4 and SSOL2/SSOL4), since TCCP/TCW2, one may not be able to use this command to list any retrieved parameter.

Synopsis 1: LIST_PARAMETER <identifier>(<phase>, <constituent array>; <digit>)

Synopsis 2: LIST_PARAMETER

Ensuing 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.

11.11.4 LIST_SYMBOL

Description: This command lists on the screen the symbols available in the current GES5 workspace, which are defined (retrieved from database in the TDB module, or directly entered in the GES module) and used in TP-Functions for the entered model parameters for various phases in the defined system.

Note that for encrypted commercial databases (except for two SGTE-owned databases, namely SSUB3 and SSOL2/SSOL4), since TCCP/TCW2, one may not be able to use this command to list any retrieved symbol.

Synopsis: LIST_SYMBOL

Ensuing 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 TDB module, or directly entered in the GES module) for the current system, which are available in the current GES5 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 *e.g.* Notepad, PFE32, Wordpad, Emacs). The default value is the screen (by pressing <RETURN>).

11.11.5 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 GES monitor to set the constitution.

Synopsis: LIST_CONSTITUENT

11.11.6 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.

Synopsis: 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 handled 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

11.12 Commands to Amend Data

11.12.1 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.

Synopsis: AMEND_ELEMENT_DATA

Ensuing 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/: <YYYYYY>

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/: <YYYYYY>

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.

Index value Symbol

0	G
1	H298
2	H0

11.12.2 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.

Since TCCQ, there have been 19 options for a phase can be possibly amended; and since TCCR, the option FRACTION_LIMITS has been added; and since TCCS, the option QUASICHEM_IONIC shall have one input for the number of atoms per atom. 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_RISID
TERNARY_EXTRAPOLAT
HKF_ELECTROSTATIC
DEFAULT_STABLE
SITE RATIOS
FRACTION LIMITS
```

For various amending options, their corresponding ensuing 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 ensuing prompt.
For HKF_ELECTROSTATIC, no ensuing prompt.
For QUASICHEM_IONIC,
    Number of bonds/atom /2/: <number of bonds per atom>
For QUASICHEM_FACT00,  no ensuing prompt.
For QUASICHEM_RISID,   no ensuing prompt.
For GLASS_TRANSITION, no ensuing prompt.
For ZRO2_TRANSITION,  no ensuing prompt.
For REMOVE_ADDITION,  no ensuing prompt.
For STATUS_BITS,
    New status word (in hexadecimal) /04204000/: <new status bits>
For DEFAULT_STABLE,   no ensuing 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 GES 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 GES 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 modeling chemical ordering/disordering phenomenon, `QUASICHEM_IONIC` for using a quasichemical 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 GES 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 post-processing). 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, since TCCR/TCW4, 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 since TCCR/TCW4) 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 Sigworth-Elliott Model (Sigworth and Elliott, 1974) for Fe-rich liquid mixture] can not 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 GES 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 **TCCS Examples Book (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], he 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 (as described/listed in *Section 11.11.6 – LIST_STATUS* or through the GES command **HELP LIST_STATUS**) 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.

Synopsis: AMEND_PHASE_DESCRIPTION

Ensuing 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, as shown below:

AMEND WHAT /COMPOSITION_SET/: ?

```
You can amend
EXCESS_MODEL
MAGNETIC_ORDERING
DEBYE_HUCKEL
STATUS_BITS
NEW_CONSTITUENT
RENAME_PHASE
COMPOSITION_SETS
GLASS_TRANSITION
DISORDERED_PART
MAJOR_CONSTITUENT
ZRO2_TRANSITION
REMOVE_ADDITIONS
QUASICHEM_IONIC
QUASICHEM_FACT00
QUASICHEM_IRSID
TERNARY_EXTRAPOLAT
HKF_ELECTROSTATIC
DEFAULT_STABLE
SITE RATIOS
FRACTION_LIMITS
```

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

For RENAME_PHASE :

If a user wants to add a new constituent to a phase this can be done by this option.

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.

This is also a way to “delete” a phase by hiding it under a new name.

For 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.

For 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.

For 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

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.

For 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.

For FRACTION_LIMITS:

Since TCCR/TCW4, this useful 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 GES 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 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			!

For 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.).

For 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 GES 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 so-called *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 in 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 (see *Section 10.3.10*). 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 x_i are calculated from the site fractions y_i . The last term means that the contribution from G_m^{ord} 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.

For 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 preset 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 (for more details also see *Section 11.3.3 – Regular/Subregular Solution Model*), 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 REDLICH-KISTER_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 new standard *Example 52* in the **TCC Examples Book** (since TCCS) 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). The first sub-option (*i.e.*, “First (the independent) constituent:” will be repeatedly prompted (*again and again*), after having specified the desired binary Excess model type, for further changes of binary excess model for other specific binary pair in the current *substitutional* solution phase; by simply typing <RETURN> at such a repeated 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), 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 (see *Section 11.3.16 – HKF and PITZ Models: for concentrated aqueous solutions*).

The FLORY-HUGGINS POLYMER MODEL is for polymers (see *Section 11.3.18 – Flory-Huggins Model: for polymers*).

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

For 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 Redlich-Kister_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 new standard *Example 52* in the **TCC Examples Book** (*since TCCS*), 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.

For DEBYE_HUCKEL:

If the user wants to use the DHLL (Debye-Hückel Limiting Law) Model for an dilute aqueous solution 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 aqueous phase.

For 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 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.

For 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.

For ZRO2_TRANSITION:

It is not implemented yet.

For 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).

For 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 species and turn on the option QUASICHEM_FACT00. Two examples concerning this quasichemical model can be found in the **TCCS Examples Book (Examples 49 and 50)**.

For QUASICHEM_ISRID :

If the user wants to use the Quasichemical Model developed by ISRID, France, he can use a normal liquid model with specified associates or species and turn on the option QUASICHEM_ISRID.

For 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.

For 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 GES module) have been defined properly!

The only two exceptions are:

- ◆ For the Kapoor-Frohberg-Gaye cell model, which can be set only as 04208000;
- ◆ For the complete Revised HKF aqueous solution model, which can be set as 02084000.

For 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.

11.12.3 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.

Synopsis: AMEND_SYMBOL

Ensuing 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 2.5250000E+01
TABLE VALUE 1.5600000E+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 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
1    298.15<T<2000.00
2    2000.00<T<4500.00
3    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	<#occurrences> F<string>@
Insert	I<string>@
Move	<+-#positions> M
Restore string	R
Substitute	S<OLD>@<NEW>@
Type string	T

where CP denotes the current position in string, # means number of, @ is a terminator of an input or search string.

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.

11.12.4 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, see *Section 11.10.5*.

Synopsis: AMEND_PARAMETER

Ensuing 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
 1 298.15<T<2000.00
 2 2000.00<T<4500.00
 3 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 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	<#occurrences> F<string>@
Insert	I<string>@
Move	<+-#positions> M
Restore string	R
Substitute	S<OLD>@<NEW>@
Type string	T

where CP denotes the current position in string, # means number of, @ is a terminator of an input or search string.

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.

11.12.5 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.

Synopsis 1: CHANGE_STATUS <keyword> <new status> <name1; name2; ...>

Synopsis 2: CHANGE_STATUS

Ensuing 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;

11.12.6 PATCH_WORKSPACES

Description: This command is only for those who really know what they are doing.

Synopsis: PATCH_WORKSPACES

11.12.7 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.

Synopsis: SET_R_AND_P_NORM

Ensuing 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 GES module.

11.13 Commands to Delete Data

11.13.1 REINITIATE

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

Synopsis: REINITIATE

Ensuing 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.

Index value Symbol

0	G
1	H298
2	H0

11.13.2 DELETE

Description: Not implemented yet.

Synopsis: DELETE

11.14 Commands to Store or Read Data

11.14.1 SAVE_GES_WORKSPACE

Description: The data area used by the GES workspace can be saved on a file (with a default extension “.GES5” under Windows, or “.ges5” under UNIX/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.

Synopsis 1: SAVE_GES_WORKSPACE <file name>

Synopsis 2: SAVE_GES_WORKSPACE

Ensuing Prompt: WORKSPACE FILE /ABCD/: <file name>

Under UNIX and Linux, this prompt should be specified with name of the file where the current GES5 workspace is written.

Under Windows Vista/XP/2000/NT4, a “Save As” window as shown in *Figure 11-1* will pop up instead. 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 “Save as type” box is the proper one for the GES5 workspace format (*i.e.*, GES5 file).



Figure 11-1. The “Save As” window: Saving the GES5 workspace file.

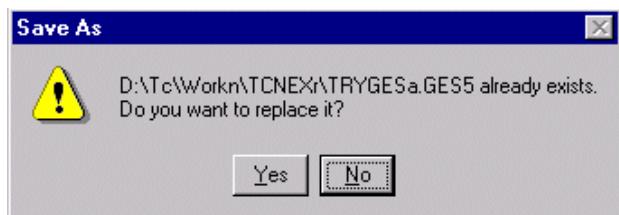


Figure 11-2. The “Save As” warning window: Proceeding or canceling the saving.

If there is already one *.GES5 file with the same name saved in the current working directory, a small screen with a warning message as shown in *Figure 11-2* will pop up. The user needs to either choose the “Yes” button to proceed a replacement of the previous *.GES5 file or the “No” button to cancel the current saving. For the later case, the user can either give another name or change the working directory in order to save the current GES5 workspace onto a file.

11.14.2 READ_GES_WORKSPACE

Description: The data area previously saved on a file by a `SAVE_GES_WORKSPACE` command is read back into the GES workspace. Note that these saved files are unique for each CPU type, therefore one saved on one CPU type can not be used on another CPU type.

Synopsis 1: `READ_GES_WORKSPACE <file name>`

Synopsis 2: `READ_GES_WORKSPACE`

Ensuing Prompt: `WORKSPACE FILE /ABCD/: <file name>`

Under UNIX and Linux, this prompt should be specified with name of the file from where a previously saved GES5 workspace is read.

Under Windows Vista/XP/2000/NT4, a “**Open file**” window as shown in *Figure 11-3* will pop up instead. 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 GES5 workspace format (*i.e.*, GES5 file).



Figure 11-3. The “Open file” window: Opening the GES5 workspace file.

11.15 Other Commands

11.15.1 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.

Synopsis: ADD_COMMENT

11.15.2 SET_INTERACTIVE

Description: This command is useful in demonstration or MACRO files in order to stop execution of the command file in the GES module. It has no meaning in interactive mode.

Synopsis: SET_INTERACTIVE

(This page is intended to be empty)

12 Optimization Module (PARROT)

12.1 Introduction

Any thermodynamic and kinetic software would be useless if without any databases of good quality; therefore, the developments and constructions of thermodynamic and kinetic databases have always been a highly-demanding and challenging task which requires expertise and experience. Since the very beginning of developments of our Thermo-Calc and DICTRA software, tremendous and endless efforts have been being put into the continuous developments of high-quality, internal-consistent, application-oriented and result-reliable thermodynamic and kinetic databases, through a great number of national and international collaboration projects. Furthermore, a powerful tool, namely the **PARROT** program, has been specifically developed for the purposes of data evaluations and critical assessments of thermodynamic/kinetic model parameters and for the establishments of various thermodynamic/kinetic databases, applying the well-known and widely-recognized **Extended Calphad Approach** in the field of Computational Thermodynamics and Kinetics. As a direct result, most of thermodynamic/kinetic databases currently available for materials R&D (which are used within the Thermo-Calc/DICTRA software, as well as within other software in the market) have been developed worldwide, using the PARROT program.

The PARROT program \Leftrightarrow has been become an important, basic and integrated module of both the Thermo-Calc (TCC) and DICTRA software. The PARROT module consists of a comprehensive subroutine package for data evaluation of thermodynamic and kinetic model parameters from experimental observations of quantities describing a set of equilibrium states or dynamic processes in various multicomponent heterogeneous interaction systems. The module is interactively connected with all other basic modules in the TCC and DICTRA software. At present moment, it is not available in the TCW software.

The PARROT module makes use of the GES module (the Gibbs Energy System) to handle the models of the various phases that might form in a multicomponent system, as well as of the POLY module (the Equilibrium Calculation Module) to store and calculate complex heterogeneous equilibria. Before using the PARROT module in Thermo-Calc, one should be familiar with the POLY module and also study the GES module. Many respects about models, parameters, functions, equilibrium states and minimization techniques are explained there (specifically *Chapters 8 & 10*, as well as the *Thermo-Calc Software System* document). In the DICTRA software, the PARROT module also utilizes the DICTRA facility to store and simulate dynamic process properties in diffusion-controlled phase transformations; for details see the **DICTRA User's Guide** and **DICTRA Examples Book**.

The objective of the PARROT module is to provide a unified set of subroutines to be used in critical assessment of various thermochemical and kinetic model parameters against available experimental and literature data. Advanced users of Thermo-Calc and DICTRA are highly encouraged to use this module in the courses of data assessment and database establishment. However, due to its special purposes on critical assessments, this module has not been included in the application programming interfaces, *i.e.*, TQ or TCAPI or TC-MATLAB Toolbox.

Similar to other modules implemented in the Thermo-Calc and DICTRA software, the PARROT module provides the users of application programs with a user interface with a set of simple and general commands for both data editing and parameter optimization. Through such a user interface, one can interactively enter and modify phase descriptions, model connections, basic thermodynamic parameters, and so on. The aim of this

\Leftrightarrow **Revision History of the PARROT Module User's Guide:**

Feb 1984	First release (Edited by Bo Jansson)
Oct 1993	Second revised release (Edited by Birgitta Jönsson and Bo Jansson)
Sept 1998	Third revised release (Edited by Bo Sundman)
Jun 2000	Fourth revised and extended release (Edited by Bo Sundman and Pingfang Shi)
Nov 2002	Fifth revised release (Edited by Bo Sundman and Pingfang Shi)
Jun 2004	Sixth revised release (Edited by Pingfang Shi); with minor changes in 2006
Jun 2008	Seventh revised release (Edited by Pingfang Shi)

User's Guide is to describe this user interface. The data editing task is done through PARROT's sub-module **ED_EXP** by calling the PARROT command **EDIT_EXPERIMENTS** (for more details about the **ED_EXP** module, please refer to *Section 12.7.7* and *Chapter 13*).

Like the POLY and GES modules, the module has its own workspaces with dynamic memories. It transfers continuously all kinds of thermodynamic/kinetic data and calculated equilibrium states or dynamic parameters (*back and forth*) with the POLY3, GES5 and DICTRA workspaces.

Normally, the so-called Global Minimization Technique is turned off during optimization [*i.e.*, the Ordinary POLY Minimization (GEM) is used for equilibrium calculations]; however, it is possible since TCCS to turn on the Global Minimization Technique for (some) individual experimental equilibrium points using the POLY/ED_EXP command-sequence **ADVANCED_OPTION TOGGLE_ALTERNATE**.

The PARROT users are highly recommended and encouraged to read the recently published book of *Computational Thermodynamics – The Calphad Method* by H.L. Lukas, S.G. Fries and B. Sundman (2007, Cambridge). This book gives a very comprehensive overview on various important and practical aspects of Computational Thermodynamics and of the Extended Calphad Approach, such as thermodynamic principles/basis, thermodynamic modeling/definitions, experimental measurements/data, thermodynamic properties/stability, thermodynamic optimizations/assessments, thermodynamic parameters/databases and thermodynamic calculations/applications. Many case studies of critical assessments using the PARROT module are described in the book, as well.

The PARROT users are also highly recommended and encouraged to carefully go through and practice the examples provided along with the standard TCC and DICTRA software installations; such examples include the **TCEX36** and **DICEX_g2**, which are documented in the *TCC Examples Book* and *DICTRA Examples Book*.

The following commands are available in the PARROT module:

PARROT : ?		
AMEND_PARAMETER	LIST_CONDITIONS	SAVE_PARROT_WORKSPACES
BACK	LIST_PARAMETER	SET_ALTERNATE_MODE
COMPILE_EXPERIMENTS	LIST_PHASE_DATA	SET_EXTERNAL_PARAMETER
CONTINUE_OPTIMIZATION	LIST_RESULT	SET_FIX_VARIABLE
CREATE_NEW_STORE_FILE	LIST_STORE_FILE	SET_INTERACTIVE
EDIT_EXPERIMENTS	LIST_SYMBOL_IN_GES	SET_OPTIMIZING_CONDITION
ENTER_PARAMETER	MACRO_FILE_OPEN	SET_OPTIMIZING_VARIABLE
EXIT	OPTIMIZE_VARIABLES	SET_OUTPUT_LEVELS
GOTO_MODULE	READ_PARROT_WORKSPACES	SET_SCALED_VARIABLE
HELP	RECOVER_VARIABLES	SET_STORE_FILE
INFORMATION	REINITIATE	
LIST_ALL_VARIABLES	RESCALE_VARIABLES	
PARROT :		

Note that since TCCS the **SET_EXTERNAL_PARAMETER** command has been added in the PARROT module, for the purpose of optimizations of parameters used in so-called external models (which may not be as fully-implemented/integrated parts inside the GES system of the Thermo-Calc software).

12.1.1 Extended Calphad Approach

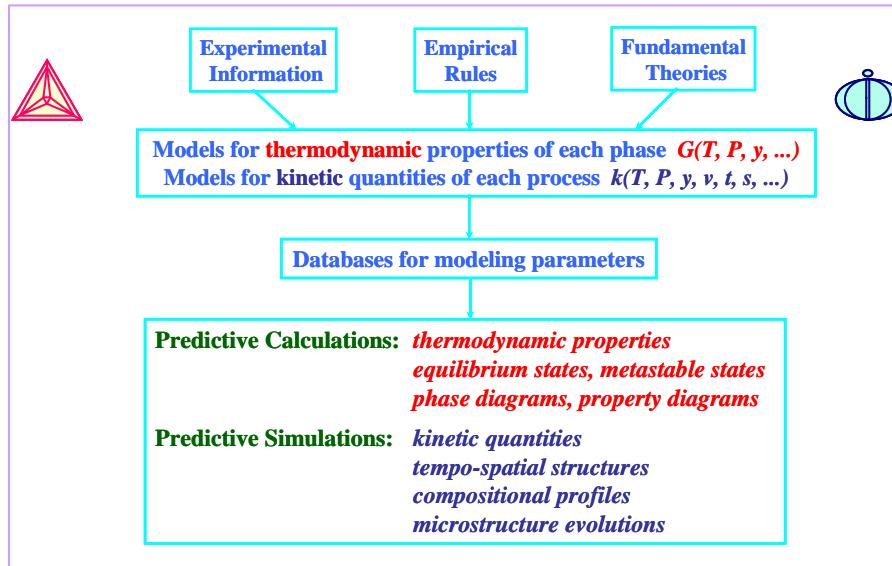
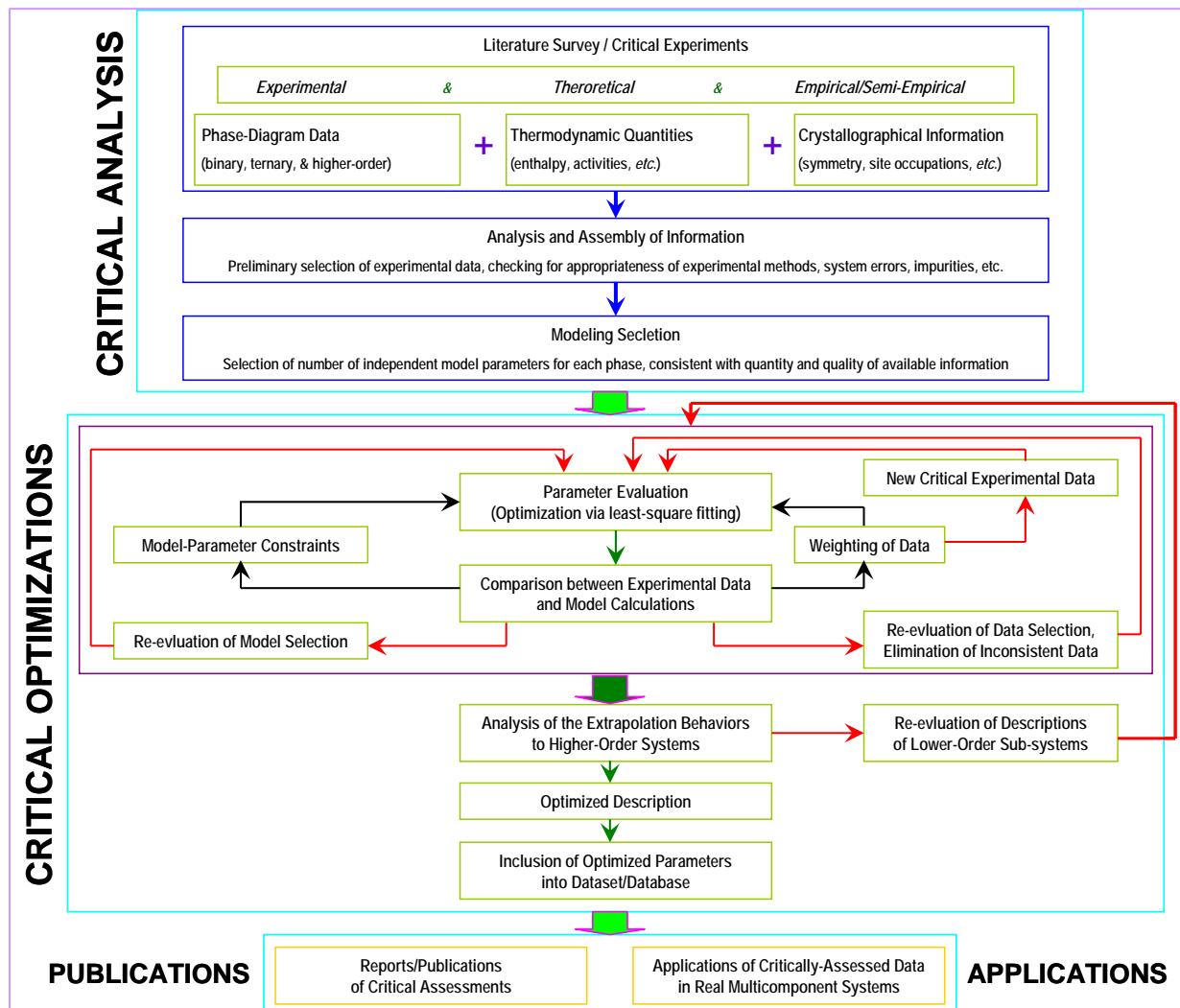
Calphad originally stands for *CALculation of PHAses Diagrams* while nowadays has expanded as to *Computer Coupling of Phase Diagrams and Thermochemistry*. The histories, principles, methodology, software/database developments and applications have been extensively described in the recently published book of *Computational Thermodynamics – The Calphad Method* by H.L. Lukas, S.G. Fries and B. Sundman (2007, Cambridge).

The Calphad approach was first generally described in the book of *Computer Calculations of Phase Diagrams* by L. Kaufman and H. Bernstein (1970), in which the important concept of *lattice stability* (that had been introduced earlier by L. Kaufman, 1959) for calculation of phase diagrams based on modeled Gibbs free energies was developed. They explained clearly how thermodynamic model parameters describing Gibbs free energies in multicomponent systems could be derived both from experimental phase diagrams and from the rudimentary first-principles techniques available at that time, and how they could be used for calculations of phase diagrams. More details of the lattice stability concept can be also read from the recent paper of *Hume-Rothery and Calphad Thermodynamics* by L. Kaufman (2002; in *Calphad and Alloy Thermodynamics*, Eds. P.E. Turchi, A. Gonis and R.D. Shull, Warrendale, PA: TMS, pp. 3-19). The lattice stability concept was essential for the development of multicomponent thermodynamic databases. The method of extrapolating solubility lines into the metastable range to obtain a thermodynamic property (such as the melting temperature of metastable FCC Cr) shows one of the important advantages of combining phase diagrams and thermodynamics.

Along the development histories of the Calphad techniques, of thermodynamic models, and of thermodynamic software, databases and application programming interfaces, more such combinations have been used in exploring various thermochemical and thermophysical properties of many systems using experimental and theoretical data in thermodynamic models, through important contributions made by Kubascheski *et al.* (1967), Hillert and Staffanson (1970), Eriksson (1971), Basebe and Nishizawa (1972), Ansara *et al.* (1973), Hillert and Sundman (1976, 1990), Sundman and Ågren (1981), Sundman *et al.* (1985, 1998, 1999), Fernández Guillermet (1982), Jansson (1983), Hillert *et al.* (1985), Andersson and Sundman (1987), Dinsdale (1991), Chase *et al.* (1995), Dupin (1995), Kattner (1997), Aldinger *et al.* (1995, 1997, 2000, 2001, 2007), Hillert (1997, 1998, 2001), Pelton *et al.* (2000, 2001), Andersson *et al.* (2002), Lukas *et al.* (1977, 2007), Shi *et al.* (2008), among many others.

The **Thermo-Calc software (TCC/TCW)**, together with many high-quality, internal-consistent, application-oriented and result-reliable *thermodynamic databases*, as well as with various *application programming interfaces* (namely **TQ**, **TCAPI** and **TC-MATLAB Toolbox**), represent the most successful developments of the Calphad techniques, which have been globally applied in materials R&D activities for more than 25 years. For various detailed information regarding the developments of the Thermo-Calc software/database/programming-interface package, thermodynamic models implemented in the software, thermodynamic databases used together with the package, and some specific applications utilizing the package, please refer to the documents of *Thermo-Calc Software System*, *Thermo-Calc Database Guide*, *TCC/TCW User's Guides*, *TCC/TCW Examples Books*, *TQ/TCAPI/Toolbox Programming Interface User's Guides and Examples* and *Thermo-Calc Reference Lists* which are also available at <http://www.thermocalc.se/DOWNLOAD AREA/Download.html>.

The Calphad approach has also been significantly extended into the field of *Computational Kinetics* since the early stages of its developments. Nowadays, it has become a powerful method in a wide range of applications where modeled Gibbs free energies and derivatives of heterogeneous systems thereof are used for calculations of phase equilibria and phase properties and for simulations of phase transformations of real multicomponent materials. For instances, chemical potentials and chemical driving forces (*i.e.*, the thermodynamic factors being as the second derivatives of Gibbs free energies) are used in simulations of diffusion-controlled and/or deformation-induced phase transformations in materials processes; chemical driving forces are also used in simulations of materials microstructures based on the Landau theory and phase-field methods; fractions of solid/liquid phases, segregations of components, and energies and chemical driving forces of metastable states are used in simulations of melt/slag solidifications and of precipitations; energies, chemical driving forces and other thermochemical/thermophysical properties are used in computational fluid dynamics (CFD) of materials processing. In other words, wherever/whenever a sophisticated thermodynamic/kinetic descriptions of heterogeneous interaction systems/processes are required, the Extended Calphad Approach can be comprehensively applied. Figures 12-1 and 12-2 illustrate the basic principles and normal scheme of the Extended Calphad Approach.

**Figure 12-1.** The Extended Calphad Approach – The Principles.**Figure 12-2.** The Extended Calphad Approach – The Scheme.

Successful developments have been made particularly on the **DICTRA** software and associated mobility databases which have been applied since 1990 in solving many scientific and engineering problems for simulating diffusion-controlled phase transformations in materials systems/processes (Ågren, 1982; Jönsson, 1992, 1994; Engström et al., 1994; Helander and Ågren, 1999; Ågren and Höglund, 2001; Andersson et al., 1990, 2002; Borgenstam et al., 2002; Gomez-Acebo et al., 2004; Campbell et al., 2004, 2005; Hallstedt et al., 2006; Turchi et al., 2007; Knezevic et al., 2008). For all kinds of detailed information of the **DICTRA** software/database package and its applications, please refer to the documents of *DICTRA User's Guide*, *DICTRA Examples Book* and *DICTRA Reference Lists* which are also available for reviewing/downloading at the web site http://www.thermocalc.se/DOWNLOAD_AREA/Download.html. Figure 12-3 shows the Extended Calphad Approach has been applied in the DICTRA kinetic/mobility assessments and diffusion simulations.

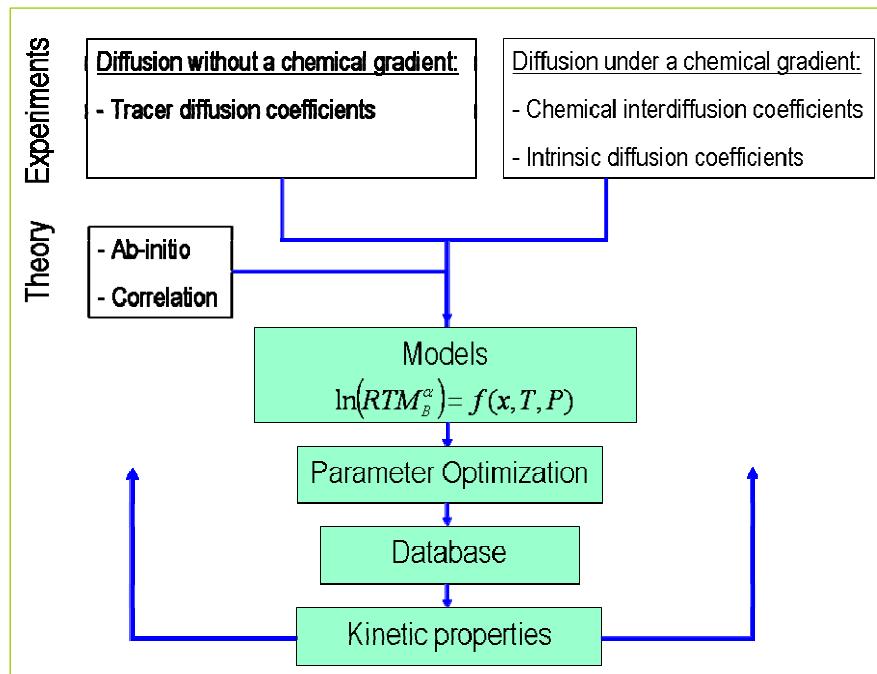


Figure 12-3. The Extended Calphad Approach – DICTRA Kinetic/Mobility Assessments.

Further developments have also been made recently on the **TC-PRISMA** software and associated property databases for simulating precipitations in materials systems/processes (which will be formally released in the fall of 2008).

Other software and associated thermochemical/thermophysical databases have also been successfully developed by some of our R&D partners and been applied in materials R&D by scientific communities and industrial organizations. Distinguished examples are the **MICRESS** software for simulating microstructure evolution in materials (<http://www.micress.de>) and the **PrecipiCalc** software for calculating precipitations in materials (http://www.questek.com/PDF/ppc_techsheet.pdf); both of which utilize the application programming interfaces (TQ and TC API, respectively) for accurate calculations of various necessary standard thermodynamic properties and derivatives based on the Thermo-Calc software and thermodynamic/mobility databases.

12.1.2 Thermodynamic and Kinetic Models

The integral Gibbs energy and derivatives for each phase in a defined system depends on temperature, pressure and compositions, and this is described by a thermodynamic model and expressed in a mathematical equation in the Thermo-Calc software system, in which a large number of thermodynamic models for various phases in different states (as listed in *Table 12-1*) have been implemented (into the GES module, as well as into the POLY, PARROT and ED_EXP modules) and have been used in widely-ranging applications. The dependence on the constitution is usually the most complicated to find a good model for. However, the dependence on temperature and pressure can be also very complicated for some phases (such as aqueous solutions).

Table 12-1. List of a wide spectrum of thermodynamic models implemented in the Thermo-Calc software

Model Name in GES	Full Model Name	Applicable Phases
Ideal	Ideal Substitutional Model	Ideal gases
Regular	Regular Solution Model	Binary alloys
Redlich-Kister	Redlich-Kister (<i>Subregular</i>) Model	Binary alloys
Polynom	Simple Polynomial Model	Binary phase with no sublattice
Legendre	Legendre Polynomial Model	Binary phase with no sublattice
Redlich-Kister_Muggiana	Muggiana Ternary Excess Model (based on R-K binary parameters)	Ternary or higher order alloys
Redlich-Kister_Kohler	Kohler Ternary Excess Model (based on R-K binary parameters)	Ternary or higher order alloys
Muggiana_Restor	Muggiana Ternary Extrapolation Model	Ternary or higher order alloys
Kohler-All	Kohler Ternary Extrapolation Model	Ternary or higher order alloys
Toop-Kohler	Toop-Kohler Ternary Extrapolation Model	Ternary or higher order alloys
CEF	Compound-Energy-Formulism (Sublattice Model)	Alloys, Liquids, Gases, Oxides, ...
ITSLM	Ionic Two-Sublattice Liquid Model	Liquid slag
Quasichem_Ionic	Quasichemical Model of ionic liquid	Liquid slag
Quasichem_FACT00	F*A*C*T Quasichemical Model for liquid	Liquid slag
Quasichem_IRSID	IRSID (Kapoor-Frohberg-Gaye Cell) Model for liquid	Liquid slag
AM	Associate Model for liquid	Liquid slag
CAM	Central Atom Model (<i>not fully implemented yet</i>)	Liquid slag
SEHM	Sigworth-Elliott_Hillert Model for dilute liquid	Dilute Fe-rich liquid
G2SLM	Generalized Two-State Liquid Model	Amorphous phases
MO	Inden Magnetic Ordering Model	Alloys
CVM	Chemical Ordering via CVM Approach	Alloys
Murnaghan	Murnaghan Model	High-PT minerals/alloys
Birch-Murnaghan	Birch-Murnaghan Model	Extra-high-PT minerals
GPVT	Generalized PVT Model	Alloys, Liquids, Minerals, etc.
SUPERFLUID	SUPERFLUID Model	Real gases & gaseous mixture
DHLL	Debye-Hückel Limiting Law	Dilute aqueous solutions
SIT	Specific Ionic Interaction Theory	Dilute aqueous solutions
PITZ	Generalized Pitzer's Formalism	Concentrated aqueous solutions
HKF	Revised Helgeson-Kirkham-Flowers Model	Concentrated aqueous solutions
FHP	Flory-Huggins Polymer Model	Polymers

For more details about available thermodynamic models used within the Thermo-Calc software (as well as the application programming interfaces), please refer to *Section 11.3 (Thermodynamic Models)* in *Chapter 11 (GES: the Gibbs Energy System Module)* and the *Thermo-Calc Software System* document, as well as to the online help of the GES module.

In the last and current major-versions of application programming interfaces [TQ6/TQ7, TCAPI4/TCAPI5 and Toolbox4/Toolbox5, that correspond to the major-versions (namely TCCR/TCW4 and TCCS/TCW5) of the Thermo-Calc software], the GES system has been extended to include kinetic models for handling diffusion equations that correspond to diffusion-controlled phase transformation problems treated by the DICTRA software and mobility databases. As a result, one can calculate various types of diffusion coefficients (*e.g.*, chemical inter-diffusion coefficients, intrinsic diffusion coefficients, tracer/dilute diffusion coefficients, self-diffusion coefficients, impurity diffusion coefficients) in accordance with chemical driving forces and atomic mobility parameters, directly in such application programming interfaces (but not inside TCCR/TCW4 and TCCS/TCW5). For details about various kinetic models for solving complex diffusion equations, please refer to the *DICTRA User's Guide* and *DICTRA Examples Book*.

For an ordinary user of the Thermo-Calc software (TCC/TCW), the existing databases will select the appropriate thermodynamic models for each of the included phases in a defined system. For an advanced and experienced user who has the expertise and wishes to conduct data evaluations and critical assessments for some specific systems using the PARROT module inside the TCC (and DICTRA) software, he/she can select specific models for certain phases.

Please note that most of the available thermodynamic models listed in *Table 12-1* can (and should) be appropriately called for the purposes of optimizing model parameters of certain phases. However, there are some specific models (such as SUPERFLUID and CAM) that have to be partially or entirely programmed into the source codes of the GES system, which makes it rather difficult to use them for further data assessments. In order to find the proper solutions for the usage of such models in data assessments, the new PARROT command `SET_EXTERNAL_PARAMETER` has been introduced since TCCS, which requires the user to first write his/her own codes for a preferred model (that will be referred as an external model as it is not within the standard GES 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 it as an 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.

12.1.3 Experimental Information

An effective thermodynamic calculation and kinetic simulation must always be based on good thermodynamic/kinetic models and databases, a good thermodynamic/kinetic database (with optimized model parameters for various simple and complex phases) has to always be based on a tremendous amount of critical assessment work, and a critical assessment of thermodynamic/kinetic model parameters ought to always be based on a wide range of reliable experimental results, literature information, empirical/semi-empirical correlations and fundamental theories.

With expertise and experience, almost all kinds of thermochemical/thermophysical data obtainable from various types of experimental measurements can be utilized all together during the process of a critical assessment of thermodynamic/kinetic model parameters for a specific system, using the PARROT module within the TCC and DICTRA software.

The following list summarizes various types of experimental results and other information that are suitable for considerations in critical assessments using the PARROT module:

- Experimental techniques used in measurements;
- Crystal structures, crystallographic data, lattice parameters, and defect/ordering/site-occupancy information (determined by X-ray, neutron diffraction, Mössbauer spectroscopy, and perturbed angular-correlation measurements);
- PVT-EOS data (molar volume, density, thermal expansion, bulk-modulus/compressibility, etc.);
- Calorimetric data for mixing and reactions (including data for enthalpy of mixing of liquids, series of mixing-enthalpy, partial enthalpy, enthalpy of congruent phase transformation, direct-reaction calorimetry, solution calorimetry, combustion calorimetry, and shape of H(X) curve);
- Calorimetric data for phase transformations (including data for drop calorimetry and scanning calorimetry);
- Heat capacity, enthalpy, entropy and Gibbs free energy of phase formations;
- Changes of enthalpy, entropy and Gibbs free energy of chemical reactions;
- Chemical potentials and activities/activity coefficients via EMF measurements of reversible galvanic cells, of vapor pressures (through direct pressure measurements, optical spectroscopy of vapor phase, gas-transport measurements, isopiestic measurements, dew-point measurements, Knudsen cell measurements), of equilibria with a well-defined gaseous mixture, and of solubility on a nearly inert solvent (treated as a Henrian solution);
- Phase-diagram data for binary, ternary, quaternary and higher-order sub-systems (in the P-T-X spaces), including DTA (differential thermal analysis) measurements, temperature-dependence and composition-dependence of enthalpy and other properties (e.g., lattice parameter, dilatometric length, electronic conductivity, magnetic susceptibility, among others), reaction types (invariant/monovariant) and phase relations (primary/secondary solidifications, allotropic/eutectic/eutectoid/peritectic/peritectoid/monotectic/metatetic/syntetic...) through qualitative/quantitative metallography [observed by optical micrography, electron micrography and scanning micrography], phase amounts/compositions through microprobe measurements [obtained by X-ray spectroscopy, TEM (transmission electron microscopy)], positions/directions of two-phase tie-lines and three-phase equilibria for ternary systems;
- Atomistic properties (such as Debye temperatures, Curie temperatures and magnetism);

- Cooling paths, frozen ranges, heat transformations, volume/density changes and segregations during solidification processes of metallic alloy melts and/or slag;
- Partition coefficients of certain components or species between two interacted/equilibrated phases, obtained from varied chemical analysis;
- Speciation, solubility, solubility products, solubility limits, activities/activity-coefficients, Eh-pH relations in aqueous electrolyte solutions;
- Diffusion-controlled composition-profiles measured by various experimental means;
- Chemical driving forces for metastable phases;
- Consistent information from first-principle calculations and other theoretical investigations;
- Consistent empirical/semi-empirical correlations;
- (If available) Uncertainties of various experimental data (as described above).

12.1.4 Thermodynamic Databases

The critical assessment of thermodynamic model parameters to describe real systems have already proved to be of industrial interest in understanding material properties, structures and processes. This is because a large effort has been spent on determining the thermodynamic model parameters in all binary and many ternary and some higher order systems with various models.

From such models and parameters, one may then predict the stable states for multicomponent systems. Thermodynamic databases with up to 10-15 elements can accurately predict the stable states of some industrial metallic systems, like steels/Fe-alloys, Al alloys, Ti alloys and Ni-based superalloys. Moreover, critical assessment work under long-term international collaborations have made it possible to establish some large-scale, internally-consistent thermodynamic databases for many kinds of applications, such as the SGTE Pure Substances Database (SSUB) and SGTE Alloy Solutions Database (SSOL), both of which include as many as 83 elements at present.

One can summarize the advantages of critically assessing thermodynamic databases compared to fitting different quantities separately as follows:

- ◆ Unique way to combine and reduce a large amount of experimental data to a few model parameters.
- ◆ Consistent assessment of experimental thermochemical and phase diagram data.
- ◆ Reliable interpolation and extrapolation of thermodynamic data.
- ◆ Useful tool for planning new experimental work.
- ◆ Basis for modeling and simulation of phase transformations.

The last point above is perhaps the most important. The understanding of many dynamic processes is often very poor, and the sensitivity to changes in input and operating conditions must often be determined by trial and error. Thermodynamics provide information not only about the stable state the system is trying to reach, but also about chemical potentials and driving forces for the metastable states. This is essential in the kinetic modeling and simulation of complex material processes.

12.1.5 Optimization Method

For equilibria where all independent state variables are determined with negligible inaccuracy, the criteria for the best fit, derived from the *maximum likelihood principle*, will be minimum in the sum of square of weighted residuals. Inaccuracies in experimental conditions can be taken into account in two ways in the PARROT module:

- a) The inaccuracies in conditions, *i.e.*, independent state variables, can be prescribed in the POLY interface. In this case, an equilibrium will be calculated with the experimental values of independent state variables. The standard deviations of the dependent state variables will be calculated by the error propagation law presuming linear dependency of the dependent state variables on the independent state variables.
- b) The “true” value of the condition can be optimized by using one of the defined variables as the condition. This can be obtained by the IMPORT command in the experimental data file. In this case,

the experimental observations of the independent state variable should be specified in the EXPERIMENT command in experimental data file. The commands that can be used in the experimental data file are a subset of the commands available in the POLY module with a few extensions, and are detailed in the *Chapter 13 (ED_EXP: the Edit-Experiment Module)*.

Both methods can be transformed to the problem of finding the minimum of the sum of squares. Method b) can be used when several experiments have been performed under the same badly determined condition. Both methods can be mixed in the same optimization run.

The set of variables that give a minimum in the sum of squares is found by subroutine VA05A from the Harwell Subroutine Library.

Please note that since TCCS it is possible to turn Global Minimization Technique on for (some) individual experimental equilibrium points during optimizations using the POLY/ED_EXP command-sequence ADVANCED-OPTION TOGGLE-ALTERNATE, while the Global Minimization Technique is normally turned off and thus the Ordinary POLY Minimization (GEM) is used for equilibrium calculations during optimizations.

12.1.6 Requirements for Using the Least Square Method

The least square method can provide a good estimate of the model parameters, provided that:

- ◆ The randomness of the observed quantities has a Gaussian probability distribution.
- ◆ The observed quantities are only subject to random errors.
- ◆ The different experiments are uncorrelated
- ◆ The standard deviation of each observation can be estimated.
- ◆ The number of observations is large.
- ◆ The considered models are exact.

Almost none of these requirements is fulfilled in a normal thermodynamic assessment. But as there is no any other better method available, one can only try to approach as close as possible.

12.1.7 Other Optimization Software

The classical program for optimization of binary systems was developed by Dr. Lukas and is known as **BINGSS**. In order to be compatible with this software, a number of utility programs to convert from BINGSS to PARROT has been developed:

- c2g The c2g (coe-to-ges) program transforms a BINGSS “COE” file to a setup file for PARROT entering all elements, phases and parameters.
- d2p The d2p (dat-to-pop) program transforms a BINGSS “DAT” file with experimental data to a “POP” file.
- t2c The t2c (tdb-to-coe) program transforms a Thermo-Calc “TDB” file to an “COE” file for BINGSS. This is a way to extract parameters for the pure elements from the Thermo-Calc database for use in BINGSS.

12.2 Get Started

This is a user's guide and not an instruction to do an assessment, but a brief outline of the different steps to perform the assessment will be given here. For details of some commands mentioned here, please refer to the corresponding sections documented in the present and next parts.

There is an example, TCEX36, of an assessment of a fictitious system provided with the Thermo-Calc software/database package, and described in the *TCCS Examples Book*. This should be studied carefully. Other published assessments should also be studied, in particular to learn about thermodynamic/kinetic models used in such data evaluations/assessments.

12.2.1 Experimental data file: the POP file

The assessor should first collect the available experimental results and literature data on the selected system, and enter all experimental data on a file, called an *.POP file (because the default extension is .POP). The experimental equilibria and measurements are described by POLY commands, with some additional features. The commands that are legal in an *.POP file are described in a special part of *Chapter 13 (ED_EXP: the Edit-Experiment Module)*.

The *.POP file is very important as it describes the known experimental data for a system. Quite often the same system must be reassessed some years later when new information is available, or if a model for a phase should be changed. As the reassessment may be done by someone else, it is important that the information in the POP file is well organized and documented. It should be readable to a human as well as to the program.

The syntax of the *.POP file is intended to be self documenting and also independent of the models used for describing the phases in the system.

The recommended way to specify an experimental equilibrium is to be as close as possible to the actual experimental conditions. Usually, the sets of stable phases are known, and also temperature, pressure and some or all compositions.

12.2.1.1 Single-phase equilibrium

Experiments with a single stable phase most often concern enthalpies of mixing or chemical potentials. As an example, this describe an experiment in the Au-Cu system:

```
CREATE-NEW-EQUILIBRIUM 1 1
CHANGE-STATUS LIQUID=FIX 1
SET-CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563
SET-REFERENCE-STATE AU LIQ * 1E5
SET-REFERENCE-STATE CU LIQ * 1E5
COMMENT Measuremnt by Topor and Kleppa, Met trans 1984
LABEL ALH
EXPERIMENT HMR=-1520:200
```

The first five commands in this example are standard POLY commands and described in the POLY user's guide (*Chapter 8*). But the first command, CREATE, is rarely used in the POLY module, and may deserve some comments. Each experimental equilibrium must start with the CREATE command, and the first integer given after the command is a unique identifier which is used later interactively to set weights for example. The second integer is an initialization code, for which 0 means that all components and phases are initially suspended, 1 means that all components are entered but all phases suspended, 2 means all components and phase are initially entered.

The last three commands, COMMENT, LABEL and EXPERIMENT, are only available for the experimental data file and in the ED_EXP module. Such commands are described in detail in *Chpater 13 (ED_EXP: the Edit-Experiment Module)*.

The EXPERIMENT command specifies the quantity to be fitted by the optimization. The syntax of this command is similar to the POLY command SET-CONDITION. It is followed by a state variable or a function name and a value and an uncertainty.

The COMMENT command is followed by a text that will be saved in the work file of the optimization. One may also give comments after a dollar sign, “\$”, but these comments are lost when the experimental data file is compiled (compilation will be described below).

The LABEL command provides a way to specify a set of equilibria that the user wants to treat as one entity when setting weights. A label is maximum four characters and must start with the letter “A”.

12.2.1.2 Two-phase equilibrium

Most experimental information, for example from the phase diagram, involves two or more phases. For example, the melting temperature of an Au-Cu alloy can be described as follows:

```
CREATE-NEW-EQUILIBRIUM 1 1
CHANGE-STATUS PHASE LIQUID FCC=FIX 1
SET-CONDITION X(FCC,CU)=0.14 P=1E5
EXPERIMENT T=970:2
COMMENT H E Bennet, J Inst of Metals 1962
LABEL ALS
SET-ALTERNATE-CONDITION X(LIQUID,CU)=0.16
```

All commands except the last was described above. The last command specifies an estimated value of the liquid composition at the equilibrium. Actually it is not necessary, except for using this experimental equilibrium in the alternate mode calculation. The alternate mode is explained in more detail below.

Note that the experiment in this case was the temperature. One could equally well describe the same melting point with the temperature as CONDITION, and the composition as EXPERIMENT, because both are measured quantities. The selection of quantities used as CONDITIONS should be based on the experimental technique. Those known with the least accuracy should be used as EXPERIEMENTS.

12.2.1.3 Invariant equilibria experiments

It is a peculiarity of the PARROT module that invariant equilibria are the most important experimental information to be provided for an assessment. It is thus recommended to have all invariant equilibria for a system on the POP file, even if some of them may not have been explicitly measured. A reasonable estimate from the available experimental data can often be sufficient. But one should be careful using a topographically drawn phase diagram where there is little data to limit the imagination of the artist. At the end of the assessment, such estimated equilibria should be excluded, but they are very useful to obtain a set of start values for the model parameters.

An example of a three phase equilibrium in a binary system (*i.e.*, presenting an invariant equilibrium) is given below:

```
CREATE-NEW-EQUILIBRIUM 1 1
CHANGE-STATUS PHASE FCC BCC LIQUID=FIX 1
SET-COND P=1E5
EXPERIMENT T=912:5
SET-ALTERNATE-CONDITION X(FCC,B)=0.1 X(BCC,B)=0.4 X(LIQ,B)=0.2
LABEL AINV
COMMENT Estimated compositions
```

The alternate conditions are needed to use this experiment for alternate mode calculations. Another example of an invariant equilibrium is a congruent transformation:

```
CREATE-NEW-EQUILIBRIUM 1 1
CHANGE-STATUS PHASE BCC LIQUID=FIX 1
SET-COND P=1E5 X(BCC,B)-X(LIQ,B)=0
EXPERIMENT T=1213:10
SET-ALTERNATE-CONDITION X(BCC,B)=0.52 X(LIQ,B)=0.52
LABEL AINV
COMMENT Estimated compositions
```

With some experience from phase diagram evaluation, it is possible to estimate metastable invariant equilibria. In particular such estimated metastable equilibria are useful to reduce the number of phases that are assessed simultaneously. One may, for example, assume that a certain intermediate phase does not form and extrapolate the liquidus curves below the stable three-phase equilibria and estimate temperatures and compositions of metastable three-phase equilibria between two other phases and the liquid.

Another useful technique is to extrapolate a liquidus line from a peritectic equilibria to estimate the congruent melting temperature of a compound. This may be more useful than the peritectic equilibria itself as it involves only two phases.

12.2.1.4 Ternary and higher order experiments

The PARROT module can handle optimization of ternary or higher order information in the same way as of binary. The only difference is that one more condition is needed for each component added.

In practice, quaternary and higher order information are used mainly to optimize binary or ternary parameters. In ternary systems, it may be more important to use the feature that one may also have uncertainties on conditions. A tie-line in a binary system is determined, if the two phases, the temperature and pressure are known, and the composition of one of the phases is measured. For a tie-line in a ternary system, one must have measured at least two compositions and often both have the same uncertainty. One may then assign an uncertainty to the composition selected as condition. For example,

```
CREATE-NEW-EQUILIBRIUM 1 1
CHANGE-STATUS FCC BCC=FIX 1
SET-CONDITION T=1273 P=1E5 X(FCC,B)=0.1:0.02
EXPERIMENT X(FCC,C)=0.12:.02
LABEL AFB
SET-ALTERNATE-COND X(BCC,B)=0.17 X(BCC,C)=0.07
```

The alternate conditions will be explained below.

One problem with binary assessments is that the experimental information can often be described almost equally well by very different sets of model parameters. It is often the extrapolation of these assessments into ternary systems that gives decisive information about which set of model parameters that is the best. Sometimes, information from several ternary systems may be needed to decide the best description of a binary system.

12.2.1.5 Simultaneous use of binary and ternary experiments

The PARROT module allows simultaneous optimization of binary and ternary (and higher order) information. By using the CHANGE_STATUS command with COMPONENT as the key word, one may have experimental data from binary and ternary systems on the same POP file.

Note that it is not a good technique to set the fraction of a third component to zero for a binary experimental information; the command CHANGE_STATUS COMPONENT C=SUS is fragile. However, it may need manual setting to work properly. An example of a binary (A-B) three-phase equilibrium (FCC-BCC-LIQ) in a ternary system (A-B-C) is:

```
CREATE-NEW-EQUILIBRIUM 1 0
CHANGE-STATUS COMPONENT A B = ENTERED
CHANGE-STATUS PHASE FCC BCC LIQ=FIX 1
SET-COND P=1E5
EXPERIMENT T=1177:10
LABEL AAB
COMMENT from A-B
```

Note the use of initialization code 0 in the CREATE-NEW-EQUILIBRIUM command. This means that all components must be entered.

12.2.2 Graphical experimental file: the EXP file

For plotting a calculated diagram together with experimental data, it is recommended that the experimental data is also written on a file according to the “DATAPOLT” graphical format. This is described in the “**DATAPLOT User’s Guide and Examples**” document (previously Chapter 15 of this User’s Guide). In some cases, tables in the POP file can be easily converted to the DATAPLOT format. The default extension of a DATAPLOT file is “.EXP” (under Windows) or “.exp” (under UNIX/Linux).

12.2.3 System definition file: the SETUP file

The second step is to create a “SETUP” file, which consists of definitions on the names of the elements, species and phases, the models for the phases, and all known information like “lattice stabilities” or Gibbs energies of formation.

Most values for the pure elements can be found in the collection made by Scientific Group Thermodata Europe (SGTE) and published by Alan Dinsdale in the CALPHAD Journal (1991, 317-425). These can also be extracted from the SGTE Pure Elements Database distributed freely with the Thermo-Calc package. In the GES module there is a command LIST-DATA with option P that can be used to create a template setup file after extracting the data from the PURE database. This template must be edited, and new phases and parameters must be added. The default extension of a SETUP file is “.TCM”, this is short for the *Thermo-Calc MACRO file*.

At the end of a SETUP (TCM) file, the use of the PARROT command CREATE-NEW-STORE-FILE will create a work file for the assessed system. See Section 12.2.4 for details of work files.

12.2.3.1 Models for phases

The literature with experimental data collected for the POP file usually contains information useful for modeling the phases. A survey of models useful to describe the thermodynamics of various phases can be found in the report from the *Second Ringberg Workshop*, published in the CALPHAD Journal (1997, 139-285). Most of the currently used models for crystalline solids can be described by the *Compound Energy Formalism* (CEF), based on the sublattice model described by Sundman and Ågren (J Phys Chem Solids, 1981, 297-301).

In order to make an assessment compatible with an existing database, the models for most solution phases often have to be taken from the database. For inter-metallic phases, it may be important to determine if a phase has the same structure as the phase in another system.

The following criteria should be considered for model selection:

- ◆ physical soundness,
- ◆ as few parameters as possible that must be optimized,
- ◆ reasonable extrapolations of the model,
- ◆ consistency with previous assessments.

12.2.3.2 Model parameters

The PARROT module reserves 99 variables for optimization, and can handle 1000 experimental measurements in a single assessment system. But there are limits on the simultaneous numbers of variables and experiments. At each optimization, the program will list these limits.

The variables are called V1 to V99, and they are used when entering functions and parameters to be optimized. For example,

```
ENTER-PARAM L(LIQUID,AU,CU;0) 298.15 V1+V2*T; 6000 N
ENTER-PARAM L(LIQUID,AU,CU;1) 298.15 V3+V4*T; 6000 N
ENTER-PARAM L(LIQUID,AU,CU;2) 298.15 V5+V6*T; 6000 N
```

makes it possible to use 3 Redlich-Kister coefficients, each linearly temperature dependent, to express the Gibbs energy for a liquid solution phase. The variables V1 to V6 can then be optimized to describe the experimental

information. In the setup file, one often introduces more variables than will be needed, as it is convenient to have them in sequential order for each phase.

In some cases, a model requires that several thermodynamic parameters are related, and this can be conveniently handled by using the same variables for several parameters. For example, the parameters for an B2-ordered phase can be described as follows:

```
ENTER-FUNCTION GAB 298.15 V10+V11*T; 6000 N
ENTER-PARAMETER G(B2,A:B) 298.15 GAB; 6000 N
ENTER-PARAMETER G(B2,B:A) 298.15 GAB; 6000 N
```

A stoichiometric compound with measured heat capacity data may require several variables to describe its temperature dependence, for example,

```
ENTER-PARAMETER G(MG2SI,MG:SI) 298.15 V20+V21*T+V22*T*LN(T)+ V23*T**(-1)
+V24*T**2+V25*T**3; 6000 N
```

It is possible to optimize all kinds of parameters in the GES systems that can be described as functions of temperature, pressure and composition. Examples are: standard enthalpy of formation, entropy, thermal capacity, molar volume, thermal expansivity, compressibility, Curie temperature for magnetic transformations, Born function, *etc.*

12.2.4 Work file or store file: the PAR file

As mentioned above in *Section 12.2.3*, at the end of a SETUP (TCM) file, the use of the PARROT command CREATE-NEW-STORE-FILE will create a work file (or called store file) for the assessed system. Such a work file can also be interactively created in the PARROT module by the same command CREATE-NEW-STORE-FILE. All current data on the definitions of the system (elements, species, components, phases), symbols (constants, variables, functions, tables), parameters, etc., which are stored in the workspace used by the GES, POLY and PARROT modules, but not any experimental information, will be automatically saved on such a work file.

The work file has a default extension “.PAR” (under Windows and UNIX/LINUX). Note that such work files cannot be edited directly by any editor. Moreover, they are hardware dependent (meaning that they are unique for each CPU type, therefore a work file saved on one CPU type can not be used on another CPU type).

12.2.5 Various file names and their relations

As all files have different extensions, it is possible to use the same name for all files for an assessment. Thus for the Au-Cu system, one may have aucu.POP, aucu.EXP, aucu.TCM and aucu.PAR. In particular, the work file (PAR) may exist in several copies during the assessment. But it is recommended to update the text files, POP, TCM and EXP, to reflect all changes made interactively in the PAR file.

The text files, *i.e.*, the *SETUP file, the *.POP file, the *.TCM files and the *.EXP file, are important documentation of the assessment. At the end of the assessment, these should be updated in such a way that it is possible to directly run the *SETUP file, to compile the experiments and to make optimization, so that it is fast to get the final results. This requires that the final weights are entered in the *.POP file and the final parameters set as start values of the parameters. With such a set of files, it is easier to reassess the system using new experimental data or new models.

Some care must also be taken of the work file (*.PAR). This always contains the last optimized set of parameters and the weights of the selected experiments. The work file contains a workspace for POLY3 and for GES5. When a diagram is calculated from the current work file, an *.POLY3 file is created. This *.POLY3 file will have a copy of the current set of parameters. If some further optimization is done, and the user then by accident tries to read the old *.POLY3 file, he may destroy the new set of parameters and overwrite them by the old set. Thus one MUST NEVER READ an *.POLY3 or *.GES5 file while running PARROT. But one may of course SAVE new *.POLY3 or *.GES5 files, for example, when calculating diagrams from the current set of parameters.

12.2.6 Interactive running of the PARROT module

With the three files, POP, SETUP and EXP, the user can start running PARROT interactively. This can be divided into some initial separate steps. Usually, these steps have to be repeated, by cyclically modifying weights, modifying models, adding new information, *etc.* It is actually difficult to decide when an assessment is finished. It is quite often that the deadline for the publication sets the limit for the work.

The commands on the SETUP file are executed by the MACRO command. There is usually a number of error messages, and the SETUP file must be corrected and rerun until there are no errors. In the PARROT module, one can list interactively the descriptions of the phases, the parameter expressions and values of the optimizing variables. After a successful run of the SETUP file, one should check that if all models and parameters have been entered correctly.

12.2.6.1 Compilation of experiments

The next step is to “compile” the experimental data file (POP). The command to do this is COMPILE. This compilation will usually also result in a number of error messages due to syntax errors. The compilation normally stops when it detects an error and gives an understandable message. Such syntax errors must be corrected in the POP file, and the file must be compiled again. It is convenient to use several windows for this, one for editing and one for compiling. Sometimes an error message is less understandable, and the error may have occurred some lines before the program actually discovers it. Consultation with an expert is usually the best way to quickly correct these kinds of problems, as it can be difficult to find the right place to look in the manual. As the SETUP file and the POP file are text files, they can be easily E-mailed to PARROT experts anywhere around the world.

12.2.6.2 Setting the Alternate Mode

When the experimental data file has been compiled correctly, there comes the first really big problem. This is to try to calculate the experimental information from the models for the phases in the system. Initially, all model parameters are zero, and in many cases it may not be possible to calculate a measured value from the model unless the parameters have some reasonable values.

Recently, this problem has been simplified in the PARROT module by introducing the *Alternate Mode*. The command SET-ALTERNATE-MODE YES means that experimental equilibria involving two or more phases are calculated with an approximate technique (which will be described in a separate section below). The user may make a selection of the experimental information to be used by setting weights in the ED_EXP module.

The OPTIMIZE command in the PARROT module is then used with the alternate mode until it has converged. Several OPTIMIZE commands are usually needed, and maybe the user will have to change the selection of experimental data. This is again done in the ED_EXP module. The result of an optimization is obtained in a readable form for a human with the command LIST-RESULT. The work file (PAR) is continuously updated and always contains the last set of optimized variables and calculated results. Sometimes the user may be interested in saving a copy of the current work file when trying various selections of experimental data or models. This is done by making a copy of the work file and give the copy another name.

12.2.6.3 Plotting intermediate results

Reading the output from the LIST-RESULT command is usually not enough to understand how good (or bad) fit one has obtained. As the PARROT module is a part of Thermo-Calc package, it is possible to use the POLY/POST modules to directly calculate the phase diagrams and other diagrams with thermodynamic properties, and to plot them together with the experimental data saved in the EXP file. As this is often done very frequently, it is convenient to have the command sequence for such calculations and plotting on MACRO files (TCM). Calculating and plotting of the phase diagram may give many surprising results when the variables to be optimized are far from their final values!

12.2.6.4 Selection of experimental data

Optimization with the alternate mode gives a set of start values for the parameters. When this has converged, the user should turn off the alternate mode and calculate all equilibria using the normal mode. This can be done in the ED_EXP module with the CALCULATE-ALL command. Several experimental equilibria may still fail to converge, and the user may have to provide manual start values or even remove some equilibrium points (by setting their weight to zero). At a later stage in the assessment, when the optimizing variables are closer to their final values, the user may be able to restore and calculate all experimental equilibria.

When the selected set of equilibria can be calculated in the ED_EXP module, the user should calculate all experimental points once again in the PARROT module using the OPTIMIZE command with only 0 iterations. The output from the LIST-RESULT command should be carefully examined.

In this output, the experimental information that is badly fitted will be marked with an asterisk "*" in the rightmost column. It is not a problem that many experimental data are badly fitted at this initial stage, but one should be careful with errors that the optimizer in PARROT may not be able to solve by itself. A typical case of such a problem is when a phase has a congruent melting and there are experimental information on both sides of the congruent transformation. If the composition of the phase is the experimental information, it may happen that the calculated equilibria is on the "wrong" side of the congruent point and thus give a large error. The user must correct such problems manually in the ED_EXP module. A similar error may occur for miscibility gaps when the experimental information is from one side and the calculation gives the composition of the other side.

12.2.6.5 Critical sets of experimental data and of parameters

When a sensible weighting of the experimental data has been found, this is called the "critical set". The following points should be taken into account when determining such a set:

- The reliability of the experimental technique.
- The agreement between independent measurements of the same quantity.
- The agreement between data obtained with different experimental methods.
- One should use only experimentally determined properties, not quantities that has been converted.
- Be careful about the estimated accuracy of the experiments.
- Correct systematic errors (temperature scale for example).
- Use your experience from previous assessments.
- Make use of "negative" information, for example, that a phase should not be stable in a certain composition or temperature region.

The assessor must also determine a reasonable set of model parameters to be optimized. With more parameters, the sum of errors usually decrease, but at the same time the parameters become less well determined. A measure of the significance of the parameters is given by the column *Relative Standard Deviation, RSD*, listed for each optimized variable. The meaning of RSD is that the parameter can be changed +/- with this factor without changing the reduced sum of error with more than one unit. A large RSD thus means that the variable is badly determined.

The value of RSD is significant only when the user has run the optimization after a RESCALE command and it has converged almost immediately, *i.e.*, the "scaling factor" is the same as the "value". If there are variables with a RSD larger than unity, it means too many variables have been used.

The values of RSD depend also on the weighting of the experiments. It may be possible to reduce the RSD by changing the weights.

If one has one or more variables with RSD larger than unity, one should try to remove one or more optimizing variables by setting them to zero, or to some value that can be determined from other information, for example, by some semi-empirical estimation methods.

12.2.6.6 Optimize and continue optimize

With the critical set of experiments which may be modified now and then, and when trying various number of model parameters to be optimized, the user must use his skills to get the best possible result. The smaller the sum of errors the better the result. By giving the commands OPTIMIZE, CONTINUE, RESCALE and OPTIMIZE again, the user may finally reach a point when PARROT states that it cannot improve the set of optimizing variables. However, this should not be trusted, and a few more OPTIMIZE commands should be made. But if PARROT converges with the same number of iterations (when there are variables to optimize), one has to accept this set.

A problem using OPTIMIZE several times to make the optimization converge is that sometimes a parameter may suddenly start to change by several orders of magnitudes. This behavior may lead to impossible parameter values, and requires a careful reconsideration of the weighting of the critical set of experiments and of the parameters that are optimized.

There is a reasonable range in which the parameter values should be restricted. To find interaction parameters of the order 1E7 for the temperature-independent part and 1E3 for the temperature-dependent part is a clear indication of a bad weighting of the experiments or using too many parameters.

If the user is still not satisfied with the overall fit, he has to change the weights or add more information to force the optimization in the right direction. The success in such manipulations depends on the skill of the assessor.

12.2.6.7 Some hints

Many problems and errors may occur during optimization, and it is not possible to give simple explanations how to handle them. The main recommendation is to exclude all experiments that give strange results but make sure that all important invariant equilibria are reasonably calculated. If some invariant equilibria cannot be calculated, it may be better to exclude the phases that are involved in these equilibria if they are intermediate phases, and just optimize the liquid and the most important solution phases in a first step. When reasonable results have been obtained for the most important phases, the intermediate phases may be put back, and be optimized while keeping the parameters for the liquid and already optimized solid phases as fixed. Consult *Section 12.4* (for the Tricks and Treats) for more hints, as it is impossible to give any more general advice at this stage.

12.2.6.8 Analysis of results

It is rarely the case that the assessor finishes an assessment with the feeling that it cannot be improved. As mentioned above, there are usually other factors that determine when the assessment should be finished. But the following analysis of the final result should be made:

- A satisfactory description of the critical set of experiments.
- A satisfactory description of data not included in the critical set.
- A reasonable set of parameter values.
- Reasonable extrapolations of the thermodynamic properties, also to higher order systems.
- A comparison with the result using other critical sets or models.
- An analysis of the statistical properties of the final set of parameters.

Such analysis should also be commented in the report and publication from the assessment.

12.2.7 Rounding off parameters

When all optimizing variables have a relative standard deviation (RSD) lower than unity, one may accept this as a final set of parameters of the assessment. However, all the parameters have many digits, and they must be rounded off in some way. The simplest rounding off when dealing with metallic system is usually to keep as many digits as will give less than one J/mol difference at 1000 K. For aqueous systems or systems assessed for very different temperature ranges, one may have to use other criteria. But a rounding off that give larger differences than one J/mol may give detectable changes in the phase diagram or with other experimental data.

There is also another philosophy when rounding off parameters. One starts by considering the variable with the highest RSD, usually larger than 0.1. Basically, this means that there is only one significant digit in this variable. One may thus round it off to just one digit and trailing zeroes. However, this will change the sum of errors and the remaining variables must be re-optimized with the rounded variable fixed. One should reach almost the same reduced sum of errors with the remaining variables; otherwise the variable set fixed was not the best to round off.

When rounding off the first variable to one digit has been successful, one may continue to round of the variable with the largest RSD in the new set, and set it fixed and re-optimize the remaining. This can continue until there is just one variable left. The final sum of errors should not deviate significantly from the initial one. The rounded parameters are easier to handle than those just rounded with many non-zero digits. A valid critique of the second type of rounding is that the variables are not dimensionless. Thus, if the Avogadros number is changed, the variables must be changed, and they will no longer have trailing zeroes.

12.2.8 Changes that are interactively done require recompilation

It is possible to change almost everything from the initial SETUP and POP files interactively. For example, one may add more parameters to be optimized, modify or add more experimental information. But there are some changes that will destroy the data structure, and it thus requires that the experimental data file (POP) is recompiled. An example of such a change is to add more composition sets to a phase. This actually changes the number of phases, and this will destroy the links between the experimental data and the thermodynamic models. Thus the POP file must be recompiled, and it is then important that the POP file reflects the changes made interactively in the ED_EXP module.

One practical problem with recompiling the POP file may also be that the experimental equilibria will be calculated with the default start values of all compositions. In some systems, the equilibria require manual start values, and the recompilation of the POP file may then require careful massaging of the experimental equilibria to converge again. To simplify the recompilation, it is recommended to set major constituents for the phases same as these are used in the default start values (see *Section 11.12.2* in the GES User's Guide for details). It is possible to provide start values of the constitutions of the phases in the POP file.

12.3 Alternate Mode

The information that the PARROT module uses to optimize the thermodynamic model parameters represent measurements at equilibrium in the system. The measurements can be in single-phase regions, *e.g.*, activities or enthalpies, or in two-phase regions, *e.g.*, solubilities or transformation temperatures, or in regions with more than two phases involved. At each equilibrium, at least one quantity must have been measured in addition to those necessary to determine the equilibrium state.

For a binary system in a single-phase region, one may have measured the temperature, pressure, composition and chemical potential. Three of these quantities are necessary to specify the equilibrium state and the forth can be used as experimental information to model the phase. In a two-phase region at given temperature and pressure, one may have determined the stable phases and the composition of one or both phases. The temperature, pressure and set of phases are sufficient to determine the equilibrium, and the compositions are then possible to use as experimental data.

Since TCCQ, the improved ALTERNATE mode is possible to include functions to be evaluated together with an alternate calculation and to select ALTERNATE mode for each experimental equilibrium.

12.3.1 Common tangent construction

It is easy to understand that there may be a problem to calculate an equilibrium between two or more phases when the model parameters for the phases are badly determined. The equilibrium calculation requires that one can find a “common tangent”, *i.e.*, that the chemical potentials for all components are the same in all phases. Such a common tangent may not exist or be at a completely wrong composition or temperature for the initial set of model parameters. For detailed information on common tangent plane and driving force, please refer to *Section 3.2.15* and *Figure 3.1*.

Instead of requiring that an equilibrium should be calculated between two or more phases, PARROT now supports an “alternate” technique to calculate such experimental information. The alternate technique calculates the thermodynamic properties for each phase separately, and the program uses as “experimental information” the difference in chemical potential for the components in each phase. The model parameters are then adjusted to make the chemical potentials of all components the same. This is not a new technique, it has also been possible to describe equilibria in this way in PARROT earlier, but it has been cumbersome and difficult. In the BINGSS software, the options IVERS=1 or IVERS=3 have been available from the beginning, and these make use of the same technique as the alternate mode to calculate chemical potentials for each phase separately rather than the full equilibrium.

12.3.2 Preparation of the POP file for Alternate Mode

An alternate calculation of the experimental equilibria with two or more stable phases is set by the SET-ALTERNATE-MODE Y command sequence in the PARROT module. The POP file will usually require additional information to handle this option because there must be information enough to calculate each phase separately. For example, if both compositions of a binary tie-line have been measured, this may be given in the POP file as below:

```
CREATE 1 1
CH-ST PH FCC BCC=FIX 1
SET-COND P=1E5 T=1000
EXPERIMENT X(BCC,B)=.2:.01 X(FCC,B)=.3:.01
```

The equilibrium above can be calculated with alternate mode without any modification. PARROT will use the values given as EXPERIMENT as condition when calculating the thermodynamic properties of each phase. But if just one side of the tie-line has been measured, one must provide an estimate of the composition of the other phase. This can be added by the SET-ALTERNATE-CONDITION command in the POP file. A command SET-ALT-COND is ignored unless the alternate mode is set. The same example as above would then be:

```
CREATE 1 1
```

```

CH-ST PH FCC BCC=FIX 1
SET-COND P=1E5 T=1000
EXPERIMENT X(BCC,B)=.2:.01
SET-ALT-COND X(FCC,B)=.3

```

When PARROT calculates the thermodynamic properties of the BCC phase, it will use the composition as provided by the EXPERIMENT command, and when calculating for the FCC it will use the composition provided with the SET-ALT-COND command.

The alternate mode should only be used in the beginning of an assessment before reasonable model parameters have been determined that makes it possible to calculate the experimental equilibria in the normal way. But it will be a big relief to new users if using the alternate mode, as it required great skill and experience to guess such start values in the normal mode.

12.3.3 More examples of the SET_ALTERNATE_CONDITION command

A three-phase equilibrium may have some compositions determined experimentally, and some can be provided as alternate conditions:

```

CREATE 1 1
CH-ST PH FCC BCC LIQ=FIX 1
SET-COND P=1E5
EXPERIMENT T=912:5 X(LIQ,B)=0.2:.02
SET-ALT-COND X(FCC,B)=0.1 X(BCC,B)=.4

```

However, the experiment below cannot be converted to alternate mode:

```

CREATE 1 1
CH-ST PH BCC LIQ=FIX 1
SET-COND P=1E5 T=1111
SET-REF-STATE B LIQ * 1E5
EXPERIMENT MUR(B)=-4300:500

```

The reason is that there are no information about the compositions of the two phases, and even if these were added as SET-ALT-COND the alternate mode will end up with 5 conditions for each phase instead of the correct 4 because the alternate mode will keep the condition MUR(B) for both phases. In a future release of Thermo-Calc, the alternate mode may be able to handle this kind of situation.

A composition of a stoichiometric phase MUST be given with 7 correct digits. For example,

```

CREATE 1 1
CH-ST PH LIQ A2B=FIX 1
SET-COND P=1E5 X(LIQ,B)=0.2
EXPERIMENT T=992:5
SET-ALT-COND X(A2B,B)=.6666667

```

Note that experimental information on equilibria involving phases with the status ENTERED or DORMANT will be ignored by the alternate mode.

12.4 Tricks and Treats

Each assessor will develop his personal relation to PARROT, because it is such a rich software with many unique features. However, there are some common tricks that may be useful to know, even before the user has developed a more intimate relation with PARROT.

- 1) Use as few experiments as possible initially to get a reasonable overall fit. In particular, the invariant equilibria are useful, also the metastable states that can be estimated by excluding some phases. The estimated equilibria should be excluded in the final optimization.
- 2) If you have 100 activity measurements in a system but only 10 composition points from the phase diagram, you may have to decrease the weights on the activity experiments.
- 3) It may be interesting to exclude some or all intermediate phases initially and just optimize the liquid and the terminal (end-member) phases (for the pure components). It is always interesting to calculate the metastable phase diagram with just these phases, also at later stages in the optimization. The metastable solubility lines should not have any strange kinks or turns.
- 4) Phases with miscibility gaps, as well metastable phases, are always problematic. Try to keep control of the miscibility gaps by some real or estimated experimental information. Unfortunately, it is not possible to calculate the top of a miscibility gap as a single equilibrium.
- 5) Phases with order/disorder transformation are also problematic. They quite often require manual start values, and sometimes the ordered state may disappear during the optimization. It may help to add an experiment that controls the state of order.
- 6) The alternate mode is a very new feature in PARROT, and no one has much experience with it. It should only be used to find an initial set of model parameters that can be used to make the experimental equilibria possible to calculate in the normal mode.
- 7) When the liquid and some solution phases have been fitted reasonably well, the optimizing variables describing these phases can be set fixed, and the variables for the intermediary phases can then be optimized.
- 8) A final optimization with all variables and all experimental information, with the appropriate weighting, should be made.
- 9) Check continuously that the optimized variables are within a reasonable range of values. When a variable starts to change several orders of magnitudes during an optimization, one must reconsider the experimental weights and if too many parameters are used.

12.4.1 Conflicting data

In some cases, one may find measurements of the same quantity that is widely different. In principle, all available experimental data should be entered into the POP file, unless there are obvious reasons, for instance, not pure samples used in the experiments, for such data some sets can be rejected.

But during the optimization, one should not include two conflicting sets at the same time but only use one at a time together with the rest of the data. This follows from a simple rule by Bo Jansson: *if you have two conflicting datasets, then either one or the other may be correct or both are wrong. Both cannot be correct.*

Hopefully, the optimization can make it clear which dataset is most in agreement with the other information on the system.

There are also cases when conflicting data cannot be detected directly. For example, activity data may not be possible to reconcile with solubility data from the phase diagram. This is indicated by large errors in the fit when both datasets are included. Thus it may be necessary to try to optimize with some datasets excluded to find these inconsistencies.

12.4.2 Lack of data or bad data

In some systems, there are very little data. That does not mean one should fit the available data uncritically with high accuracy without considering the possible errors. One may look for data in systems with similar elements to get an idea about reasonable estimates of lacking data or use *ab initio* calculations or semi-empirical methods for estimations.

12.4.3 Time estimate

One can say that a perfect assessment result for a complex system can hardly be obtained, especially when time is concerned. A rough estimate of the time spent on an assessment is:

- ◆ 25% for collecting available experimental data and creating the POP, SETUP and EXP files, correcting and updating them.
- ◆ 25% for “optimizing the weights” to find the correct balance between various experimental information and to select a critical set of experimental information.
- ◆ 25% for optimizing the parameter values.
- ◆ 25% for writing the report. Quite often one has to go back and re-optimize when one finds that a selection or decision made during the optimization cannot be explained or defended in the report.

12.4.4 Number of parameters

Usually, an assessment is considered better when fewer adjustable variables are needed to get the same level of fitting. Of course, it does not matter much if 24 or 25 parameters were needed.

But if one can get almost the same fit with 8 parameters instead of 12, then the assessment with 8 can be considered a superior fit. However, in an assessment, one may put different weights on different information and it is very difficult to compare assessments. For example, the Fe-Al system may be assessed by one person for incorporation in a database for Al, and by another assessor for incorporation in a database for Fe. Quite likely the assessors will have very different opinions on which information that is important to fit, and may end up with a very different number of parameters.

Another very important reason to use a small number of parameters is that the assessment will be used for extrapolations into higher order systems. By experience, it has been found that the fewer parameters that is used in the binary systems the fewer problems occur in higher order systems.

12.5 Command Structure

12.5.1 Definitions of some terms

A model parameter, *i.e.*, a TP-function in GES, is called a “PARAMETER” in the PARROT module. An entity that can be optimized must have a single numeric value and is called a “VARIABLE”. In the present version of the PARROT module, 99 variables, named V1 to V99 are predefined. The variables are entered into the symbol list in the GES5 workspace, and can thus be used in the definition of model parameter expressions in the PARROT workspace. The variables can also be used in the equilibrium description by the IMPORT command. The user can prescribe fixed values for the variables or evaluate their “best” value in an optimization. At initiation, all variables are set fixed with the numerical value equal to zero.

12.5.2 Command connections with other modules

The PARROT workspace is very closely connected to the GES5, POLY3 and DICTRA workspaces, and a PARROT work file (*.PAR) normally contains relevant parts for PARROT, GES, POLY and DICTRA modules simultaneously.

The system definition (*.SETUP) file will usually initialize the PARROT workspace, as well the GES5 and POLY3 workspaces which are associated with the PARROT workspace.

Because the ED_EXP module is actually a sub-module of the PARROT module, the performance of any ED_EXP commands or experimental data (*.POP) file commands will modify in the PARROT workspace.

The graphical experimental data (*.EXP) does not influence the PARROT workspace.

For user's convenience, many GES commands are also available in the PARROT module.

12.5.3 User interface

The user interface of the PARROT module is constructed to be used in an interactive session on a time sharing computer. The main feature of its user interface is a command monitor, and the user can control the actions of the program by typing the appropriate commands.

The commands of the PARROT module often consists of several words separated by a hyphen “-” or an underscore “_” character. The hyphen and underscore characters are treated as identical. The commands can be abbreviated until they become ambiguous, and each part between hyphens or underscore can be shortened. Trailing parts of a command can be omitted if they are not necessary to distinguish between commands.

Many of the commands require that some parameter values are specified. If the <RETURN> key is pressed after the command, the program will ask for these values. Usually the program suggests a default value given between slashes “//”, and the user can accept this value by just hitting the <RETURN> key, or the user specify another value. If the user knows which parameters are required in consequence by a command, all parameters values can be entered on the same line as the command.

12.6 General Commands

These commands are found in almost all modules in the Thermo-Calc and DICTRA software. All such commands perform normally the same in different modules, except for the INFORMATION command which shows information on the switched module.

There are two commands for help in general. However, the most important help facility is the possibility to type a question mark, “?”, anytime you are asked a question you do not fully understand. You should then be given a concise description of what the program wants you to give as input. In some cases a more extensive description can be obtained by typing two question marks, “??”.

12.6.1 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 (also see Section 5.4.11).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing Prompt: MODULE NAME:

```
NO SUCH MODULE, USE ANY OF THESE:
SYSTEM_UTILITIES
GIBBS_ENERGY_SYSTEM
TABULATIONREACTION
POLY_3
BINARY_DIAGRAM_EASY
DATABASE_RETRIEVAL
REACTOR_SIMULATOR_3
PARROT
POTENTIAL_DIAGRAM
SCHEIL_SIMULATION
POURBAIX DIAGRAM
TERNARY_DIAGRAM
MODULE NAME: <module name>
```

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

12.6.2 HELP

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

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

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

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

12.6.3 INFORMATION

Description: Basic information about various PARROT subjects (concepts and models) can be obtained with this command for a number of subjects, as they are described in different parts of this chapter.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: <subject name>

The name of the subject must be given. Extensive information is available for various subjects as listed below (this list can be seen if typing a question mark “?”):

INTRODUCTION	GETTING STARTED	OPTIMIZING PROCEDURE
DATAMANAGEMENT AND FILES	DEFINITION OF SOME TERMS	
USER INTERFACE	ALTERNATE MODE	

12.6.4 BACK

Description: This command switches control back to the most recent module. See also GOTO_MODULE.

Synopsis: BACK

12.6.5 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.

Synopsis: SET_INTERACTIVE

12.6.6 EXIT

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

Synopsis: EXIT

12.7 Most Frequently Used Commands

12.7.1 CREATE_NEW_STORE_FILE

Description: The user must create a file to be used as a “work” file (or called store file) before any optimization can be done. The workspace used by the GES, POLY and PARROT modules will be stored automatically on the work file but not any experimental information. The work file has a default extension “.PAR” under Windows and UNIX/Linux. 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 (see *Section 12.7.2*).

Synopsis: CREATE_NEW_STORE_FILE

Ensuing Prompt: STORE FILE /ABCD/: <file name>

Under UNIX and Linux, this prompt should be specified with name of the desired store file where the current PARROT/GES5/POLY3 workspace is written. If one work file has been previously created (by this command), or set (by the SET-STORE-FILE command) or opened (by the READ_WORKSPACE command), its name will be set as default.

Under Windows Vista/XP/2000/NT4 environments, a “Save As” window as shown in *Figure 12-4* will pop up instead. 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 “Save as type” box is the proper one for the PARROT workspace format (*i.e.*, PAR file).

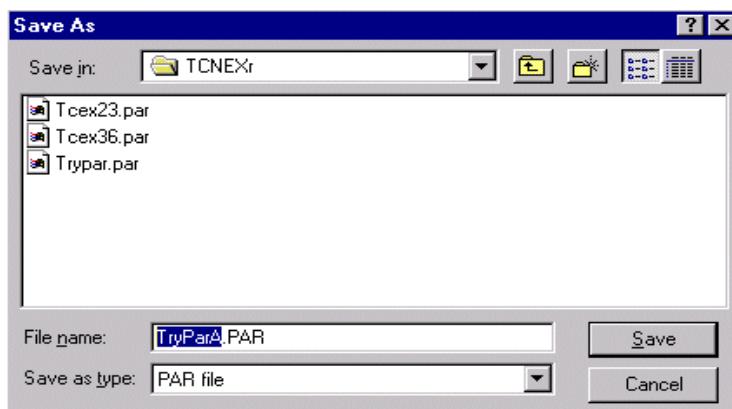


Figure 12-4. The “Save As” window: Creating a stored work file.

If there is already one work (PAR) file with the same name saved in the current working directory, a small screen with a warning message as shown in *Figure 12-5* will pop up. The user needs to either choose the Yes button to proceed a replacement of the previous PAR file or the No button to cancel the current saving. For the later case, the user can either give another name or change the working directory in order to save the current PARROT workspace onto a file.

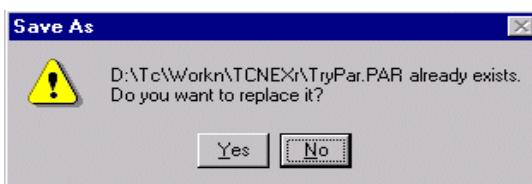


Figure 12-5. The “Save As” warning window: Proceeding or cancelling the creating.

12.7.2 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 file must have been created as a store file, see the CREATE-NEW-STORE-FILE command. The workspace used by GES, POLY and PARROT is read from the specified store file (with a default extension “. PAR”).

Synopsis: SET_STORE_FILE

Ensuing Prompt: STORE FILE /ABCD/: <file name>

Under UNIX and Linux, this prompt should be specified with name of the stored work file from where a previously saved PARROT/GES5/POLY3 workspace is read. If one work file has been previously set (by this command), or created (by the CREATE-NEW-STORE-FILE command) or opened (by the READ_WORKSPACE command), its name will be set as default.

Under Windows Vista/XP/2000/NT4 environments, an “**Open file**” window as shown in *Figure 12-6* will pop up instead. 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 PARROT workspace format (*i.e.*, PAR file).



Figure 12-6. The “Open file” window: Setting a stored PARROT workspace file.

12.7.3 COMPILE_EXPERIMENT

Description: The description of the experimental equilibria saved on a POP file, given as commands in the POLY and ED_EXP syntax, is compiled into the PARROT structured data (*i.e.*, stored into the current work file which has been created or opened or set prior to this command).

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 by using a text editor) and recompile it with the work (store) file.

Synopsis: COMPILE_EXPERIMENT

Ensuing Prompt: INPUT FILE /ABCD/: <file name>

Under UNIX and Linux, this prompt should be specified with name of the experimental data file (POP) from where all description of experimental equilibrium information will be read and compiled by this command. If one POP file has been previously compiled (by this command), its name will be set as default.

Under Windows Vista/XP/2000/NT4 environments, an “**Open file**” window as shown in *Figure 12-7* will pop up instead. 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 experimental data format (*i.e.*, POP file).

OUTPUT TO SCREEN OR FILE /SCREEN/ : <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 the list file under a specific file name which can later on be opened and edited by simple text editor.

INITIATE STORE FILE /Y/ : <Y or N>

After a successful compilation, the data in the workspace of POLY3 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.

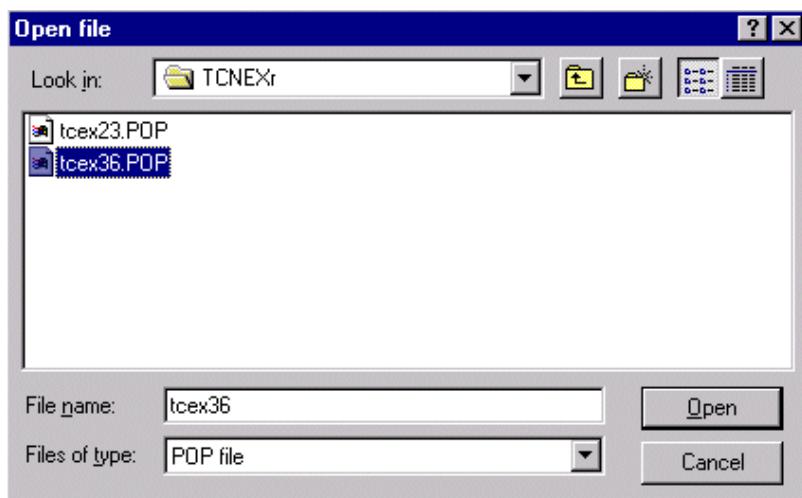


Figure 12-7. The “Open file” window: Opening an experimental data *.POP file.

12.7.4 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.

Since TCCQ, the improved ALTERNATE mode is possible to include functions to be evaluated together with an alternate calculation and to select ALTERNATE mode for each experimental equilibrium.

Synopsis: SET_ALTERNATE_MODE

Ensuing Prompt: On /Y/ : <Y or N>
The default answer Y will turn on the alternate mode. Anything else will turn it off.

12.7.5 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 (see *Section 12.7.6*) 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 file or interactively entered in the GES module) can be obtained by using the LIST-ALL-VARIABLE command (see *Section 12.7.11*).

Synopsis: SET_OPTIMIZING_VARIABLE

Ensuing 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.

12.7.6 SET_FIX_VARIABLE

Description: 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 (see *Section 12.7.5*).

Synopsis: SET_FIX_VARIABLE

Ensuing 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.

12.7.7 EDIT_EXPERIMENTS

Description: By this command, the sub-module for editing experimental equilibria, i.e., the ED_EXP module 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.

For details about the ED_EXP module, see *Part 13*.

Synopsis: EDIT_EXPERIMENTS

12.7.8 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.

Improvements have been made since TCCS, so that the optimizer tries a little harder in cases when the initial set of optimizing variables is far away from the solution.

Synopsis: OPTIMIZE_VARIABLES

Ensuing 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.

12.7.9 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.

Synopsis: CONTINUE_OPTIMIZATION

Ensuing 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.

12.7.10 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 (see Section 12.8.1).

Since TCCQ, the LIST_RESULT command has a new 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.

Since TCCS, the option G has been improved so that the plot is automatically plotted and the user can scale it giving commands in the POST module, and the option D (new since TCCR) has been slightly modified so that it 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 (column6).

Synopsis: LIST_RESULT

Ensuing 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 (new since TCCR, and modified since TCCS) lists all experiments, including those with fulfilled inequalities which are suppressed with the default option C.
- The F (full) format is now obsolete but the question has been retained for backward compatibility.
- The G (graphical) format (new since TCCQ, and improved since TCCS) 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. Since TCCS, 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

12.7.11 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 (see *Section 12.8.1*).

Synopsis: LIST_ALL_VARIABLES

Ensuing 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

```

12.7.12 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.

Synopsis: RESCALE_VARIABLES

12.8 Other Commands

12.8.1 SET_OUTPUT_LEVELS

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

Synopsis: SET_OUTPUT_LEVELS

Ensuing 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/:

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/:

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/:

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

LIST ALL PARAMETERS: /N/:

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/:

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).

12.8.2 SAVE_PARROT_WORKSPACES

Description: If the user is not happy with the late changes made through various PARROT/GES/POLY commands, he can use this command to save the current workspace (*i.e.*, data area) used by the PARROT program (also including the current GES5 and POLY3 workspaces), onto the present work 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 opening of a SETUP TCM file). Note that the PARROT/GES5/POLY3 workspaces are updated after each PARROT/GES5/POLY3 command, and by this command the current PARROT/GES5/POLY3 workspaces will be always updated onto the latest work file associated with the latest command, either SET_STORE_FILE or CREATE_NEW_STORE_FILE.

Unlike the SAVE commands in other modules (*e.g.*, GES or PLOY), this command will not ask for the file name where to save the current PARROT/GES5/POLY3 workspaces, for the reason described above. One can not use this command if there is no work file opened or created previously.

Synopsis: SAVE_PARROT_WORKSPACE

12.8.3 READ_PARROT_WORKSPACES

Description: If the user is not happy with the late changes made through various PARROT/GES/POLY commands, he can read the previous PARROT/GES5/POLY3 workspaces back to replace the current PARROT/GES5/POLY3 workspaces. The previous PARROT/GES5/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 opening 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.*, GES or PLOY), this command will not ask for the file name where to read a previously opened/created/updated PARROT/GES5/POLY3 workspaces, for the reason described above. One can not use this command if there is no work file opened or created previously.

Synopsis: `READ_PARROT_WORKSPACE`

12.8.4 MACRO_FILE_OPEN

Description: MACRO is an extraordinary convenient and easier way of pre-defining sequences of various legal TCC (and DICTRA) commands stored in a so-called **MACRO file** (which is a simple textual file normally with the default extension “TCM” for the TCC software 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 TCC software; SYS, POLY, PARROT and DICTRA_Monitor module in the DICTRA software).

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 TCC software, if in the SYS module the `SET_LOG_FILE` command is used and a **LOG file** name is given before any other SYS/TDB/GES/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 current TCC session 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 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 TCC (and DICTRA) program.

All kinds of legal TCC (and DICTRA) commands and their required inputs can be stored into a MACRO file. A MACRO file must be terminated with the `EXIT` command, or can be interrupted (for the purposes of allowing further interactive operations by the user) in the SYS, GES, POLY, PARROT or POST modules with the `SET_INTERACTIVE` command.

Within a MACRO file (for previous versions up to TCCR), one could have as many as possible **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 TCC commands when the file is called by the TCC software.

However, this might be considered as somewhat tedious. Therefore, it is now (since TCCS) possible to have multiple-line **comment-blocks** inside a MACRO file, using the paired “@(*“* and “@*”* signs (*i.e.*, the indicators that starting the lines for the “*begin comment*” and “*finish comment*”, respectively). 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 TCC 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 2nd 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, a textual copy of 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, e.g.

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

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 TCC software from temporarily stopping at any pause by simply typing any character (except for the Y character) after specifying the name of a MACRO file.

Since TCCN, 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 TCC 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_Module command **SET_ECHO** at the every 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 TCC (and DICTRA) software, that are enforced according to the MACRO file(s).

Synopsis 1: MACRO_FILE_OPEN <name of a Macro file>

Synopsis 2: MACRO_FILE_OPEN

Ensuing Prompt: Macro filename: <name of a Macro file>
Specify the filename with the MACRO command. The default extension is “TCM”.

Notes: Under Windows Vista/XP/2000/NT4 environments, if an appropriate MACRO file is not given after the command, an **Open file** 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 12-8*.

The file type (*i.e.*, TCM, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with executing various Thermo-Calc commands. The user may also cancel such an **Open file** window session, and thus the current MACRO file will not be opened.

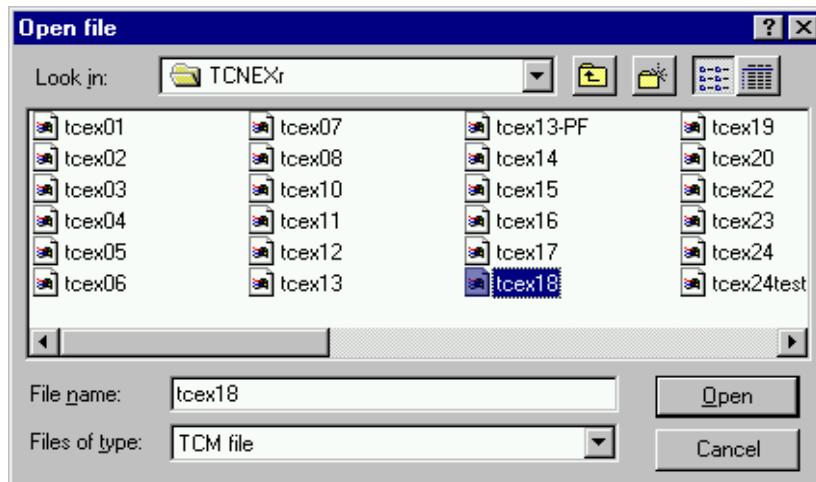


Figure 12-8. The “Open file” window: Opening an existing TCM file.

If the MACRO file contains some SYS/TDB/TAB/GES/POLY/POST/PARROT/ED_EXP-module commands for setting *.LOG files, saving/reading GES5/POLY3/PARROT workspaces, switching USER databases, compiling experiments (from existing *.POP files), creating new *.PAR files, appending experimental data *.EXP files, plotting/dumping diagrams, *etc.*, a corresponding window (*e.g.*, **Save As**, **Open file**, **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.

More Notes: 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, *i.e.*, 1 (under Windows Vista/XP/2000/NT4) or 9 (under Linux and all types of UNIX platforms).

For an extensive example on using MACRO features for the PARROT-Module, please refer to the **TCEX36.TCM** on the TCC installation area **\TCEX** and in the **TCCS Examples Book**. For more details on how a MACRO file is constructed, please refer to *Sections 8.10.26 and 14.2.8*.

12.8.5 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 GES5/POLY3 workspaces.

Synopsis: REINITIATE

12.8.6 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.

Synopsis: SET_OPTIMIZING_CONDITION

Ensuing 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.

12.8.7 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.

Synopsis: LIST_CONDITIONS

Ensuing 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.

Example Output: The following is the listed:

```

== 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

== LIST STATUS ==
LIST INCREMENT = 1
LIST STATUS
SUM OF SQUARES: Y
SCALED VARIABLES: Y
WEIGHTED RESIDUALS: Y
ALL PARAMETERS: Y
CORRELATION MATRIX: Y

```

12.8.8 LIST_STORE_FILE

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

Synopsis: LIST_STORE_FILE

12.8.9 SET_SCALED_VARIABLE

Description: This command is similar to the SET_OPTIMIZING_VARIABLE command (see *Section 12.7.5*) 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 (see *Section 12.7.6*) 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 file or interactively entered in the GES module) can be obtained by using the LIST-ALL-VARIABLE command (see *Section 12.7.11*).

Please note that since TCCS the SET_SCALED_VARIABLE command always works correctly for specifying a lower and upper limit for an optimizing variable.

Synopsis: SET_SCALED_VARIABLE

Ensuing 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.

12.8.10 RECOVER_VARIABLES

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

Synopsis: RECOVER_VARIABLES

12.8.11 SET_EXTERNAL_PARAMETER

Description: With this command (new since TCCS), the user can additionally optimize model parameters of the so-called external models, which may not be as fully-implemented/integrated parts inside the GES 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.

Synopsis: SET_EXTERNAL_PARAMETER

Ensuing 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.

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
```

IMPORTANT NOTES: In order to be able to build the user-specified external model and to use this new 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 GES 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.

12.9 Commands from the Gibbs Energy System (GES)

The commands below actually belong to the Gibbs Energy System (GES module). As a service to the user, it is also possible to give them directly from the PARROT module. One can always refer to the related sections in the *Part 11* (the GES Module) for details of these commands. However, for your convenience, they are described here.

12.9.1 LIST_PHASE_DATA

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. Also see *Section 11.11.2*.

Synopsis: LIST_PHASE_DATA

Ensuing Prompt: PHASE NAME: <phase name>
Specify a phase name.

12.9.2 ENTER_PARAMETER

Description: This command is identical to the ENTER_PARAMETER command in the GES module (see *Section 11.10.5*).

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 GES workspaces.

The composition-dependence of the Gibbs energy is described in the GES 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 (see *Section 11.12.4*).

As described in *Section 11.4 (Thermodynamic Parameters)*, 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 (see more details as described in *Section 11.4 – Thermodynamic Parameters*):

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

Examples of GES parameter name:

G(GAS,C1O2)	The Gibbs energy of formation of a CO ₂ 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 new types of identifiers are available since TCCN/TCW1 and are further expanded since TCCR/TCW4:

- ☞ 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.

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 UPPER-cases is recommended; however, if one would prefer to write it as a mixture of UPPER-case and lower-case, it will then automatically convert all lower-cases to UPPER-cases, as the GES module only recognizes UPPER-case 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, 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 $[1 - \sum Y(\text{specified constituents})]$, which implies that in an A-B binary system the following three L parameters are identical (but in higher-order systems, they are different):

$L(\text{phase}, A, B)$ is multiplied with $X(A) * X(B)$
 $L(\text{phase}, A, *)$ is multiplied with $X(A) * (1 - X(A))$
 $L(\text{phase}, B, *)$ 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.

Synopsis: ENTER_PARAMETER

Ensuing 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 UPPER-cases is recommended; however, if one would prefer to write it as a mixture of UPPER-case and lower-case, it will then automatically convert all lower-cases to UPPER-cases, as the GES module only recognizes UPPER-case 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, 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 $[1 - \sum y(\text{specified constituents})]$, 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 GES 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 composition-dependent 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 TP-Function. *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.

12.9.3 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 GES module. Please consult the description of the ENTER_PARAMETER command for details on how a parameter is defined (*Section 11.10.5* or *12.9.2*) or the AMEND_PARAMETER command in the GES module (see *Section 11.12.4*), or use this command inside the GES module if you are not sure about the questions.

Synopsis: AMEND_PARAMETER

Ensuing 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
  4  298.15<T<2000.00
  5  2000.00<T<4500.00
  6  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 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	<#occurrences> F<string>@
Insert	I<string>@
Move	<+-#positions> M
Restore string	R
Substitute	S<OLD>@<NEW>@
Type string	T

where CP denotes the current position in string, # means number of, @ is a terminator of an input or search string.

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.

12.9.4 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 GES module (see *Section 11.11.3*).

The parameter name is (see more details as described in *Section 11.4*):

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

Examples of parameter names:

G(GAS,C1O2)	The Gibbs energy of formation of a CO ₂ 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).

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

- ☞ 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.

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 $[1 - \sum y(\text{specified constituents})]$, 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 (except for two SGTE-owned databases, namely SSUB3 and SSOL2/SSOL4), since TCCP/TCW2, one may not be able to use this command to list any retrieved parameter.

Synopsis 1: `LIST_PARAMETER <identifier>(<phase>, <constituent array>; <digit>)`

Synopsis 2: `LIST_PARAMETER`

Ensuing 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.

12.9.5 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 GES5 workspace. This command provides a way to find out how the functions depend on the optimizing variables.

This command is nearly identical to the LIST_SYMBOL command in the GES module (see Section 11.11.4), but in PARROT module the list will only be shown on screen, not written to any file.

Synopsis: LIST_SYMBOL_IN_GES

Ensuing 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.

12.10 Connections with Other Modules and Interfaces

Internally, the PARROT module has very tight but smooth connections with many other modules inside the Thermo-Calc and DICTRA software. In the future development of our software/database/interface packages, it will be implemented into the graphical user interfaces.

12.10.1 Connections with the ED_EXP modules

As mentioned above and it will be described in more details in the next part in this User's Guide, the EX_EXP module acts as the sub-module of the PARROT program. Through the `COMPILE_EXPERIMENTS` command in the PARROT module, all kinds of experimental information (equilibrium or kinetic data) can be compiled into the PARROT workspace. During a particular assessment, the user will need to go forth and back between the PARROT and EX_EXP modules, in order to make appropriate adjustments on experimental selection (such as data inclusions, relative weights, measurement errors, *etc.*) and on optimization settings (such as optimizing or fixed variables, starting values, *etc.*).

12.10.2 Connections with the GES and POLY modules

As illustrated in the above sections, the PARROT data structure is closely associated with the GES5, POLY3 and DICTRA workspaces. Whenever an optimization run is enforced during a critical assessment, the PARROT program will call the GES module for stored system definition data and model parameters, as well the POLY or DICTRA module for equilibrium calculations or kinetic simulations, and then compare the calculated results with experimental data. One can always go back forth and back among the PARROT, GES, POLY and DICTRA modules, but one must be careful not to destroy the PARROT workspace.

Therefore, from a stored PARROT work file (PAR), one can always trace the system's GES, POLY and DICTRA data area, and, if desired, can save such data in some individual GES or POLY or DICTRA files (*i.e.*, *.GES5/*.TDB/*.DAT, or *.POLY3 or *.DIC files). Moreover, thanks to the availability of the POST post-processor as a sub-module of the POLY and DICTRA modules, one can conveniently call the POST module at any time, to plot for the optimizing system the phase diagrams, property diagrams or composition profiles; consequently, one superimpose various experimental data (which are normally used in the current optimization exercise) and describing texts onto such plotted graphs.

12.10.3 Connections with the DICTRA software

The DICTRA software utilizes the PARROT module in the same way as the Thermo-Calc software, in which the PARROT workspace also shares some parts with the DICTRA data area. Because DICTRA uses Thermo-Calc as its engine for all kinds of thermodynamic data treatments and calculations on thermodynamic properties, stable or metastable equilibria, and local and partial equilibria in dynamic process simulation, the optimization procedure on both thermodynamic and kinetic model parameters can be paralleled with calculations of phase diagrams or property diagrams or composition profiles. For more details, please refer to the DICTRA manuals.

12.10.4 Availability in the TCW software

At the present moment, the PARROT facilities are not available in the graphic user interface of the Thermo-Calc software. However, along the future software development, it is planned to integrate the PARROT module into the TCW software under Windows.

13 Edit-Experiment Module (ED-EXP)

13.1 Introduction

The **ED_EXP** module \Leftrightarrow is actually a sub-module of the PARROT module, and is an essential and powerful tool in the Thermo-Calc and DICTRA software for editing various types of experimental data used in critical assessment of thermodynamic and kinetic model parameters. It is always connected with and entered from the PARROT module. During an assessment course, it normally requires frequent communications back and forth between the PARROT and ED_EXP modules, in order to make appropriate selections and settings and to adjust optimum starting values for the experimental points in the optimization.

The ED_EXP module uses the POLY module in a very special way so that each experimental data point can be calculated as an equilibrium. Therefore, many ED_EXP commands are the same as those in the POLY module, while their performance may be slightly different (see *Section 13.2* for more comments). Some POLY commands are absent in this module, and some new commands (as described in *Section 13.3* in this part) have been added to the ED_EXP monitor.

Before the ED_EXP module is entered (through the EDIT_EXPERIMENTS command from the PARROT module) for the first time in running Thermo-Calc or DICTRA, an experimental data (*.POP for Thermo-Calc or *.DOP for DICTRA) file has normally already been compiled with success.

Similar to other modules implemented in the Thermo-Calc and DICTRA software, the ED_EXP module provides the users of application programs with a user interface with a set of simple and general commands for experimental data editing and intermediate equilibrium calculations during an optimization procedure. Through such a user interface, one can easily conduct all kinds of editing tasks interactively (see the functionality described in *Sections 13.2* through *13.4*), which are necessary for a good optimization.

The ED_EXP module utilizes the POLY3 workspace part of an in-use PARROT work file (*.PAR). The two POLY commands, READ_WORKSPACE and SAVE_WORKSPACE, interact with the POLY3 workspace part, and thus they are only associated with the current PARROT work file rather than directly with any specific *.POLY file. By the SAVE command, it transfers promptly all the calculated experimental equilibrium points or dynamic steps into the POLY3 (and then DICTRA) workspaces. In the DICTRA software, some DICTRA commands are also available in the ED_EXP module; and for details of its performance in DICTRA, please refer to the *DICTRA User's Guide*.

The following commands are available in the ED_EXP module of the Thermo-Calc software:

ED_EXP : ?		
ADVANCED_OPTIONS	GRAPHICS_PLOT	SAVE_WORKSPACES
BACK	HELP	SELECT_EQUILIBRIUM
CHANGE_STATUS	IMPORT	SET_ALL_START_VALUES
COMMENT	INFORMATION	SET_ALTERNATE_CONDITION
COMPUTE_ALL_EQUILIBRIA	LABEL_DATA	SET_CONDITION
COMPUTE_EQUILIBRIUM	LIST_ALL_EQUILIBRIA	SET_NUMERICAL_LIMITS
CREATE_NEW_EQUILIBRIUM	LIST_CONDITIONS	SET_REFERENCE_STATE
DEFINE_COMPONENTS	LIST_EQUILIBRIUM	SET_START_CONSTITUTION
DELETE_SYMBOL	LIST_STATUS	SET_START_VALUE
ENTER_SYMBOL	LIST_SYMBOLS	SET_WEIGHT
EVALUATE_FUNCTIONS	MAKE_POP_FILE	SHOW_VALUE
EXPERIMENT	READ_WORKSPACES	STORE_ALL_WEIGHTS
EXPORT	REINITIATE_MODULE	TABLE_HEAD
FLUSH_BUFFER	RESTORE_ALL_WEIGHTS	TRANSFER_START_VALUES
ED_EXP :		

Note that the GRAPHICS_PLOT, LIST_ALL_EQUILIBRIA, RESTORE_ALL_WEIGHTS, and STORE_ALL_WEIGHTS commands are the new ED-EXP commands since TCC P; however, the GRAPHICS_PLOT command can only be used in POP files since TCCQ. The ADVANCED_OPTIONS command replaces the old SPECIAL_OPTIONS command since TCCS.

\Leftrightarrow Revision History of the ED_EXP Module User's Guide:

Feb 1984	First release (Edited by Bo Jansson)
Oct 1993	Second revised release (Edited by Birgitta Jönsson and Bo Jansson)
Sept 1998	Third revised release (Edited by Bo Sundman)
Jun 2000	Fourth revised and extended release (Edited by Pingfang Shi)
Nov 2002	Fifth revised and extended release (Edited by Pingfang Shi)
Jun 2004	Sixth revised release (Edited by Pingfang Shi); with minor changes in 2006/2008

13.2 Performance of POLY Commands in the ED_EXP Module

As mentioned in the Introduction (*Section 13.1*), many POLY commands can be directly used in the ED_EXP module and *.POP or *.DOP files. The functionality of such commands in the ED_EXP module is normally identical to those in the PLOY module. However, the actual performance of some of such commands may differ slightly from in each other module.

In *Table 13-1*, the functionality and performance of all such POLY commands used in ED_EXP module and *.POP or *.DOP files are summarized:

Table 13-1. Summary of the Performance of POLY Commands in the ED_EXP Module

Command	Functionality	Performance
HELP	Get help on a specific command.	On all kinds of commands available in the module as listed.
INFORMATION	Get information on a specific subject.	On all types of POLY and ED-EXP subjects.
BACK	Get back to the PARROT module.	Always to the PARROT module. Important Note: Do not forget to give a SAVE-WORKSPACE command before this.
DEFINE_COMPONENT	Define new components.	On the current experimental point.
ENTER_SYMBOL	Enter symbols (constants, variables, functions or tables).	On all experimental points in the current work file.
DELETE_SYMBOL	Delete existing symbols (constants, variables, functions or tables).	On all experimental points in the current work file.
LIST_SYMBOLS	List existing symbols (constants, variables, functions or tables).	On all experimental points in the current work file.
CHANGE_STATUS	Alternate status for phases, species or components.	On the current experimental point.
LIST_STATUS	List status for all phases, species and components on screen.	On the current experimental point.
SET_REFERENCE_STATE	Set reference state for a component.	On the current experimental point.
SET_CONDITION	Set equilibrium conditions (which will alternate the same conditions for the selected experimental equilibrium).	On the current experimental point.
LIST_CONDITION	List equilibrium conditions for the current experimental point.	On the current experimental point.
SET_START_VALUE	Set start values.	On the current experimental point.
SET_START_CONSTITUTION	Set start constitutions.	On the current experimental point.
SET_ALL_START_VALUES	Set all start values and constitutions.	On the current experimental point.
SET_NUMERICAL_LIMITS	Set various numerical limits for the equilibrium calculation procedure.	On all experimental points in the current work file.
SELECT_EQUILIBRIUM	Select an identified experimental point from the presently-read and possibly-modified data block from the current work file.	The selected experimental point becomes the “current” or “present” point referred by the ED_EXP module. Important Note: After a command COMPUTE_ALL_EQUILIB, the last experimental data point becomes the “present” point. In such a case, one has to use this SELECT_EQUILIB command for further editing.

COMPUTE_EQUILIBRIUM	Calculate equilibrium at a certain experimental point.	On the current experimental point; For a range or all non-zero-weighted experimental points, use the ED_EXP command COMPUTE ALL EQUILIB.
LIST_EQUILIBRIUM	List the calculated equilibrium state of the current experimental point.	On the current experimental point.
CREATE_NEW_EQUILIBRIUM	Create a new equilibrium points.	Based on the current experimental point, but will be given a different set of system definition, condition values, etc.
EVALUATE_FUNCTIONS	Evaluate value(s) for one or several or all defined functions.	On the current experimental point.
SHOW_VALUE	Show value for one state variable or entered function or variable.	On the current experimental point.
READ_WORKSPACES	Read an experimental data block from a previously saved POLY3 workspace from the current work *.PAR file. Actually, this command in the ED_EXP command was previously named as READ_BLOCK, and it always arises a sub-prompt asking from which data block the experimental data points are to be edited. See <i>Section 13.3.6 (READ_BLOCK or READ_WORKSPACE)</i> for details.	On all experimental points in a specific data block from the current work file. Important Note: This command must be used prior to any other ED_EXP command if the module is entered for the first time, or 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.
SAVE_WORKSPACES	Save the editing changes into the POLY3 workspace in the current work file.	On all experimental points in the current work file. Important Note: Do not forget to give this command before going BACK to the PARROT module.
ADVANCED_OPTIONS	Specify an advanced option for a specific solution phase	On all experimental points in the current work file.
REINITIATE_MODULE	Reinitiate the ED_EXP and POLY modules to the states as they were first entered.	On all experimental points in the current work file. All the read experimental data blocks, defined components and conditions, specified start values, changed status, entered symbols, calculated or created equilibria, etc. will be removed.

Please note that there have been three major modifications/improvements since TCCS:

- The ADVANCED_OPTIONS command replaces the old SPECIAL_OPTIONS command;
- It is possible to turn Global Minimization Technique on for (some) individual experimental equilibrium points during optimizations using the POLY/ED_EXP command-sequence ADVANCED-OPTION TOGGLE-ALTERNATE, while the Global Minimization Technique is normally turned off and thus the Ordinary POLY Minimization (GEM) is used for equilibrium calculations during optimizations.
- In the POP file and in the ED-EXP module, one can use the new type of symbols in the ENTER-SYMBOL command for ordered phases (treated by the so-called *Four Substitutional Sublattice Ordering Model*) which have the phase-type of F or B. This is called INTERNAL and can be specified as L12, L10, D019, B19, D03, L21, B2 or B32; in addition, a phase and its composition-set must be specified. The symbol value will be zero if the phase is not ordered according to the specification; otherwise, it will have a positive value depending on the degree of order, with a maximum value of unity. This is useful when optimizing ordered phases (such as ordered FCC, HCP and BCC phases), as the order may change during the optimization.

13.3 Special Commands only available in the ED_EXP Module

Some special commands, as listed below and in-detail described in the following sections, are designed commands only for the ED_EXP module. As mentioned above, they can also be used in *.POP or *.DOP files.

```
COMPUTE_ALL_EQUILIBRIA
EXPERIMENT
EXPORT
IMPORT
LABEL_DATA
MAKE_POP_FILE
READ           (i.e., READ_BLOCK or READ_WORKSPACE)
SET_ALTERNATE_CONDITION
SET_WEIGHT
TRANSFER_START_VALUES
LIST_ALL_EQUILIBRIA
STORE_ALL_WEIGHTS
RESTORE_ALL_WEIGHTS
GRAPHICS_PLOT
```

Note that the GRAPHICS_PLOT, LIST_ALL_EQUILIBRIA, RESTORE_ALL_WEIGHTS, and STORE_ALL_WEIGHTS commands are the new ED-EXP commands since TCCP.

The GRAPHICS_PLOT commands has been completely rewritten in TCCQ (and the late sub-versions/patches) and it now has a total different functionality; although it is still visible in the ED-EXP module, it only works as a special command in POP files. For details please refer to the section 13.4.4.

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 (as listed in the Introduction, see *Section 13.1*), they can not be used in the ED_EXP module; they are only functional in an *.POP or *.DOP file.

13.3.1 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 (see *Section 8.11.2*), 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 (see *Section 8.11.10*) so that the current point will be switched to a desired one.

Synopsis: 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 (see Section 11.3.5), 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 (see Section 11.4.2) 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).

Example List: Below is an example of listed information following the command as interactively run the ED_EXP module, after running the TCEX36a.TCM and TCEX36b.TCM files and some sequent actions in PARROT and ED_EXP modules:

Eq	Lab	Iter	Weight	Temp	Fix phases or comments
1	AINV	1	1.	1189.1	LIQUID A2B BCC
2	AINV	1	1.	1340.1	LIQUID A2B
3	AINV	1	1.	1052.6	LIQUID A2B BCC
4	AINV	1	1.	1207.9	LIQUID BCC FCC
5	AINV	1	1.	718.1	A2B BCC BCC#2
6	AINV		< unused >		BCC BCC#2
					Failed using alternate, setting weight to zero
10	ALF	1	1.	1594.0	LIQUID FCC
					Failed using alternate, setting weight to zero
11	ALF	1	1.	1548.0	LIQUID FCC
					Failed using alternate, setting weight to zero
12	ALF	1	1.	1499.0	LIQUID FCC
					Failed using alternate, setting weight to zero
13	ALF	1	1.	1438.0	LIQUID FCC
20	ATIE	1	1.	1413.0	LIQUID FCC
21	ATIE	1	1.	1337.0	LIQUID FCC
22	ATIE	1	1.	1213.0	LIQUID FCC
23	ATIE	1	1.	1100.0	LIQUID BCC
100	AA	2	1.	1573.0	LIQUID
101	AA	2	1.	1573.0	LIQUID
102	AA	2	1.	1573.0	LIQUID
103	AA	2	1.	1573.0	LIQUID
104	AA	2	1.	1573.0	LIQUID
105	AA	2	1.	1573.0	LIQUID
106	AA	2	1.	1573.0	LIQUID
107	AA	2	1.	1573.0	LIQUID
108	AA	2	1.	1573.0	LIQUID
110	AA	1	1.	1773.0	LIQUID
111	AA	1	1.	1773.0	LIQUID
112	AA	1	1.	1773.0	LIQUID
113	AA	1	1.	1773.0	LIQUID
114	AA	1	1.	1773.0	LIQUID
115	AA	1	1.	1773.0	LIQUID
116	AA	1	1.	1773.0	LIQUID
117	AA	1	1.	1773.0	LIQUID
118	AA	1	1.	1773.0	LIQUID

13.3.2 EXPERIMENT

Description: This command is mainly used in the POP or DOP file, but can also be given interactively, 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 (see Sections 8.3.5 and 8.7.1) 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 either an absolute value or a relative value (in percentage, $x\%$).

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.

Synopsis:

EXPERIMENT <experiment>

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

The syntax of an `experiment` is similar to that of a `condition`. In the most simple case, it is a state variable set equal to a value with a given uncertainty which is specified after the colon “:”. For instance, `T=1273.15:5`,

An inequality, “<” or “>”, can also be used in an `experiment` syntax, *e.g.*, `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 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 in both cases. The second experiment is that the activity of PB should be 0.8 with an uncertainty of 5 per cent. 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
```

13.3.3 EXPORT

Description: This command makes it possible to transfer a calculated value from an equilibrium to a optimizing variable (an “V” 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.

Synopsis 1: EXPORT <function name>#<n>

Synopsis 2: EXPORT

Ensuing 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/lower-case 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”).

13.3.4 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 23* as given in the /TCEX/ subdirectory of a Thermo-Calc installation and described in the *Thermo-Calc Examples Book*.

Synopsis 1: IMPORT <constant name>#<n>

Synopsis 2: IMPORT

Ensuing 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 #.

Important note: the constant name is UPPER/lower-case 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.

13.3.5 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.

Synopsis 1: LABEL_DATA A<bCd>

Synopsis 2: LABEL_DATA

Ensuing 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).

13.3.6 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). In order to document or transfer the current or final edited experimental data to an *.POP/*.*.DOP file, one may use this command to write it to a new *.POP/*.*.DOP file or overwrite it on an old *.POP/*.*.DOP file. Note that the experimental data file (the *.POP/*.*.DOP file) is a simple text file, whilst the work file (the *.PAR file) is not a text file and is dependent on the computer hardware.

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 exiting *.POP file name is specified, it will be overwritten. In DICTRA, an *.DOP file is written or overwritten.

Synopsis 1: MAKE_POP_FILE <POP/DOP-file name>

Synopsis 2: MAKE_POP_FILE

Ensuing 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.

13.3.7 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.

Synopsis: READ

Ensuing 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.

13.3.8 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 (see *Sections 12.3 and 12.7.4*).

The command syntax for this command is the almost the same as for the POLY command SET-CONDITION (see *Section 8.7.1*), while the uncertainty should also be specified.

Synopsis 1: `SET_ALTERNATE_CONDITION <alternate condition(s)>`
 Each alternate condition (including the normally POLY-module condition plus uncertainty; 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. See details for conditions described in *Section 8.3.5*.

Example:

```
SET_ALT_COND T=1273:5, W(C)=.0015:.0001, X(LIQ,CR)=.22:5%
or
```

```
SET_ALT_COND T=1273:5
SET_ALT_COND W(C)=.0015:.0001, X(LIQ,CR)=.22:5%
```

In this example, the user sets the temperature to 1273 K with a uncertainty of 5 K, the mass (weight) fraction of C to .0015 with an absolute uncertainty of .0001 wt%, and the mole fraction of Cr to .22 with a relative uncertainty of 5%.

Synopsis 2: `SET_ALTERNATE_CONDITION`

Ensuing 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>)	mole 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,

$$X(\text{LIQ}, S) - X(\text{PYRRHOTITE}, S) = 0$$

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:

$$2 * \text{MUR}(\text{FE}) + 3 * \text{MUR}(\text{O}) = -35000$$

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 be either an absolute value or a relative value (in percentage, x%).

13.3.9 SET_WEIGHT

Description: Each experimental value has an uncertainty associated with it, specified by the value after the colon “:” in the EXPERIMENT command (see *Section 13.3.1*). 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.

Synopsis 1: SET_WEIGHT <weight value> <selection>

Synopsis 2: SET_WEIGHT

Ensuing 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.

13.3.10 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>.

Synopsis 1: TRANSFER_START_VALUE <equilibrium number>

Synopsis 2: TRANSFER_START_VALUE

Ensuing 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.

13.3.11 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.

Synopsis 1: STORE_ALL_WEIGHTS <weight-set name>

Synopsis 2: STORE_ALL_WEIGHTS

Ensuing Prompt: Command line: <weight-set name>
Give a weight-set name (comment line texts) for the current weight set for all experimental points.

13.3.12 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.

Synopsis 1: RESTORE_ALL_WEIGHTS <weight-set number>

Synopsis 2: RESTORE_ALL_WEIGHTS

Ensuing 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.

13.3.13 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.

Synopsis: LIST_ALL_EQUILIBRIA

13.4 Other Commands in the Experimental Data (POP or DOP) Files

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 part (*i.e.*, in Sections 13.3.1 through 13.3.3) 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.

Please note that there have been three major modifications/improvements since TCCS:

- The ADVANCED_OPTIONS command replaces the old SPECIAL_OPTIONS command;
- It is possible to turn Global Minimization Technique on for (some) individual experimental equilibrium points during optimizations using the POLY/ED_EXP command-sequence ADVANCED-OPTION TOGGLE-ALTERNATE, while the Global Minimization Technique is normally turned off and thus the Ordinary POLY Minimization (GEM) is used for equilibrium calculations during optimizations.

- In the POP file and in the ED-EXP module, one can use the new type of symbols in the ENTER-SYMBOL command for ordered phases (treated by the so-called *Four Substitutional Sublattice Ordering Model*) which have the phase-type of F or B. This is called INTERNAL and can be specified as L12, L10, D019, B19, D03, L21, B2 or B32; in addition, a phase and its composition-set must be specified. The symbol value will be zero if the phase is not ordered according to the specification; otherwise, it will have a positive value depending on the degree of order, with a maximum value of unity. This is useful when optimizing ordered phases (such as ordered FCC, HCP and BCC phases), as the order may change during the optimization.

Most of the special commands for the ED_EXP module are also often used in a 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           (i.e., 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.

13.4.1 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.

Synopsis:

```
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 TABLE-HEAD 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 an @ 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.

13.4.2 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.

Synopsis: COMMENT <comment>
A comment can have maximum about 60 characters, written on the same line.

13.4.3 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-EXPERIMENT 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.

Synopsis: FLUSH_BUFFER

13.4.4 GRAPHICS_PLOT

Description: Previously in TCCP (only some early sub-versions), this command was useful in the ED-EXP module for directly conducting graphical plotting and it was equivalent to the POLY command STEP_WITH_OPTIONS in order to make a STEPPING calculation based on the currently-selected equilibrium point and to plot the desired property diagrams. For the details, please refer to the *TCCP Users' Guide*.

Since TCCQ (and late sub-versions/patches of TCCP), it has been completely rewritten, and it now has totally different functionality. It can now help 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.

Synopsis 1: 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.

14 System Utility Module (SYS)

14.1 Introduction

The classical versions of both Thermo-Calc and DICTRA software have a so-called **System Utility Module** \Leftrightarrow (under the SYS prompt), which provides the primary controls on inter-module communication, MACRO-file creation and operation, working and plotting environmental setting, and command information searching. They are essential for properly performing ordinary calculations, desirably obtaining calculated results, and easily conducting various tasks.

It also facilitates some odd features, such as user interface setting, command unit setting, error reporting preference, terminal characteristics definition, workspace listing, open or close of a file through a unit, interactive calculator, news retrieval, etc. Some of such odd commands are used for performance preference of the users, and some are designed for debugging of the programmers. Few odd commands are included only for some special purposes, which might have been obsolete in later versions.

The following commands are available in the SYS module:

SYS:?		
BACK	INFORMATION	SET_LOG_FILE
CLOSE_FILE	MACRO_FILE_OPEN	SET_PLOT_ENVIRONMENT
EXIT	OPEN_FILE	SET_TC_OPTIONS
GOTO_MODULE	SET_COMMAND_UNITS	SET_TERMINAL
HELP	SET_ECHO	STOP_ON_ERROR
HP_CALCULATOR	SET_INTERACTIVE_MODE	
SYS:		

Notes for versions since TCCR:

- \Leftrightarrow Five SYS commands have been taken away from the SYS-Module monitor (and also eliminated from this chapter in the *TCC User's Guide*), because they are normally for system debugging purpose only, or obsolete now, or totally irrelevant. Such commands include: NEWS, LIST_FREE_WORKSPACE, PATCH, SET_ERROR_MESSAGE_UNIT and TRACE.
- \Leftrightarrow The *Section 14.3 (Display of General Information)* has been completely removed from this chapter in the *TCC User's Guide*. However, for the purpose of obtaining information about various subjects in the SYS module, please use the on-line help feature of the TCCR software, i.e., typing the INFORMATION <SUBJECT> command sequence in the SYS module, that will follow a display on screen of some brief information on a certain subject related to the Thermo-Calc software/database/interface package.
- \Leftrightarrow The FOP Module (FUNC_OPT_PLOT, the simple function optimizer and plotting facility; for function/parameter mathematic-fitting and plotting) has been completely eliminated from the TCC software. One should instead use the PARROT module to perform critical thermodynamic assessments.
- \Leftrightarrow In the meantime, the SYS command SET_ECHO (that has been *blindly* existing in the TCC and DICTRA software for a very long time) is documented in the current edition of this chapter in the *TCC User's Guide*; see Sections 14.2.6 and 14.2.8.

Also note that since TCCS the following two more SYS commands have been added to the SYS-Module monitor:

- \Leftrightarrow SET_ECHO command for showing the full meaning of each of executed commands enforced by MACRO files;
- \Leftrightarrow SET_TC_OPTIONS for user-specified definitions of the default values for three general options for performing single-points, stepping and mapping calculations, i.e., "USE_GLOBAL_MINIMIZATION /YES/", "GLOBAL_TEST_INTERVAL /0/" and "AUTOMATICALLY_ADD_INITIAL_EQUILIBRIA /YES/".

\Leftrightarrow Revision History of the SYS Module User's Guide:

Mar 1985	First release (Edited by Bo Sundman)
Oct 1993	Second revised release (Edited by Bo Sundman)
Sept 1996	Third revised release (Edited by Mikael Schalin and Bo Sundman)
Jun 2000	Fourth revised and extended release (Edited by Pingfang Shi)
Nov 2002	Fifth revised release (Edited by Pingfang Shi)
Jun 2004	Sixth revised and extended release (Edited by Pingfang Shi)
Aug 2006	Seven revised and simplified release (Edited by Pingfang Shi); with minor changes in 2008

14.2 General Commands

14.2.1 HELP

Description: This command lists the available commands or gives an explanation of the specified command. An extended description of the specified command will be given.

The commands can be abbreviated. The underscore character or the hyphen can be used to separate the words of the command. The commands are terminated by a space and then the user may type any argument to the command on the same line. The user may also hit the <RETURN> key after the command and will then be prompted for the arguments. The arguments must be given in the order specified by the prompting of the program.

Synopsis 1: HELP <command name>

Synopsis 2: HELP

Ensuing Prompt: COMMAND: <command name>

Options: command name -- the name of an SYS-module command to obtain help.

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

14.2.2 INFORMATION

Description: This command gives information about the features of the Thermo-Calc software system. Basic information about various subjects (concepts and models) can be obtained, as they are described in different parts of this manual. For brief information displayed by this command, please also see *Section 14.4*.

Synopsis: INFORMATION

Ensuing Prompt: WHICH SUBJECT /PURPOSE/: <subject name>

The name of the subject must be given. Extensive information is available for various subjects as listed below (this list can be seen if typing a question mark "?"):

PURPOSE (Introducing the THERMO-CALC Software Package)	
COMPUTATIONAL THERMODYNAMICS	
TCC - THERMO-CALC CLASSIC	TCW - THERMO-CALC WINDOWS
TC4A - THERMO-CALC FOR ACADEMIC	TC4U - THERMO-CALC FOR UNIVERSITY
MODELS IN THERMO-CALC	MODULES OF THERMO-CALC
DATABASES IN THERMO-CALC	FUNCTIONALITY OF THERMO-CALC
STATE VARIABLES	DERIVED VARIABLES
PHASE DIAGRAMS	PROPERTY DIAGRAMS
TDB (DATABASE RETRIEVAL)	GES (GIBBS_ENERGY_SYSTEM)
POLY (EQUILIBRIUM CALCULATIONS)	POST (POST_PROCESSOR)
PARROT (ASSESSMENT)	ED_EXP (EDIT_EXPERIMENT)
BIN (BINARY_DIAGRAM)	TERN (TERNARY_DIAGRAM)
POT (POTENTIAL_DIAGRAM)	POURBAIX (POURBAIX_DIAGRAM)
TAB (TABULATION)	CHEMICAL EQUATION
SCHEIL (SCHEIL_SIMULATION)	REACTOR (REACTOR_SIMULATOR)
SYS (SYSTEM.Utility)	FOP (FUNCTION_OPT_PLOT)
USER INTERFACE OF THERMO-CALC	GUI (GRAPHICAL_USER_INTERFACE)
APPLICATIONS OF THERMO-CALC	THERMO-CALC ENGINE
API - PROGRAMMING INTERFACE	TQ/TCAPL INTERFACES
TC-TOOLBOX IN MATLAB SOFTWARE	TCMI MATERIALS INTERFACE
GLOBAL (Global Minimization Technique in Thermo-Calc software)	
DICTRA (Diffusion-Controlled Transformation Simulation Software)	
HELP (How to get on-line help in the TCC software)	
NEWS (Revision History and New Features of the TCC Software)	

14.2.3 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 (also see Section 5.4.11).

Synopsis 1: GOTO_MODULE <module name>

Synopsis 2: GOTO_MODULE

Ensuing Prompt: MODULE NAME:

```
NO SUCH MODULE, USE ANY OF THESE:  
SYSTEM_UTILITIES  
GIBBS_ENERGY_SYSTEM  
TABULATIONREACTION  
POLY_3  
BINARY_DIAGRAM_EASY  
DATABASE_RETRIEVAL  
REACTOR_SIMULATOR_3  
PARROT  
POTENTIAL_DIAGRAM  
SCHEIL_SIMULATION  
POURBAIX_DIAGRAM  
TERNARY_DIAGRAM  
MODULE NAME: <module name>
```

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

14.2.4 BACK

Description: This command gives the control back to the most recent module. It will, in some cases, be necessary to switch back and forth between two modules of the Thermo-Calc software, and this command is then simpler than the more general command GOTO_MODULE.

Synopsis: BACK

14.2.5 EXIT

Description: This command terminates the program and returns to the operating system. Unless a SAVE command has been executed (in either the GES, POLY3, PARROT or REACTOR module), all data and results will be lost.

Synopsis: EXIT

14.2.6 SET_ECHO

Description: This command is specially designed for uses within MACRO files of the TCC (and DICTRA) software. By simply adding this command at the every 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 details of various commands in all the sequential operations in the TCC (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.

Synopsis: SET_ECHO

14.2.7 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. See more details in *Section 14.2.7*.

Synopsis: SET_LOG_FILE

Ensuing Prompt: OUTPUT FILE: <file name>

The name of the file must be given, where all the sequent commands will be saved.

Notes: Under Windows Vista/XP/2000/NT4 environments, a **Save As** window will be popped 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 properly specified, as shown in *Figure 14-1*. 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 **Save as type** box) can not be changed. By pressing the **Save** button, the program proceeds with saving all the subsequent commands in the specified *.LOG file.

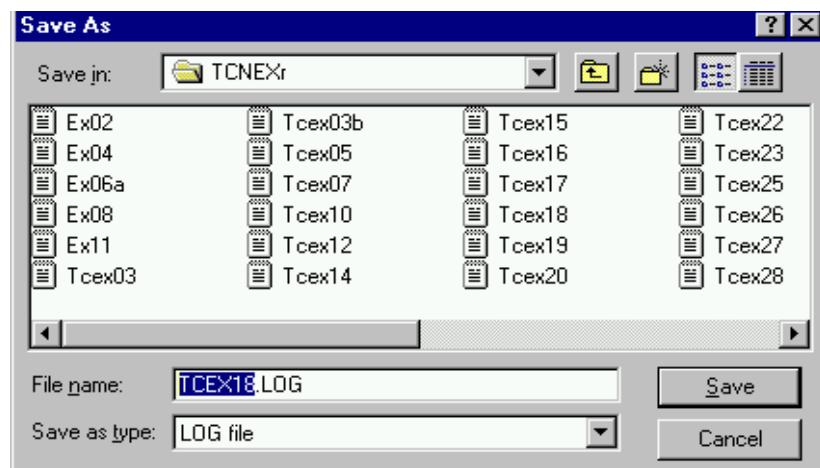


Figure 14-1. The “Save As” window: Saving all the subsequent commands in an *.LOG file.

14.2.8 MACRO_FILE_OPEN

Description: MACRO is an extraordinary convenient and easier way of pre-defining sequences of various legal TCC (and DICTRA) commands stored in a so-called **MACRO file** (which is a simple textual file normally with the default extension “TCM” for the TCC software 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 TCC software; SYS, POLY, PARROT and DICTRA_Monitor module in the DICTRA software).

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 TCC software, if in the SYS module the **SET_LOG_FILE** command is used and a **LOG file** name is given before any other SYS/TDB/TAB/GES/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 current TCC session 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 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 TCC (and DICTRA) program.

All kinds of legal TCC (and DICTRA) commands and their required inputs can be stored into a MACRO file. A MACRO file must be terminated with the **EXIT** command, or can be interrupted (for the purposes of allowing further interactive operations by the user) in the SYS, GES, POLY, PARROT or POST modules with the **SET_INTERACTIVE** command.

Within a MACRO file (for previous versions up to TCCR), one could have as many as possible **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 TCC commands when the file is called by the TCC software.

However, this might be considered as somewhat tedious. Therefore, it is now (since TCCS) possible to have multiple-line **comment-blocks** inside a MACRO file, using the paired “@(“ and “@)” signs (*i.e.*, the indicators that starting the lines for the “*begin comment*” and “*finish comment*”, respectively). 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 TCC 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 2nd 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, a textual copy of 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, e.g.

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

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 TCC software from temporarily stopping at any pause by simply typing any character (except for the Y character) after specifying the name of a MACRO file.

Since TCCN, 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 TCC 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_Module command **SET_ECHO** at the every 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 TCC (and DICTRA) software, that are enforced according to the MACRO file(s).

Synopsis 1: MACRO_FILE_OPEN <name of a Macro file>

Synopsis 2: MACRO_FILE_OPEN

Ensuing Prompt: Macro filename: <name of a Macro file>

Specify the filename with the MACRO command. The default extension is “TCM”.

Notes: Under Windows Vista/XP/2000/NT4 environments, if an appropriate MACRO file is not given after the command, an **Open file** 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 14-2*.

The file type (*i.e.*, TCM, as in the **Files of type** box) can not be changed. By pressing the **Open** button, the program proceeds with executing various Thermo-Calc commands. The user may also cancel such an **Open file** window session, and thus the current MACRO file will not be opened.

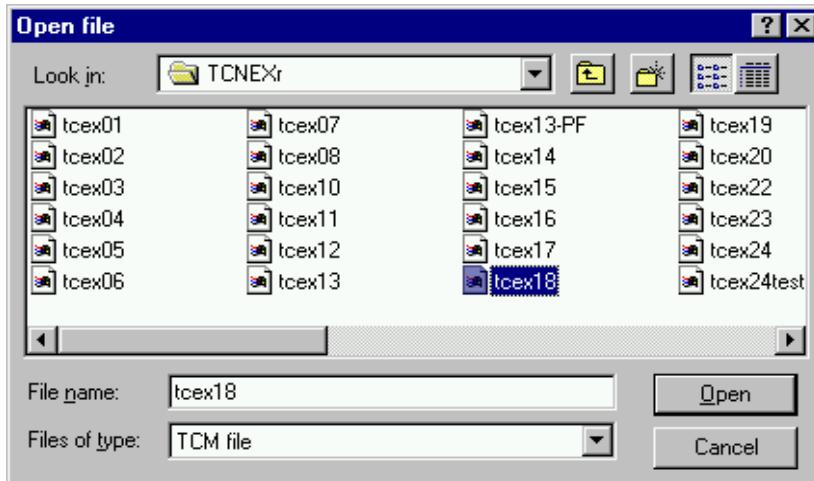


Figure 14-2. The “Open file” window: Opening an existing TCM file.

If the MACRO file contains some SYS/TDB/TAB/GES/POLY/POST/PARROT/ED_EXP-module commands for setting *.LOG files, saving/reading GES5/POLY3/PARROT workspaces, switching USER databases, compiling experiments (from existing *.POP files), creating new *.PAR files, appending experimental data *.EXP files, plotting/dumping diagrams, *etc.*, a corresponding window (*e.g.*, **Save As**, **Open file**, **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.

More Notes: 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, *i.e.*, 1 (under Windows Vista/XP/2000/NT4) or 9 (under Linux and all types of UNIX platforms).

For more extensive examples, please refer to various TCM files on the TCC installation area \TCEX\ and in the **TCCS Examples Book**.

14.2.9 SET_TC_OPTIONS

Description: This new command since TCCS/TCW5 lets the user to pre-define the “default” values of three general options for performing all the sequential single-points, stepping and mapping calculations.

Synopsis: **SET_TC_OPTIONS**

Ensuing 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 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 nth 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 (TCCS/TCW5) is installed 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 two (2) axis-variables have already been defined by the POLY command SET_AXIS_VARIABLE.

14.2.10 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: This command must be terminated with an empty line or two commas.

Synopsis: `SET_PLOT_ENVIRONMENT`

Ensuing Prompt: `DEFAULT PLOTDEVICE NUMBER /default number/ : <device number>`

The number given here is unique for each type of graphical device. For plotting Thermo-Calc graphs on SCREEN, the default device number should normally be selected, *e.g.*, 1 for MS-Windows or IBM-VGA on PC Windows Vista/XP/2000/NT4, and 9 for X-Windows on PC-Linux and all types of UNIX platforms. Other available device numbers (*e.g.*, PostScript portrait mode, PostScript landscape mode, HP7550 plotter, *etc.*) are for hard copies.

A question mark "?" will always give you a list of all available graphical devices used in Thermo-Calc.

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 Output: `SET_PLOT_ENVIRONMENT 1 lasp 5 a0tr,,`

This example sets the default plot device number to 1 (normally as 1 for PC-Windows, and 9 for PC-Linux and all types of UNIX platforms), and defines an alias named `lasp` with plot device 5 and connected to a printer named `a0tr`.

14.3 Odd Commands

14.3.1 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.

Synopsis: SET_INTERACTIVE_MODE

14.3.2 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 (see *Section 14.3.8*).

NOTE: The first two lines of an input file will be skipped before any input is read from an input file.

Synopsis: SET_COMMAND_UNIT

Ensuing 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.

14.3.3 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.

Synopsis: STOP_ON_ERROR <argument>

Give the preferred argument (ON or OFF) after the STOP command.

14.3.4 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.

Synopsis: OPEN_FILE

Ensuing Prompt: FILE NAME: <file name>

A legal file name must be specified.

14.3.5 CLOSE_FILE

Description: A previously opened text file is closed by this command.

Synopsis: CLOSE_FILE

Ensuing Prompt: UNIT NUMBER: <unit number>

A unit number previously given by the program in an OPEN_FILE command must be specified.

14.3.6 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.

Synopsis: SET_TERMINAL

Ensuing 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).

14.3.7 HP_CALCULATOR

Description: This is the QBA simple interactive calculator using reverse polish notations:

```
REVERSE POLISH CALCULATOR
0.0000000E+00
HPC>?
This is a calculator with HP flavour
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

Example input: 30000 8 / 1273 / chs 1.5 3 ^ + exp 2 *
Computes 2*EXP(1.5**3-30000/(8*1273))
```

Synopsis: HP_CALCULATOR

Notes: All available OPCODEs (HPC codes) are listed below:

HPC>HELP					
BACK	CLX	HELP	RCL_REG	SQRT	[ACOS
CHSIGN	COS	LN	ROT_STACK	STO_REG	[ASIN
CLEAR_REG	DISPLAYREG	LOG	SHOW_STACK	SWITCH_XY	[ATAN
CLSTACK	EXP	POWER_2	SIN	TAN	

HPC>

One must use the BACK command to quit the HP_CALCULATOR, in order to go back to the SYS module.

15 Dataplot Graphical Language (DATAPLOT)

Started from the TCCR version, this chapter had been extracted and separately prepared as an individual document “**DATAPLOT User’s Guide**”.

>>> Open the document [DATAPLOT User’s Guide & Examples](#)

(This page is intended to be empty)

16 Reference Lists on Thermo-Calc Package and Applications

Started from the TCCR version, this chapter had been extracted and separately prepared as an individual document “*Thermo-Calc Reference List*”.

>> Open the document [Thermo-Calc Reference List](#)

(This page is intended to be empty)

17 Appendices and Indices

The following appendices are included in this chapter:

- Appendix A. *Thermo-Calc Classic File Types and Relations*
- Appendix B. *Thermo-Calc Classic Quick Reference Card*
- Appendix C. *Units for State Variables and Derived Variables*
- Appendix D. *On Reference States and Standard States*

This chapter additionally gives two index lists for *all Thermo-Calc commands* (*used via various TCC modules and in various types of textual files*, respectively), as well as lists of *figures and tables* located in varied parts of this TCCS User's Guide (and those in chapters that had been extracted from the previous edition of the TCC User's Guide and had been moved to four separate documents since TCCR).

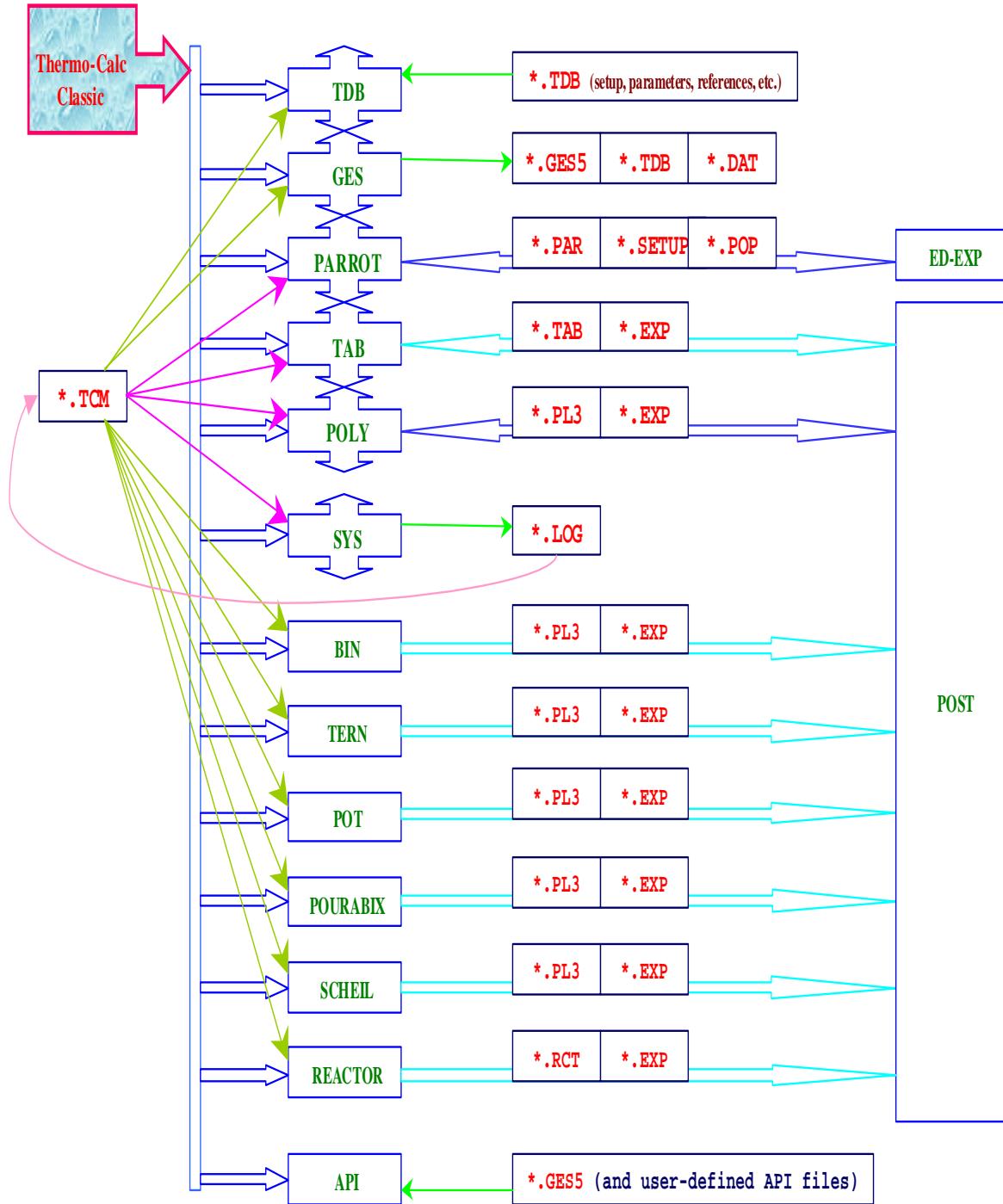
The indices on some terms used in this manual (that used to be within the TCCP/TCCQ User's Guides) have been taken away from this edition.

Some other forms (*e.g., TCC/TCW/DICTRA Course Application Form, TCS Order Option Form, and License Delete Guarantee*) may be published on our web site (www.thermocalc.com). If interested and desired, you can download and fill the forms, and then return to us.

For other forms that you might require, please contact us at:

- ☞ info@thermocalc.se for information on various software/databases/interfaces and their applications;
- ☞ support@thermocalc.se for technical issues, and for bug-problem reporting and suggestions/comments;
- ☞ consult@thermocalc.se for consultancy specifications.

Appendix A. Thermo-Calc Classic File Types and Relations



Appendix B. Thermo-Calc Classic Quick Reference Card

Thermo-Calc Classic Quick Reference Card

Examples of simple MACRO files:

Example 1: Calculation of the binary Al-Si phase diagram

```

go data
switch KP
define_system al_si
get
  ... go to the TDB (database retrieval) module;
  ... switch to the Kaufman binary alloy database;
  ... define the Al-Si binary system;
  ... get data from the database for the system;
  ... go to the POLY (equilibrium calculation) module;
  ... set mole fraction of Si as one axis variable
  ... and temperature the other for mapping;
  ... save the POLY workspace to a file called alsi
  ... map the diagram;
  ... go to the POST (post-processor) module;
  ... set the Si weight percent as one diagram axis
  ... and temperature (°C) as the other;
  ... plot the diagram on screen;
  ... set on the interactive mode.

```

Example 2: Calculation and plot of oxide layers on a stainless steel

```

go data
switch SSOL
define_system fe c cr v mn si
rej ph /all
res ph fcc hcp cem m23 m7 gra ...
reject all phases;
get
  ... get the steel data from the SSOL database;
append SSUB
  ... append oxide/gas data from SGTE substance database;
define_system fe c cr v mn si
get
  ... define the steel Fe-Cr-Mn-Si system;
  ... get the oxide & gas data from the SSUB database;
  ... go to the POLY (equilibrium calculation) module;
  ... set the equilibrium conditions T-P and acr(O);
  ... set the composition (in weight percent) of the alloying
  ... elements Cr-C-V-Mn-Si in the steel;
  ... set the rest as Fe composition, equivalent to bf(e)=82.3;
  ... set the ref-state for O as in gas at current T & 1 bar;
  ... calculate the equilibrium;
  ... list the calculated equilibrium;
  ... set acr(o) as the stepping variable;
    (note that 2* means a step as a factor of 2)
  ... go to the POST (post-processor) module;
  ... set the acr(o) as one diagram axis,
  ... and mass percent of all stable phases as the other;
  ... step the diagram (with the option NORMAL) for an
  ... automatic calculation of several equilibria;
  ... go to the POST (post-processor) module;
  ... set the acr(o) as one diagram axis,
  ... and mass percent of all stable phases as the other;
  ... set the curve-labeling option D On;
  ... set the x-axis scaling in common logarithmic;
  ... plot the diagram on screen;
  ... set on the interactive mode.

```

```

save oxides
step normal

```

```

post
set_dia_axis x acr(o)
set_dia_axis y bp(*)
set_label curve d
set_axis_type x log
plot screen
set_interactive

```

For more information, please consult the TCC User's Guide, Examples Book, and technical service via
Tel: +46-8-790 9718; Fax: +46-8-673 3718; E-mail: support@thermocalc.se; http://www.thermocalc.se

Modules, user interface and commands

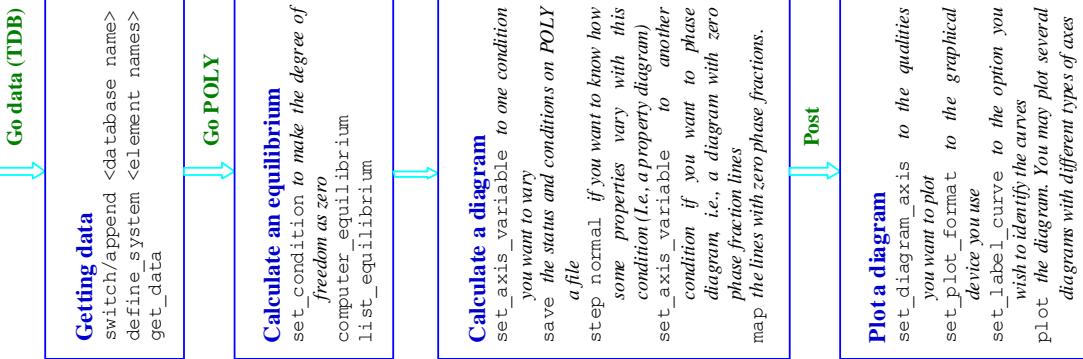
The Thermo-Calc consists of several modules and the most important are the database module, the POLY module for equilibrium calculations, the post processor for plotting and the Gibbs energy system (GES) for listing and manipulating data. In all modules there is a HELP command which describes the commands and an INFORMATION command which gives more general information. Most commands are quite long and may consist of several words connected with a hyphen or underscore (hyphen and underscore are treated as identical). The commands can be abbreviated and each word can be abbreviated separately. The command may ask questions and on-line help is provided by pressing "?". The GOTO and BACK commands are used to go back and forth between various modules.

Setting conditions

The whole secret of how to use Thermo-Calc is to learn and understand the very flexible way you can set conditions. Each condition is separate and set by specifying values of state variables. Thermo-Calc has a large number of state variables (see next section) to be used. Thermo-Calc also provides predefined ways in some advanced and easy-to-use modules to calculate specific things such as binary and ternary phase diagrams, potential diagrams, Poubalax diagram, Scheil-Gulliver solidification path, etc. In ordinary situations, you must yourself set the appropriate conditions for each calculation. However, you can create MACRO files with the commands you need for specific calculations, the enclosed two examples on the other side of this page are simple MACRO files.

The number of conditions to be set are given by the Gibbs phase rule, the number of components plus two (for temperature and pressure). With two components you must thus set 4 conditions. The simplest conditions are the temperature, pressure and the amounts of each component. In some cases one prefers to use fractions of the minor components rather than total amounts and then one may specify that the system is closed by giving the total size of the system in moles or mass.

TCC Calculation Flowchart



But you may use any kind of combination of conditions on the activities, chemical potentials, enthalpies, etc. Note that the components are the elements unless you define other yourself. You can set amounts of other species than the components by the `set_input_amount` command! You can delete a condition by setting the condition value to NONE.

State variables

In Thermo-Calc you use thermodynamic state variables to set conditions, for example `set_condition t=1.273`. They are also used in the post processor to plot results and their values can be obtained with the `show` command.

The state variables are named according to generally accepted standards. The most important state variables are: `T` for temperature (K), `P` for pressure (Pa), `N` for total number of moles, `ACR(<component>)` for chemical potential (`J/mol`), `X(<component>)` for overall mole fraction of a component, `W(<component>)` for overall mass (weight) fraction of a component.

Note that there is no way to specify percent of mole or mass as a condition but it can be obtained on output with the `set_diagram_axis` command, or by a function.

The value for any thermodynamic quantity can be set or obtained by using a symbolic name, for example (all values in SI units): `G` for the Gibbs energy, `S` for the entropy, `H` for the enthalpy, `V` for the volume, `A` for the Helmholtz energy.

The state variables `G`, `H`, etc. are the values for the whole system. You may normalize these quantities per mole, mass or volume using for example `GW` for the Gibbs energy of the system per mole of components, `GW` for the Gibbs energy for the system per mass and `GV` for the Gibbs energy for the system per volume. You may set or obtain the values for individual phases by using `G(<phase>)` which is the Gibbs energy for the current amount of `<phase>`. (Note that `G(<phase>)` is zero if the amount of the phase is zero). A normalized quantity is independent of the amount thus `GW(<phase>)` will be the Gibbs energy per mole component, `GW(<phase>)` the Gibbs energy per mass and `GR(<phase>)` the Gibbs energy for `<phase>` per formula unit, etc.

The amount of a component can be given by `N(<component>)` for the number of moles or `B(<component>)` for the mass of `<component>`. Just `N` or `B` represents the total size of the system.

The amount of a phase can be set or obtained as the amount of moles of components in the phase by the state variable `NP(<phase>)` or in mass of components in the phase by `BP(<phase>)`. The value of `NP` is independent of the formula unit of the phase but may depend on what components you have defined. `BP` and `NP` can be normalized, `BPV(<phase>)` is the density of `<phase>`.

The driving force of a phase can be obtained by the state variables `DGM(<phase>)` or `DGF(<phase>)`. The value is divided by RT and thus dimensionless. Note that `DGM` and `DGF` cannot be used as conditions.

The fraction of the different constituents of a phase is a model dependent quantity but can be set or obtained with the state variable `Y(<phase>,<species>)`. When you must specify the sublattice use a hash sign, "#", followed by the sublattice number.

A unique feature of Thermo-Calc is the facility to set conditions by giving of quantities for individual phases. In addition to the thermodynamic quantities above you may use, for example:

`W(<phase>,<component>)` for the mass fraction of `<component>` in `<phase>` `X(<phase>,<component>)` for the mole fraction of `<component>` in `<phase>`

You may calculate partial derivatives of state variables by using the dot notation "...". For example `H,T` is the heat capacity. You may calculate more elaborate derivatives for example `BP(LIQUID)` `W(C)` which will give how much the mass of liquid will change with a unit change of the mass fraction of carbon. The state variables can be used as conditions or as axes in the post processor. In that case it is often interesting to use an asterisk, "*" to denote all phases or components. You may define functions of state variables and use these as axes in the post processor.

Hints and Frequently Asked Questions (FAQ)

How can I get the partial pressure of a gas species? **Answer:** You may enter a function `P* Y(GAS, <species>)` and this will be the partial pressure of `<species>` if the gas is stable. If gas unstable, you must enter a function `P* Y(GAS, <species>)*EXP(DGF(GAS))`.

The activity of any species, for example `H2O`, can be obtained by entering a function as `ACR(H)*2*ACR(O)`. Remember to set `_reference_state` for the components `H` and `O`! Note that `ACR` can only be used for the components.

What is wrong when I get crossing tie-lines in my phase diagram but no miscibility gap? **Answer:** The crossing tie-lines are signs of a miscibility gap. However, you have to inform Thermo-Calc in advance that you know, or suspect, that there is a miscibility gap in phase. This can be done with the `POLY` command `special_option (with the set_misibility_gap option) or GES command _amend_phase_description (with the composition_sets option) where you can give part two or more composition sets. Then you must start the calculation where the miscibility gap is stable and use the set_all_start_values command to make both composition sets stable but with different initial compositions.`

How can I obtain the value of the excess Gibbs energy? **Answer:** There is no way to obtain the excess Gibbs energy from `POLY` as this is a model dependent quantity. You may obtain the Gibbs energy of mixing, and for a substitutional regular solution model that is the same thing. In order to have, for example, the enthalpy of mixing in the liquid for a ternary system A-B-C, give the following commands:
`SET REF STATE A LIQUID * 1E5
SET REF STATE B LIQUID * 1E5
SET REF STATE C LIQUID * 1E5
SHOW HMR(LIQUID)`

The suffix R requires that the user defines which reference states should be used. But if you have sublattices with mixing on two or more sublattices the Gibbs energy of mixing is different from the excess Gibbs energy.

How do I calculate the melting temperature of an alloy? **Answer:** Calculate first an equilibrium with the given composition at a temperature well below the melting temperature. Then give the commands `change_status liqid=fix 0, set_condition t=none and compute_equiv`. The temperature is obtained with a `show T` or `list_equilibrium` command.

How do I get partial quantities for a component? **Answer:** Taking a Fe-bearing alloy as an example, the partial quantities of the component Fe can be obtained by the `SHOW <value> command, with the value MUR(FE) for its partial Gibbs energy, MUR(FFE).T for partial entropy, and HFE for partial enthalpy (note you have to first define HFE by ENTER VARIABLE HFE = MUR(FFE)-T*MUR(FFE).T).`

Appendix C. Units of State Variables and Derived Variables

General Introduction

Units for various thermodynamic, physical, chemical and transport quantities are important concepts for appropriately setting standard state variables (intensive and extensive) and derived variables, utilizing pre-defined symbols, and entering user-defined symbols (constants, functions, variables and tables).

A comprehensive understanding on units of various quantities for a system, for a system component, for a phase, and/or for a phase constituent (on a sublattice) can really help in setting calculation/simulation conditions, defining stepping/mapping variables, in tabulating substance/reaction properties, and in generating graphical and textual outputs with calculated/simulated results.

A user may also get detailed information on unit definitions for various quantities by looking through the [Part 2 in the document Thermo-Calc Software System](#) and [Chapter 8 \(Section 8.3\) of the TCCS User's Guide](#), or by entering the following subject keywords, which are relevant to the UNITS subject:

STATE VARIABLES	For Standard State Variables
INTENSIVE VARIABLES	For Various Intensive State Variables
EXTENSIVE VARIABLES	For Various Extensive State Variables
DERIVED VARIABLES	For Derived Variables (Partial Derivatives)
CONDITIONS	For Condition Settings
AXIS-VARIABLES	For Stepping/Mapping Variable Settings

Unlike in the TCW software (where different units can be chosen for various quantities), the TCC (and DICTRA) software always uses the default units for standard state variables and derived variables in various modules. However, it allows users to call some pre-defined symbols (but only legal in the POST_PROCESSOR module) and/or to additionally define desired user-specified symbols (constants, functions, variables or tables) using the valid standard state variables and derived variables.

For details on default units of standard state variables, derived variables and pre-defined symbols, as well as on alternative units of user-specified symbols, for a system, for system components, for phases, for components in a given phase and for species in a given phase, please refer to the subject keywords below:

BASIC UNITS	For Variables that are globally used
SYSTEM UNITS	For Variables of a Defined System
COMPONENT UNITS	For Variables of a Specific Component in the System
PHASE UNITS	For Variables of a Phase in the System
PHASE-COMPONENT UNITS	For Variables of a Specific Component in a Phase
SPECIES UNITS	For Variables of a Specific Species in a Phase
USER UNITS	For Symbols that are defined by a user

Basic Units: For Variables that are Globally Used

Within the POLY module (and its sub-module POST_PROCESSOR), the following basic units are mostly in SI units and can be used (through various standard state variables, derived variables, and pre-defined symbols):

Quantity	Mnemonic	Basic Unit
Temperature	T	Kelvin (K)
Pressure	P	Pascal (Pa)
Amount	N	moles (mol)
	M_F	mole fraction (<i>dimensionless</i>) (only in POST module)
	M_P	mole percent (mol%) (only in POST module)
	B	mass (grams, g)
	W_F	mass fraction (<i>dimensionless</i>) (only in POST module)
	W_P	mass percent (mass% = wt%) (only in POST module)

Volume	V	cubic meters (m^3)
Energy	G, A, U, H	Joules (J)
	S	Joules/Kelvin (J/K)
Curie Temperature	TC	Kelvin (K)
Bohr magneton number	BMAG	(dimensionless)

Notes: The M-F, M-P, W-F and W-P quantities can be used only in the POST module

System Units: For Variables of a Defined System

The default units associated with standard state variables, derived variables, and pre-defined symbols of a system are defined as listed below:

Quantity	SV & Unit [§]	DV & Unit [¶]	PS & Unit ^θ
Temperature	T K		T_C C (Celsius) T_F F (Fahrenheit)
Pressure	P Pa		
Amount	N mol	NW mol/g NV mol/m ³	
	B g	BM g/mol BV g/m ³	
Volume	V m ³	VM m ³ /mol VW m ³ /g	
Gibbs Energy	G J	GM J/mol GW J/g GV J/m ³	
Helmholtz Energy	A J	AM J/mol AW J/g AV J/m ³	
Internal Energy	U J	UM J/mol UW J/g UV J/m ³	
Enthalpy	H J	HM J/mol HW J/g HV J/m ³	
Entropy	S J/K	SM J/mol/K SW J/g/K SV J/m ³ /K	
Heat Capacity		HM.T J/mol/K HW.T J/g/K HV.T J/m ³ /K	
Thermal Expansivity		VM.T m ³ /mol/K VW.T m ³ /g/K	
Isothermal Compressibility		VM.P m ³ /mol/Pa VW.P m ³ /g/Pa	

Notes:

[§] SV stands for standard state variables of the whole system, and they are used in the POLY and POST modules;

[¶] DV stands for derived variables of the whole system, and they are used in the POLY and POST modules;

^θ PS stands for pre-defined variables of the whole system, and they can only be in the POST module;

Normalizing suffixes like M (per mole of components), W (per mass in grams) or V (per volume in m^3) can be applied to extensive quantities V, G, A, U, H and S of the whole system (except for VW).

The extensive quantity N can have normalizing suffixes W or V (but not M), while B can have M or V (but not W), and under such circumstances they present certain special quantities of the whole system, e.g.,

NW	= moles per mass of the system	(mol/g)
NV	= moles per volume of the system	(mol/m ³)
BM	= mass per mole of the system	(g/mol)
BV	= mass per volume of the system	(g/m ³ , i.e., density)

NM and BW for overall amount (size) of the system are always as unity (so it is unnecessary to evaluate them). NW and BM should not be set as conditions.

All such normalized quantities are calculated via the first derivatives of the corresponding system extensive variable with regard to the total system size [in terms of N or B or V] in the whole system.

The heat capacity of the system stands for C_p (when pressure is a condition) or for C_v (when volume is a condition).

Component Units: For Variables of a Specific Component in the System

The default units associated with standard state variables, derived variables, and pre-defined symbols of a specific system component in the system are defined as listed below:

Quantity	SV & Unit [§]	DV & Unit [¶]	PS & Unit [◊]
Amount	N (c) mol	X (c) dimensionless	M-F (c) dimensionless
			M-P (c) dimensionless
		NM (c) dimensionless	
		NW (c) mol/g	
		NV (c) mol/m ³	
	B (c) g	W (c) dimensionless	W-F (c) dimensionless
			W-P (c) dimensionless
		BM (c) g/mol	
		BW (c) dimensionless	
		BV (c) g/m ³	
Chemical Potential	MU (c) J/mol		
	MUR (c) J/mol		
Activity	AC (c) dimensionless		
	ACR (c) dimensionless		
ln(Activity)		LNAC (c) dimensionless	
		LNACR (c) dimensionless	

Notes:

[§] SV stands for standard state variables of the specific system component, and they are used in the POLY and POST modules;

[¶] DV stands for derived variables of the specific system component, and they are used in the POLY and POST modules;

[◊] PS stands for pre-defined variables of the specific system component, and they can only be in the POST module; Normalizing suffixes like M (per mole of components), W (per mass in grams) or V (per volume in m³) can be applied to extensive quantities N (c) and B (c) of the specific system component; and under such circumstances they present certain special quantities of the specific system component, e.g.,

$$NM(c) = \text{mole fraction of the component} \quad (\text{dimensionless})$$

$$= X(c)$$

$$NW(c) = \text{moles of the component per mass of the system} \quad (\text{mol/g})$$

$$NV(c) = \text{moles of the component per volume of the system} \quad (\text{mol/m}^3)$$

$$BM(c) = \text{mass of the component per mole of the system} \quad (\text{g/mol})$$

$$BW(c) = \text{mass fraction of the component} \quad (\text{dimensionless})$$

$$= W(c)$$

$$BV(c) = \text{mass of the component per volume of the system} \quad (\text{g/m}^3)$$

NW (c) and BM (c) should not be set as conditions.

All such normalized quantities are calculated via the first derivatives of the corresponding component extensive variable with regard to the total system size [in terms of N or B or V] in the whole system.

The suffix R presents that intensive quantities MU(c), AC(c) and LNAC(c) to be calculated with respect with a certain reference state chosen for the system component, such as MUR(c), ACR(c) and LNACR(c).

Phase Units: For Variables of a Specific Phase in the System

The default units associated with standard state variables and derived variables of a specific phase in the system are defined as listed below:

<i>Quantity</i>	<i>SV & Unit</i> [§]	<i>DV & Unit</i> [¶]
Amount	NP(ph) mol	NPM(ph) dimensionless
		NPW(ph) mol/g
		NPV(ph) mol/m ³
	BP(ph) g	BPM(ph) g/mol
		BPW(ph) dimensionless
		BPV(ph) g/m ³
Volume	VP(ph) m ³	VPM(ph) m ³ /mol
		VPW(ph) m ³ /g
		VPV(ph) dimensionless
	V(ph) m ³	VM(ph) m ³ /mol
		VW(ph) m ³ /g
		VF(ph) m ³ /mole_formula_unit
Gibbs Energy	G(ph) J	GM(ph) J/mol
		GW(ph) J/g
		GV(ph) J/m ³
		GF(ph) J/mole_formula_unit
Helmholtz Energy	A(ph) J	AM(ph) J/mol
		AW(ph) J/g
		AV(ph) J/m ³
		AF(ph) J/mole_formula_unit
Internal Energy	U(ph) J	UM(ph) J/mol
		UW(ph) J/g
		UV(ph) J/m ³
		UF(ph) J/mole_formula_unit
Enthalpy	H(ph) J	HM(ph) J/mol
		HW(ph) J/g
		HV(ph) J/m ³
		HF(ph) J/mole_formula_unit
Entropy	S(ph) J/K	SM(ph) J/mol/K
		SW(ph) J/g/K
		SV(ph) J/m ³ /K
		J/mole_formula_unit/K
Curie Temperature	TC(ph) K	
Bohr Magneton Number	BMAG(ph) dimensionless	
Phase Stability Function	QF(ph) dimensionless	
Driving Force		DGM(ph) J/mol
		DGW(ph) J/g

		DGV (ph) J/m ³
		DGF (ph) J/mole_formula_unit
Heat Capacity		HM (ph) .T J/mol/K
		HW (ph) .T J/g/K
		HV (ph) .T J/m ³ /K
		HF (ph) .T J/mole_formula_unit/K
Thermal Expansivity		VM (ph) .T m ³ /mol/K
		VW (ph) .T m ³ /g/K
		VF (ph) .T m ³ /mole_formula_unit/K
Isothermal Compressibility		VM (ph) .P m ³ /mol/Pa
		VW (ph) .P m ³ /g/Pa
		VF (ph) .P m ³ /mole_formula_unit/Pa

Notes:

§ SV stands for standard state variables of the specific phase, and they are used in the POLY and POST modules;
 ¶ DV stands for derived variables of the specific phase, and they are used in the POLY and POST modules;

Normalizing suffixes like M (per mole of components), W (per mass in grams), V (per volume in m³) or F (per mole formula unit) can be applied to extensive quantities V(ph), G(ph), A(ph), U(ph), H(ph) and S(ph) of the specific phase [except for VW(ph)].

The extensive quantity NP(ph), BP(ph) and VP(ph) can have normalizing suffixes M, W or V, and under such circumstances they present certain special quantities of the specific phase, e.g.,

NPM (ph)	= moles of the phase per mole of the system	(dimensionless)
NPW (ph)	= moles of the phase per mass of the system	(mol/g)
NPV (ph)	= moles of the phase per volume of the system	(mol/m ³)
BPM (ph)	= mass of the phase per mole of the system	(g/mol)
BPW (ph)	= mass of the phase per mass of the system	(dimensionless)
BPV (ph)	= mass of the phase per volume of the system	(g/m ³)
VPM (ph)	= volume of the phase per mole of the system	(m ³ /mol)
VPW (ph)	= volume of the phase per mass of the system	(m ³ /g)
VPV (ph)	= volume of the phase per volume of the system	(dimensionless)

NP(ph), BP(ph) and VP(ph), as well as all their M/W/V/F-suffixed quantities of the phase should not be set as equilibrium conditions. Instead, one can use the CHANGE_STATUS command to set a relevant condition, e.g., CHANGE_STATUS phase <phase name> = fix <amount> where the fixed amount is NPF(ph).

DGM(ph), DGW(ph), DGV(ph) and DGF(ph) can not be set as conditions, as they can only be calculated under a certain type of equilibrium state.

All such normalized quantities are calculated via the first derivatives of the corresponding phase extensive variable with regard to the phase amount [in terms of NP(ph), BP(ph) or VP(ph)], except for the normalized NPM(ph), BPW(ph) and VPV(ph) quantities [as well as for the normalized DGM(ph), DGW(ph), DGV(ph) and DGF(ph) properties] where the suffixes M, W, and V still mean the normalizations are performed with regard to the total system size (in terms of N, B and V, respectively).

The special quantities, QF(ph) [the phase stability function of a specific phase in an equilibrium state, which is negative when the phase composition is inside a spinodal and positive everywhere else], TC(ph) [Curie temperature of a specific phase in an equilibrium state] and BMAG(ph) [Bohr magneton number of a specific phase in an equilibrium state], are dependent upon the thermodynamic model chosen for the phase. They cannot have any normalizing suffix (M/W/V/F); while the reference-state suffix R is legal to be added to such special variables but a R-suffixed quantity is always identical to its original variable [e.g., QFR(ph) = QF(ph)] under a specific condition in a defined system, i.e., the phase stability functions, Curie temperatures and Bohr magneton numbers of various phases do not change along with any possibly re-defined reference states of system components, when the definitions of the system conditions remain unchanged. Additionally, they cannot be used as conditions, as they can only be calculated under a certain type of equilibrium state.

The heat capacity of the phase stands for C_p (when pressure is a condition) or for C_v (when volume is a condition).

Phase-Component Units: For Variables of a Specific Component in a Phase

The default units associated with standard state variables, derived variables, and pre-defined symbols of a specific component in a given phase are defined as listed below:

Quantity	SV & Unit [§]	DV & Unit [¶]
Amount	N(ph, c) mol	X(ph, c) dimensionless NM(ph, c) dimensionless NW(ph, c) mol/g NV(ph, c) mol/m ³ NF(ph, c) mol/mole_formula_unit
	B(ph, c) g	W(ph, c) dimensionless BM(ph, c) g/mol BW(ph, c) dimensionless BV(ph, c) g/m ³ BF(ph, c) g/mole_formula_unit

Notes:

[§] SV stands for standard state variables of the specific phase, and they are used in the POLY and POST modules;

[¶] DV stands for derived variables of the specific phase, and they are used in the POLY and POST modules;

Normalizing suffixes like M (per mole of components), W (per mass in grams), V (per volume in m³) or F (per mole formula unit) can be applied to extensive quantities N(ph, c) and B(ph, c) of the specific system component in the given phase; and under such circumstances they present certain special quantities of the system component in the phase, e.g.,

NM(ph, c) = mole fraction of the component in the phase (dimensionless)
= X(ph, c)

NW(ph, c) = moles of the component per mass of the phase (mol/g)

NV(ph, c) = moles of the component per volume of the phase (mol/m³)

NF(ph, c) = moles of the component per mole formula unit of the phase (mol/mole_formula_unit)

BM(ph, c) = mass of the component per mole of the phase (g/mol)

BW(ph, c) = mass fraction of the component in the phase (dimensionless)
= W(ph, c)

BV(ph, c) = mass of the component per volume of the phase (g/m³)

BF(ph, c) = mass of the component per mole formula unit of the phase (g/mole_formula_unit)

NW(ph, c) and BM(ph, c) should not be set as conditions.

All such normalized quantities are calculated via the first derivatives of the corresponding phase-component extensive variable with regard to the phase amount [in terms of NP(ph), BP(ph) or VP(ph)].

Phase-Species Units: For Variables of a Specific Species in a Phase

The default units associated with standard state variables of a specific species in *or related to* a given phase [*i.e.*, a specific phase constituent (on a certain sublattice) in a given phase] are defined as listed below:

Quantity	SV & Unit [§]	DV & Unit [¶]
Amount	$y(\text{ph}, \text{sp})$ dimensionless	
Chemical Potential	$\text{MU}(\text{sp}, \text{ph})$ J/mol	
	$\text{MUR}(\text{sp}, \text{ph})$ J/mol	
Activity	$\text{AC}(\text{sp}, \text{ph})$ dimensionless	
	$\text{ACR}(\text{sp}, \text{ph})$ dimensionless	
ln(Activity)		$\text{LNAC}(\text{sp}, \text{ph})$ dimensionless
		$\text{LNACR}(\text{sp}, \text{ph})$ dimensionless

Notes:

[§] SV stands for standard state variables of the specific phase, and they are used in the POLY and POST modules;

[¶] DV stands for derived variables of the specific phase, and they are used in the POLY and POST modules;

The suffix R presents that intensive quantities MU(sp,ph), AC(sp,ph) and LNAC(sp,ph) to be calculated with respect with certain reference states chosen for the system components that built up the specific species, such as MUR(sp,ph), ACR(sp,ph) and LNACR(sp,ph); normally, MU(sp,ph)=MUR(sp,ph), AC(sp,ph)=ACR(sp,ph), LNAC(sp,ph)=LNACR(sp,ph).

In case there are two or more sublattices in the given phase, the syntax “sp” should be specified as “constituent#sublattice” indicating that the specific constituent is on the given sublattice or site (denoted by the # sign and a digit) in the phase. For instance,

$y(\text{BCC}, \text{C}\#2)$ stands for the site fraction of the C species on the second sublattice site of the BCC_A2 solution phase;

MUR(AQUEOUS, FE+3) and ACR(AQUEOUS, FE+3) stand for the chemical potential and activity of the Fe⁺³ cation in the (*single-substitutional-lattice* modelled) AQUEOUS solution phase, for which the reference states for components (Fe and electron) have been set to certain phases (with defined reference temperature and pressure).

Currently, MU(sp,ph), AC(sp,ph) and LNAC(sp,ph) can be obtained only for *single-substitutional-lattice* solution phases (such as AQUEOUS solution and GASEous mixture) and for those interacting species on the substitutional sublattice of two-sublattice solution phases (such as BCC_A2 and M6C phases). For instance, one can obtain the values of AC(sp,AQ) for Fe⁺³ and Fe species *related to* the AQUEOUS solution phase. One may also obtain the values of MU(sp,BCC) for Fe and W species *related to* the BCC_A2 phase defined in the Fe-W-C ternary system, but not for C (which is on the interstitial sublattice site); however, one can not get any value for MU(sp,M6C) of the Fe-W-C ternary system, due to that the M6C phase has a phase definition of (Fe) 2 (Fe) 2 (Fe, W) 2 (C) 1.

Special Notes (on units for variables of specific species in a phase):

Unlike in POLY3 (and POST) module, there are other quantities in the GES, TDB, PARROT and ED_EXP modules, that describe various thermodynamic, physical, chemical and transport properties for a compound phase, and for a certain species or a given species combination (binary, ternary or higher-order) in a specific solution phase. Such quantities can not be directly called in the POLY (and POST) module. They are:

G(ph, sp), or G(ph, sp; 0), or G(ph, sp1; sp2; ...; 0)
for Gibbs energy (J/mol) of a pure substance or end-member;
G(ph, sp1, sp2, ...) or L(ph, sp1, sp2, ...), or
G(ph, sp1, sp2, ...; 0) or L(ph, sp1, sp2, ...; 0), or
G(ph, sp1, sp2, ...; 0) or L(ph, sp1, sp2, ...; 0)
for zero-order interaction parameter (J/mol) of a specified binary, ternary or higher-order interactions
on a certain sublattice site in a solution phase.
G(ph, sp1, sp2, ...; i) or L(ph, sp1, sp2, ...; i), or
G(ph, sp1, sp2, ...; i) or L(ph, sp1, sp2, ...; i)
for ith-order interaction parameter (J/mol) of a specified binary, ternary or higher-order interactions on
a certain sublattice site in a solution phase.

TC (ph, sp), or TC (ph, sp; 0), or TC (ph, sp1; sp2; ...; 0)	for Curie temperature (K) of a pure substance or end-member;
TC (ph, sp1, sp2, ...), or TC (ph, sp1, sp2, ...; 0), or TC (ph, sp1, sp2, ...; ...; 0)	for zero-order Curie-temperature contribution (K) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
TC (ph, sp1, sp2, ...; i), or TC (ph, sp1, sp2, ...; ...; i)	for ith-order Curie-temperature contribution (K) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
BMAGN (ph, sp), or BMAGN (ph, sp; 0), or BMAGN (ph, sp1; sp2; ...; 0)	for Bohr magneton number (dimensionless) of a pure substance or end-member;
BMAGN (ph, sp1, sp2, ...), or BMAGN (ph, sp1, sp2, ...; 0), or BMAGN (ph, ...; sp1, sp2, ...; 0)	for zero-order Bohr magneton number (dimensionless) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
BMAGN (ph, sp1, sp2, ...; i), or BMAGN (ph, sp1, sp2, ...; ...; i)	for ith-order Bohr magneton number (dimensionless) to a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
V0 (ph, sp), or V0 (ph, sp; 0), or V0 (ph, sp1; sp2; ...; 0)	for molar volume (m^3) at 1 bar and reference T_0 of a pure substance or end-member;
V0 (ph, sp1, sp2, ...), or V0 (ph, sp1, sp2, ...; 0), or V0 (ph, sp1, sp2, ...; ...; 0)	for zero-order composition-dependent molar volume (m^3) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
V0 (ph, sp1, sp2, ...; i), or V0 (ph, sp1, sp2, ...; ...; i)	for ith-order composition-dependent molar volume (m^3) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VA (ph, sp), or VA (ph, sp; 0), or VA (ph, sp1; sp2; ...; 0)	for integrated thermal expansivity ($\text{m}^3/\text{mol}/\text{K}$) of a pure substance or end-member;
VA (ph, sp1, sp2, ...), or VA (ph, sp1, sp2, ...; 0), or VA (ph, sp1, sp2, ...; ...; 0)	for zero-order composition-dependent thermal expansivity ($\text{m}^3/\text{mol}/\text{K}$) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VA (ph, sp1, sp2, ...; i), or VA (ph, sp1, sp2, ...; ...; i)	for ith-order composition-dependent thermal expansivity ($\text{m}^3/\text{mol}/\text{K}$) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VC (ph, sp), or VB (ph, sp; 0), or VB (ph, sp1; sp2; ...; 0)	for isothermal compressibility ($\text{m}^3/\text{mol}/\text{Pa}$) of a pure substance or end-member;
VC (ph, sp1, sp2, ...), or VB (ph, sp1, sp2, ...; 0), or VB (ph, sp1, sp2, ...; ...; 0)	for zero-order composition-dependent isothermal compressibility ($\text{m}^3/\text{mol}/\text{Pa}$) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VC (ph, sp1, sp2, ...; i), or VB (ph, sp1, sp2, ...; ...; i)	for ith-order composition-dependent isothermal compressibility ($\text{m}^3/\text{mol}/\text{Pa}$) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VK (ph, sp), or VB (ph, sp; 0), or VB (ph, sp1; sp2; ...; 0)	for high-pressure fitting parameter (m^3/mol) of a pure substance or end-member;
VK (ph, sp1, sp2, ...), or VB (ph, sp1, sp2, ...; 0), or VB (ph, sp1, sp2, ...; ...; 0)	for zero-order composition-dependent high-pressure fitting parameter (m^3/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
VK (ph, sp1, sp2, ...; i), or VB (ph, sp1, sp2, ...; ...; i)	for ith-order composition-dependent high-pressure fitting parameter (m^3/mol) of a specified binary, ternary or higher-order interactions on a certain sublattice site in a solution phase.
WR (ph, sp), or WR (ph, sp; 0)	for standard Born function (J/mol) of a specific aqueous species in the AQUEOUS solution phase (always with a single sublattice) under the reference-state temperature and pressure.

User-Specified Units: For Functions that are Defined by a User

Additionally, you can also use your desired units (other than the basic and default units) for some calculated/plotted/tabulated properties, by properly entering user-defined symbols (usually as functions) through the command

```
ENTER_SYMBOL FUNCTION <Name> = < Expression of state variables, derived variables  
and/or already-entered symbols > ;
```

Using your own entered symbols, you may find it very convenient and useful to present various calculated/simulated results in different desired units. Note that a symbol name must be unique in the current workspace, that starts 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. For details on how to define symbols, please refer to [Section 8.10.12](#), or type the HELP ENTER_SYMBOL command or type the ? mark under the ENTER_SYMBOL command.

Examples:

ENT_SYM FUN PB = P/1E5 ;	for pressure (in bar)
ENT_SYM FUN LnFBO2 = LNAC(O2,GAS)+LOG(P)+11.52608845 ;	for ln[Fugacity(O2,GAS)] (in bar)
ENT_SYM FUN Cp = Hm.T ;	for heat capacity of system (in J/mol/K)
ENT_SYM FUN CpFCC = Hm(FCC).T ;	for heat capacity of FCC phase (in J/mol/K)

However, when a converted property is a function of one or more derivatives, one must enter a symbol as a variable instead of a function. For instance, a symbol presenting heat capacity of a system or of a phase in a unit of cal/mol/K must be entered through e.g.,

ENT_SYM VAR Cp2 = Hm.T/4.1858 ;	for heat capacity of system (in cal/mol/K)
ENT_SYM VAR Cp2FCC = Hm(FCC).T/4.1858 ;	for heat capacity of FCC phase (in cal/mol/K)

Please notice that any of user-defined symbols cannot be used for setting equilibrium conditions or stepping/mapping variables, because of that only standard state variables (and some derived variables) can be used for such purposes. However, you can utilize any of the pre-defined variables and user-defined symbols for plotting various phase diagrams and property diagrams as you want. Moreover, some advanced modules (such as POURBAIX and SCHEIL modules) also generated some already-defined symbols (constants, functions, variables and tables) for various interesting properties, and such symbols can be used only for graphical/tabular outputs as well.

Given on the following pages are just a few of suggested examples for entering such conventional functions (in 8 different categories listed below) and their associated units in various possible ways:

- ☞ User-specified units for intensive variables of a defined system
- ☞ User-specified units for extensive variables of a defined system
- ☞ User-specified units for intensive variables of a system component
- ☞ User-specified units for extensive variables of a system component
- ☞ User-specified units for extensive variables of a phase
- ☞ User-specified units for extensive variables of a component in a phase
- ☞ User-specified units for intensive variables of a species in a phase
- ☞ User-specified units for extensive variables of a species in a phase

User-specified units for intensive variables of a defined system

Quantity	Suggested Name & Expression	Unit
Temperature	TempC = T-273.15	Celsius (C)
	TempF = 1.8*T-459.67	Fahrenheit (F)
	TempF = 1.8*TempC+32	
Pressure	PB = P/1E5	bars (bar)
	PKb = P/1E8	kilobars (kbar)
	PAtm = P/101325	atmosphere (atm)
	PSI = P/6894.76	pounds/sq. inch (psi)
	PIHg = P/3342.11	inches of Hg
	PTor = P/133.322	Tors (millimeters of Hg)

User-specified units for extensive variables of a defined system

Quantity	Suggested Name & Expression	Unit
Amount	BKg = 1E-3*B	kilograms (kg)
	BTon = 1E-6*B	tons
Volume	VDM = 1E-3*V	cubic decimeters (dm ³ , l)
	VCM = 1E-6*V	cubic centimeters (cm ³)
	VMM = 1E-9*V	cubic millimeters (mm ³)
	VCI = 1.6387064E-5*V	cubic inches (in ³)
	VLi = 1E-3*V	liters (l)
Energy	GCal = G/4.1858	cal
	ACal = A/4.1858	cal
	UCal = U/4.1858	cal
	HCal = H/4.1858	cal
	SCal = S/4.1858	Cal/K
Heat Capacity	Cp2 = HM.T/4.1858	cal/mol/K
	Cp3 = HW.T/4.1858	cal/g/K
	Cp4 = HV.T/4.1858	cal/m ³ /K
Thermal Expansion	A2 = VM.T*1E-6	cm ³ /mol/K
	A3 = VW.T*1E-6	cm ³ /g/K
Isothermal Compressibility	B2 = VM.P*1E-9	mm ³ /mol/Pa
	B3 = VW.P*1E-9	mm ³ /g/Pa

Notes: One can also define similar symbols for certain quantities of a system using normalizing suffixes [M (per mole of components), W (per mass in grams), V (per volume in m³) or F (per mole formula unit)], e.g., in the above-mentioned symbols,

- replacing B by BM or BV
- replacing V by VM or VW
- replacing G by GM, GW or GV
- replacing A by AM, AW or AV
- replacing U by UM, UW or UV
- replacing H by HM, HW or HV
- replacing S by SM, SW or SV

Remember that all such normalized quantities are calculated via the first derivatives of the corresponding system extensive variable with regard to the total system size [in terms of N, B or V] of the whole system.

User-specified units for intensive variables of a system component

Quantity	Suggested Name & Expression	Unit
Chemical Potential	MUi = MU(c) / 4.1858	cal/mol
	MURI = MUR(c) / 4.1858	cal/mol
Special quantity for aqueous solution	pH = -LOG10(ACR(H+1))	dimensionless
	Eh = MUR(ZE) / RNF where RNF=96485.309	volts (V)

Notes: “i” stands for an auxiliary index or letter that corresponds to the component c.

User-specified units for extensive variables of a system component

Quantity	Suggested Name & Expression	Unit
Amount	MF _i = N(c) / N = X(c)	dimensionless
	MP _i = N(c) / N * 100	dimensionless
	WF _i = B(c) / B = W(c)	dimensionless
	WP _i = B(c) / B * 100	dimensionless
	BKgi = 1E-3 * B(c)	kilograms (kg)
	BToni = 1E-6 * B(c)	tons

Notes: “i” stands for an auxiliary index or letter that corresponds to the component c.

One can also define similar symbols for certain quantities of a system component using normalizing suffixes [M (per mole of components), W (per mass in grams) or V (per volume in m³), e.g., in the above-mentioned symbols,

replacing N(c) by NM(c), NW(c) or NV(c)
replacing B(c) by BM(c), BW(c) or BV(c)

Remember that all such normalized quantities are calculated via the first derivatives of the corresponding system component extensive variable with regard to the total system size [in terms of N, B or V] of the whole system.

User-specified units for extensive variables of a phase

Quantity	Suggested Name & Expression	Unit
Amount	MF _j = NP(ph) / N	dimensionless
	MP _j = NP(ph) / N * 100	dimensionless
	WF _j = BP(ph) / B	dimensionless
	WP _j = BP(ph) / B * 100	dimensionless
	BKgj = 1E-3 * BP(ph)	kilograms (kg)
	BTonj = 1E-6 * BP(ph)	tons
Volume	VCM _j = 1E-3 * V(ph)	cubic decimeters (dm ³ , l)
	VCM _j = 1E-6 * V(ph)	cubic centimeters (cm ³)
	VMM _j = 1E-9 * V(ph)	cubic millimeters (mm ³)
Energy	GCal _j = G(ph) / 4.1858	cal
	ACal _j = A(ph) / 4.1858	cal
	UCal _j = U(ph) / 4.1858	cal
	HCal _j = H(ph) / 4.1858	cal
	SCal _j = S(ph) / 4.1858	Cal/K
Driving Force	DG2j = DGM(ph) / 4.1858	cal/mol

	DG3j = DGW(ph) / 4.1858	cal/g
	DG4j = DGV(ph) / 4.1858	Cal/m³
	DG5j = DGF(ph) / 4.1858	cal/mole formula unit
Heat Capacity	Cp2j = HM(ph).T/4.1858	cal/mol/K
	Cp3j = HW(ph).T/4.1858	cal/g/K
	Cp4j = HV(ph).T/4.1858	cal/m³/K
	Cp5j = HF(ph).T/4.1858	cal/mole formula unit/K
Thermal Expansion	A2j = VM(ph).T*1E-6	cm³/mol/K
	A3j = VW(ph).T*1E-6	cm³/g/K
	A4j = VF(ph).T*1E-6	cm³/mole formula unit/K
Isothermal Compressibility	B2j = VM(ph).P*1E-9	mm³/mol/Pa
	B3j = VW(ph).P*1E-9	mm³/g/Pa
	B4j = VF(ph).P*1E-9	mm³/mole formula unit/Pa

Notes: "j" stands for an auxiliary index or letter that corresponds to the phase ph.

One can also define similar symbols for certain quantities of a phase using normalizing suffixes [M (per mole of components), W (per mass in grams), V (per volume in m³) or F (per mole formula unit)], e.g., in the above-mentioned symbols,

replacing NP(ph) by NPM(ph), NPW(ph) or NPV(ph)
 replacing BP(ph) by BPM(ph), BPW(ph) or BPV(ph)
 replacing VP(ph) by VPM(ph), VPW(ph) or VPV(ph)
 replacing V(ph) by VM(ph), VW(ph) or VF(ph)
 replacing G(ph) by GM(ph), GW(ph), GV(ph) or GV(ph)
 replacing A(ph) by AM(ph), AW(ph), AV(ph) or AV(ph)
 replacing U(ph) by UM(ph), UW(ph), UV(ph) or UV(ph)
 replacing H(ph) by HM(ph), HW(ph), HV(ph) or HV(ph)
 replacing S(ph) by SM(ph), SW(ph), SV(ph) or SV(ph)

Remember that all such normalized quantities are calculated via the first derivatives of the corresponding phase extensive variable with regard to the phase amount [in terms of NP(ph), BP(ph) or VP(ph)] of the phase, except for the normalized NPM(ph), BPW(ph) and VPV(ph) quantities [as well as for the normalized DGM(ph), DGW(ph), DGV(ph) and DGF(ph) properties] where the suffixes M, W, and V still mean the normalizations are performed with regard to the total system size (in terms of N, B and V, respectively).

User-specified units for extensive variables of a component in a phase

Quantity	Suggested Name & Expression	Unit
Amount	MFij = N(ph,c) /NP(ph)	dimensionless
	= X(ph,c)	
	MPij = N(ph,c) /NP(ph) *100	dimensionless
	WFi = B(ph,c) /BP(ph)	dimensionless
	= W(ph,c)	
	WPij = B(ph,c) /BP(ph) *100	dimensionless
	BKgij = 1E-3*B(ph,c)	kilograms (kg)
	BTonij = 1E-6*B(ph,c)	tons

Notes: "ij" stands for an auxiliary index or letter that corresponds to the component c in the phase ph.

One can also define similar symbols for certain quantities of a system component using normalizing suffixes [M (per mole of components), W (per mass in grams), V (per volume in m³) or F (per mole formula unit)], e.g., in the above-mentioned symbols,

replacing N(ph,c) by NM(ph,c), NW(ph,c), NV(ph,c) or NF(ph,c)
 and NP(ph) by NPM(ph), NPW(ph), NPV(ph) or NP(ph)/NA
 replacing B(ph,c) by BM(ph,c), BW(ph,c), BV(ph,c) or BF(ph,c)
 and BP(ph) by BPM(ph), BPW(ph), BPV(ph) or BP(ph)/NA
 (where NA stands for total atomic number in the phase formula)

Remember that all the normalized Z(ph,c) quantities are calculated via the first derivatives of the corresponding system component extensive variable Z with regard to the total system size [in terms of

N , B or V] of the whole system; whilst all the the $Z(\text{ph})$ quantities are calculated via the first derivatives of the corresponding phase extensive variable Z with regard to the phase amount [in terms of $\text{NP}(\text{ph})$, $\text{BP}(\text{ph})$ or $\text{VP}(\text{ph})$] of the phase, except for the normalized $\text{NPM}(\text{ph})$, $\text{BPW}(\text{ph})$ and $\text{VPV}(\text{ph})$ quantities [as well as for the normalized $\text{DGM}(\text{ph})$, $\text{DGW}(\text{ph})$, $\text{DGV}(\text{ph})$ and $\text{DGF}(\text{ph})$ properties] where the suffixes M, W, and V still mean the normalizations are performed with regard to the total system size (in terms of N , B and V , respectively).

User-specified units for intensive variables of a species in a phase

Quantity	Suggested Name & Expression	Unit
Chemical Potential	$\text{MUCalij} = \text{MU}(\text{sp}, \text{ph}) / 4.1858$	cal/mol
	$\text{MUErgij} = \text{MU}(\text{sp}, \text{ph}) * 1E7$	cal/mol
Fugacity	$\text{FAij} = \text{AC}(\text{sp}, \text{ph}) * \text{PATM}$	atmosphere (atm)
	$\text{FBij} = \text{AC}(\text{sp}, \text{ph}) * \text{PB}$	bars (bar)
	$\text{FCij} = \text{AC}(\text{sp}, \text{ph}) * \text{PKB}$	kilobars (kbar)
	$\text{FTij} = \text{AC}(\text{sp}, \text{ph}) * \text{PTOR}$	Tors (millimeters of Hg)
ln(Fugacity)	$\text{LnFAij} = \text{LNAC}(\text{sp}, \text{ph}) + \text{LN}(\text{PATM})$	atmosphere (atm)
	$\text{LnFBij} = \text{LNAC}(\text{sp}, \text{ph}) + \text{LN}(\text{PB})$	bars (bar)
	$\text{LnFCij} = \text{LNAC}(\text{sp}, \text{ph}) + \text{LN}(\text{PKB})$	kilobars (kbar)
	$\text{LnFTij} = \text{LNAC}(\text{sp}, \text{ph}) + \text{LN}(\text{PTOR})$	Tors (millimeters of Hg)
Special quantity for aqueous solution	$\text{AIij} = \text{ACR}(\text{FE}+3, \text{AQ}) * \text{AH2O}$	dimensionless
	$\text{RCij} = \text{ACR}(\text{FE}+3, \text{AQ}) * \text{YH2O} / \text{Y(AQ, FE+3)}$ where $\text{AH2O}=55.508435$ $\text{YH2O}=\text{Y(AQ, H2O)}$	dimensionless
	$\text{WRCalij} = \text{WR}(\text{AQ}, \text{sp}) / 4.1858$	cal/mol

Notes: “ij” stands for an auxiliary index or letter that corresponds to the component c in the phase ph .
 Born Function $\text{WR}(\text{AQ}, \text{sp})$ is only valid for aqueous species in system where the AQUEOUS solution phase is considered.

User-specified units for extensive variables of a species in a phase

Quantity	Suggested Name & Expression	Unit
Special quantity for aqueous solution	$\text{MLij} = \text{Y(AQ, FE+3)} * \text{AH2O} / \text{YH2O}$	molality (m)
	$\text{ISTR} = 1 * \text{IS1} + 1 * \text{IS2} + 1 * \text{IS3}$ where $\text{AH2O}=55.508435$ $\text{YH2O}=\text{Y(AQ, H2O)}$ $\text{ISn} = .5 * \text{MLi} * \text{Zi}^{**2} + .5 * \text{MLj} * \text{Zj}^{**2} + .5 * \text{MLk} * \text{Zk}^{**2}$	molality (m)

Notes: “ij” stands for an auxiliary index or letter that corresponds to the component c in the phase ph .

Appendix D. On Reference States and Standard States

(Within the Framework of Thermo-Calc Software/Databases)

Brief Introduction

As in general, the concepts of “reference state” and “standard state” for various components, species and phases in various systems described by certain models are really difficult to fully understand and rather easy to get confused, and thus there are possible difficulties and confusions in setting/changing reference states and in interpreting calculation results. As the first suggestion, we would usually encourage our users to go through the Section 2.13 of the document ***Thermo-Calc Software System*** and Section 10.3 of the ***TCCS User’s Guide***, and read some sophistic textbooks of Thermodynamics, prior to using the `SET_REFERENCE_STATE` command.

How Does Thermo-Calc Handle Reference States

From a practical point of view, Gibbs free energies are always given related to some references in one or other way. By default, a Thermo-Calc database always uses elements (at their standard states, *i.e.*, in stable structured phases normally under 298.15 K and 1 bar) as the default components, and thus sets the reference states for all components as the so-called **SER** (*Stable Element Reference*), which builds up a certain “Gibbs free energy reference plane” in a defined system. When the temperature-pressure conditions have been defined (in GES/POLY module), a corresponding “Gibbs free energy surface” over the entire composition space will be formed. It is very important to note that:

- (1) The Gibbs free energy expressions (for both standard and excess properties) are results of assessments that are based on available experimental info (which are usually measured at temperatures above 273.15 K and pressures equal/above 1 bar) and sometimes theoretical data;
- (2) As certain types of polynomials, such expressions normally should not be extrapolated to temperatures below 298.15 K (especially too far below 298.15 K, it is not appropriate to do so; this is because at such low temperature range it may results in very strange values of C_p , H and S, and thus of G);
- (3) Of course, some assessment work are done by considering existing experimental and theoretical data below 298.15 K, and thus one may have accordingly set the lowest applicable temperature of assessed G/L expressions; in such cases, it would be safer and appropriate to extrapolate to a reasonable lower temperature;
- (4) For some special pure substances and/or solution phases (normally their major phase constituents), the Thermo-Calc software can employ advanced thermodynamic models in the GES system that allows appropriate calculations in a certain temperature range below 298.15 K; for instance, the complete Revised HKF Model (Helgeson-Kirkham-Flowers) can calculate thermodynamic and transport properties of H_2O (in ice, aqueous and gas/fluid states) over a wide temperature-pressure range.
- (5) Nowadays, first-principle calculations may play a unique and efficient role in calculating various thermodynamic and chemical/physical properties down to 0 K; by incorporating such results together with CALPHAD assessments into a database, it will be possible to make calculations from 0 K to high temperatures.

Another very important matter: as a default setting in a Thermo-Calc database, the `DEFINE_SYSTEM_DEFAULT` command has always an associated “G-reference type index” when `<element>` is used as the keyword, that globally defines the reference state type of all elements in the database. Three types (as in integer index) of element reference states can be chosen from:

1 ==> G	G refers to $G(298.15,1) = H(298.15,1) - T \cdot S(298.15,1)$
2 ==> H298	G refers to $H(298.15,1)$ [assuming $S(0,1)=0$]
3 ==> H0	G refers to $H(0,1)$ [assuming $S(0,1)=0$]

A substance or alloy solution database (in the Thermo-Calc database spectrum, or in the SGTE recommended database format) normally uses the second type, *i.e.*, H298 [referring to $H(298.15\text{K},1\text{bar})$ and assuming $S(0\text{K},1\text{bar})=0$]. This means that the standard Gibbs free energy expression

(stored in the database) for a substance or for a species in a phase corresponding to the SER state will always be interpreted as referring to H(298.15,1) which is the enthalpy of formation from all relevant elements in their SER states at 298.15 K and 1 bar, *i.e.*,

$$\begin{aligned} G(FCC_A1,FE:C;0)-H298(GRAPHITE,C;0)-H298(BCC_A2,FE;0) \\ = +GFEFCC+GHSERCC+GPCFCC+77207-15.877*T \\ G(M7C3,FE:C;0)-3 H298(GRAPHITE,C;0)-7 H298(BCC_A2,FE;0) \\ = +7*GHSERFE+3*GHSERCC+75000-48.2168*T \end{aligned}$$

This is why that, for convenience in publications, such a reference state is usually abbreviated as HSER, and accordingly the Gibbs free energy expression may be simply written as

$$\begin{aligned} G0(FCC_A1,FE:C)-HSER &= +GFEFCC+GHSERCC+GPCFCC+77207-15.877*T \\ G0(M7C3,FE:C)-HSER &= +7*GHSERFE+3*GHSERCC+75000-48.2168*T \end{aligned}$$

Therefore, with the HSER state as the default-set in a database that is implied by the TDB command

DEFINE_SYSTEM_DEFAULT ELEMENT 2 !

the stored G/L expressions (as $f(T,P,X)$, for standard and excess properties) will build a certain Gibbs free energy reference plane for a defined system, that is always refer to the HSER state.

However, Thermo-Calc software (in many of its basic modules such as GES, POLY, PARROT/ED_EXP and POST) also allows the user to re-define system components other than elements that are default-defined in TDB (while inside some of its special/advanced modules such as POURBAIX, it may also enforce some necessary changes in definitions of system components in accordance with the chosen model and for the purposes of calculating some specific properties *e.g.* pH and Eh), and to choose other appropriate reference states for the defined system components, in terms of their reference phases, reference temperature (which should always be higher than 298.14 K) and reference pressure (which should be equal to or higher than 1 bar). By doing so, Thermo-Calc will build up a new “Gibbs free energy reference plane”, and then will properly calculate various thermodynamic properties, such as:

- ☞ G-H-S and Gm-Hm-Sm of the defined system;
- ☞ G-H-S (phase) and Gm-Hm-Sm (phase) of various phases;
- ☞ MU (comp) and AC (comp): potentials and activities of system components with respect to the default reference states of components (SER)
 $MU(\text{comp}) = RT\ln AC(\text{comp})$
- ☞ MUR (comp) and ACR (comp): potentials and activities of system components with respect to the re-defined reference states of components
 $MUR(\text{comp}) = RT\ln ACR(\text{comp})$
- ☞ DGM (comp): driving forces (/RT) of system components
- ☞ *and so on.*

Please also note that:

- (1) From a chosen database (or a certain database combination), the entire Gibbs free energy surface (G or Gm) calculated for a defined system (under a certain T-P-X space) is always the same, regardless how the reference states of system components are defined; this is because that it is always determined by the Gibbs free energy expressions (both standard properties and excess properties) stored in the retrieved database. This also implies that the contributions of each and all system components (regardless how their reference states are defined) to various phases and the entire system (under a certain T-P-X space) will always be the same, and further implies that changing definitions of system components and their reference states will not introduce variation of mass balance (and global Gibbs free energy minimum) of a defined system (under a certain T-P-X space).
- (2) When changing a reference state for a specific component, its chemical potential and activity with respect to such a reference state in a defined system (under a certain T-P-X space) will be altered. Therefore, an appropriate definition of reference state for a component is always required by a chosen thermodynamic model, and thus should usually be related to measurements of potential or activity for that component (which again refer to a particular reference phase in experiments).

Some Important Issues about Chemical Potential and Activity

- (1) These two concepts are always related to certain definitions of reference states and standard states, and they will be meaningless when references states and standard states have not been defined appropriately. They can be applied to not only system components [*i.e.*, MUR(comp) or MU(comp), ACR(comp) or AC(comp)] but also phase constituents (species) [*i.e.*, MUR(sp,ph) or MU(sp,ph), ACR(sp,ph) or AC(sp,ph)].

(2) In case of species, Thermo-Calc can calculate chemical potentials and activities of species *related to* solution phases where there is only a *single substitutional lattice* (and of course there is a corresponding solution model to handle such property calculations), for instance:

GAS mixture:	MUR (<i>species, GAS</i>) and ACR (<i>species, GAS</i>) treated by ideal EOS and ideal mixing model or an implemented non-ideal model.
AQUEOUS solution:	MUR (<i>species, AQUEOUS</i>) and ACR (<i>species, AQUEOUS</i>) treated by DHLL, SIT, PITZ or HKF model
LIQUID solution: ACR (<i>species, LIQUID</i>) (metallic)	MUR (<i>species, LIQUID</i>) and treated by RKM or another model

However, it is impossible to calculate such properties if a solution phase has two or more sublattice sites:

Non-substitutional phase:

e.g., FCC: (Fe,Cr)1(C,N,Va)1

Ionic Two-sublattice phase:

e.g., ION_LIQ: (Cr+3,Fe+2)1(O-2,FeO3/2,VA)1

SPINEL: (Fe+3, Fe+2)1 (Fe+3, Fe+2, Va)2 (Fe+2, Va)2 (O-2)4

- (3) In case of system components, MU(comp)_HSER and AC(comp)_HSER [i.e., MU(comp) and AC(comp) with respect to the HSER reference state] always remain the same (when T-P-X conditions are fixed), regardless how the reference states of system components are defined; this is because the MU(comp)_HSER and AC(comp)_HSER quantities are calculated using the store Gibbs free energy expressions in the database which are already based on the HSER reference state (see above). Therefore, the AC(comp)_HSER value does not have any physical meaning.

Furthermore, when the default SER (and implicitly HSER) reference state is used and PT conditions are defined, the retrieved (from the TDB module) and calculated (in the GES/POLY modules) values for G_0 (phase, species) and G_0 (phase) are:

$$\begin{aligned}\delta G_0(\text{ph}, \text{sp})_{\text{PT}} &= G_0(\text{ph}, \text{sp})_{\text{PT}} - H298(\text{SER}) \\ &= \langle \text{as in TDB/GES} \rangle \\ \delta G_0(\text{ph})_{\text{PT}} &= G_0(\text{ph})_{\text{PT}} - H298(\text{SER}) \\ &= \langle \text{as in TDB/GES} \rangle\end{aligned}$$

and it is always true that

MUR(comp) _HSER = MU(comp@HSER) = MU0(comp) @HSER
 ACR(comp) HSER = AC(comp) HSER

After having appropriately set a reference state (other than the SER/HSER state), a new “Gibbs free energy reference plane” is built up for the defined system, and the MUR(*comp*)_RefSt property is a quantity measuring the difference between the “Gibbs free energy surface” (corresponding to a defined temperature-pressure-composition condition) and “Gibbs free energy reference plane”, and thus its ACR(*comp*)_RefSt value becomes physically meaningful. For instance, by setting the reference state of a component *related to* a given phase (normally the stable phase SP of the component at 298.15 and 1 bar, but not always necessary SP) at the current temperature * (or 298.15 K or 970 °C) and at 1 bar:

RefSt = SP * 101325 (donated as _1T)
(i.e., by "s-r-s comp SP * 101325")

$$\delta G_0(\text{ph}, \text{sp})_{-\text{PT}} = [G_0(\text{ph}, \text{sp})_{-\text{PT}} - H_298(\text{SER})]$$

```

          - [G0(ph,sp)-1T - H298(SER)]
          = G0(ph,sp)-PT - G0(ph,sp)-1T
δG0(ph)-PT = [G0(ph)-PT - H298(SER)] - [G0(ph)-1T - H298(SER)]
          = G0(ph)-PT - G0(ph)-1T
MUR(comp)-1T = delta{δG(sys)-PT} / delta{δN(comp)}
          = MU(comp)-PT - MU0(comp)-1T
ACR(comp)-1T = exp[MUR(comp)-1T / RT]
-----
RefSt = SP 298.15 101325 (donated as RPT, i.e., as room-PT)
          (i.e., by "s-r-s comp SP 298.15 101325")
δG0(ph,sp)-PT = [G0(ph,sp)-PT - H298(SER)]
          - [G0(ph,sp)-RPT - H298(SER)]
          = G0(ph,sp)-PT - G0(ph,sp)-RPT
δG0(ph)-PT = [G0(ph)-PT - H298(SER)] - [G0(ph)-RPT - H298(SER)]
          = G0(ph)-PT - G(ph)-RPT
MUR(comp)-RPT = delta{δG(sys)-PT} / delta{δN(comp)}
          = MU(comp)-PT - MU0(comp)-RPT
ACR(comp)-RPT = exp[MUR(comp)-RPT / RT]
-----
```

- (4) During a critically-assessment, one should always extremely careful in using activity information: a measured activity is always based on an experimentally-chosen reference state for that particular component/species.

Some Specific Examples

Now, let's come to some examples, to see how to properly set the reference states for system components and to correctly obtain and interpret the activity properties. Below is a detailed list of such example calculation results using the SSOL2 database (showing both the correct and wrong settings):

```
=====
A) Pure C System:
Condition P=101325 Pa, T=1000 K, N=1
-----
ref_ph ref_T/P      AC(C)   ACR(C)   MU(C)    MUR(C)   Gm(gra)  NP(gra)
                           Gm
-----
SER                    .21818  .21818 -12658.3 -12658.3 -12658.3   1
                                         -12658.3
-----
Why? Graphite (stable element) is stable at 1000 K
MU0(C)@HSER = -12658.3
MU0(C@1000K) = -12658.3
MU(C@1000K) = MU0(C@1000K) = MU0(C)@HSER
               = -12658.3
--> MU(C) = MU(C@1000K) = -12658.3
     AC(C) = exp[MU(C)/RT] = 0.21818
     MUR(C) = MU(C@1000K) = -12658.3
     ACR(C) = exp[MUR(C)/RT] = 0.21818
-----
gra * 101325 .21818 1      -12658.3      0      -12658.3   1
                                         -12658.3
-----
Why? Graphite (stable element) is stable at 1000 K
MU0(C)@1000K = -12658.3
MU0(C@1000K) = -12658.3
MU(C@1000K) = MU0(C@1000K)
               = -12658.3
--> MU(C) = MU(C@1000K) = -12658.3
     AC(C) = exp[MU(C)/RT] = 0.21818
     MUR(C) = MU(C@1000K) - MU0(C)@1000K
               = (-12658.3) - (-12658.3) = 0
     ACR(C) = exp[MUR(C)/RT] = 1
```

```
-----
gra    298.15 101325 .21818 .26805 -12658.3 -10946.8 -12658.3   1
                                         -12658.3

-----
```

Why? Graphite (stable element) is stable
 $MU_0(C)@298.15 = -1711.5$
 $MU_0(C@1000K) = -12658.3$
 $MU(C@1000K) = MU_0(C@1000K)$
 $= -12658.3$

--> $MU(C) = MU(C@1000K) = -12658.3$
 $AC(C) = \exp[MU(C)/RT] = 0.21818$
 $MUR(C) = MU(C@1000K)-MU_0(C)@298.15$
 $= (-12658.3) - (-1711.5) = -10946.8$
 $ACR(C) = \exp[MUR(C)/RT] = 0.26805$

```
---- But the following extrapolations are not appropriate at all !!!
gra    200    101325 .21818 .25465 -12658.3 -11373.2 -12658.3 -12658.3
gra    100    101325 .21818 .22604 -12658.3 -12364.1 -12658.3 -12658.3
gra    95     101325 .21818 .21765 -12658.3 -12678.5 -12658.3 -12658.3
gra    10     101325 .21818 0      -12658.3 -9609698 -12658.3 -12658.3
=====
```

B) Pure Fe System:Condition $P=101325 \text{ Pa}$, $T=1000 \text{ K}$, $N=1$

ref_ph	ref_T/P	AC(Fe)	ACR(Fe)	MU(Fe)	MUR(Fe)	Gm(bcc)	NP(bcc)
						Gm	
SER		.00619	.00619	-42271.7	-42271.7	-42271.7	1
							-42271.7

Why? BCC_Fe (stable element) is stable at 1000 K
 $MU_0(Fe)@HSER = -42271.7$
 $MU_0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU_0(Fe@1000K) = MU_0(Fe)@HSER$
 $= -42271.7$

--> $MU(Fe) = MU(Fe@1000K) = -42271.7$
 $AC(Fe) = \exp[MU(Fe)/RT] = 0.00619$
 $MUR(Fe) = MU(Fe@1000K) = -42271.7$
 $ACR(Fe) = \exp[MUR(Fe)/RT] = 0.00619$

```
bcc    *    101325 .00619 1      -42271.7      0      -42271.7   1
                                         -42271.7
```

Why? BCC_Fe (stable element) is stable at 1000 K
 $MU_0(Fe)@1000K = -42271.7$
 $MU_0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU_0(Fe@1000K)$
 $= -42271.7$

--> $MU(Fe) = MU(Fe@1000K) = -42271.7$
 $AC(Fe) = \exp[MU(Fe)/RT] = 0.00619$
 $MUR(Fe) = MU(Fe@1000K)-MU_0(Fe)@1000K$
 $= (-42271.7) - (-42271.7) = 0$
 $ACR(Fe) = \exp[MUR(Fe)/RT] = 1$

```
bcc    298.15 101325 .00619 .01674 -42271.7 -34139.0 -42271.7   1
                                         -42271.7
```

Why? BCC_Fe (stable element) is stable at 1000 K
 $MU_0(Fe)@298.15 = -8132.1$
 $MU_0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU_0(Fe@1000K)$
 $= -42271.7$

--> $MU(Fe) = MU(Fe@1000K) = -42271.7$
 $AC(Fe) = \exp[MU(Fe)/RT] = 0.00619$
 $MUR(Fe) = MU(Fe@1000K)-MU_0(Fe)@298.15$
 $= (-42271.7) - (-8132.1) = -34139.0$
 $ACR(Fe) = \exp[MUR(Fe)/RT] = 0.01674$

---- But the following extrapolations are not appropriate at all !!!

bcc	200	101325	.00619	.01258	-42271.7	-36380.0	-42271.7	-42271.7
bcc	100	101325	.00619	.01086	-42271.7	-37604.0	-42271.7	-42271.7
bcc	50	101325	.00619	.01042	-42271.7	-37947.2	-42271.7	-42271.7
bcc	25	101325	.00619	.00933	-42271.7	-38863.4	-42271.7	-42271.7
bcc	11	101325	.00619	.00625	-42271.7	-42201.9	-42271.7	-42271.7

C) Binary Fe-C System:

Condition P=101325 Pa, T=1000 K, N=1, X(C)=0.5

ref_ph	ref_T/P	AC(C)	ACR(C)	MU(C)	MUR(C)	Gm(gra)	NP(gra)
		AC(Fe)	ACR(Fe)	MU(Fe)	MUR(Fe)	Gm(bcc)	NP(bcc)

Gm

C-SER		.21818	.21818	-12658.3	-12658.3	-12658.3	.49963875
Fe-SER		.00619	.00619	-42277.8	-42277.8	-42256.4	.50036125
							-27468.1

Why? Graphite & BCC_Fe are stable at 1000 K
 $MU_0(C) @ HSER = -12658.3$
 $MU_0(C@1000K) = -12658.3$
 $MU(C@1000K) = MU_0(C@1000K) + 'ex-contribution(C)'$
 $= -12658.3$
 $MU_0(Fe) @ HSER = -42271.7$
 $MU_0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU_0(Fe@1000K) + 'ex-contribution(Fe)'$
 $= -42277.8$

--> $MU(C) = MU(C@1000K) = -12658.3$
 $AC(C) = \exp[MU(C)/RT] = 0.21818$
 $MUR(C) = MU(C@1000K) = -12658.3$
 $ACR(C) = \exp[MUR(C)/RT] = 0.21818$

--> $MU(Fe) = MU(Fe@1000K) = -42277.8$
 $AC(Fe) = \exp[MU(Fe)/RT] = 0.00619$
 $MUR(Fe) = MU(Fe@1000K) = -42277.8$
 $ACR(Fe) = \exp[MUR(Fe)/RT] = 0.00619$

C-gra *	101325	.21818	1	-12658.3	0	-12658.3	.49963875
Fe-bcc *	101325	.00619	.99928	-42277.8	-6.1	-42256.4	.50036125
							-27468.1

Why? Graphite & BCC_Fe are stable at 1000 K
 $MU_0(C) @ 1000K = -12658.3$
 $MU_0(C@1000K) = -12658.3$
 $MU(C@1000K) = MU_0(C@1000K) + 'ex-contribution(C)'$
 $= -12658.3$
 $MU_0(Fe) @ 1000K = -42271.7$
 $MU_0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU_0(Fe@1000K) + 'ex-contribution(Fe)'$
 $= -42277.8$

--> $MU(C) = MU(C@1000K) = -12658.3$
 $AC(C) = \exp[MU(C)/RT] = 0.21818$
 $MUR(C) = MU(C@1000K) - MU_0(C) @ 1000K$
 $= (-12658.3) - (-12658.3) = 0$
 $ACR(C) = \exp[MUR(C)/RT] = 1$

--> $MU(Fe) = MU(Fe@1000K) = -42277.8$
 $AC(Fe) = \exp[MU(Fe)/RT] = 0.00619$
 $MUR(Fe) = MU(Fe@1000K) - MU_0(Fe) @ 1000K$
 $= (-42277.8) - (-42271.7) = -6.1$
 $ACR(Fe) = \exp[MUR(Fe)/RT] = 0.99928$

C-gra	298.15	101325	.21818	.26805	-12658.3	-10946.8	-12658.3	.49963875
Fe-bcc	298.15	101325	.00619	.01646	-42277.8	-34145.0	-42256.4	.50036125
								-27468.1

Why? Graphite & BCC_Fe are stable at 1000 K
 $MU_0(C) @ 298.15 = -1711.5$
 $MU_0(C@1000K) = -12658.3$
 $MU(C@1000K) = MU_0(C@1000K) + 'ex-contribution(C)'$

```

= -12658.3
MU0(Fe) 298.15 = -8132.1
MU0(Fe@1000K) = -42271.7
MU(Fe@1000K) = MU0(Fe@1000K) + 'ex-contribution(Fe)'
= -42277.8
--> MU(C) = MU(C@1000K) = -12658.3
AC(C) = exp[MU(C)/RT] = 0.21818
MUR(C) = MU(C@1000K) - MU0(C)@298.15
= (-12658.3) - (-1711.5) = -10946.8
ACR(C) = exp[MUR(C)/RT] = 0.26805
--> MU(Fe) = MU(Fe@1000K) = -42277.8
AC(Fe) = exp[MU(Fe)/RT] = 0.00619
MUR(Fe) = MU(Fe@1000K) - MU0(Fe)@298.15
= (-42277.8) - (-8132.1) = -34139.0
ACR(Fe) = exp[MUR(Fe)/RT] = 0.01674
-----
---- But the following extrapolations are not appropriate at all !!!
gra 200 101325 .21818 .25465 -12658.3 -11373.2 -12658.3
bcc 200 101325 .00619 .01258 -42277.8 -36380.0 -42256.4
                                         -27468.1
=====
=====
```

D) Binary Fe-Cr-C System:

Condition P=101325 Pa, T=1000 K, N=1, X(C)=0.1, X(Cr)=0.2

<i>ref_ph</i> <i>ref_T/P</i>	<i>AC(C)</i>	<i>ACR(C)</i>	<i>MU(C)</i>	<i>MUR(C)</i>	<i>Gm(gra)</i>	<i>NP(gra)</i>
	<i>AC(Cr)</i>	<i>ACR(Cr)</i>	<i>MU(Cr)</i>	<i>MUR(Cr)</i>	<i>Gm(bcc)</i>	<i>NP(bcc)</i>
	<i>AC(Fe)</i>	<i>ACR(Fe)</i>	<i>MU(Fe)</i>	<i>MUR(Fe)</i>	<i>Gm(m7c3)</i>	<i>NP(m7c3)</i>
	<i>Gm</i>					
C-SER	.01840	.01840	-33218.3	-33218.3	-12658.3	0
Cr-SER	.00132	.00132	-55129.9	-55129.9	-42863.3	.66692425
Fe-SER	.00602	.00602	-42503.4	-42503.4	-46577.0	.33307575
						-44100.2

Why? BCC & M7C3 are stable at 1000 K
 $MU0(C) @HSER = -12658.3$
 $MU0(C@1000K) = -12658.3$
 $MU(C@1000K) = MU0(C@1000K) + 'ex-contribution(C)'$
 $= -33218.3$
 $MU0(Cr) @HSER = -36693.7$
 $MU0(Cr@1000K) = -36693.7$
 $MU(Cr@1000K) = MU0(Cr@1000K) + 'ex-contribution(Cr)'$
 $= -55129.9$
 $MU0(Fe) @HSER = -42271.7$
 $MU0(Fe@1000K) = -42271.7$
 $MU(Fe@1000K) = MU0(Fe@1000K) + 'ex-contribution(Fe)'$
 $= -42503.4$

--> MU(C) = MU(C@1000K) = -33218.3
AC(C) = exp[MU(C)/RT] = 0.0184
MUR(C) = MU(C@1000K) = -33218.3
ACR(C) = exp[MUR(C)/RT] = 0.0184
--> MU(Cr) = MU(Cr@1000K) = -55129.9
AC(Cr) = exp[MU(Cr)/RT] = 0.00132
MUR(Cr) = MU(Cr@1000K) = -55129.9
ACR(Cr) = exp[MUR(Cr)/RT] = 0.00132
--> MU(Fe) = MU(Fe@1000K) = -42503.4
AC(Fe) = exp[MU(Fe)/RT] = 0.00602
MUR(Fe) = MU(Fe@1000K) = -42503.4
ACR(Fe) = exp[MUR(Fe)/RT] = 0.00602

C-gra *	101325	.01840	.08435	-33218.3	-20560.0	-12658.3	0
Cr-bcc *	101325	.00132	.10889	-55129.9	-18436.2	-42863.3	.66692425
Fe-bcc *	101325	.00602	.97252	-42503.4	-231.694	-46577.0	.33307575
							-44100.2

Why? BCC & M7C3 are stable at 1000 K
 $MU0(C) @1000K = -12658.3$
 $MU0(C@1000K) = -12658.3$

```

    MU(C@1000K) = MU0(C@1000K) + 'ex-contribution(C)'
    = -33218.3
    MU0(Cr)@1000K = -36693.7
    MU0(Cr@1000K) = -36693.7
    MU(Cr@1000K) = MU0(Cr@1000K) + 'ex-contribution(Cr)'
    = -55129.9
    MU0(Fe)@1000K = -42271.7
    MU0(Fe@1000K) = -42271.7
    MU(Fe@1000K) = MU0(Fe@1000K) + 'ex-contribution(Fe)'
    = -42503.4
--> MU(C) = MU(C@1000K) = -33218.3
    AC(C) = exp[MU(C)/RT] = 0.0184
    MUR(C) = MU(C@1000K) - MU0(C)@1000K
    = (-33218.3) - (-12658.3) = -20560.0
    ACR(C) = exp[MUR(C)/RT] = 0.08435
--> MU(Cr) = MU(Cr@1000K) = -55129.9
    AC(Cr) = exp[MU(Cr)/RT] = 0.00132
    MUR(Cr) = MU(Cr@1000K) - MU0(Cr)@1000K
    = (-55129.9) - (-36693.7) = -18436.2
    ACR(Cr) = exp[MUR(Cr)/RT] = 0.10889
--> MU(Fe) = MU(Fe@1000K) = -42503.4
    AC(Fe) = exp[MU(Fe)/RT] = 0.00602
    MUR(Fe) = MU(Fe@1000K) - MU0(Fe)@1000K
    = (-42503.4) - (-42271.7) = -231.694
    ACR(Fe) = exp[MUR(Fe)/RT] = 0.97252
-----
C-gra 298.15 101325 .01840 .02261 -33218.3 -31506.7 -12658.3 0
Cr-bcc 298.15 101325 .00132 .00307 -55129.9 -48110.8 -42863.3 .66692425
Fe-bcc 298.15 101325 .00602 .01602 -42503.4 -34370.7 -46577.0 .33307575
                                         -44100.2
-----
```

Why? BCC & M7C3 are stable at 1000 K

```

    MU0(C)@298.15 = -1711.5
    MU0(C@1000K) = -1265.8
    MU(C@1000K) = MU0(C@1000K) + 'ex-contribution(C)'
    = -33218.3
    MU0(Cr)@298.15 = -7019.1
    MU0(Cr@1000K) = -36693.7
    MU(Cr@1000K) = MU0(Cr@1000K) + 'ex-contribution(Cr)'
    = -55129.9
    MU0(Fe)@298.15 = -8132.1
    MU0(Fe@1000K) = -42271.7
    MU(Fe@1000K) = MU0(Fe@1000K) + 'ex-contribution(Fe)'
    = -42277.8
--> MU(C) = MU(C@1000K) = -33218.3
    AC(C) = exp[MU(C)/RT] = 0.0184
    MUR(C) = MU(C@1000K) - MU0(C)@298.15
    = (-33218.3) - (-1711.5) = -31506.7
    ACR(C) = exp[MUR(C)/RT] = 0.02261
--> MU(Cr) = MU(Cr@1000K) = -55129.9
    AC(Cr) = exp[MU(Cr)/RT] = 0.00132
    MUR(Cr) = MU(Cr@1000K) - MU0(Cr)@298.15
    = (-55129.9) - (-7019.1) = -48110.8
    ACR(Cr) = exp[MUR(Cr)/RT] = 0.00307
--> MU(Fe) = MU(Fe@1000K) = -42503.4
    AC(Fe) = exp[MU(Fe)/RT] = 0.00602
    MUR(Fe) = MU(Fe@1000K) - MU0(Fe)@298.15
    = (-42503.4) - (-8132.1) = -34370.7
    ACR(Fe) = exp[MUR(Fe)/RT] = 0.01602
-----
```

Some Typical Questions about Reference States

Questions:

Why do I get a different activity when I set reference state as gas at temperature versus when I use the command "show acr(o,gas)"?

What reference states should I use when comparing activity related to a gas versus activity related to a solid so that I am comparing the same types of number?

If I have a gas system (H, N, O, C) and set the reference state for O as "gas" I get one activity, but if I type the command "show acr(o,gas)" I get another value of activity which is equal to the partial pressure of O (not O_2). The same is true for H and N . So what is the difference in reference state when I "set ref state" versus specify the reference state in the "show" command?

Explanations:

As described above, there are two different concepts of activity (AC/ACR - with/without referring to defined reference states):

- For system components: $\text{AC}(\text{comp})$ and $\text{ACR}(\text{comp})$
e.g., AC(O) and ACR(O)
 $\text{AC(O}_2)$ and $\text{ACR(O}_2)$ (if O₂ is set as a system component)
 - For species (phase constituents): $\text{AC(sp, ph)} = \text{ACR(sp, ph)}$
*e.g., $\text{AC(O, GAS)} = \text{ACR(O, GAS)}$,
 $\text{AC(O}_2, \text{GAS}) = \text{ACR(O}_2, \text{GAS})$
 $\text{AC(O}_2, \text{AQ}) = \text{ACR(O}_2, \text{AQ})$*

Normally, $\text{ACR}(\text{O})$ is not the same as $\text{ACR}(\text{O}, \text{GAS})$, because they have totally different meanings; their difference could end up to many orders of magnitude. Since the ideal EOS and ideal mixing model is used for the GAS phase in SSUB3, the $\text{ACR}(\text{O}, \text{GAS})$ is always equal to the $y(\text{O}, \text{GAS})$ quantity.

The proper way for setting reference states for system components O, H and N, e.g., in a defined He-O-H-C-N+Me system, is normally as follows:

S-R-S O GAS * 101325
S-R-S H GAS * 101325
S-R-S N GAS * 101325

And if graphite (or C_S) is included in a defined system, for the component C the reference state is usually defined as:

S-R-S C GRA * 101325

Note that it is often to set the constant pressure as 1E5 Pascal (*i.e.*, 1 bar), instead of 101325 Pascal (1 atm), for convenience and simplicity.

Furthermore, $AC(O)$ is unique for a defined system, while $ACR(sp, ph)$ for O related to GAS is different from that for O related to alloy phases. Please note that, for alloy solution phases with more than one sublattice sites, e.g., FCC_A1 defined as $(Fe, Cr, O) 1 (C, N, Va) 1$, it is impossible to calculate $ACR(O, FCC)$.

If you want to investigate the tendency of gas absorption/extraction in metallic alloy or other solid phases, you can first calculate e.g. ACR(O) quantity *related to* gaseous mixture (only) and metallic alloy/solid phases (without gas phase included), in separate calculations but using the same settings of reference states (for O, H and N). Then, by comparing the ACR(O) quantity in two separate cases, you can judge the tendency. Under such a circumstance, the gas phase is unnecessary in equilibrium with the alloy/solid phases if including all of them in another equilibrium calculation (see below).

The other way is like that demonstrated by the TCC Examples 11 and 32, *i.e.*, including gaseous mixture and alloy/solid phases in a system. Under such a situation, the calculation is made in a sense of full equilibrium in the combined system (the gaseous mixture is either stable or unstable along with the considered alloy/solid phases in the defined system).

The *TCC Example 11 (oxidation of Cu₂S in an H-O gas mixture)* is an equilibrium calculation of appropriate input amounts of H and O (and their ratio) to avoid Cu₂O formation (from Cu₂S). The calculation is intended to only show the input amounts of H and O (and their ratio); but if having set their reference state (as mentioned above), one can list out the ACR(O) and ACR(H), as well as ACR(O, GAS) and ACR(H, GAS), under two different equilibria (one as Cu₂O already formed, and the other Cu₂O starts to form), so that one can understand the redox condition in the system.

The *TCC Example 32* is similar to Example 11, but the former one is to calculate the ACR(O, GAS) and ACR(O₂, GAS) conditions (*i.e.*, partial pressures of O and O₂ in the gaseous mixture) for forming Cr₂O₃ and other oxides on the surface layer of a steel. The calculation is made after setting the reference state of system component O as the GAS phase under current temperature of a constant pressure of 1 bar. As indicated above, such a calculation does not necessarily require the GAS mixture phase being stable in the system, since we are only interested in knowing the partial pressures of O (and O₂) in the atmosphere (which is sustained by the considered gas species and often plus some inactive species such as He), which is related to the ACR(O) quantity in the whole system (controlling the O behaviour in both GAS mixture and alloy/solid phases).

Index on Thermo-Calc Commands in Various Modules

Module Command	Brief Description	Module	Page
ADD_COMMENT	Add/amend/delete comments on parameters and others.	GES	
ADD_INITIAL_EQUILIBRIUM	Add initial equilibrium points for mapping.	POLY	
ADD_LABEL_TEXT	Add labeling text(s) on the current diagram, at specified coordinates. Optionally, one can also let the program automatically add a text by first calculating an equilibrium at the specified coordinates and then making a label out of the stable phase names.	POST	
ADVANCED_OPTIONS	Set advanced options for the current system and calculations. Note that since TCCS this new command is replacing the old commands SPECIAL_OPTIONS and SET_MINIMIZATION_OPTIONS.	POLY	
		ED-EXP	
AMEND_ELEMENT_DATA	Change the data for a specific element.	GES	
AMEND_PARAMETER	Change the temperature-pressure function of a specific parameter.	GES	
		PARROT	
AMEND_PHASE_DESCRIPTION	Specify the phase description of a specific phase if it has a miscibility gap, special excess model or that it has a specific additional contribution to its Gibbs energy.	GES	
AMEND_RECORD	Amend some of the already-created record, <i>but has not been implemented yet.</i>	REACTOR	
AMEND_SELECTION	Change the pre-defined system (elements or species or system that have been defined with the command DEFINE_ELEMENTS or DEFINE_SPECIES or DEFINE_SYSTEM).	TDB	
AMEND_STORED_EQUILIBRIA	Give information about the calculated regions after a STEP or MAP command, so parts of a calculation that are redundant or where metastable equilibria have been calculated can then be suspended.	POLY	
AMEND_SYMBOL	Calculate the current values of a specific 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 possibly modify the definitions of variables or functions.	GES	
APPEND_DATABASE	Append data from a file or additional database to the current set of data already read from another database.	TDB	
APPEND_EXPERIMENTAL_DATA	Append an EXP file (experimental data and text, that are saved in a file which is prepared according to syntax of the DATAPLOT graphical language) onto the current diagram.	POST	
BACK	Switch control back to the most recent module. (Notes: From POST module, it always returns to the POLY or TAB module; from ED-EXP module, it always returns to the PARROT module)	SYS TDB GES TAB POLY POST PARROT ED-EXP REACTOR	

CHANGE_STATUS	Change the status for a specific phase, component (element) or species.	GES	
		POLY	
		ED-EXP	
CLOSE_FILE	Close an unit number previously opened by the program with an OPEN_FILE command.	SYS	
COMMENT	Write brief comments or description on a newly created equilibrium (experimental) data point or a set of points given in a table (but only used in POP or DOP files).	ED-EXP	
COMPILE_EXPERIMENTS	Start compiling all experimental equilibrium data saved on a POP file, with the current SETUP file.	PARROT	
COMPUTE_ALL_EQUILIBRIA	Calculate all the equilibria from the current to the last experimental points.	ED-EXP	
COMPUTE_EQUILIBRIUM	Calculate the current equilibrium.	POLY	
		ED-EXP	
COMPUTE_TRANSITION	As a combination of the CHANGE_STATUS and SET_CONDITION commands, it allows direct calculation when a phase may form by varying one condition.	POLY	
CONTINUE_OPTIMIZATION	Continues the optimization using the same Hessian matrix.	PARROT	
CREATE_3D_PLOTFILE	Automatically convert all the default/predefined graphical definition information and selected data points (expressed and stored in specific tables or *.TAB files, which are taken from the current and/or previous MAP/STEP calculations) into an appropriate format, and creates an *.WRL (Virtual Reality Markup Language) file for viewing by a VRML viewer such as plugins to web browsers or standalone programs.	POST	
CREATE_DIVIDERS	Define the segment boundaries and the ways on how to distribute phases between various segments and to split the feeds of heat and mass.	REACTOR	
CREATE_NEW_EQUILIBRIUM	Create a new equilibrium with a different set of conditions and phases (but normally with the same set of components).	POLY	
		ED-EXP	
CREATE_NEW_STORE_FILE	Create a PAR file to store workspaces used by the GES, POLY and PARROT modules.	PARROT	
CREATE_PIPES	Define the pipes between various segments and the ways on how the mass and heat transport between the surroundings and the reactor, among various segments in the reactor, and along various dividers.	REACTOR	
CREATE_PROBE	Create probe, <i>but has not been implemented yet.</i>	REACTOR	
CREATE_STAGE_BOXES	Create 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.	REACTOR	
DATABASE_INFORMATION	Display a short description of the current database (including information on covered systems, used models, valid ranges for parameters in temperature and composition, major applications, and so on).	TDB	

DEFINE_COMPONENTS	Define the current set of components in the system.	POLY	
		ED-EXP	
DEFINE_ELEMENT	Define the system in terms of elements.	TDB	
DEFINE_DIAGRAM	As an extension of the DEFINE_MATERIAL command, it allows automatic calculation and plotting of a diagram.	POLY	
DEFINE_MATERIAL	Read a system from a specified database, set the composition and temperature (and pressure equal to 1 bar) and calculate the equilibrium.	POLY	
DEFINE_SPECIES	Define the system in terms of species.	TDB	
DEFINE_SYSTEM	Define the system in terms of either elements or species, depending on the default value of the keyword of the current database.	TDB	
DELETE	<i>Not implemented yet</i>	GES	
DELETE_INITIAL_EQUILIBRIUM	Delete all or some specified initial equilibria before the MAP and STEP commands.	POLY	
DELETE_SYMBOL	Delete symbols (constants, variables, functions or tables that are entered by the command ENTER-SYMBOL).	POLY	
		ED-EXP	
DUMP_DIAGRAM	Similar to the PRINT_DIAGRAM command, quick save (dump) the current diagram to a file in a specified graphical format, but with average quality.	POST	
EDIT_EXPERIMENTS	Switch to the sub-module for editing experimental equilibria, <i>i.e.</i> , the ED_EXP module.	PARROT	
ENTER_FUNCTION	Define a new thermodynamic function to be included in the sixth column of tabular output.	TAB	
ENTER_ELEMENT	Enter element(s) into the current system.	GES	
ENTER_PARAMETER	Enter TP-function(s) for a specific parameter for a specified phase.	GES	
		PARROT	
ENTER_PHASE	Enter a certain phase (and specify all the details in phase type, sublattices, constitutions, <i>etc.</i>) into the current system.	GES	
ENTERREACTION	Identical to the TABULATEREACTION command, tabulate the thermodynamic property changes for an entered reaction, as a table, and plotted as a graph or saved as an experimental file (with an extension of EXP).	TAB	
ENTER_SPECIES	Enter species that are made up from the already entered elements into the current system.	GES	
ENTER_SYMBOL	Define quantities (constants, variables, functions or tables) as symbols that are convenient for the user.	GES	
		POLY	
		POST	
		ED-EXP	
EVALUATE_FUNCTIONS	Evaluate and list the value(s) of one or more or all entered functions or variables.	POLY	
		ED-EXP	
EXECUTE_UNUSED_SPECIES	Exclude species that do not enter any of your current defined phases from the species list, in order to save program workspace. (Disabled since TCCN)	TDB	
EXECUTE_POLY3_COMMAND	Prompt for directly utilizing all kinds of POLY commands inside the REACTOR module.	REACTOR	

EXIT	Terminate the program and returns to the operating system. Unless a SAVE command has been executed (in various modules), all data and results will be lost.	SYS	
		TDB	
		GES	
		TAB	
		POLY	
		POST	
		PARROT	
		REACTOR	
EXPERIMENT	Change the value or uncertainty of an experiment or add more experimental information.	ED-EXP	
EXPORT	Transfer a calculated value from an equilibrium to a optimizing variable (an "V" variable) in the PARROT workspace.	ED-EXP	
FLUSH_BUFFER	Terminate the current block, save it to the work file and start a new block (but only used in POP or DOP files).	ED-EXP	
GET_DATA	Retrieve all information concerning a defined system from the current database, and enter the defined system's elements, species, phases, and the connected parameters obtained from either the primarily switched or additionally appending database(s) to the GES5 and/or DICTRA workspace.	TDB	
GOTO_MODULE	Switch between various modules.	SYS	
		TDB	
		GES	
		TAB	
		POLY	
		PARROT	
		REACTOR	
GRAPHICS_PLOT	Equivalent to the STEP_WITH_OPTIONS command, it makes a STEPPING calculation based on the currently-selected equilibrium point.	ED-EXP	
HELP	List the available commands or give an explanation of a specified command of the current module.	SYS	
		TDB	
		GES	
		TAB	
		POLY	
		POST	
		PARROT	
		ED-EXP	
		REACTOR	
HP_CALCULATOR	Open the QBA simple interactive calculator using reverse polish notations.	SYS	
IMPORT	Transfer the value of one of the optimizing variables to a constant (opposite to the EXPORT command).	ED-EXP	

INFORMATION	Give basic information about various subjects in the current module.	SYS	
		TDB	
		GES	
		POLY	
		PARROT	
		ED-EXP	
		REACTOR	
LABEL_DATA	Add a label to the experimental equilibrium point, either as a single point or several points given individually or in a table.	ED-EXP	
LIST_ALL_EQUILIBRIA	List all the calculated the details of all equilibrium points from the current one to the last.	ED-EXP	
LIST_ALL_VARIABLES	List the values and status of all variables in the PARROT workspace, on screen or on a text file.	PARROT	
LIST_AXIS_VARIABLE	List all the axes already set by the command SET_AXIS_VARIABLE.	POLY	
LIST_CONDITIONS	List all conditions that are set by the command SET_CONDITION and the command sequence CHANGE_STATUS PHASE ... =FIX 0 or 1.	POLY	
	List the present values of the output and optimizing conditions on screen or a simple text file which can be opened and edited by a text editor later on.	ED-EXP	
LIST_CONSTITUITION	List all the constitution (as site-fractions) for all the phases in the defined system on screen.	GES	
LIST_DATA	List all data for the entered system in a readable manner on the output file or on screen.	GES	
LIST_DATA_TABLE	List various properties (<i>i.e.</i> , the Y-axis variable on a property diagram) that change with an independent variable (<i>i.e.</i> , the X-axis variable on a property diagram), which have already been defined in the POST module 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).	POST	
LIST_DATABASE	List all elements, species, phases or phase constituents in the current database.	TDB	
LIST DESIGN	List the design details of the creator, <i>but has not been implemented yet.</i>	REACTOR	
LIST_EQUILIBRIUM	List the result from the last calculated equilibrium.	POLY	
		ED-EXP	
LIST_INITIAL_EQUILIBRIA	List all the initial equilibria that are added with the ADD-INITIAL-EQUILIBRIUM command.	POLY	
LIST_PARAMETER	List the TP-function(s) of a specific parameter for a specified phase on screen.	GES	
		PARROT	
LIST_PHASE_DATA	List all data for a specific phase in a readable manner on screen.	GES	
		PARROT	
LIST_PLOT_SETTINGS	List the present values of most parameters specifying the type of diagram to be plotted.	POST	
LIST_RECORDS	List all the already-created records for the reactor (including stage boxes, dividers, pipes, <i>etc.</i>).	REACTOR	

LIST_RESULT	List result of the current optimization run on screen or a specified file.	PARROT	
LIST_STATUS	List status of components, species and/or phases.	GES	
		POLY	
		ED-EXP	
LIST_STORE_FILE	Write on screen the name of the store file and its full path.	PARROT	
LIST_SUBSTANCE	List all species in the current database with a certain set of elements.	TAB	
LIST_SYMBOL LIST_SYMBOLS	List the definitions of a specified or all defined constants, functions, variables and tables.	GES	
		POLY	
		POST	
		ED-EXP	
LIST_SYMBOL_IN_GES	List TP-function(s) for the entered model parameters for phases in the system on screen.	PARROT	
LIST_SYSTEM	List all elements, species, phases or phase constituents in the defined system.	TDB	
LOAD_INITIAL_EQUILIBRIUM	Copy all the conditions and calculated results from a previously added initial equilibrium to the current equilibrium. The current conditions and results are lost.	POLY	
MACRO_FILE_OPEN	Open a MACRO file that predefines sequences of commands, and then execute them. The MACRO must be terminated with the EXIT command, or in the SYS, GES, POLY, PARROT or POST module with the SET-INTERACTIVE command.	SYS	
		TAB	
		POLY	
		PARROT	
MAKE_EXPERIMENTAL_DATAFILE	Write the graphical information on screen or an EXP file using the DATAPLOT format.	POST	
MAKE_POP_FILE	Write the interactively-made changes and additions of experimental information to a new POP/DOP file or overwrite them on an old POP/DOP file.	ED-EXP	
MAP	Map a phase diagram from one or more initial equilibria.	POLY	
MERGE_WITH_DATABASE	Get data from a file or additional database and merge them with the current set of data already read from another database. (<i>Disabled since TCCP</i>).	TDB	
MODIFY_LABEL_TEXT	Move a label created with the ADD_LABEL command to another position, or replace its text with another one.	POST	
NEW_DIRECTORY_FILE	Open a new database initiation file (or called database directory file) generated by a local database manager or user, for accessing additional databases that are not predefined in the original database initiation file.	TDB	
OPEN_FILE	Specify a text file to be opened for uses in other commands where a unit number is necessary.	SYS	
OPTIMIZE_VARIABLES	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.	PARROT	
PATCH PATCH_WORKSPACE	<i>It is only for those who think they know what they are doing! It has been taken away from the SYS module since TCCR.</i>	TAB	
		GES	
		POLY	
		POST	
		REACTOR	

PLOT_DIAGRAM	Plot graphic information on a specific graphic device using the plot format set by the SET_PLOT_FORMAT command.	POST	
POST	Switch to the post-processor, the POST module.	POLY	
PRINT_DIAGRAM	Print the current diagram directly on a connected printer with high quality.	POST	
QUICK_EXPERIMENTAL_PLOT	Similar to the APPEND EXPERIMENTAL DATA command, but it can be used when there is no graphical information to be plotted in the POLY 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) file.	POST	
READ_WORKSPACE(S)	Read the workspaces and calculated results from a module-related and specifically formatted file that was saved previously with a SAVE_WORKSPACES command.	POLY	
		ED-EXP	
		REACTOR	
READ_GES_WORKSPACE	Read the GES workspaces from a GES file that was saved previously with a SAVE_GES_WORKSPACES command.	GES	
READ_PARROT_WORKSPACES	Read the PARROT workspaces from an *.PAR file that was saved previously with a SAVE_PARROT_WORKSPACES command.	PARROT	
RECOVER_START_VALUES	Recover the start values in equilibrium calculations. Note that it has already been disabled since TCCS.	POLY	
RECOVER_VARIABLES	Set the values of all variables back to their start values.	PARROT	
REINITIATE REINITIATE_MODULE	Reinitiate the current module to the state it was when first entered. All data in the module-related workspaces are erased, and all variables are initiated to their default values.	GES	
		POLY	
		PARROT	
		ED-EXP	
REINITIATE_PLOT_SETTINGS	Define all the parameters describing a diagram with default values.	POST	
REJECT	Reject elements, species, phases or phase constituents that can form from the defined elements and species.	TDB	
RESCALE_VARIABLES	Copy the current values of all the parameters to their start values and the scaling factors for further optimization.	PARROT	
RESTORE	Restore already explicitly rejected elements, species, phases or constituents; but it can not restore a completely rejected system.	TDB	
RESTORE_ALL_WEIGHTS	Restores 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	ED-EXP	
RESTORE_PHASE_IN_PLOT	Restore a phase that has been previously suspended from the plot with the SUSPEND_PHASE_IN_PLOT command.	POST	
SAVE_WORKSPACE(S)	Save the current status and module-related workspaces of the program on a specifically formatted file.	POLY	
		ED-EXP	
		REACTOR	
SAVE_GES_WORKSPACE	Save the current status and GES workspaces of the program on a GES file.	GES	

SAVE_PARROT_WORKSPACES	Save the current status and PARROT workspaces of the program on a PAR file.	PARROT	
SELECT_EQUILIBRIUM	Switch between the previously created initial equilibria.	POLY	
		ED-EXP	
SET_ALL_START_VALUES	Set all necessary start values, e.g., temperature, pressure, and for each phase as if it should be stable and its constitution.	POLY	
		ED-EXP	
SET_ALTERNATE_MODE	Turn on or off the alternate mode.	PARROT	
SET_ALTERNATE_CONDITION	Specify the alternate conditions when the <i>alternate mode</i> is set on in the PARROT module. The command syntax is the almost the same as for the SET_CONDITION command, while the uncertainty should also be specified.	ED-EXP	
SET_AUTO_APPEND_DATABASE	Decide on automatically appending data from an additional database (to the defined system with data from the current database).	TDB	
SET_AXIS_LENGTH	Change the real length of a specific axis in inches.	POST	
SET_AXIS_PLOT_STATUS	Specify whether the axis title texts and axis labeling texts are to be plotted or not on a diagram.	POST	
SET_AXIS_TEXT_STATUS	Change the text for a specific axis to the automatic text given by axis specification or a text given by user.	POST	
SET_AXIS_TYPE	Change the axis type of a specific axis between linear, logarithmic and inverse axis.	POST	
SET_AXIS_VARIABLE	Set an axis for calculating (STEP or MAP) a diagram.	POLY	
SET_COLOR	Select different colors or line types on some types of information on a diagram.	POST	
SET_COMMAND_UNITS	Set input unit number (for reading inputs already prepared by a textual editor on a file) and output unit number (for returning outputs).	SYS	
SET_CONDITION	Specify the one or several equilibrium conditions for the current equilibrium calculation, or reset them with new values.	POLY	
		ED-EXP	
SET_CORNER_TEXT	Write texts in the corners (normally as subtitles) on a plotted diagram.	POST	
SET_DIAGRAM_AXIS	Specify an axis variable for a plot or a listing.	POST	
SET_DIAGRAM_TYPE	Choose the diagram type as perpendicular plot or triangular plot (Gibbs triangle, especially for ternary systems).	POST	
SET_ENERGY_UNIT	Set the energy unit as calories or joule, in all the subsequent outputs (tables, graphs, and files) in the current calculation operation.	TAB	
SET_EXTERNAL_PARAMETER	This new command since TCCS allows the optimizations of parameters used in so-called external models (which may not be as fully-implemented/integrated parts inside the GES system).	PARROT	
SET_FIX_VARIABLE	Prescribe a fixed value to a variable that will be considered as a constant at the optimization.	PARROT	
SET_FONT	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).	POST	
SET_INPUT_AMOUNTS	Specify how a system is made up from mixing of various substances.	POLY	

SET_INTERACTIVE	Reset the input and output units to their initial values, <i>i.e.</i> , keyboard and screen. Remember to add this as the last command to MACRO files.	SYS	
SET_INTERACTIVE_MODE		GES	
		TAB	
		POLY	
		POST	
		PARROT	
SET_LABEL_CURVE_OPTION	Identify the curves with digits and then list the meaning of these digits beside the plot.	POST	
SET_LOG_FILE	Save everything the user types in the TCC software, on a simple textual file that can be edited as a MACRO file.	SYS	
SET_MINIMIZATION_OPTIONS	Decide the options for adjusting Global Minimization settings, and other for adjusting the normal POLY optimization routine (general calculations). Note that it has already been merged/replaced by the new ADVANCED_OPTIONS command since TCCS .	POLY	
SET_NUMERICAL_LIMITS	Change the criteria for convergence.	ED-EXP	
SET_OPTIMIZING_CONDITION	Specify the conditions for the optimization.	PARROT	
SET_OPTIMIZING_VARIABLE	Specify a variable to be estimated at the optimization (and a start value for it).	PARROT	
SET_OUTPUT_LEVELS	Choose what type of information the PARROT module should give during the optimization procedure and when the result is listed.	PARROT	
SET_PLOT_ENVIRONMENT	Set the plot devices.	SYS	
SET_PLOT_FORMAT	Set the format of the graphical output to different graphical devices.	POST	
SET_PLOT_OPTION	Toggle on (Y) or off (N) the plotting of some options on a plotted diagram.	TAB	
SET_PLOT_SIZE	Change the size of the diagram by specifying a global plot size (as a relative scaling factor). Note the real size of the plotted diagram depends on what output device the user has chosen by the SET_PLOT_FORMAT command.	POST	
SET_PREFIX_SCALING	Set the prefix scaling of remote exponents with certain powers for a specific axis.	POST	
SET_R_AND_P_NORM	Change the default values of the gas constant (R) and/or the pressure 1 bar (P) for interpreting the energy and pressure units.	GES	
SET_RASTER_STATUS	Select to have a raster (<i>i.e.</i> , with gridlines on both axis directions) plotted in the current diagram.	POST	
SET_REFERENCE_STATE	Set the reference state for a component (referred to a selected phase, temperature and pressure) if the default reference state in the used database(s) does not suit.	POLY	
SET_SCALING_STATUS	Set the scaling status of a specified axis variable as automatic or manual scaling. For manual scaling, a minimum and a maximum value must be given.	POST	
SET_SCALED_VARIABLE	Specify the start value for a specific optimizing variable (similar to the SET_OPTIMIZING_VARIABLE command, but it is functional for only one optimizing variable at one time), as well as its minimum and maximum values.	ED-EXP	
		POST	
		PARROT	

SET_START_CONSTITUITION	Set the start values of constitutions for an individual phase.	POLY	
		ED-EXP	
SET_START_VALUE	Set a start value of an individual variable.	POLY	
		ED-EXP	
SET_STORE_FILE	Specify a store file (a work PAR file, that was previously generated by the command CREATE_NEW_STORE_FILE) to be used for compilation and optimization.	PARROT	
SET_TC_OPTIONS	This new command since TCCS lets the user to pre-define the “default” values of three general options for performing all the sequential single-points, stepping and mapping calculations.	SYS	
SET_TERMINAL	Define the characteristics of the terminal.	SYS	
SET_TIC_TYPE	Change the drawing of the axis tics (the placement of tic marks on the axis, e.g., inside or outside or no tic marks).	POST	
SET_TIELINE_STATUS	Select to plot the tie-lines in two-phase fields, if the tie-lines are in the plane of the calculation.	POST	
SET_TITLE	Specify a title that will appear on all listed table and plotted diagrams from the POST module in the current run of Thermo-Calc.	POST	
SET_TRUE_MANUAL_SCALLING	Toggle the manual scaling on or off for a specific axis.	POST	
SET_WEIGHT	Change the contribution scale of the absolute difference between the experimental and calculated values to the sum of error (the uncertainty) for a single experimental point or for a set of equilibria.	ED-EXP	
SHOW_PROBE	Show the details of the created probe, <i>but has not been implemented yet.</i>	REACTOR	
SHOW_VALUE	Show on the screen the current value of any state variable, function or variable.	POLY	
		ED-EXP	
SPECIAL_OPTIONS	Set special options for the current system and calculations. <i>Note that it has already been merged/replaced by the new ADVANCED_OPTIONS command since TCCS .</i>	POLY	
		ED-EXP	
START_SIMULATION	Start the reactor simulation, and list the results for all iterations.	REACTOR	
STEP_WITH_OPTIONS	Initiate the stepping procedure with a specific option.	POLY	
STOP_ON_ERROR	Define the way to prevent that an erroneous sequence of commands to the program will cause waste of computer activity: either to terminate after an illegal or ambiguous command (ON), or to reset the effect of such a command (OFF).	SYS	
STORE_ALL_WEIGHTS	Store a new weight-set that is the current situation of specified weights for all the different experimental points read from the currently-selected experimental data block in an assessment.	ED-EXP	
SWITCH_DATABASE	Switch from the current database to a new one, and reinitializes the entire TDB module for defining a system (or the entire TAB module for defining a substance or reaction) and the GESS workspace for storing retrieved data.	TDB	
		TAB	
SUSPEND_PHASE_IN_PLOT	Specify the lines originating from the presence of a certain phase shall not be plotted on a diagram.	POST	

TABLE_HEAD	Enter many experimental measurements of the same type in a table format. These commands should always be used together and in a sequence (but only used in POP or DOP files).	ED-EXP	
TABULATE			POLY
			POST
TABULATE_DERIVATIVES	Automatically calculate all partial derivatives for a phase at a given composition. <i>It is mainly for system debugging purposes !</i>	TAB	
TABULATEREACTION	Tabulate thermodynamic property changes for a chemical reaction as a table, or one of the property's variation can be plotted against temperature as a graph, or saved as an experimental file (with an extension of EXP).	TAB	
TABULATESUBSTANCE	Tabulate the thermodynamic data of a substance (and of a solution phase at a fixed composition) as a table, or one of the property's variation can be plotted against temperature as a graph, or saved as an experimental file (with an extension of EXP).	TAB	
TRANSFER_START_VALUES	Transfer the site fractions from a previous successfully-calculated equilibria to the present experimental point.	ED-EXP	

Notes:

- All module commands can be used in a MACRO file (*.TCM).
- Some of the above module commands can be used in a relevant module-dependent file (such as *.TDB/*.DAT, *.SETUP, *.POP/*.DOP).

Index on Thermo-Calc Commands in Various Types of Files

File Command	Brief Description	File Ext.	Page
ADD_CONSTITUENT	TDB file syntax for adding constituents of a phase that has already some constituents.	TDB/DAT	
ADD_REFERENCES	TDB file syntax for adding more reference lists	TDB/DAT	
ALLOTROPIC_PHASE	TDB file syntax for defining a compound phase (same as the COMPOUND_PHASE keyword but does not enter the constituent as a species to the data structure; it should therefore be used if the species is already defined).	TDB/DAT	
ASSESSED_SYSTEMS	TDB file syntax for describing some special options when the TDB, GES and POLY models deal with the existing systems with assessed data.	TDB/DAT	
CASE / ENDCASE	TDB file syntax for enforcing a definition on which type of the following boolean algebra will operate.	TDB/DAT	
COMMENT	POP/DOP file syntax for writing brief comments or descriptive information on a newly created equilibrium (experimental) data point or a set of points given in a table.	POP/DOP	
COMPOUND_PHASE	TDB file syntax for defining a compound phase (The species name and stoichiometric formula must be identical. The phase has this species as its only constituent).	TDB/DAT	
CONSTITUENT	TDB file syntax for defining constituents of a phase.	TDB/DAT	
DATABASE_INFORMATION	TDB file syntax for defining a text for the detailed description of the current database, that can be listed with the TDB command DATABASE_INFORMATION.	TDB/DAT	
DEFAULT_COMMAND	TDB file syntax for defining default commands to be executed by the TDB module at database initialization.	TDB/DAT	
DEFINE_SYSTEM_DEFAULT	TDB file syntax for defining default value to ELEMENT or SPECIES in the TDB command DEFINE_SYSTEM.	TDB/DAT	
DIFFUSION	TDB file syntax for specifying what type of DICTRA diffusion model to use for a phase if the default model is not desired.	TDB/DAT	
ELEMENT	TDB file syntax for defining an element.	TDB/DAT	
FLUSH_BUFFER	POP/DOP file syntax for terminating the current block, saving it to the work file and start a new block.	POP/DOP	
FTP_FILE	TDB file syntax for specifying a special function random file whose function names correspond to the record numbers where these record names and their functions are stored.	TDB/DAT	
FUNCTION	TDB file syntax for defining a function.	TDB/DAT	
LIST_OF_REFERENCE	TDB file syntax for starting a reference list that is directly coded in the database definition file.	TDB/DAT	
OPTION	TDB file syntax for defining an “alloy” (with a name, a major component and a number of alloying elements).	TDB/DAT	
PARAMETER	TDB file syntax for defining a normal GES parameter for thermodynamic data, as well as five special extensions suitable for kinetic data used in the DICTRA software.	TDB/DAT	
PHASE	TDB file syntax for defining a phase.	TDB/DAT	
REFERENCE_FILE	TDB file syntax for specifying a special reference file containing a complete structured list of the references for the various parameters (and sometimes functions) in the database.	TDB/DAT	

SET_INTERACTIVE	TCM file syntax for resetting the input and output units to their initial values, <i>i.e.</i> , keyboard and screen, and letting the prompt be in the TAB, POLY, POST, GES, PARROT or SYS module. Remember to add this as the last command to MACRO files.	TCM/LOG	
SPECIES	TDB file syntax for defining a species.	TDB/DAT	
TABLE	TDB file syntax for defining a table of Gibbs energy as a function of temperature where the values are given from the start temperature to the end temperature, at a step of the delta temperature.	TDB/DAT	
TABLE_HEAD TABLE_VALUES TABLE_END	POP/DOP file syntax for entering many experimental measurements of the same type in a table format. These commands should always be used together and in a sequence.	POP/DOP	
TEMPERATURE_LIMIT	TDB file syntax for defining the default upper and lower temperature limits used by GES for Gibbs energy parameters and functions.	TDB/DAT	
TYPE_DEFINITION	TDB file syntax for coupling phases to an action performed by the TDB module when the TDB command GET_DATA is executed.	TDB/DAT	
VERSION_DATA	TDB file syntax for specifying the version date of the database.	TDB/DAT	
ZERO_VOLUME_SPECIES	TDB file syntax for specifying a list of which species are to be considered as zero volume ones in DICTRA simulations.	TDB/DAT	
File Command	Brief Description	File	Page
PROLOGUE	PROLOGUE command for indicating the beginning of a number of consequent lines of user-defined diagram layout manipulating commands.	EXP	
XSCALE	PROLOGUE command for setting the scaling in word coordinates of the X-axis.	EXP	
YSCALE	PROLOGUE command for setting the scaling in word coordinates of the Y-axis.	EXP	
XTEXT	PROLOGUE command for setting the X-axis text.	EXP	
YTEXT	PROLOGUE command for setting the Y-axis text.	EXP	
XTYPE	PROLOGUE command for setting the X-axis type as linear (default), logarithmic or inverse.	EXP	
YTYPE	PROLOGUE command for setting the Y-axis type as linear (default), logarithmic or inverse.	EXP	
XLENGTH	PROLOGUE command for setting the X-axis length to approximately # centimeters.	EXP	
YLENGTH	PROLOGUE command for setting the Y-axis length to approximately # centimeters.	EXP	
TIC_TYPE	PROLOGUE command for setting the relative length of the tic marks.	EXP	
TITLE	PROLOGUE command for setting the title text string to be printed above the diagram.	EXP	
DIAGRAM_TYPE	PROLOGUE command for setting the diagram type to square (which is default) or triangular.	EXP	
DATASET	DATASET command for indicating the beginning of a number of consequent lines comprising a set of user-defined data.	EXP	

BLOCK / BLOCKEND	DATASET command for defining how the current numeric data block (starts with BLOCK and ends with BLOCKEND) shall be interpreted.	EXP	
DATAPPOINT	<i>Not actually a DATASET command, but as the basic DATAPLOT command, it performs an action at the current point determined by the specified X/Y-coordinates.</i>	EXP	
CLIP	DATASET command for turning the clipping on or off.	EXP	
ATTRIBUTE	DATASET command for specifying where the current XY position is in the character or symbol plotbox.	EXP	
LINETYPE	DATASET command for redefining the current linetype in the plot.	EXP	
DRAWLINE	DATASET command for drawing a line starting at (x1, y1) to (xn, yn) through (x2, y2) → (x(n-1), y(n-1)).	EXP	
CHARSIZE	DATASET command for redefining the default size of the characters in the plot.	EXP	
SYMBOLSIZE	DATASET command for redefining the current symbol size setting.	EXP	
GLOBAISIZE	DATASET command for redefining the default global size of the plot.	EXP	
COLOR	DATASET command for redefining the current color setting.	EXP	
FONT	DATASET command for redefining the default font setting in the POST-processor.	EXP	
STRING	DATASET command for defining a string containing a text and operation codes (e.g., to change the default font settings).	EXP	
TEXT	DATASET command for writing the text or string "string_name" following the keyword TEXT, at the current position,	EXP	
FUNCTION	DATASET command for defining and plotting a user-defined function.	EXP	
PCFUNCTION	DATASET command for appending a user-defined function to the current path.	EXP	
PAINT	DATASET command for painting the area enclosed by the current path in the current pattern.	EXP	
INCLUDE	DATASET command for including a file into the current input stream.	EXP	

Notes:

- All file commands (file syntaxes) can be used only in *certain types of textual files* (such as TDB/DAT, POP/DOP, TCM/LOG, EXP) as stated in the table.
- For *non-textual file types* (such as GES5, POLY3, PAR), the files are restrictively structured in binary codes, and therefore are absolutely impossible be opened/edited outside the Thermo-Calc software (TCC/TCW), DICTRA software and/or various Thermo-Calc programming interfaces.
- Since some chapters that are relevant to the aforementioned textual file commands have been extracted from the TCCR User's Guide and have been moved to other four separate documents, for details of the following file commands, please refer to their corresponding individual documents:
 - ☞ TDB/DAT → *Thermo-Calc Database Guide*;
 - ☞ EXP → *DATAPLOT Language User's Guide and Examples*.

List of Figures

Number	Figure Caption	Page
1-1	Examples of calculations with Thermo-Calc	
1-2	Thermo-Calc related software, databases and interfaces	
1-3	General structure of the Thermo-Calc package	
1-4	Build your own application programs using the Thermo-Calc Engine	
2-1	Modifying Your TCCP Shortcuts (with recommended values)	
2-2	Working with a Linux/UNIX version of TCCP in TeraTerm windows (accessed from Windows NT/2000/XP computers)	
3-1	Definition of driving force	
3-2	Extended CALPHAD method in the Thermo-Calc and DICTRA packages	
3-3	Interaction between the user and modules in the Thermo-Calc package	
3-4	TCW -- The easiest way to interact with Thermo-Calc (<i>an example</i>)	
3-5	Thermo-Calc graphs in 3D viewed by a VRML web browser	
3-6	Thermo-Calc coupled with DICTRA for diffusion simulations	
3-7	Programming and graphical interfaces of the Thermo-Calc package	
3-8	An advanced example of using the TQ interface: MICRESS	
3-9	A simple example of using the TCAPI interface	
3-10	An advanced example of using the TCAPI interface: PrecipiCalc	
3-11	The TC MATLAB Toolbox in the MATLAB® software package	
3-12	The liquidus surface of the A-B-C system calculated and plotted, using the TC-Toolbox in the MATLAB software	
3-13a	A Particle Dissolution During a Heating Sequence, using the TC MATLAB Toolbox.	
3-13b	The code and output of a sample program calculating the molar Gibbs energy curves, using the TC MATLAB Toolbox.	
8-1	The principle for calculating T_o temperature in a multicomponent system	
8-2	Ternary Fe-M-C diagram where M is a substitutional alloying element and C is the fast diffusing carbon	
8-3	Phase diagram for an Fe-1C-0.5Mn-0.5Si-0.2C (wt%) steel. The calculated paraequilibrium state and T_o temperature line are also imposed on the diagram	
8-4	The “Save As” window: Saving the POLY3/GES5 workspaces as an *.POLY3 file	
8-5	The warming message: If saving the POLY3/GES5 workspaces on an existing *.POLY3 file	
8-6	The “Open file” window: Reading the POLY3/GES5 workspaces from an *.POLY3 file	
8-7	Paraequilibrium calculation of isopleths: Formation of para-pearlite in two Fe-based alloys, with equilibrium results appended.	
9-1	The “ThermoCalc - graph” window: Options for graphical outputs	
9-2	The “Save Metafile As” window: Save plotted diagram as an EMF file	
9-3	The “Font” window: Globally change font for plotted diagram	
9-4	The “Color” window: Adjust color for background and other purposes	
9-5	The “Add Text to Diagram” window and its sub-window “Font”: Easily and flexibly edit desired labelling/identifying texts	
9-6	The The “Edit Arrow” window: Edit an arrow appearance	

9-7	The “Set Symbol Number and Size” window: Choose a desired symbol	
9-8	The “Print” window: Choices on connected printers and properties	
9-9	The “Ghostscript” window: Proceeding printing	
9-10	The “Save As” window: Saving the dumped graphical file	
9-11	The “Open file” window: Opening an existing *.EXP file	
9-12	An axample of triangular TC diagram viewed in 3D by Cortona VRML Client	
9-13	An axample of tetrahedron TC diagram viewed in 3D by Cortona VRML Client	
9-14	An axample of triangular TC diagram viewed in 3D by SIM VRMLView	
9-15	An axample of tetrahedron TC diagram viewed in 3D by SIM VRMLView	
10-1	Phase diagram, Gibbs energy curves and phase-fraction plot of the Fe-Cr binary system, automatically calculated by the BIN module (outputs corresponding to the three main options, P, G and F, respectively)	
10-2	Other types of phase diagrams and property diagrams of the Fe-Cr binary system, generated by the POST module, based on calculation results from the main options (P, G and F)	
10-3	Isothermal section (at 1000 K) and projected monovariant lines of the liquidus surface for the Al-Mg-Si ternary system, automatically calculated by the TERN module (outputs corresponding to the two main options, P and M, respectively)	
10-4	Other types of phase diagrams and property diagrams of the Al-Mg-Si ternary system, generated by the POST module, based on calculation results from the main options (P & M)	
10-5	Potential diagram of the interaction between Fe and SO ₂ /O ₂ -bearing gas at 1000 K and 1 bar, automatically calculated and plotted by the POT module	
10-6	Other types of potential diagrams and property plots of the Fe-S-O system, generated by the POST module (directly after the POT-module calculation), or recalculated by the POLY module at different pressures and/or temperatures	
10-7	Pourbaix diagram of Fe (with 0.001 mole Fe in 1 kg of pure water at 25°C and 1 bar; gas excluded in the calculation)	
10-8	The pH-Eh diagrams for pure water at 25°C and 1 bar (<i>in 8a</i>), and for a 3mNaCl-0.001mCO ₂ -0.001mSO ₂ aqueous solution at 150°C and 100 bar (<i>in 8b</i>). Note that gas is included in the calculation	
10-9	Pourbaix diagram of Fe (with 0.001 mole Fe in 1 kg of pure water at 25°C and 1 bar; gas included in the calculation)	
10-10	Pourbaix diagrams of Fe (for varied Fe actively interacted with pure water or aqueous solution at different temperature and pressure conditions; gas included in the calculations)	
10-11	Pourbaix diagrams and property plots of two Fe-based alloys, actively interacted with an aqueous solution (1 kg of water with 3 m NaCl or 1E-5 m S ₂) at constant temperature and pressure conditions	
10-12	Property diagrams for the stable phase assemblages (aqueous solution involving) magnetite+linnaeite+FCC(Ni-Co-Fe) and magnetite+hematite+pyrite+linnaeite+millerite in the heterogeneous interaction between an Fe-based alloy (1E-3 m Fe, 5E-5 m Co and 5E-5 m Ni) and an aqueous solution (1 kg of water, with 1E-6 m S ₂ or open to S ₂) at a pressure of 1 kbar	
10-13	Scheil-Gulliver Model for simulations of alloy solidification process	
10-14	Some SCHEIL-module simulation results for the solidification process of the Al-4Mg-2Si alloy	
10-15	Some comparisons among calculated results on solidification path of an Fe-10Cr-C alloy, using various approaches	
10-16	Solidification path simulation by both TCC (the SCHEIL module) and DICTRA, for an	

	Fe-18%Cr-8%Ni alloy	
10-17	Schematic Illustration of Multi-Staged Steady-State Reactor Concepts	
10-18	The “Open file” window: Reading the REACTOR workspace from an * .RCT file	
10-19	The “Save As” window: Saving the REACTOR workspace as an * .RCT file	
10-20	The warming message: If saving the REACTOR workspace on an existing * .RCT file	
11-1	The “Save As” window: Saving the GES workspace file	
11-2	The “Save As” warning window: Proceeding or canceling the saving	
11-3	The “Open file” window: Opening the GES workspace file	
12-1	The Extended Calphad Approach – The Principles	
12-2	The Extended Calphad Approach – The Scheme	
12-3	The Extended Calphad Approach – DICTRA Kinetic/Mobility Assessments	
12-4	The “Save As” window: Creating a stored work file	
12-5	The “Save As” warning window: Proceeding or canceling the creating	
12-6	The “Open file” window: Setting a stored PARROT workspace file	
12-7	The “Open file” window: Opening an experimental data * .POP file	
14-1	The “Save As” window: Saving all the subsequent commands in an * .LOG file	
14-2	The “Open file” window: Opening an existing * .TCM file	
15-1a	Examples of all DIGLIB symbols (printed from TC Graph Window)	
15-1b	Examples of all DIGLIB symbols (in EMF format)	
15-1c	Examples of all DIGLIB symbols (in PostScript format)	
15-1d	Examples of all DIGLIB symbols (in LaTeX format)	
15-2	Examples of all DIGLIB fonts and line types	
15-3	PostScript outputs of the available PostScript fonts	
15-4a	Current ENCODING PostScript paint patterns (NORMAL video status)	
15-4b	Current ENCODING PostScript paint patterns (INVERSE video status)	
A_A-1	File types and their relations in the Thermo-Calc Classic software	

Notes:

- Since some chapters have been extracted from the TCC User’s Guide and have been moved to other four separate documents, for details of some above-listed diagrams, please refer to their corresponding individual documents:
 - ☞ Chapter index as 3 (e.g. 3.2) → *Thermo-Calc Software System*;
 - ☞ Chapter index as 4 or 6 → *Thermo-Calc Database Guide*;
 - ☞ Chapter index as 15 (e.g. 15.4) → *DATAPLOT Language User’s Guide and Examples*.

List of Tables

Number	Table Title	Page
1-1	Revision history of the Thermo-Calc software/database/interface package	
1-2	Available hardware suitable for the Thermo-Calc Classic software (<i>version R</i>)	
1-3	A benchmark calculation of the Thermo-Calc Classic software (<i>version R</i>)	
3-1	State variables that can be used in the Thermo-Calc package	
3-2	Derived variables predefined for aqueous solution and gaseous mixture phases	
3-3	Various phases handled by the Thermo-Calc package	
3-4	Different levels of the improvement of the Thermo-Calc user interface	
3-5	Modules and application programming interfaces of the Thermo-Calc package	
3-6	Major functionality of the Thermo-Calc software and interfaces	
3-7	Some existing applications of the Thermo-Calc software and databases	
4-1	List of the currently available Thermo-Calc databases	
8-1	Handling of various phase and property diagrams for ternary systems in the TCC software	
8-2	Examples of derived variables for aqueous solution and gaseous mixture phases	
9-1	List of major graphic formats used for dumping a Thermo-Calc graph	
10-1	Four main options in the POURBAIX module	
11-1	List of a wide spectrum of thermodynamic models implemented in the GES module	
13-1	Summary of the performance of POLY commands in the ED-EXP module	
15-1	List of the available PostScript fonts	
15-2a	Current ENCODING vectors used by PostScript fonts 1-28 & 30-42	
15-2b	Current ENCODING vectors used by PostScript font 29	
15-2c	Current ENCODING vectors used by PostScript font 43	

Notes:

- Since some chapters have been extracted from the TCC User's Guide and have been moved to other four separate documents, for details of some above-listed tables, please refer to their corresponding individual documents:
 - ☞ Chapter index as 3 (e.g. 3.2) → *Thermo-Calc Software System*;
 - ☞ Chapter index as 4 or 6 → *Thermo-Calc Database Guide*;
 - ☞ Chapter index as 15 (e.g. 15.4) → *DATAPLOT Language User's Guide and Examples*.