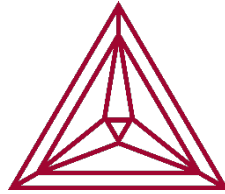




TC-API: SDK Programmer's Guide
Version 2016b



TC-API
7.5-SNAPSHOT

Generated by Doxygen 1.8.9.1

Tue Oct 11 2016 11:14:45

Contents

1	Thermo-Calc c-api	1
1.1	Installed files	1
1.1.1	TC-API libraries.	1
1.1.2	Source folder.	1
1.1.3	Project folders for building the example code	1
1.2	Explicit loading or Dynamic linking	1
1.3	About the source code	2
1.3.1	The dynamic linking examples use the files	2
1.3.2	The explicit loading examples use the files	2
1.4	When starting developing proprietary projects the following code is needed	2
1.4.1	The dynamic linking examples use the files	2
1.4.2	The explicit loading examples use the files	2
2	Class Index	5
2.1	Class List	5
3	File Index	7
3.1	File List	7
4	Class Documentation	9
4.1	<code>_tc_function_library</code> Struct Reference	9
4.1.1	Detailed Description	10
4.1.2	Member Data Documentation	10
4.1.2.1	<code>tc_append_database</code>	10
4.1.2.2	<code>tc_check_license</code>	10
4.1.2.3	<code>tc_component_status</code>	10
4.1.2.4	<code>tc_compute_equilibrium</code>	10
4.1.2.5	<code>tc_create_new_equilibrium</code>	10
4.1.2.6	<code>tc_database</code>	11
4.1.2.7	<code>tc_define_components</code>	11
4.1.2.8	<code>tc_degrees_of_freedom</code>	11
4.1.2.9	<code>tc_deinit</code>	11

4.1.2.10	<code>tc_delete_condition</code>	11
4.1.2.11	<code>tc_delete_symbol</code>	11
4.1.2.12	<code>tc_element</code>	11
4.1.2.13	<code>tc_element_reject</code>	11
4.1.2.14	<code>tc_element_select</code>	11
4.1.2.15	<code>tc_enter_ges5_parameter</code>	11
4.1.2.16	<code>tc_enter_symbol</code>	11
4.1.2.17	<code>tc_error</code>	11
4.1.2.18	<code>tc_ges5</code>	12
4.1.2.19	<code>tc_get_data</code>	12
4.1.2.20	<code>tc_get_derivatives</code>	12
4.1.2.21	<code>tc_get_ges5_parameter</code>	12
4.1.2.22	<code>tc_get_value</code>	12
4.1.2.23	<code>tc_init_root</code>	12
4.1.2.24	<code>tc_init_root3</code>	12
4.1.2.25	<code>tc_list_component</code>	12
4.1.2.26	<code>tc_list_conditions</code>	12
4.1.2.27	<code>tc_list_phase</code>	12
4.1.2.28	<code>tc_list_species</code>	12
4.1.2.29	<code>tc_list_symbols</code>	12
4.1.2.30	<code>tc_nr_of_constituents_in_phase</code>	13
4.1.2.31	<code>tc_open_database</code>	13
4.1.2.32	<code>tc_phase</code>	13
4.1.2.33	<code>tc_phase_all_constituents</code>	13
4.1.2.34	<code>tc_phase_constituents</code>	13
4.1.2.35	<code>tc_phase_reject</code>	13
4.1.2.36	<code>tc_phase_select</code>	13
4.1.2.37	<code>tc_phase_status</code>	13
4.1.2.38	<code>tc_phase_structure</code>	13
4.1.2.39	<code>tc_poly3</code>	13
4.1.2.40	<code>tc_put_sitefractions</code>	13
4.1.2.41	<code>tc_read_poly3_file</code>	13
4.1.2.42	<code>tc_reject_constituent</code>	14
4.1.2.43	<code>tc_reset_error</code>	14
4.1.2.44	<code>tc_restore_constituent</code>	14
4.1.2.45	<code>tc_save_poly3_file</code>	14
4.1.2.46	<code>tc_select_equilibrium</code>	14
4.1.2.47	<code>tc_set_component_status</code>	14
4.1.2.48	<code>tc_set_condition</code>	14
4.1.2.49	<code>tc_set_license_code</code>	14

4.1.2.50	tc_set_minimization_option	14
4.1.2.51	tc_set_phase_addition	14
4.1.2.52	tc_set_phase_status	14
4.1.2.53	tc_set_start_value	14
4.1.2.54	tc_species_status	15
4.1.2.55	tc_version	15
4.2	tc_components_strings Struct Reference	15
4.2.1	Detailed Description	15
4.2.2	Member Data Documentation	15
4.2.2.1	component	15
4.3	tc_conditions_as_arrays_of_strings Struct Reference	15
4.3.1	Detailed Description	15
4.3.2	Member Data Documentation	16
4.3.2.1	condition	16
4.4	tc_constituents_strings Struct Reference	16
4.4.1	Detailed Description	16
4.4.2	Member Data Documentation	16
4.4.2.1	constituent	16
4.5	tc_databases_strings Struct Reference	16
4.5.1	Detailed Description	16
4.5.2	Member Data Documentation	16
4.5.2.1	database	16
4.6	tc_elements_strings Struct Reference	17
4.6.1	Detailed Description	17
4.6.2	Member Data Documentation	17
4.6.2.1	element	17
4.7	tc_phases_strings Struct Reference	17
4.7.1	Detailed Description	17
4.7.2	Member Data Documentation	17
4.7.2.1	phase	17
4.8	tc_reference_strings Struct Reference	17
4.8.1	Detailed Description	18
4.8.2	Member Data Documentation	18
4.8.2.1	reference	18
4.9	tc_species_strings Struct Reference	18
4.9.1	Detailed Description	18
4.9.2	Member Data Documentation	18
4.9.2.1	specie	18

5 File Documentation

19

5.1	example1.c File Reference	19
5.2	example2.c File Reference	19
5.2.1	Function Documentation	19
5.2.1.1	main	19
5.3	example3.c File Reference	19
5.3.1	Typedef Documentation	20
5.3.1.1	ivect	20
5.3.1.2	rvect	20
5.3.1.3	str8	20
5.3.1.4	strvect	20
5.3.2	Function Documentation	20
5.3.2.1	main	20
5.4	libtc.c File Reference	20
5.4.1	Function Documentation	20
5.4.1.1	importFunctions	20
5.4.1.2	tloadfunc	21
5.5	libtc.h File Reference	21
5.5.1	Typedef Documentation	22
5.5.1.1	BoolFuncIntPStringInt	22
5.5.1.2	BoolFuncStringStringStrLen	22
5.5.1.3	FloatFuncString	22
5.5.1.4	IntFuncNoParams	22
5.5.1.5	IntFuncString	22
5.5.1.6	IntFuncStringInt	22
5.5.1.7	IntFuncStringIntIntP	22
5.5.1.8	IntFuncStringIntPStringIntFloatP	22
5.5.1.9	IntFuncStringIntPStringStrLenFloatP	22
5.5.1.10	IntFuncStringString	22
5.5.1.11	StringFuncString	22
5.5.1.12	tc_function_library	23
5.5.1.13	VoidFuncInt	23
5.5.1.14	VoidFuncIntPIntPIntPIntP	23
5.5.1.15	VoidFuncNoParams	23
5.5.1.16	VoidFuncString	23
5.5.1.17	VoidFuncStringFloat	23
5.5.1.18	VoidFuncStringFloatP	23
5.5.1.19	VoidFuncStringFloatPFloatP	23
5.5.1.20	VoidFuncStringInt	23
5.5.1.21	VoidFuncStringIntInt	23
5.5.1.22	VoidFuncStringIntString	23

5.5.1.23	VoidFuncStringString	23
5.5.1.24	VoidFuncStringStringFloat	23
5.5.1.25	VoidFuncStringStringInt	24
5.5.1.26	VoidFuncStringStringIntIntFloatString	24
5.5.2	Function Documentation	24
5.5.2.1	importFunctions	24
5.6	ReadMe.txt File Reference	24
5.7	ReadMe.txt File Reference	24
5.8	tc_data_defs.h File Reference	24
5.8.1	Macro Definition Documentation	25
5.8.1.1	BOOL_FUNC_WIN	25
5.8.1.2	DllExport	25
5.8.1.3	false	25
5.8.1.4	FLOAT_FUNC_WIN	26
5.8.1.5	INTEGER_FUNC	26
5.8.1.6	INTEGER_FUNC_GNU	26
5.8.1.7	INTEGER_FUNC_WIN	26
5.8.1.8	TC_EPS	26
5.8.1.9	TC_MAX_NR_OF_AXES	26
5.8.1.10	TC_MAX_NR_OF_CONST_PER_SUBLATTICE	26
5.8.1.11	TC_MAX_NR_OF_CONST_PER_SUBLATTICE_IN_IDEAL_GAS	26
5.8.1.12	TC_MAX_NR_OF_CONSTITUENTS	26
5.8.1.13	TC_MAX_NR_OF_DATABASES	26
5.8.1.14	TC_MAX_NR_OF_ELEMENTS	26
5.8.1.15	TC_MAX_NR_OF_PHASES	26
5.8.1.16	TC_MAX_NR_OF_SPECIES	27
5.8.1.17	TC_MAX_NR_OF_SUBLATTICES	27
5.8.1.18	TC_NWSE	27
5.8.1.19	TC_NWSG	27
5.8.1.20	TC_STRLEN_COMPONENTS	27
5.8.1.21	TC_STRLEN_CONSTITUENTS	27
5.8.1.22	TC_STRLEN_DATABASE	27
5.8.1.23	TC_STRLEN_ELEMENTS	27
5.8.1.24	TC_STRLEN_MAX	27
5.8.1.25	TC_STRLEN_PATH_MAX	27
5.8.1.26	TC_STRLEN_PHASES	27
5.8.1.27	TC_STRLEN_REFERENCE	27
5.8.1.28	TC_STRLEN_SPECIES	28
5.8.1.29	TC_STRLEN_STOICHIOMETRY	28
5.8.1.30	TC_VARS	28

5.8.1.31	TCFuncExport	28
5.8.1.32	TCHANDLE	28
5.8.1.33	true	28
5.8.1.34	VOID_FUNC_WIN	28
5.8.2	Typedef Documentation	28
5.8.2.1	pointer	28
5.8.2.2	TC_BOOL	28
5.8.2.3	tc_components_strings	28
5.8.2.4	tc_conditions_as_arrays_of_strings	28
5.8.2.5	tc_constituents_strings	28
5.8.2.6	tc_databases_strings	29
5.8.2.7	tc_elements_strings	29
5.8.2.8	TC_FLOAT	29
5.8.2.9	TC_IARR	29
5.8.2.10	TC_INT	29
5.8.2.11	TC_LABEL_STRING	29
5.8.2.12	tc_phases_strings	29
5.8.2.13	tc_reference_strings	29
5.8.2.14	tc_species_strings	29
5.8.2.15	TC_STRING	29
5.8.2.16	TC_STRING_LENGTH	29
5.9	tcapi.h File Reference	29
5.9.1	Function Documentation	31
5.9.1.1	tc_append_database	31
5.9.1.2	tc_check_license	31
5.9.1.3	tc_component_status	31
5.9.1.4	tc_compute_equilibrium	31
5.9.1.5	tc_create_new_equilibrium	31
5.9.1.6	tc_database	31
5.9.1.7	tc_define_components	31
5.9.1.8	tc_degrees_of_freedom	31
5.9.1.9	tc_deinit	31
5.9.1.10	tc_delete_condition	31
5.9.1.11	tc_delete_symbol	32
5.9.1.12	tc_element	32
5.9.1.13	tc_element_reject	32
5.9.1.14	tc_element_select	32
5.9.1.15	tc_enter_ges5_parameter	32
5.9.1.16	tc_enter_symbol	32
5.9.1.17	tc_error	32

5.9.1.18	tc_ges5	32
5.9.1.19	tc_get_data	32
5.9.1.20	tc_get_derivatives	32
5.9.1.21	tc_get_ges5_parameter	32
5.9.1.22	tc_get_value	33
5.9.1.23	tc_init_root	33
5.9.1.24	tc_init_root3	33
5.9.1.25	tc_list_component	33
5.9.1.26	tc_list_conditions	33
5.9.1.27	tc_list_phase	33
5.9.1.28	tc_list_species	33
5.9.1.29	tc_list_symbols	33
5.9.1.30	tc_nr_of_constituents_in_phase	33
5.9.1.31	tc_open_database	33
5.9.1.32	tc_phase	33
5.9.1.33	tc_phase_all_constituents	34
5.9.1.34	tc_phase_constituents	34
5.9.1.35	tc_phase_reject	34
5.9.1.36	tc_phase_select	34
5.9.1.37	tc_phase_status	34
5.9.1.38	tc_phase_structure	34
5.9.1.39	tc_poly3	34
5.9.1.40	tc_put_sitefractions	34
5.9.1.41	tc_read_poly3_file	34
5.9.1.42	tc_reject_constituent	34
5.9.1.43	tc_reset_error	34
5.9.1.44	tc_restore_constituent	35
5.9.1.45	tc_save_poly3_file	35
5.9.1.46	tc_select_equilibrium	35
5.9.1.47	tc_set_component_status	35
5.9.1.48	tc_set_condition	35
5.9.1.49	tc_set_license_code	35
5.9.1.50	tc_set_minimization_option	35
5.9.1.51	tc_set_phase_addition	35
5.9.1.52	tc_set_phase_status	35
5.9.1.53	tc_set_start_value	35
5.9.1.54	tc_species_status	35
5.9.1.55	tc_version	35
5.10	tcExamples.c File Reference	36
5.10.1	Typedef Documentation	36

5.10.1.1	ivect	36
5.10.1.2	rvect	36
5.10.1.3	str8	36
5.10.1.4	strvect	36
5.10.2	Function Documentation	36
5.10.2.1	example1	36
5.10.2.2	example2	36
5.10.2.3	example3	37
5.11	tcExamples.h File Reference	37
5.11.1	Function Documentation	37
5.11.1.1	example1	37
5.11.1.2	example2	37
5.11.1.3	example3	37
5.12	tcMain.c File Reference	37
5.12.1	Macro Definition Documentation	38
5.12.1.1	TC_API_LIBRARY_NAME	38
5.12.2	Function Documentation	38
5.12.2.1	importLibThermoCalc	38
5.12.2.2	loadTCLibraryInCurrentDir	38
5.12.2.3	main	38
5.13	tcutils.c File Reference	38
5.13.1	Function Documentation	38
5.13.1.1	getTempEnvironmentPath	38
5.13.1.2	getThermoCalcEnvironmentPath	38
5.14	tcutils.h File Reference	38
5.14.1	Macro Definition Documentation	39
5.14.1.1	getCurrentWorkingDir	39
5.14.1.2	SLASH	39
5.14.1.3	TC61_HOME	39
5.14.1.4	TCPATH	39
5.14.1.5	TEMP	39
5.14.2	Function Documentation	39
5.14.2.1	getTempEnvironmentPath	39
5.14.2.2	getThermoCalcEnvironmentPath	39
Index		41

Chapter 1

Thermo-Calc c-api

The main part of this manual is a technical description of the TC-API. To find details on the Thermodynamic applications of each library function see the section on [tcapi.h](#).

1.1 Installed files

In the distribution of Thermo-Calc c-api, the following folders and files can be found.

1.1.1 TC-API libraries.

These could be .lib, .dll, .so -files depending on your installation.
E.g. on a 64 bit Windows system you will find:
tcapi-win-x64-7.5-SNAPSHOT.dll and tcapi-win-x64-7.5-SNAPSHOT.lib

1.1.2 Source folder.

This is the c code for running the the different examples. Some of this code can be reused in user-written projects.

1.1.3 Project folders for building the example code

Linux:

-> Linux/Linux-Dynamic-Linking

-> Linux/Linux-Explicit-Loading

Windows:

-> Windows-Mingw32-Explicit-Loading

-> Windows-Studio-Project-Dynamic-Linking

-> Windows-Studio-Project-Explicit-Loading

1.2 Explicit loading or Dynamic linking

When using Dynamic Linking, the libraries (.so or .lib) and the header-files of the tcapi are needed when the program is being built.

When using Explicit Loading no libraries are needed for the build. They are loaded at runtime (.so or .dll).

1.3 About the source code

1.3.1 The dynamic linking examples use the files

Simple thermodynamic calculations to demonstrate the use of this api.

[example1.c](#)

[example2.c](#)

[example3.c](#)

Utility functions for finding the correct environment variables

[tcutils.c](#)

[tcutils.h](#)

Declarations and documentation on all TC-API functions

[tcapi.h](#)

Thermo-Calc proprietary declarations and definitions. DO NOT EDIT.

[tc_data_defs.h](#)

1.3.2 The explicit loading examples use the files

Utilities to simplify working with explicitly loaded dll.

[libtc.c](#)

[libtc.h](#)

Simple thermodynamic calculations to demonstrate the use of this api.

[tcExamples.c](#)

[tcExamples.h](#)

Main program to exemplify how this api can be used

[tcMain.c](#)

Utility functions for finding the correct environment variables

[tcutils.c](#)

[tcutils.h](#)

Thermo-Calc proprietary declarations and definitions. DO NOT EDIT.

[tc_data_defs.h](#)

1.4 When starting developing proprietary projects the following code is needed

1.4.1 The dynamic linking examples use the files

[tcapi.h](#)

[tc_data_defs.h](#)

1.4.2 The explicit loading examples use the files

[libtc.c](#)

[libtc.h](#)

[tc_data_defs.h](#)

([tcMain.c](#) can give an idea of how the above files can be used)

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

_tc_function_library	9
tc_components_strings	15
tc_conditions_as_arrays_of_strings	15
tc_constituents_strings	16
tc_databases_strings	16
tc_elements_strings	17
tc_phases_strings	17
tc_reference_strings	17
tc_species_strings	18

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

example1.c	19
example2.c	19
example3.c	19
libtc.c	20
libtc.h	21
tc_data_defs.h	24
tcapi.h	29
tcExamples.c	36
tcExamples.h	37
tcMain.c	37
tcutils.c	38
tcutils.h	38

Chapter 4

Class Documentation

4.1 `_tc_function_library` Struct Reference

```
#include <libtc.h>
```

Public Attributes

- `VoidFuncString tc_append_database`
- `BoolFuncStringStringStrLen tc_check_license`
- `StringFuncString tc_component_status`
- `VoidFuncNoParams tc_compute_equilibrium`
- `VoidFuncInt tc_create_new_equilibrium`
- `IntFuncStringInt tc_database`
- `VoidFuncStringIntInt tc_define_components`
- `IntFuncNoParams tc_degrees_of_freedom`
- `VoidFuncNoParams tc_deinit`
- `VoidFuncString tc_delete_condition`
- `VoidFuncString tc_delete_symbol`
- `IntFuncStringInt tc_element`
- `VoidFuncString tc_element_reject`
- `VoidFuncString tc_element_select`
- `VoidFuncStringString tc_enter_ges5_parameter`
- `VoidFuncStringStringIntIntFloatString tc_enter_symbol`
- `BoolFuncIntPStringInt tc_error`
- `VoidFuncString tc_ges5`
- `VoidFuncNoParams tc_get_data`
- `VoidFuncStringFloatPFloatP tc_get_derivatives`
- `VoidFuncStringStringInt tc_get_ges5_parameter`
- `FloatFuncString tc_get_value`
- `IntFuncNoParams tc_init_root`
- `IntFuncStringString tc_init_root3`
- `VoidFuncStringInt tc_list_component`
- `VoidFuncStringInt tc_list_conditions`
- `VoidFuncStringInt tc_list_phase`
- `VoidFuncStringInt tc_list_species`
- `IntFuncStringIntIntP tc_list_symbols`
- `IntFuncString tc_nr_of_constituents_in_phase`
- `VoidFuncString tc_open_database`
- `IntFuncStringInt tc_phase`

- [IntFuncStringIntPStringIntFloatP tc_phase_all_constituents](#)
- [IntFuncStringIntPStringIntFloatP tc_phase_constituents](#)
- [VoidFuncString tc_phase_reject](#)
- [VoidFuncString tc_phase_select](#)
- [StringFuncString tc_phase_status](#)
- [IntFuncStringIntPStringStrLenFloatP tc_phase_structure](#)
- [VoidFuncString tc_poly3](#)
- [VoidFuncStringFloatP tc_put_sitefractions](#)
- [VoidFuncString tc_read_poly3_file](#)
- [VoidFuncStringIntString tc_reject_constituent](#)
- [VoidFuncNoParams tc_reset_error](#)
- [VoidFuncStringIntString tc_restore_constituent](#)
- [VoidFuncString tc_save_poly3_file](#)
- [VoidFuncInt tc_select_equilibrium](#)
- [VoidFuncStringString tc_set_component_status](#)
- [VoidFuncStringFloat tc_set_condition](#)
- [VoidFuncInt tc_set_license_code](#)
- [VoidFuncIntPIntPIntPIntP tc_set_minimization_option](#)
- [VoidFuncStringFloat tc_set_phase_addition](#)
- [VoidFuncStringStringFloat tc_set_phase_status](#)
- [VoidFuncStringFloat tc_set_start_value](#)
- [StringFuncString tc_species_status](#)
- [VoidFuncStringInt tc_version](#)

4.1.1 Detailed Description

Definition at line 67 of file libtc.h.

4.1.2 Member Data Documentation

4.1.2.1 VoidFuncString _tc_function_library::tc_append_database

Definition at line 68 of file libtc.h.

4.1.2.2 BoolFuncStringStringStrLen _tc_function_library::tc_check_license

Definition at line 69 of file libtc.h.

4.1.2.3 StringFuncString _tc_function_library::tc_component_status

Definition at line 70 of file libtc.h.

4.1.2.4 VoidFuncNoParams _tc_function_library::tc_compute_equilibrium

Definition at line 71 of file libtc.h.

4.1.2.5 VoidFuncInt _tc_function_library::tc_create_new_equilibrium

Definition at line 72 of file libtc.h.

4.1.2.6 `IntFuncStringInt _tc_function_library::tc_database`

Definition at line 73 of file `libtc.h`.

4.1.2.7 `VoidFuncStringIntInt _tc_function_library::tc_define_components`

Definition at line 74 of file `libtc.h`.

4.1.2.8 `IntFuncNoParams _tc_function_library::tc_degrees_of_freedom`

Definition at line 75 of file `libtc.h`.

4.1.2.9 `VoidFuncNoParams _tc_function_library::tc_deinit`

Definition at line 76 of file `libtc.h`.

4.1.2.10 `VoidFuncString _tc_function_library::tc_delete_condition`

Definition at line 77 of file `libtc.h`.

4.1.2.11 `VoidFuncString _tc_function_library::tc_delete_symbol`

Definition at line 78 of file `libtc.h`.

4.1.2.12 `IntFuncStringInt _tc_function_library::tc_element`

Definition at line 79 of file `libtc.h`.

4.1.2.13 `VoidFuncString _tc_function_library::tc_element_reject`

Definition at line 80 of file `libtc.h`.

4.1.2.14 `VoidFuncString _tc_function_library::tc_element_select`

Definition at line 81 of file `libtc.h`.

4.1.2.15 `VoidFuncStringString _tc_function_library::tc_enter_ges5_parameter`

Definition at line 82 of file `libtc.h`.

4.1.2.16 `VoidFuncStringStringIntIntFloatString _tc_function_library::tc_enter_symbol`

Definition at line 83 of file `libtc.h`.

4.1.2.17 `BoolFuncIntPStringInt _tc_function_library::tc_error`

Definition at line 84 of file `libtc.h`.

4.1.2.18 VoidFuncString _tc_function_library::tc_ges5

Definition at line 85 of file libtc.h.

4.1.2.19 VoidFuncNoParams _tc_function_library::tc_get_data

Definition at line 86 of file libtc.h.

4.1.2.20 VoidFuncStringFloatPFloatP _tc_function_library::tc_get_derivatives

Definition at line 87 of file libtc.h.

4.1.2.21 VoidFuncStringStringInt _tc_function_library::tc_get_ges5_parameter

Definition at line 88 of file libtc.h.

4.1.2.22 FloatFuncString _tc_function_library::tc_get_value

Definition at line 89 of file libtc.h.

4.1.2.23 IntFuncNoParams _tc_function_library::tc_init_root

Definition at line 90 of file libtc.h.

4.1.2.24 IntFuncStringString _tc_function_library::tc_init_root3

Definition at line 91 of file libtc.h.

4.1.2.25 VoidFuncStringInt _tc_function_library::tc_list_component

Definition at line 92 of file libtc.h.

4.1.2.26 VoidFuncStringInt _tc_function_library::tc_list_conditions

Definition at line 93 of file libtc.h.

4.1.2.27 VoidFuncStringInt _tc_function_library::tc_list_phase

Definition at line 94 of file libtc.h.

4.1.2.28 VoidFuncStringInt _tc_function_library::tc_list_species

Definition at line 95 of file libtc.h.

4.1.2.29 IntFuncStringIntIntP _tc_function_library::tc_list_symbols

Definition at line 96 of file libtc.h.

4.1.2.30 `IntFuncString _tc_function_library::tc_nr_of_constituents_in_phase`

Definition at line 97 of file `libtc.h`.

4.1.2.31 `VoidFuncString _tc_function_library::tc_open_database`

Definition at line 98 of file `libtc.h`.

4.1.2.32 `IntFuncStringInt _tc_function_library::tc_phase`

Definition at line 99 of file `libtc.h`.

4.1.2.33 `IntFuncStringIntPStringIntFloatP _tc_function_library::tc_phase_all_constituents`

Definition at line 100 of file `libtc.h`.

4.1.2.34 `IntFuncStringIntPStringIntFloatP _tc_function_library::tc_phase_constituents`

Definition at line 101 of file `libtc.h`.

4.1.2.35 `VoidFuncString _tc_function_library::tc_phase_reject`

Definition at line 102 of file `libtc.h`.

4.1.2.36 `VoidFuncString _tc_function_library::tc_phase_select`

Definition at line 103 of file `libtc.h`.

4.1.2.37 `StringFuncString _tc_function_library::tc_phase_status`

Definition at line 104 of file `libtc.h`.

4.1.2.38 `IntFuncStringIntPStringStrLenFloatP _tc_function_library::tc_phase_structure`

Definition at line 105 of file `libtc.h`.

4.1.2.39 `VoidFuncString _tc_function_library::tc_poly3`

Definition at line 106 of file `libtc.h`.

4.1.2.40 `VoidFuncStringFloatP _tc_function_library::tc_put_sitefractions`

Definition at line 107 of file `libtc.h`.

4.1.2.41 `VoidFuncString _tc_function_library::tc_read_poly3_file`

Definition at line 108 of file `libtc.h`.

4.1.2.42 VoidFuncStringIntString _tc_function_library::tc_reject_constituent

Definition at line 109 of file libtc.h.

4.1.2.43 VoidFuncNoParams _tc_function_library::tc_reset_error

Definition at line 110 of file libtc.h.

4.1.2.44 VoidFuncStringIntString _tc_function_library::tc_restore_constituent

Definition at line 111 of file libtc.h.

4.1.2.45 VoidFuncString _tc_function_library::tc_save_poly3_file

Definition at line 112 of file libtc.h.

4.1.2.46 VoidFuncInt _tc_function_library::tc_select_equilibrium

Definition at line 113 of file libtc.h.

4.1.2.47 VoidFuncStringString _tc_function_library::tc_set_component_status

Definition at line 114 of file libtc.h.

4.1.2.48 VoidFuncStringFloat _tc_function_library::tc_set_condition

Definition at line 115 of file libtc.h.

4.1.2.49 VoidFuncInt _tc_function_library::tc_set_license_code

Definition at line 116 of file libtc.h.

4.1.2.50 VoidFuncIntPIntPIntPIntP _tc_function_library::tc_set_minimization_option

Definition at line 117 of file libtc.h.

4.1.2.51 VoidFuncStringFloat _tc_function_library::tc_set_phase_addition

Definition at line 118 of file libtc.h.

4.1.2.52 VoidFuncStringStringFloat _tc_function_library::tc_set_phase_status

Definition at line 119 of file libtc.h.

4.1.2.53 VoidFuncStringFloat _tc_function_library::tc_set_start_value

Definition at line 120 of file libtc.h.

4.1.2.54 StringFuncString_tc_function_library::tc_species_status

Definition at line 121 of file libtc.h.

4.1.2.55 VoidFuncStringInt_tc_function_library::tc_version

Definition at line 122 of file libtc.h.

The documentation for this struct was generated from the following file:

- [libtc.h](#)

4.2 tc_components_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [component](#) [TC_STRLEN_COMPONENTS]

4.2.1 Detailed Description

Definition at line 111 of file tc_data_defs.h.

4.2.2 Member Data Documentation

4.2.2.1 char tc_components_strings::component[TC_STRLEN_COMPONENTS]

Definition at line 113 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.3 tc_conditions_as_arrays_of_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [condition](#) [TC_STRLEN_MAX]

4.3.1 Detailed Description

Definition at line 98 of file tc_data_defs.h.

4.3.2 Member Data Documentation

4.3.2.1 char tc_conditions_as_arrays_of_strings::condition[TC_STRLEN_MAX]

Definition at line 100 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.4 tc_constituents_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [constituent](#) [TC_STRLEN_CONSTITUENTS]

4.4.1 Detailed Description

Definition at line 130 of file tc_data_defs.h.

4.4.2 Member Data Documentation

4.4.2.1 char tc_constituents_strings::constituent[TC_STRLEN_CONSTITUENTS]

Definition at line 132 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.5 tc_databases_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [database](#) [TC_STRLEN_DATABASE]

4.5.1 Detailed Description

Definition at line 136 of file tc_data_defs.h.

4.5.2 Member Data Documentation

4.5.2.1 char tc_databases_strings::database[TC_STRLEN_DATABASE]

Definition at line 138 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.6 tc_elements_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [element](#) [TC_STRLEN_ELEMENTS]

4.6.1 Detailed Description

Definition at line 105 of file tc_data_defs.h.

4.6.2 Member Data Documentation

4.6.2.1 char tc_elements_strings::element[TC_STRLEN_ELEMENTS]

Definition at line 107 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.7 tc_phases_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [phase](#) [TC_STRLEN_PHASES]

4.7.1 Detailed Description

Definition at line 124 of file tc_data_defs.h.

4.7.2 Member Data Documentation

4.7.2.1 char tc_phases_strings::phase[TC_STRLEN_PHASES]

Definition at line 126 of file tc_data_defs.h.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.8 tc_reference_strings Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [reference](#) [TC_STRLEN_REFERENCE]

4.8.1 Detailed Description

Definition at line 142 of file `tc_data_defs.h`.

4.8.2 Member Data Documentation

4.8.2.1 char `tc_reference_strings::reference`[TC_STRLEN_REFERENCE]

Definition at line 144 of file `tc_data_defs.h`.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

4.9 `tc_species_strings` Struct Reference

```
#include <tc_data_defs.h>
```

Public Attributes

- char [specie](#) [TC_STRLEN_SPECIES]

4.9.1 Detailed Description

Definition at line 117 of file `tc_data_defs.h`.

4.9.2 Member Data Documentation

4.9.2.1 char `tc_species_strings::specie`[TC_STRLEN_SPECIES]

Definition at line 119 of file `tc_data_defs.h`.

The documentation for this struct was generated from the following file:

- [tc_data_defs.h](#)

Chapter 5

File Documentation

5.1 example1.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "tcapi.h"
#include "tcutils.h"
Include dependency graph for example1.c:
```

5.2 example2.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "tcapi.h"
#include "tcutils.h"
Include dependency graph for example2.c:
```

Functions

- int [main](#) (int argc, char **argv)

5.2.1 Function Documentation

5.2.1.1 int main (int argc, char ** argv)

Definition at line 14 of file example2.c.

5.3 example3.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "tcapi.h"
#include "tcutils.h"
Include dependency graph for example3.c:
```

Typedefs

- typedef char [str8](#)[8]
- typedef [str8](#) [strvect](#)[100]
- typedef [TC_INT](#) [ivect](#)[50]
- typedef [TC_FLOAT](#) [rvect](#)[50]

Functions

- int [main](#) (int argc, char **argv)

5.3.1 Typedef Documentation

5.3.1.1 typedef [TC_INT](#) [ivect](#)[50]

Definition at line 16 of file [example3.c](#).

5.3.1.2 typedef [TC_FLOAT](#) [rvect](#)[50]

Definition at line 17 of file [example3.c](#).

5.3.1.3 typedef char [str8](#)[8]

Definition at line 14 of file [example3.c](#).

5.3.1.4 typedef [str8](#) [strvect](#)[100]

Definition at line 15 of file [example3.c](#).

5.3.2 Function Documentation

5.3.2.1 int [main](#) (int *argc*, char ** *argv*)

Definition at line 19 of file [example3.c](#).

5.4 libtc.c File Reference

```
#include "libtc.h"  
Include dependency graph for libtc.c:
```

Functions

- void * [tloadfunc](#) ([TCHANDLE](#) *tcHandle*, char **function_name*)
- int [importFunctions](#) ([TCHANDLE](#) *tcHandle*, [tc_function_library](#) **tc*, char **message*)

5.4.1 Function Documentation

5.4.1.1 int [importFunctions](#) ([TCHANDLE](#) *tcHandle*, [tc_function_library](#) * *tc*, char * *message*)

Definition at line 23 of file [libtc.c](#).

5.4.1.2 void* tloadfunc (TCHANDLE tcHandle, char * function_name)

Definition at line 11 of file libtc.c.

5.5 libtc.h File Reference

```
#include "tc_data_defs.h"
#include <dlfcn.h>
#include <errno.h>
#include <string.h>
```

Include dependency graph for libtc.h: This graph shows which files directly or indirectly include this file:

Classes

- struct [_tc_function_library](#)

Typedefs

- typedef void(* [VoidFuncNoParams](#)) ()
- typedef void(* [VoidFuncString](#)) (TC_STRING)
- typedef void(* [VoidFuncInt](#)) (TC_INT)
- typedef void(* [VoidFuncStringString](#)) (TC_STRING, TC_STRING)
- typedef void(* [VoidFuncStringFloat](#)) (TC_STRING, TC_FLOAT)
- typedef void(* [VoidFuncStringFloatP](#)) (TC_STRING, TC_FLOAT *)
- typedef void(* [VoidFuncStringInt](#)) (TC_STRING, TC_INT)
- typedef void(* [VoidFuncStringStringInt](#)) (TC_STRING, TC_STRING, TC_INT)
- typedef void(* [VoidFuncStringStringFloat](#)) (TC_STRING, TC_STRING, TC_FLOAT)
- typedef void(* [VoidFuncStringIntInt](#)) (TC_STRING, TC_INT, TC_INT)
- typedef void(* [VoidFuncStringFloatPFloatP](#)) (TC_STRING, TC_FLOAT *, TC_FLOAT *)
- typedef void(* [VoidFuncStringIntString](#)) (TC_STRING, TC_INT, TC_STRING)
- typedef void(* [VoidFuncIntPIntPIntP](#)) (TC_INT *, TC_INT *, TC_INT *, TC_INT *)
- typedef void(* [VoidFuncStringStringIntIntFloatString](#)) (TC_STRING, TC_STRING, TC_INT, TC_INT, TC_FLOAT, TC_STRING)
- typedef [TC_INT](#)(* [IntFuncNoParams](#)) ()
- typedef [TC_INT](#)(* [IntFuncString](#)) (TC_STRING)
- typedef [TC_INT](#)(* [IntFuncStringString](#)) (TC_STRING, TC_STRING)
- typedef [TC_INT](#)(* [IntFuncStringInt](#)) (TC_STRING, TC_INT)
- typedef [TC_INT](#)(* [IntFuncStringIntIntP](#)) (TC_STRING, TC_INT, TC_INT *)
- typedef [TC_INT](#)(* [IntFuncStringIntPStringStrLenFloatP](#)) (TC_STRING, TC_INT *, TC_STRING, TC_STRING_LENGTH, TC_FLOAT *)
- typedef [TC_INT](#)(* [IntFuncStringIntPStringIntFloatP](#)) (TC_STRING, TC_INT *, TC_STRING, TC_INT, TC_FLOAT *)
- typedef [TC_FLOAT](#)(* [FloatFuncString](#)) (TC_STRING)
- typedef [TC_STRING](#)(* [StringFuncString](#)) (TC_STRING)
- typedef [TC_BOOL](#)(* [BoolFuncIntPStringInt](#)) (TC_INT *, TC_STRING, TC_INT)
- typedef [TC_BOOL](#)(* [BoolFuncStringStringStrLen](#)) (TC_STRING, TC_STRING, TC_STRING_LENGTH)
- typedef struct [_tc_function_library](#) [tc_function_library](#)

Functions

- int [importFunctions](#) (TCHANDLE tcHandle, [tc_function_library](#) *tc, char *message)

5.5.1 Typedef Documentation

5.5.1.1 typedef TC_BOOL(* BoolFuncIntPStringInt) (TC_INT *, TC_STRING, TC_INT)

Definition at line 55 of file libtc.h.

5.5.1.2 typedef TC_BOOL(* BoolFuncStringStringStrLen) (TC_STRING, TC_STRING, TC_STRING_LENGTH)

Definition at line 56 of file libtc.h.

5.5.1.3 typedef TC_FLOAT(* FloatFuncString) (TC_STRING)

Definition at line 51 of file libtc.h.

5.5.1.4 typedef TC_INT(* IntFuncNoParams) ()

Definition at line 43 of file libtc.h.

5.5.1.5 typedef TC_INT(* IntFuncString) (TC_STRING)

Definition at line 44 of file libtc.h.

5.5.1.6 typedef TC_INT(* IntFuncStringInt) (TC_STRING, TC_INT)

Definition at line 46 of file libtc.h.

5.5.1.7 typedef TC_INT(* IntFuncStringIntIntP) (TC_STRING, TC_INT, TC_INT *)

Definition at line 47 of file libtc.h.

5.5.1.8 typedef TC_INT(* IntFuncStringIntPStringIntFloatP) (TC_STRING, TC_INT *, TC_STRING, TC_INT, TC_FLOAT *)

Definition at line 49 of file libtc.h.

5.5.1.9 typedef TC_INT(* IntFuncStringIntPStringStrLenFloatP) (TC_STRING, TC_INT *, TC_STRING, TC_STRING_LENGTH, TC_FLOAT *)

Definition at line 48 of file libtc.h.

5.5.1.10 typedef TC_INT(* IntFuncStringString) (TC_STRING, TC_STRING)

Definition at line 45 of file libtc.h.

5.5.1.11 typedef TC_STRING(* StringFuncString) (TC_STRING)

Definition at line 53 of file libtc.h.

5.5.1.12 `typedef struct _tc_function_library tc_function_library`

5.5.1.13 `typedef void(* VoidFuncInt) (TC_INT)`

Definition at line 27 of file libtc.h.

5.5.1.14 `typedef void(* VoidFuncIntPlntPlntPlntP) (TC_INT *, TC_INT *, TC_INT *, TC_INT *)`

Definition at line 40 of file libtc.h.

5.5.1.15 `typedef void(* VoidFuncNoParams) ()`

Definition at line 24 of file libtc.h.

5.5.1.16 `typedef void(* VoidFuncString) (TC_STRING)`

Definition at line 26 of file libtc.h.

5.5.1.17 `typedef void(* VoidFuncStringFloat) (TC_STRING, TC_FLOAT)`

Definition at line 30 of file libtc.h.

5.5.1.18 `typedef void(* VoidFuncStringFloatP) (TC_STRING, TC_FLOAT *)`

Definition at line 31 of file libtc.h.

5.5.1.19 `typedef void(* VoidFuncStringFloatPFloatP) (TC_STRING, TC_FLOAT *, TC_FLOAT *)`

Definition at line 37 of file libtc.h.

5.5.1.20 `typedef void(* VoidFuncStringInt) (TC_STRING, TC_INT)`

Definition at line 32 of file libtc.h.

5.5.1.21 `typedef void(* VoidFuncStringIntInt) (TC_STRING, TC_INT, TC_INT)`

Definition at line 36 of file libtc.h.

5.5.1.22 `typedef void(* VoidFuncStringIntString) (TC_STRING, TC_INT, TC_STRING)`

Definition at line 38 of file libtc.h.

5.5.1.23 `typedef void(* VoidFuncStringString) (TC_STRING, TC_STRING)`

Definition at line 29 of file libtc.h.

5.5.1.24 `typedef void(* VoidFuncStringStringFloat) (TC_STRING, TC_STRING, TC_FLOAT)`

Definition at line 35 of file libtc.h.

5.5.1.25 `typedef void(* VoidFuncStringStringInt) (TC_STRING, TC_STRING, TC_INT)`

Definition at line 34 of file libtc.h.

5.5.1.26 `typedef void(* VoidFuncStringStringIntIntFloatString) (TC_STRING, TC_STRING, TC_INT, TC_INT, TC_FLOAT, TC_STRING)`

Definition at line 41 of file libtc.h.

5.5.2 Function Documentation

5.5.2.1 `int importFunctions (TCHANDLE tcHandle, tc_function_library * tc, char * message)`

Definition at line 23 of file libtc.c.

5.6 ReadMe.txt File Reference

5.7 ReadMe.txt File Reference

5.8 tc_data_defs.h File Reference

This graph shows which files directly or indirectly include this file:

Classes

- struct [tc_conditions_as_arrays_of_strings](#)
- struct [tc_elements_strings](#)
- struct [tc_components_strings](#)
- struct [tc_species_strings](#)
- struct [tc_phases_strings](#)
- struct [tc_constituents_strings](#)
- struct [tc_databases_strings](#)
- struct [tc_reference_strings](#)

Macros

- `#define TCHANDLE void*`
- `#define TC_NWSG 4000000`
- `#define TC_NWSE 500000`
- `#define TC_STRLEN_SPECIES 25`
- `#define TC_STRLEN_PHASES 25`
- `#define TC_STRLEN_ELEMENTS 3`
- `#define TC_STRLEN_COMPONENTS 25`
- `#define TC_STRLEN_CONSTITUENTS 25`
- `#define TC_STRLEN_DATABASE 9`
- `#define TC_STRLEN_STOICHIOMETRY 81`
- `#define TC_STRLEN_MAX 256`
- `#define TC_STRLEN_PATH_MAX 512`
- `#define TC_STRLEN_REFERENCE 1024`
- `#define TC_MAX_NR_OF_ELEMENTS 40`

- #define TC_MAX_NR_OF_SPECIES 5000
- #define TC_MAX_NR_OF_SUBLATTICES 10
- #define TC_MAX_NR_OF_CONSTITUENTS 200
- #define TC_MAX_NR_OF_CONST_PER_SUBLATTICE 200
- #define TC_MAX_NR_OF_CONST_PER_SUBLATTICE_IN_IDEAL_GAS 5000
- #define TC_MAX_NR_OF_DATABASES 130
- #define TC_MAX_NR_OF_AXES 5
- #define TC_MAX_NR_OF_PHASES 4000 /* check ITDBPX in tdbmax.inc */
- #define TC_EPS 1.00E-8
- #define TC_VARS
- #define true 1
- #define false 0
- #define DIExport __declspec(dllexport)
- #define TCFuncExport extern DIExport
- #define INTEGER_FUNC extern TC_INT
- #define INTEGER_FUNC_WIN INTEGER_FUNC
- #define INTEGER_FUNC_GNU INTEGER_FUNC
- #define VOID_FUNC_WIN extern void
- #define BOOL_FUNC_WIN extern TC_BOOL
- #define FLOAT_FUNC_WIN extern TC_FLOAT

Typedefs

- typedef long TC_INT
- typedef long pointer
- typedef double TC_FLOAT
- typedef TC_INT TC_BOOL
- typedef char * TC_STRING
- typedef long TC_STRING_LENGTH
- typedef struct tc_conditions_as_arrays_of_strings tc_conditions_as_arrays_of_strings
- typedef struct tc_elements_strings tc_elements_strings
- typedef struct tc_components_strings tc_components_strings
- typedef struct tc_species_strings tc_species_strings
- typedef struct tc_phases_strings tc_phases_strings
- typedef struct tc_constituents_strings tc_constituents_strings
- typedef struct tc_databases_strings tc_databases_strings
- typedef struct tc_reference_strings tc_reference_strings
- typedef TC_INT TC_IARR[4]
- typedef char TC_LABEL_STRING[127]

5.8.1 Macro Definition Documentation

5.8.1.1 #define BOOL_FUNC_WIN extern TC_BOOL

Definition at line 196 of file tc_data_defs.h.

5.8.1.2 #define DIExport __declspec(dllexport)

Definition at line 170 of file tc_data_defs.h.

5.8.1.3 #define false 0

Definition at line 161 of file tc_data_defs.h.

5.8.1.4 #define FLOAT_FUNC_WIN extern TC_FLOAT

Definition at line 197 of file tc_data_defs.h.

5.8.1.5 #define INTEGER_FUNC extern TC_INT

Definition at line 191 of file tc_data_defs.h.

5.8.1.6 #define INTEGER_FUNC_GNU INTEGER_FUNC

Definition at line 194 of file tc_data_defs.h.

5.8.1.7 #define INTEGER_FUNC_WIN INTEGER_FUNC

Definition at line 193 of file tc_data_defs.h.

5.8.1.8 #define TC_EPS 1.00E-8

Definition at line 77 of file tc_data_defs.h.

5.8.1.9 #define TC_MAX_NR_OF_AXES 5

Definition at line 74 of file tc_data_defs.h.

5.8.1.10 #define TC_MAX_NR_OF_CONST_PER_SUBLATTICE 200

Definition at line 71 of file tc_data_defs.h.

5.8.1.11 #define TC_MAX_NR_OF_CONST_PER_SUBLATTICE_IN IDEAL_GAS 5000

Definition at line 72 of file tc_data_defs.h.

5.8.1.12 #define TC_MAX_NR_OF_CONSTITUENTS 200

Definition at line 70 of file tc_data_defs.h.

5.8.1.13 #define TC_MAX_NR_OF_DATABASES 130

Definition at line 73 of file tc_data_defs.h.

5.8.1.14 #define TC_MAX_NR_OF_ELEMENTS 40

Definition at line 67 of file tc_data_defs.h.

5.8.1.15 #define TC_MAX_NR_OF_PHASES 4000 /* check ITDBPX in tdbmax.inc */

Definition at line 76 of file tc_data_defs.h.

5.8.1.16 `#define TC_MAX_NR_OF_SPECIES 5000`

Definition at line 68 of file tc_data_defs.h.

5.8.1.17 `#define TC_MAX_NR_OF_SUBLATTICES 10`

Definition at line 69 of file tc_data_defs.h.

5.8.1.18 `#define TC_NWSE 500000`

Definition at line 54 of file tc_data_defs.h.

5.8.1.19 `#define TC_NWSG 4000000`

Definition at line 53 of file tc_data_defs.h.

5.8.1.20 `#define TC_STRLEN_COMPONENTS 25`

Definition at line 59 of file tc_data_defs.h.

5.8.1.21 `#define TC_STRLEN_CONSTITUENTS 25`

Definition at line 60 of file tc_data_defs.h.

5.8.1.22 `#define TC_STRLEN_DATABASE 9`

Definition at line 61 of file tc_data_defs.h.

5.8.1.23 `#define TC_STRLEN_ELEMENTS 3`

Definition at line 58 of file tc_data_defs.h.

5.8.1.24 `#define TC_STRLEN_MAX 256`

Definition at line 63 of file tc_data_defs.h.

5.8.1.25 `#define TC_STRLEN_PATH_MAX 512`

Definition at line 64 of file tc_data_defs.h.

5.8.1.26 `#define TC_STRLEN_PHASES 25`

Definition at line 57 of file tc_data_defs.h.

5.8.1.27 `#define TC_STRLEN_REFERENCE 1024`

Definition at line 65 of file tc_data_defs.h.

5.8.1.28 #define TC_STRLEN_SPECIES 25

Definition at line 56 of file tc_data_defs.h.

5.8.1.29 #define TC_STRLEN_STOICHIOMETRY 81

Definition at line 62 of file tc_data_defs.h.

5.8.1.30 #define TC_VARS

Definition at line 151 of file tc_data_defs.h.

5.8.1.31 #define TCFuncExport extern DIExport

Definition at line 182 of file tc_data_defs.h.

5.8.1.32 #define TCHANDLE void*

Definition at line 36 of file tc_data_defs.h.

5.8.1.33 #define true 1

Definition at line 157 of file tc_data_defs.h.

5.8.1.34 #define VOID_FUNC_WIN extern void

Definition at line 195 of file tc_data_defs.h.

5.8.2 Typedef Documentation**5.8.2.1 typedef long pointer**

Definition at line 20 of file tc_data_defs.h.

5.8.2.2 typedef TC_INT TC_BOOL

Definition at line 41 of file tc_data_defs.h.

5.8.2.3 typedef struct tc_components_strings tc_components_strings

Definition at line 110 of file tc_data_defs.h.

5.8.2.4 typedef struct tc_conditions_as_arrays_of_strings tc_conditions_as_arrays_of_strings

Definition at line 97 of file tc_data_defs.h.

5.8.2.5 typedef struct tc_constituents_strings tc_constituents_strings

Definition at line 129 of file tc_data_defs.h.

5.8.2.6 typedef struct tc_databases_strings tc_databases_strings

Definition at line 135 of file tc_data_defs.h.

5.8.2.7 typedef struct tc_elements_strings tc_elements_strings

Definition at line 104 of file tc_data_defs.h.

5.8.2.8 typedef double TC_FLOAT

Definition at line 31 of file tc_data_defs.h.

5.8.2.9 typedef TC_INT TC_IARR[4]

Definition at line 147 of file tc_data_defs.h.

5.8.2.10 typedef long TC_INT

Definition at line 19 of file tc_data_defs.h.

5.8.2.11 typedef char TC_LABEL_STRING[127]

Definition at line 148 of file tc_data_defs.h.

5.8.2.12 typedef struct tc_phases_strings tc_phases_strings

Definition at line 123 of file tc_data_defs.h.

5.8.2.13 typedef struct tc_reference_strings tc_reference_strings

Definition at line 141 of file tc_data_defs.h.

5.8.2.14 typedef struct tc_species_strings tc_species_strings

Definition at line 116 of file tc_data_defs.h.

5.8.2.15 typedef char* TC_STRING

Definition at line 42 of file tc_data_defs.h.

5.8.2.16 typedef long TC_STRING_LENGTH

Definition at line 50 of file tc_data_defs.h.

5.9 tcapi.h File Reference

```
#include "tc_data_defs.h"
```

Include dependency graph for tcapi.h: This graph shows which files directly or indirectly include this file:

Functions

- void `tc_append_database` (TC_STRING name)
- TC_BOOL `tc_check_license` (TC_STRING name, TC_STRING message, TC_STRING_LENGTH strlen_↔ message)
- TC_STRING `tc_component_status` (TC_STRING component_name)
- void `tc_compute_equilibrium` ()
- void `tc_create_new_equilibrium` (TC_INT equilibrium)
- TC_INT `tc_database` (TC_STRING datan, TC_INT linelen)
- void `tc_define_components` (TC_STRING component_name, TC_INT strlen, TC_INT number_of_↔ components)
- TC_INT `tc_degrees_of_freedom` ()
- void `tc_deinit` ()
- void `tc_delete_condition` (TC_STRING condition)
- void `tc_delete_symbol` (TC_STRING symbol)
- TC_INT `tc_element` (TC_STRING elements, TC_INT linelen)
- void `tc_element_reject` (TC_STRING element_name)
- void `tc_element_select` (TC_STRING element_name)
- void `tc_enter_ges5_parameter` (TC_STRING parameter, TC_STRING expression)
- void `tc_enter_symbol` (TC_STRING symbol, TC_STRING type, TC_INT argument_type, TC_INT integer_↔ argument, TC_FLOAT double_argument, TC_STRING string_argument)
- TC_BOOL `tc_error` (TC_INT *error_number, TC_STRING message, TC_INT strlen)
- void `tc_ges5` (TC_STRING command)
- void `tc_get_data` ()
- void `tc_get_derivatives` (TC_STRING phase_name, TC_FLOAT *arr1, TC_FLOAT *arr2)
- void `tc_get_ges5_parameter` (TC_STRING parameter, TC_STRING expression, TC_INT strlenExpression)
- TC_FLOAT `tc_get_value` (TC_STRING symbol)
- TC_INT `tc_init_root` ()
- TC_INT `tc_init_root3` (TC_STRING tmpopath, TC_STRING topath)
- TC_INT `tc_list_component` (TC_STRING component_name, TC_INT strlen)
- TC_INT `tc_list_conditions` (TC_STRING conditions, TC_INT strlen)
- TC_INT `tc_list_phase` (TC_STRING phase_name, TC_INT strlen)
- TC_INT `tc_list_species` (TC_STRING species_name, TC_INT strlen)
- TC_INT `tc_list_symbols` (TC_STRING symbols, TC_INT strlen, TC_INT *type)
- TC_INT `tc_nr_of_constituents_in_phase` (TC_STRING phase_name)
- void `tc_open_database` (TC_STRING name)
- TC_INT `tc_phase` (TC_STRING phases, TC_INT linelen)
- TC_INT `tc_phase_all_constituents` (TC_STRING phase_name, TC_INT *constituent_array, TC_STRING_↔ G element_array, TC_INT strLenElem, TC_FLOAT *number_of_sites)
- TC_INT `tc_phase_constituents` (TC_STRING phase_name, TC_INT *constituent_array, TC_STRING_↔ G element_array, TC_INT strLenElem, TC_FLOAT *number_of_sites)
- void `tc_phase_reject` (TC_STRING phase_name)
- void `tc_phase_select` (TC_STRING phase_name)
- TC_STRING `tc_phase_status` (TC_STRING phase_name)
- TC_INT `tc_phase_structure` (TC_STRING phase_name, TC_INT *constituent_array, TC_STRING species_↔ _array, TC_STRING_LENGTH strLenSpecies, TC_FLOAT *number_of_sites)
- void `tc_poly3` (TC_STRING command)
- void `tc_put_sitefractions` (TC_STRING phase_name, TC_FLOAT *sfarr)
- void `tc_read_poly3_file` (TC_STRING filename)
- void `tc_reject_constituent` (TC_STRING phase_name, TC_INT sublattice, TC_STRING constituent)
- void `tc_reset_error` ()
- void `tc_restore_constituent` (TC_STRING phase_name, TC_INT sublattice, TC_STRING constituent)
- void `tc_save_poly3_file` (TC_STRING filename)
- void `tc_select_equilibrium` (TC_INT equilibrium)
- void `tc_set_component_status` (TC_STRING component_name, TC_STRING status)

- void `tc_set_condition` (`TC_STRING` condition, `TC_FLOAT` value)
- void `tc_set_license_code` (`TC_INT` code)
- void `tc_set_minimization_option` (`TC_INT` *global_flag, `TC_INT` *max_gridpoints, `TC_INT` *frequency, `TC_INT` *mesh_flag)
- void `tc_set_phase_addition` (`TC_STRING` phase_name, `TC_FLOAT` addition)
- void `tc_set_phase_status` (`TC_STRING` phase_name, `TC_STRING` status, `TC_FLOAT` value)
- void `tc_set_start_value` (`TC_STRING` state_variable, `TC_FLOAT` starting_value)
- `TC_STRING` `tc_species_status` (`TC_STRING` species_name)
- void `tc_version` (`TC_STRING` str, `TC_INT` str_len)

5.9.1 Function Documentation

5.9.1.1 void `tc_append_database` (`TC_STRING` name)

Appends the named database

5.9.1.2 `TC_BOOL` `tc_check_license` (`TC_STRING` name, `TC_STRING` message, `TC_STRING_LENGTH` strlen_message)

Returns true if the checked license is available and valid, currently accepts: `TC_DLL`, `TC_GUI` and `TC_TC4U`

5.9.1.3 `TC_STRING` `tc_component_status` (`TC_STRING` component_name)

Returns the status of "component_name" where status may be one of "ENTERED" or "SUSPENDED"

5.9.1.4 void `tc_compute_equilibrium` ()

Computes the equilibrium in POLY-3 using the currently set conditions

5.9.1.5 void `tc_create_new_equilibrium` (`TC_INT` equilibrium)

Creates a new equilibrium in POLY-3 with number "equilibrium number".

5.9.1.6 `TC_INT` `tc_database` (`TC_STRING` datan, `TC_INT` linelen)

Returns the number of databases in the system and their names in "datan"

5.9.1.7 void `tc_define_components` (`TC_STRING` component_name, `TC_INT` strlen, `TC_INT` number_of_components)

Redefines the components in the system to the components in "component_name".

5.9.1.8 `TC_INT` `tc_degrees_of_freedom` ()

Returns the degrees of freedom in the system. This must be zero in order to perform an equilibrium calculation.

5.9.1.9 void `tc_deinit` ()

5.9.1.10 void `tc_delete_condition` (`TC_STRING` condition)

Deletes the condition for the expression in "condition".

5.9.1.11 void `tc_delete_symbol` (*TC_STRING symbol*)

Deletes a symbol in the system.

5.9.1.12 *TC_INT* `tc_element` (*TC_STRING elements*, *TC_INT lincelen*)

Returns the number of elements in the database and their names in "elements"

5.9.1.13 void `tc_element_reject` (*TC_STRING element_name*)

Rejects "element_name" in the currently selected database.

5.9.1.14 void `tc_element_select` (*TC_STRING element_name*)

Selects "element_name" in the currently selected database.

5.9.1.15 void `tc_enter_ges5_parameter` (*TC_STRING parameter*, *TC_STRING expression*)

Enters a parameter expression

5.9.1.16 void `tc_enter_symbol` (*TC_STRING symbol*, *TC_STRING type*, *TC_INT argument_type*, *TC_INT integer_argument*, *TC_FLOAT double_argument*, *TC_STRING string_argument*)

Enters a symbol in the system, the symbol type may be one of "CONSTANT", "VARIABLE", "FUNCTION" or "TABLE", "argument_type" defines which of the following arguments will be used, 1 indicates the integer argument, 2 the double argument and 3 the string argument.

5.9.1.17 *TC_BOOL* `tc_error` (*TC_INT * error_number*, *TC_STRING message*, *TC_INT strlen*)

Returns true if an error has been set, returning the error number in "error_number" and its corresponding message in "message"

5.9.1.18 void `tc_ges5` (*TC_STRING command*)

Sends a command to the GES5 module as defined in the argument "command"

5.9.1.19 void `tc_get_data` ()

Executes the database command "GET_DATA"

5.9.1.20 void `tc_get_derivatives` (*TC_STRING phase_name*, *TC_FLOAT * arr1*, *TC_FLOAT * arr2*)

Returns Gm and the first derivatives with respect to site-fractions in "arr1" and the second derivatives in "arr2" as GM.Y1.Y1, GM.Y1.Y2, GM.Y2.Y2, GM.Y1.Y3, GM.Y2.Y3, GM.YN.YN

5.9.1.21 void `tc_get_ges5_parameter` (*TC_STRING parameter*, *TC_STRING expression*, *TC_INT strlenExpression*)

Retrieves the expression of a parameter name

5.9.1.22 TC_FLOAT tc_get_value (TC_STRING *symbol*)

Retrieves the symbol or state variable value from the POLY-3 module.

5.9.1.23 TC_INT tc_init_root ()

Initializes the Thermo-Calc system. This function (or tc_init_root3) must be called prior to anything else.

5.9.1.24 TC_INT tc_init_root3 (TC_STRING *tmppath*, TC_STRING *tcpath*)

Initializes the Thermo-Calc system. This function (or tc_init_root) must be called prior to anything else. *tmppath* Path to directory for log file *tcpath* Path to Thermo-Calc installation (Used to find databases)

5.9.1.25 TC_INT tc_list_component (TC_STRING *component_name*, TC_INT *strlen*)

Returns the number of components in the system and their names in "*component_name*".

5.9.1.26 TC_INT tc_list_conditions (TC_STRING *conditions*, TC_INT *strlen*)

Returns the number of conditions set and their values in "*conditions*".

5.9.1.27 TC_INT tc_list_phase (TC_STRING *phase_name*, TC_INT *strlen*)

Returns the number of phases in the system and their names in "*phase_name*".

5.9.1.28 TC_INT tc_list_species (TC_STRING *species_name*, TC_INT *strlen*)

Returns the number of species in the system and their names in "*species_name*".

5.9.1.29 TC_INT tc_list_symbols (TC_STRING *symbols*, TC_INT *strlen*, TC_INT * *type*)

Returns the number of defined symbols in the system with their expression and value in "*symbols*" and their corresponding type in "*type of symbol*", where the type may be one of 1="CONSTANT", 2="VARIABLE" 3="FUNCTION" 4="TABLE"

5.9.1.30 TC_INT tc_nr_of_constituents_in_phase (TC_STRING *phase_name*)**5.9.1.31 void tc_open_database (TC_STRING *name*)**

Opens the named database "*name_of_database*".

5.9.1.32 TC_INT tc_phase (TC_STRING *phases*, TC_INT *linelen*)

Returns the number of phases in the system with the selected elements. NOTE: the routine returns the number of all available phases.

5.9.1.33 `TC_INT tc_phase_all_constituents (TC_STRING phase_name, TC_INT * constituent_array, TC_STRING element_array, TC_INT strLenElem, TC_FLOAT * number_of_sites)`

Returns the number of sublattices in the phase (including phases with the status SUSPENDED), the number of constituents on each sublattice in "constituent_array", the name of the selected species on each sublattice one after each other in "element_array" and the "number_of_sites" on each sublattice.

5.9.1.34 `TC_INT tc_phase_constituents (TC_STRING phase_name, TC_INT * constituent_array, TC_STRING element_array, TC_INT strLenElem, TC_FLOAT * number_of_sites)`

Returns the number of sublattices in the phase, the number of constituents on each sublattice in "constituent_array", the name of the selected species on each sublattice one after each other in "element_array" and the "number of sites" on each sublattice.

5.9.1.35 `void tc_phase_reject (TC_STRING phase_name)`

Rejects the phase in "phase_name"

5.9.1.36 `void tc_phase_select (TC_STRING phase_name)`

Selects the phase in "phase_name"

5.9.1.37 `TC_STRING tc_phase_status (TC_STRING phase_name)`

Returns the status of "phase_name" where status may be one of "FIXED", "SUSPENDED" or "ENTERED"

5.9.1.38 `TC_INT tc_phase_structure (TC_STRING phase_name, TC_INT * constituent_array, TC_STRING species_array, TC_STRING_LENGTH strLenSpecies, TC_FLOAT * number_of_sites)`

Returns the number of sublattices in the phase, the number of constituents on each sublattice in "constituent_array", the name of the species on each sublattice one after each other in "species_array" and the number of sites in "number of sites".

5.9.1.39 `void tc_poly3 (TC_STRING command)`

Sends a command to POLY-3 module as defined in the argument "command"

5.9.1.40 `void tc_put_sitefractions (TC_STRING phase_name, TC_FLOAT * sfarr)`

5.9.1.41 `void tc_read_poly3_file (TC_STRING filename)`

Loads the workspace from file "filename" in POLY-3.

5.9.1.42 `void tc_reject_constituent (TC_STRING phase_name, TC_INT sublattice, TC_STRING constituent)`

Rejects the constituent "constituent" on sublattice "sublattice" from phase "phase_name".

5.9.1.43 `void tc_reset_error ()`

Resets the error if an error has been set

5.9.1.44 void `tc_restore_constituent` (*TC_STRING phase_name*, *TC_INT sublattice*, *TC_STRING constituent*)

Restores the constituent "constituent" on sublattice "sublattice" from phase "phase_name".

5.9.1.45 void `tc_save_poly3_file` (*TC_STRING filename*)

Stores/overwrites the current workspace in POLY-3 on the file "filename".

5.9.1.46 void `tc_select_equilibrium` (*TC_INT equilibrium*)

Selects an equilibrium in POLY-3 with number "equilibrium".

5.9.1.47 void `tc_set_component_status` (*TC_STRING component_name*, *TC_STRING status*)

Sets the status of "component_name" to "status" to one of "ENTERED" or "SUSPENDED"

5.9.1.48 void `tc_set_condition` (*TC_STRING condition*, *TC_FLOAT value*)

Sets a condition for the expression in "condition" to value in "value".

5.9.1.49 void `tc_set_license_code` (*TC_INT code*)

5.9.1.50 void `tc_set_minimization_option` (*TC_INT * global_flag*, *TC_INT * max_gridpoints*, *TC_INT * frequency*,
*TC_INT * mesh_flag*)

Sets parameters for global minimization

5.9.1.51 void `tc_set_phase_addition` (*TC_STRING phase_name*, *TC_FLOAT addition*)

Sets the addition "addition" to the Gibbs

5.9.1.52 void `tc_set_phase_status` (*TC_STRING phase_name*, *TC_STRING status*, *TC_FLOAT value*)

Sets the status of "phase_name" to "status" to one of "FIXED", "SUSPENDED", "DORMANT" or "ENTERED".

5.9.1.53 void `tc_set_start_value` (*TC_STRING state_variable*, *TC_FLOAT starting_value*)

Sets a starting value for the "state_variable" to "start_value".

5.9.1.54 *TC_STRING* `tc_species_status` (*TC_STRING species_name*)

Returns the status of "species_name" where status may be one of "ENTERED" or "SUSPENDED"

5.9.1.55 void `tc_version` (*TC_STRING str*, *TC_INT str_len*)

Returns the version of Thermo-Calc in "str"

5.10 tcExamples.c File Reference

```
#include <stdio.h>
#include "tcExamples.h"
#include "tcutils.h"
Include dependency graph for tcExamples.c:
```

Typedefs

- typedef char [str8](#)[8]
- typedef [str8](#) [strvect](#)[100]
- typedef [TC_INT](#) [ivect](#)[50]
- typedef [TC_FLOAT](#) [rvect](#)[50]

Functions

- void [example1](#) ([tc_function_library](#) *[tc](#))
- void [example2](#) ([tc_function_library](#) *[tc](#))
- void [example3](#) ([tc_function_library](#) *[tc](#))

5.10.1 Typedef Documentation

5.10.1.1 typedef [TC_INT](#) [ivect](#)[50]

Definition at line 288 of file [tcExamples.c](#).

5.10.1.2 typedef [TC_FLOAT](#) [rvect](#)[50]

Definition at line 289 of file [tcExamples.c](#).

5.10.1.3 typedef char [str8](#)[8]

Definition at line 286 of file [tcExamples.c](#).

5.10.1.4 typedef [str8](#) [strvect](#)[100]

Definition at line 287 of file [tcExamples.c](#).

5.10.2 Function Documentation

5.10.2.1 void [example1](#) ([tc_function_library](#) * [tc](#))

[tcExamples.h](#)

Definition at line 9 of file [tcExamples.c](#).

5.10.2.2 void [example2](#) ([tc_function_library](#) * [tc](#))

Definition at line 201 of file [tcExamples.c](#).

5.10.2.3 void example3 (tc_function_library * tc)

Definition at line 291 of file tcExamples.c.

5.11 tcExamples.h File Reference

```
#include "libtc.h"
```

Include dependency graph for tcExamples.h: This graph shows which files directly or indirectly include this file:

Functions

- void [example1](#) (tc_function_library *tc)
- void [example2](#) (tc_function_library *tc)
- void [example3](#) (tc_function_library *tc)

5.11.1 Function Documentation

5.11.1.1 void example1 (tc_function_library * tc)

[tcExamples.h](#)

Definition at line 9 of file tcExamples.c.

5.11.1.2 void example2 (tc_function_library * tc)

Definition at line 201 of file tcExamples.c.

5.11.1.3 void example3 (tc_function_library * tc)

Definition at line 291 of file tcExamples.c.

5.12 tcMain.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "tcExamples.h"
#include "tcutils.h"
```

Include dependency graph for tcMain.c:

Macros

- #define [TC_API_LIBRARY_NAME](#) "libtcapi-linux-ia32-gfortran-7.5-SNAPSHOT.so"

Functions

- [TCHANDLE loadTCLibraryInCurrentDir](#) ()
- int [importLibThermoCalc](#) (tc_function_library *tc, char *message)
- int [main](#) (int argc, char *argv[])

5.12.1 Macro Definition Documentation

5.12.1.1 `#define TC_API_LIBRARY_NAME "libtcapi-linux-ia32-gfortran-7.5-SNAPSHOT.so"`

Definition at line 28 of file tcMain.c.

5.12.2 Function Documentation

5.12.2.1 `int importLibThermoCalc (tc_function_library * tc, char * message)`

Definition at line 66 of file tcMain.c.

5.12.2.2 `TCHANDLE loadTCLibraryInCurrentDir ()`

Definition at line 38 of file tcMain.c.

5.12.2.3 `int main (int argc, char * argv[])`

Definition at line 100 of file tcMain.c.

5.13 tcutils.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "tcutils.h"
Include dependency graph for tcutils.c:
```

Functions

- void [getThermoCalcEnvironmentPath](#) (char *pathBuffer)
- void [getTempEnvironmentPath](#) (char *pathBuffer)

5.13.1 Function Documentation

5.13.1.1 `void getTempEnvironmentPath (char * pathBuffer)`

Get path to temp directory If it can't find it - default to current working directory

Definition at line 29 of file tcutils.c.

5.13.1.2 `void getThermoCalcEnvironmentPath (char * pathBuffer)`

Get path from the values of Thermo-Calc environment variables TC61_HOME TCPATH Older versions and fallback

Definition at line 12 of file tcutils.c.

5.14 tcutils.h File Reference

```
#include <unistd.h>
Include dependency graph for tcutils.h: This graph shows which files directly or indirectly include this file:
```

Macros

- #define [getCurrentWorkingDir](#) getcwd
- #define [TCPATH](#) "TCPATH"
- #define [TC61_HOME](#) "TC61_HOME"
- #define [TEMP](#) "TMPDIR"
- #define [SLASH](#) "/"

Functions

- void [getThermoCalcEnvironmentPath](#) (char *pathBuffer)
- void [getTempEnvironmentPath](#) (char *pathBuffer)

5.14.1 Macro Definition Documentation

5.14.1.1 #define [getCurrentWorkingDir](#) getcwd

Definition at line 6 of file tcutils.h.

5.14.1.2 #define [SLASH](#) "/"

Definition at line 21 of file tcutils.h.

5.14.1.3 #define [TC61_HOME](#) "TC61_HOME"

Definition at line 10 of file tcutils.h.

5.14.1.4 #define [TCPATH](#) "TCPATH"

Definition at line 9 of file tcutils.h.

5.14.1.5 #define [TEMP](#) "TMPDIR"

Definition at line 15 of file tcutils.h.

5.14.2 Function Documentation

5.14.2.1 void [getTempEnvironmentPath](#) (char * *pathBuffer*)

Get path to temp directory If it can't find it - default to current working directory

Definition at line 29 of file tcutils.c.

5.14.2.2 void [getThermoCalcEnvironmentPath](#) (char * *pathBuffer*)

Get path from the values of Thermo-Calc environment variables TC61_HOME TCPATH Older versions and fallback

Definition at line 12 of file tcutils.c.

Index

- [_tc_function_library](#), 9
 - [tc_append_database](#), 10
 - [tc_check_license](#), 10
 - [tc_component_status](#), 10
 - [tc_compute_equilibrium](#), 10
 - [tc_create_new_equilibrium](#), 10
 - [tc_database](#), 10
 - [tc_define_components](#), 11
 - [tc_degrees_of_freedom](#), 11
 - [tc_deinit](#), 11
 - [tc_delete_condition](#), 11
 - [tc_delete_symbol](#), 11
 - [tc_element](#), 11
 - [tc_element_reject](#), 11
 - [tc_element_select](#), 11
 - [tc_enter_ges5_parameter](#), 11
 - [tc_enter_symbol](#), 11
 - [tc_error](#), 11
 - [tc_ges5](#), 11
 - [tc_get_data](#), 12
 - [tc_get_derivatives](#), 12
 - [tc_get_ges5_parameter](#), 12
 - [tc_get_value](#), 12
 - [tc_init_root](#), 12
 - [tc_init_root3](#), 12
 - [tc_list_component](#), 12
 - [tc_list_conditions](#), 12
 - [tc_list_phase](#), 12
 - [tc_list_species](#), 12
 - [tc_list_symbols](#), 12
 - [tc_nr_of_constituents_in_phase](#), 12
 - [tc_open_database](#), 13
 - [tc_phase](#), 13
 - [tc_phase_all_constituents](#), 13
 - [tc_phase_constituents](#), 13
 - [tc_phase_reject](#), 13
 - [tc_phase_select](#), 13
 - [tc_phase_status](#), 13
 - [tc_phase_structure](#), 13
 - [tc_poly3](#), 13
 - [tc_put_sitefractions](#), 13
 - [tc_read_poly3_file](#), 13
 - [tc_reject_constituent](#), 13
 - [tc_reset_error](#), 14
 - [tc_restore_constituent](#), 14
 - [tc_save_poly3_file](#), 14
 - [tc_select_equilibrium](#), 14
 - [tc_set_component_status](#), 14
 - [tc_set_condition](#), 14
 - [tc_set_license_code](#), 14
 - [tc_set_minimization_option](#), 14
 - [tc_set_phase_addition](#), 14
 - [tc_set_phase_status](#), 14
 - [tc_set_start_value](#), 14
 - [tc_species_status](#), 14
 - [tc_version](#), 15
 - [BOOL_FUNC_WIN](#)
 - [tc_data_defs.h](#), 25
 - [BoolFuncIntPStringInt](#)
 - [libtc.h](#), 22
 - [BoolFuncStringStringStrLen](#)
 - [libtc.h](#), 22
 - [component](#)
 - [tc_components_strings](#), 15
 - [condition](#)
 - [tc_conditions_as_arrays_of_strings](#), 16
 - [constituent](#)
 - [tc_constituents_strings](#), 16
 - [database](#)
 - [tc_databases_strings](#), 16
 - [DllExport](#)
 - [tc_data_defs.h](#), 25
 - [element](#)
 - [tc_elements_strings](#), 17
 - [example1](#)
 - [tcExamples.c](#), 36
 - [tcExamples.h](#), 37
 - [example1.c](#), 19
 - [example2](#)
 - [tcExamples.c](#), 36
 - [tcExamples.h](#), 37
 - [example2.c](#), 19
 - [main](#), 19
 - [example3](#)
 - [tcExamples.c](#), 36
 - [tcExamples.h](#), 37
 - [example3.c](#), 19
 - [ivect](#), 20
 - [main](#), 20
 - [rvect](#), 20
 - [str8](#), 20
 - [strvect](#), 20
- [FLOAT_FUNC_WIN](#)
 - [tc_data_defs.h](#), 25
- [false](#)

- tc_data_defs.h, 25
- FloatFuncString
 - libtc.h, 22
- getCurrentWorkingDir
 - tcutils.h, 39
- getTempEnvironmentPath
 - tcutils.c, 38
 - tcutils.h, 39
- getThermoCalcEnvironmentPath
 - tcutils.c, 38
 - tcutils.h, 39
- INTEGER_FUNC
 - tc_data_defs.h, 26
- INTEGER_FUNC_GNU
 - tc_data_defs.h, 26
- INTEGER_FUNC_WIN
 - tc_data_defs.h, 26
- importFunctions
 - libtc.c, 20
 - libtc.h, 24
- importLibThermoCalc
 - tcMain.c, 38
- IntFuncNoParams
 - libtc.h, 22
- IntFuncString
 - libtc.h, 22
- IntFuncStringInt
 - libtc.h, 22
- IntFuncStringIntIntP
 - libtc.h, 22
- IntFuncStringIntPStringIntFloatP
 - libtc.h, 22
- IntFuncStringIntPStringStrLenFloatP
 - libtc.h, 22
- IntFuncStringString
 - libtc.h, 22
- ivect
 - example3.c, 20
 - tcExamples.c, 36
- libtc.c, 20
 - importFunctions, 20
 - tloadfunc, 20
- libtc.h, 21
 - BoolFuncIntPStringInt, 22
 - BoolFuncStringStringStrLen, 22
 - FloatFuncString, 22
 - importFunctions, 24
 - IntFuncNoParams, 22
 - IntFuncString, 22
 - IntFuncStringInt, 22
 - IntFuncStringIntIntP, 22
 - IntFuncStringIntPStringIntFloatP, 22
 - IntFuncStringIntPStringStrLenFloatP, 22
 - IntFuncStringString, 22
 - StringFuncString, 22
 - tc_function_library, 22
 - VoidFuncInt, 23
 - VoidFuncIntPIntPIntPIntP, 23
 - VoidFuncNoParams, 23
 - VoidFuncString, 23
 - VoidFuncStringFloat, 23
 - VoidFuncStringFloatP, 23
 - VoidFuncStringFloatPFloatP, 23
 - VoidFuncStringInt, 23
 - VoidFuncStringIntInt, 23
 - VoidFuncStringIntString, 23
 - VoidFuncStringString, 23
 - VoidFuncStringStringFloat, 23
 - VoidFuncStringStringInt, 23
 - VoidFuncStringStringIntIntFloatString, 24
- loadTCLibraryInCurrentDir
 - tcMain.c, 38
- main
 - example2.c, 19
 - example3.c, 20
 - tcMain.c, 38
- phase
 - tc_phases_strings, 17
- pointer
 - tc_data_defs.h, 28
- ReadMe.txt, 24
- reference
 - tc_reference_strings, 18
- rvect
 - example3.c, 20
 - tcExamples.c, 36
- SLASH
 - tcutils.h, 39
- specie
 - tc_species_strings, 18
- str8
 - example3.c, 20
 - tcExamples.c, 36
- StringFuncString
 - libtc.h, 22
- strvect
 - example3.c, 20
 - tcExamples.c, 36
- TC61_HOME
 - tcutils.h, 39
- TC_API_LIBRARY_NAME
 - tcMain.c, 38
- TC_BOOL
 - tc_data_defs.h, 28
- TC_EPS
 - tc_data_defs.h, 26
- TC_FLOAT
 - tc_data_defs.h, 29
- TC_IARR
 - tc_data_defs.h, 29

TC_INT
tc_data_defs.h, 29

TC_LABEL_STRING
tc_data_defs.h, 29

TC_MAX_NR_OF_AXES
tc_data_defs.h, 26

TC_MAX_NR_OF_CONST_PER_SUBLATTICE
tc_data_defs.h, 26

TC_MAX_NR_OF_CONST_PER_SUBLATTICE_IN_IDEAL_GAS
tc_data_defs.h, 26

TC_MAX_NR_OF_CONSTITUENTS
tc_data_defs.h, 26

TC_MAX_NR_OF_DATABASES
tc_data_defs.h, 26

TC_MAX_NR_OF_ELEMENTS
tc_data_defs.h, 26

TC_MAX_NR_OF_PHASES
tc_data_defs.h, 26

TC_MAX_NR_OF_SPECIES
tc_data_defs.h, 26

TC_MAX_NR_OF_SUBLATTICES
tc_data_defs.h, 27

TC_NWSE
tc_data_defs.h, 27

TC_NWSG
tc_data_defs.h, 27

TC_STRING
tc_data_defs.h, 29

TC_STRING_LENGTH
tc_data_defs.h, 29

TC_STRLEN_COMPONENTS
tc_data_defs.h, 27

TC_STRLEN_CONSTITUENTS
tc_data_defs.h, 27

TC_STRLEN_DATABASE
tc_data_defs.h, 27

TC_STRLEN_ELEMENTS
tc_data_defs.h, 27

TC_STRLEN_MAX
tc_data_defs.h, 27

TC_STRLEN_PATH_MAX
tc_data_defs.h, 27

TC_STRLEN_PHASES
tc_data_defs.h, 27

TC_STRLEN_REFERENCE
tc_data_defs.h, 27

TC_STRLEN_SPECIES
tc_data_defs.h, 27

TC_STRLEN_STOICHIOMETRY
tc_data_defs.h, 28

TC_VARS
tc_data_defs.h, 28

TCFuncExport
tc_data_defs.h, 28

TCHANDLE
tc_data_defs.h, 28

TCPATH
tcutils.h, 39

TEMP
tcutils.h, 39

tc_append_database
_tc_function_library, 10
tcapi.h, 31

tc_check_license
_tc_function_library, 10
tcapi.h, 31

tc_component_status
_tc_function_library, 10
tcapi.h, 31

tc_components_strings, 15
component, 15
tc_data_defs.h, 28

tc_compute_equilibrium
_tc_function_library, 10
tcapi.h, 31

tc_conditions_as_arrays_of_strings, 15
condition, 16
tc_data_defs.h, 28

tc_constituents_strings, 16
constituent, 16
tc_data_defs.h, 28

tc_create_new_equilibrium
_tc_function_library, 10
tcapi.h, 31

tc_data_defs.h, 24
BOOL_FUNC_WIN, 25
DllExport, 25
FLOAT_FUNC_WIN, 25
false, 25
INTEGER_FUNC, 26
INTEGER_FUNC_GNU, 26
INTEGER_FUNC_WIN, 26
pointer, 28
TC_BOOL, 28
TC_EPS, 26
TC_FLOAT, 29
TC_IARR, 29
TC_INT, 29
TC_LABEL_STRING, 29
TC_MAX_NR_OF_AXES, 26
TC_MAX_NR_OF_CONST_PER_SUBLATTICE, 26
TC_MAX_NR_OF_CONST_PER_SUBLATTICE_IN_IDEAL_GAS, 26
TC_MAX_NR_OF_CONSTITUENTS, 26
TC_MAX_NR_OF_DATABASES, 26
TC_MAX_NR_OF_ELEMENTS, 26
TC_MAX_NR_OF_PHASES, 26
TC_MAX_NR_OF_SPECIES, 26
TC_MAX_NR_OF_SUBLATTICES, 27
TC_NWSE, 27
TC_NWSG, 27
TC_STRING, 29
TC_STRING_LENGTH, 29
TC_STRLEN_COMPONENTS, 27

- TC_STRLLEN_CONSTITUENTS, [27](#)
- TC_STRLLEN_DATABASE, [27](#)
- TC_STRLLEN_ELEMENTS, [27](#)
- TC_STRLLEN_MAX, [27](#)
- TC_STRLLEN_PATH_MAX, [27](#)
- TC_STRLLEN_PHASES, [27](#)
- TC_STRLLEN_REFERENCE, [27](#)
- TC_STRLLEN_SPECIES, [27](#)
- TC_STRLLEN_STOICHIOMETRY, [28](#)
- TC_VARS, [28](#)
- TCFuncExport, [28](#)
- TCHANDLE, [28](#)
- tc_components_strings, [28](#)
- tc_conditions_as_arrays_of_strings, [28](#)
- tc_constituents_strings, [28](#)
- tc_databases_strings, [28](#)
- tc_elements_strings, [29](#)
- tc_phases_strings, [29](#)
- tc_reference_strings, [29](#)
- tc_species_strings, [29](#)
- true, [28](#)
- VOID_FUNC_WIN, [28](#)
- tc_database
 - [_tc_function_library](#), [10](#)
 - [tcapi.h](#), [31](#)
- tc_databases_strings, [16](#)
 - database, [16](#)
 - [tc_data_defs.h](#), [28](#)
- tc_define_components
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [31](#)
- tc_degrees_of_freedom
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [31](#)
- tc_deinit
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [31](#)
- tc_delete_condition
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [31](#)
- tc_delete_symbol
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [31](#)
- tc_element
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_element_reject
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_element_select
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_elements_strings, [17](#)
 - element, [17](#)
 - [tc_data_defs.h](#), [29](#)
- tc_enter_ges5_parameter
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_enter_symbol
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_error
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_function_library
 - [libtc.h](#), [22](#)
- tc_ges5
 - [_tc_function_library](#), [11](#)
 - [tcapi.h](#), [32](#)
- tc_get_data
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [32](#)
- tc_get_derivatives
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [32](#)
- tc_get_ges5_parameter
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [32](#)
- tc_get_value
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [32](#)
- tc_init_root
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_init_root3
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_list_component
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_list_conditions
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_list_phase
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_list_species
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_list_symbols
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_nr_of_constituents_in_phase
 - [_tc_function_library](#), [12](#)
 - [tcapi.h](#), [33](#)
- tc_open_database
 - [_tc_function_library](#), [13](#)
 - [tcapi.h](#), [33](#)
- tc_phase
 - [_tc_function_library](#), [13](#)
 - [tcapi.h](#), [33](#)
- tc_phase_all_constituents
 - [_tc_function_library](#), [13](#)
 - [tcapi.h](#), [33](#)
- tc_phase_constituents
 - [_tc_function_library](#), [13](#)

- tcapi.h, 34
- tc_phase_reject
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_phase_select
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_phase_status
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_phase_structure
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_phases_strings, 17
 - phase, 17
 - tc_data_defs.h, 29
- tc_poly3
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_put_sitefractions
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_read_poly3_file
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_reference_strings, 17
 - reference, 18
 - tc_data_defs.h, 29
- tc_reject_constituent
 - _tc_function_library, 13
 - tcapi.h, 34
- tc_reset_error
 - _tc_function_library, 14
 - tcapi.h, 34
- tc_restore_constituent
 - _tc_function_library, 14
 - tcapi.h, 34
- tc_save_poly3_file
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_select_equilibrium
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_component_status
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_condition
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_license_code
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_minimization_option
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_phase_addition
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_phase_status
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_set_start_value
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_species_status
 - _tc_function_library, 14
 - tcapi.h, 35
- tc_species_strings, 18
 - specie, 18
 - tc_data_defs.h, 29
- tc_version
 - _tc_function_library, 15
 - tcapi.h, 35
- tcExamples.c, 36
 - example1, 36
 - example2, 36
 - example3, 36
 - ivect, 36
 - rvect, 36
 - str8, 36
 - strvect, 36
- tcExamples.h, 37
 - example1, 37
 - example2, 37
 - example3, 37
- tcMain.c, 37
 - importLibThermoCalc, 38
 - loadTCLibraryInCurrentDir, 38
 - main, 38
 - TC_API_LIBRARY_NAME, 38
- tcapi.h, 29
 - tc_append_database, 31
 - tc_check_license, 31
 - tc_component_status, 31
 - tc_compute_equilibrium, 31
 - tc_create_new_equilibrium, 31
 - tc_database, 31
 - tc_define_components, 31
 - tc_degrees_of_freedom, 31
 - tc_deinit, 31
 - tc_delete_condition, 31
 - tc_delete_symbol, 31
 - tc_element, 32
 - tc_element_reject, 32
 - tc_element_select, 32
 - tc_enter_ges5_parameter, 32
 - tc_enter_symbol, 32
 - tc_error, 32
 - tc_ges5, 32
 - tc_get_data, 32
 - tc_get_derivatives, 32
 - tc_get_ges5_parameter, 32
 - tc_get_value, 32
 - tc_init_root, 33
 - tc_init_root3, 33
 - tc_list_component, 33

- tc_list_conditions, 33
- tc_list_phase, 33
- tc_list_species, 33
- tc_list_symbols, 33
- tc_nr_of_constituents_in_phase, 33
- tc_open_database, 33
- tc_phase, 33
- tc_phase_all_constituents, 33
- tc_phase_constituents, 34
- tc_phase_reject, 34
- tc_phase_select, 34
- tc_phase_status, 34
- tc_phase_structure, 34
- tc_poly3, 34
- tc_put_sitefractions, 34
- tc_read_poly3_file, 34
- tc_reject_constituent, 34
- tc_reset_error, 34
- tc_restore_constituent, 34
- tc_save_poly3_file, 35
- tc_select_equilibrium, 35
- tc_set_component_status, 35
- tc_set_condition, 35
- tc_set_license_code, 35
- tc_set_minimization_option, 35
- tc_set_phase_addition, 35
- tc_set_phase_status, 35
- tc_set_start_value, 35
- tc_species_status, 35
- tc_version, 35
- tclloadfunc
 - libtc.c, 20
- tcutils.c, 38
 - getTempEnvironmentPath, 38
 - getThermoCalcEnvironmentPath, 38
- tcutils.h, 38
 - getCurrentWorkingDir, 39
 - getTempEnvironmentPath, 39
 - getThermoCalcEnvironmentPath, 39
 - SLASH, 39
 - TC61_HOME, 39
 - TCPATH, 39
 - TEMP, 39
- true
 - tc_data_defs.h, 28
- VOID_FUNC_WIN
 - tc_data_defs.h, 28
- VoidFuncInt
 - libtc.h, 23
- VoidFuncIntPIntPIntPIntP
 - libtc.h, 23
- VoidFuncNoParams
 - libtc.h, 23
- VoidFuncString
 - libtc.h, 23
- VoidFuncStringFloat
 - libtc.h, 23
- VoidFuncStringFloatP
 - libtc.h, 23
- VoidFuncStringFloatPFloatP
 - libtc.h, 23
- VoidFuncStringInt
 - libtc.h, 23
- VoidFuncStringIntInt
 - libtc.h, 23
- VoidFuncStringIntString
 - libtc.h, 23
- VoidFuncStringString
 - libtc.h, 23
- VoidFuncStringStringFloat
 - libtc.h, 23
- VoidFuncStringStringInt
 - libtc.h, 23
- VoidFuncStringStringIntIntFloatString
 - libtc.h, 24