

GENIUS TOOLS for Creo

10.0.1.0

User manual

© 2023 INNEO Solutions GmbH



Table of Contents

1	Product overview	11
2	Full details of all functions	13
3	What's new in release 10	18
3.1	Dimension	18
3.2	Parameter	19
3.3	UDF Forms	19
3.4	Utilities	19
3.5	Value Transfer	20
3.6	Javascript	21
4	License-dependent functions	22
5	Assembly Report	25
5.1	Fundamentals	26
5.1.1	Glossary	26
5.1.2	Standard behavior	27
5.2	Usage	28
5.2.1	User interface	30
5.2.2	Command bar	30
5.2.3	Report options	32
5.2.4	Report definitions	34
5.2.5	Model selection and view	34
5.2.6	Model list	35
5.2.7	Report table	36
5.2.8	Saving	38
5.3	Configuration	39
5.3.1	Assembly Report Editor	39
5.3.1.1	User interface	39
5.3.1.2	Command bar	40
5.3.1.3	General information	41
5.3.1.4	Creating table parameters	42
5.3.1.5	Configuring report table	49
5.3.1.6	Use cases	54
5.3.2	Configuring export of reports	59
5.3.3	Creating an Excel template	61
5.3.4	Automatic update of report definitions	62
5.3.5	Using position values from Windchill	63
5.3.6	Use cases	65
5.3.6.1	Removing single lines from parts list	65
5.3.6.2	Filtering out subcomponents of purchased parts	69
6	Dimension	74

Table of Contents

6.1	Starting the program	74
6.2	User interface	76
6.2.1	Command bar	77
6.2.2	Filter and search	77
6.2.3	Model selection	79
6.3	Dimensions table	80
6.3.1	Dimension symbols and context menu	81
6.3.2	Input fields	82
6.3.3	Colors in input fields	83
6.3.4	Tolerance types	83
7	Forms	85
7.1	Fundamentals	86
7.2	Usage	86
7.2.1	User interface	87
7.2.2	Command bar	88
7.2.3	Form display	91
7.2.4	Using a form	93
7.2.4.1	Filling input and selection fields	94
7.2.4.2	Modifying components	95
7.2.4.3	Automatic setting of properties	95
7.2.4.4	Setting multiple properties	96
7.3	Configuration	96
7.3.1	Forms Editor	97
7.3.1.1	User interface	98
7.3.1.2	Command bar	98
7.3.1.3	Managing form elements	99
7.3.1.4	Describing a form	103
7.3.1.5	Creating form elements	105
7.3.1.6	Editing form display	114
7.3.1.7	JavaScript in Forms	116
7.3.1.8	Automatic load of external data	117
7.3.2	Use cases	118
7.3.2.1	Creating and applying a Form	118
7.3.2.2	Automatically set tolerance from variant table	118
8	Inspect	129
8.1	Fundamentals	129
8.2	Usage	131
8.2.1	User interface	132
8.2.2	Command bar	132
8.2.3	Positioning and Overview tabs	133
8.2.4	Usage examples	136
8.2.4.1	Positioning	136

Table of Contents

8.2.4.2	Numbering	137
8.2.4.3	Overview	140
8.2.4.4	Hide and unhide	143
8.2.4.5	Export to Excel	144
8.3	Configuration	147
8.3.1	Inspect Editor	147
8.3.1.1	User interface	148
8.3.1.2	Command bar	148
8.3.1.3	Inspect configuration	149
8.3.1.4	Symbols and tables tab	149
8.3.1.5	Overview tab	156
8.3.1.6	Attachments tab	156
8.3.1.7	Filter tab	157
8.3.2	Creating a Creo symbol for Inspect	158
8.3.3	Adding parameter values to a symbol	163
8.3.4	Creating change symbols and tables	165
8.3.5	Customize report tables	168
8.4	Inspect Revision	169
8.4.1	Fundamentals	169
8.4.2	Starting the program	170
8.4.3	User interface	170
8.4.4	Command bar	171
8.4.5	Select configuration file	172
8.4.6	Select tree view	173
8.4.7	Creating snapshots	173
8.4.8	History of snapshots and symbols	175
8.4.9	Drawing revision parameter	176
8.4.10	Export snapshot history	177
9	Library	181
9.1	Fundamentals	182
9.1.1	Glossary	183
9.1.2	Use cases	184
9.2	Usage	186
9.2.1	User interface	189
9.2.2	Tool menu	189
9.2.3	Search and filter	190
9.2.4	List of search operations	195
9.2.5	Navigation and object display	198
9.2.6	Favorites	200
9.2.7	Detail window	201
9.2.8	Forms search and read values from XML	206
9.2.9	Use cases	207
9.2.9.1	Using library objects	207

Table of Contents

9.3	Configuration	212
9.3.1	Areas in the resource directory	212
9.3.2	Import data from PDM/PLM	212
9.3.3	Multilingual use	213
9.3.4	Library Editor	215
9.3.4.1	User interface	216
9.3.4.2	Command bar	216
9.3.4.3	Category tree	217
9.3.4.4	Category view	219
9.3.4.5	Object details	222
9.3.4.6	Actions on library objects	236
9.3.4.7	Wizard: Batch mode	240
9.3.4.8	Wizard: Import	243
9.3.4.9	Wizard: MNU export	243
9.3.4.10	Wizard: Object collector	244
9.3.4.11	Cleaning up data	247
9.3.4.12	Customizing the search function	247
9.3.5	XML interface for Form values	248
9.3.6	Data structure	249
9.3.7	Setting configuration options	251
9.3.8	Use cases	251
9.3.8.1	Creating a new library	251
9.3.8.2	Creating a new category	252
9.3.8.3	Creating a new library object	252
9.3.8.4	Importing family tables	254
9.3.8.5	Updating objects	256
9.3.8.6	Importing additional object information	258
9.3.8.7	Importing Library Viewer libraries	260
9.3.8.8	Importing MUI files	265
10	Material	267
10.1	Introduction	267
10.2	Functioning	268
10.3	Usage	269
10.3.1	User interface	271
10.3.2	Selecting materials	272
10.3.3	Assigning materials	274
10.3.4	Deleting material	277
10.3.5	Updating materials	278
10.3.6	Checking materials in a model	279
10.4	Configuration	281
10.4.1	Material Selector Editor	281
10.4.1.1	Starting the program	281
10.4.1.2	User interface	282

Table of Contents

10.4.1.3	Material directory	283
10.4.1.4	Defining selectable materials	284
10.4.1.5	Display settings of Material Selector dialog	286
10.4.2	Activating material check	288
10.4.3	Edit material files	291
10.4.3.1	Edit material files with GENIUS TOOLS Material Browser	291
10.4.3.2	Edit material files in Windchill	292
11	Multibody to Assembly	294
11.1	Usage	294
11.1.1	User interface	296
11.1.2	Command bar	297
11.1.3	Body export settings	298
11.1.4	Evaluate, Export and Update	300
11.1.5	Use Case	301
11.1.5.1	Export of referenced bodies	301
12	Name Generator	306
12.1	Usage	306
12.1.1	User interface	310
12.2	Configuration	311
12.2.1	Name Generator Editor	312
12.2.1.1	User interface	313
12.2.1.2	Available name configurations	313
12.2.1.3	Configuration details	314
12.2.2	Use cases	316
12.2.2.1	Global name configurations with fallback	316
12.2.2.2	Global name configurations with Windchill	318
12.2.2.3	Integrating name configurations in Quick Access	320
12.2.2.4	Using project numbers	321
13	Parameter	327
13.1	Fundamentals	327
13.1.1	Glossary	328
13.1.2	Parameter management concept	329
13.1.3	Mechanisms on starting and saving	330
13.2	Usage	332
13.2.1	User interface	334
13.2.2	Command bar	335
13.2.3	Reading parameter values from text files	338
13.2.4	Status dialog	338
13.2.5	Model selection	339
13.2.6	Model list	340
13.2.7	Parameter form	342
13.2.7.1	Calculating / assigning masses	345

Table of Contents

13.2.8	Field types	346
13.3	Configuration	349
13.3.1	Regular Expressions	349
13.3.2	Parameter Editor	351
13.3.2.1	User interface	352
13.3.2.2	Command bar	353
13.3.2.3	Element list	353
13.3.2.4	Parameter definitions	355
13.3.2.5	Adding Parameters	359
13.3.2.6	Adding separators	375
13.3.2.7	Editing parameter definition list	376
13.3.3	Use cases	377
13.3.3.1	Creating a simple parameter definition	379
13.3.3.2	Creating a Parameter definition with tabs	390
13.3.3.3	Importing old parameter definitions	391
13.3.3.4	Auto-select parameter definitions	394
13.3.3.5	Using JavaScript with Parameter	396
14	Quick Access	405
14.1	Usage	405
14.1.1	Starting the program	405
14.1.2	User interface	406
14.1.3	Mode and selection dependence	407
14.1.4	Usage examples	408
14.2	Configuration	408
14.2.1	Different types of use	408
14.2.2	Quick Access Editor	411
14.2.2.1	User interface	411
14.2.2.2	Commands overview	412
14.2.2.3	Command details	413
14.2.2.4	Creating commands	415
14.2.2.5	Defining commands	415
14.2.2.6	Defining mapkeys	416
14.2.2.7	Export and import	422
14.2.3	Changing the start key	423
14.2.4	Inserting project images	424
14.2.5	Adapting zoom levels and icons	425
14.2.6	Use cases	427
14.2.6.1	Recording a mapkey	427
15	Value Transfer	430
15.1	Starting the program	430
15.2	User interface	431
15.3	Command bar	432

Table of Contents

15.4	Transfer list	432
15.5	Filter list	434
15.6	Table of results	435
16	UDF Forms	437
16.1	Fundamentals	437
16.2	Usage	438
16.2.1	User interface	439
16.2.2	Command bar	440
16.2.3	UDF selection	443
16.2.4	Placement	443
16.2.5	Form section	445
16.2.6	Use cases	446
16.2.6.1	Linking UDFs to drawing tables	446
16.2.6.2	Linking UDFs to drawing symbols	447
16.3	Configuration	449
16.3.1	UDF Forms Editor	450
16.3.1.1	User interface	451
16.3.1.2	Command bar	451
16.3.1.3	Details of UDF	453
16.3.1.4	References and variables tab	454
16.3.1.5	Description tab	463
16.3.2	Updating UDF definitions	465
17	Utilities	469
17.1	3D Note Form	470
17.1.1	User interface	471
17.1.2	Displaying and editing notes	473
17.1.3	Usage example	474
17.2	Close All Other Windows	477
17.3	Command Control	478
17.4	Component Parameter	479
17.4.1	User interface	480
17.4.2	Editing position number parameters	481
17.4.3	Creating component parameters	483
17.4.4	XML configuration	483
17.5	Copy Component Parameter To Substitution Component Parameter	485
17.6	Create Search.pro	487
17.7	Create Tolerance Table	491
17.8	CS Assembler	493
17.9	Default Text Editor	495
17.9.1	Create default text	496
17.9.2	Select default text	497

Table of Contents

17.10	Export Points	498
17.10.1	Exporting points	499
17.10.2	Settings for points on curves	501
17.10.3	Export options	502
17.10.4	Creating a template	503
17.10.5	Example template	504
17.11	Export Table One-To-One to Excel	506
17.12	Export Table to CSV	507
17.13	Export Table to Excel	509
17.13.1	Creating reports	510
17.13.2	Create template	512
17.13.2.1	Step-by-step guide	512
17.13.2.2	More options for templates	516
17.13.2.3	Export data from several GENIUS-TOOLS-components	517
17.14	Extend Relations	519
17.15	Extended Dimension Functions	524
17.16	Feature Regeneration Profiler	526
17.17	Full Backup	527
17.17.1	Collect objects for backup list	528
17.17.2	Choose levels of dependencies	530
17.18	Load Save Converter	533
17.19	Open Base Model	535
17.20	Open/Create Drawing	536
17.21	Select Surfaces by Color	537
17.22	Show Information	538
17.23	Show Thread Size	540
17.24	Show Pitch	542
17.25	Toggle Symbol Variants	545
17.26	Work Dir Manager	546
18	Configuring GENIUS TOOLS for Creo	550
18.1	Configuration and start	550
18.2	Configuration Utility	558
18.2.1	Starting the program	559
18.2.2	User interface	559
18.2.3	Configuration overview	561
18.2.4	Searching configuration options	562
18.2.5	Editing configuration options	565
18.2.6	Saving changes	569
18.2.7	Directly applying changes	571
18.2.8	Resetting configuration options	571
18.2.9	Using comments	572

Table of Contents

18.2.10	Viewing configuration files and layers	573
18.2.11	Environment variables	574
18.2.12	Use cases	575
18.2.12.1	Set access to programm editors	575
18.3	Database Version Control	576
18.3.1	Usage	576
18.3.2	Database and product versions	577
18.4	Javascript	577
18.4.1	JavaScript Editor	577
18.4.2	List of Javascript funtions	580
18.4.3	Explanations and examples	582
18.5	Color selector	590
18.6	Display Embedded Data	592
18.6.1	User interface	592
18.7	Configuration Options	593
18.7.1	General Configuration Options	593
18.7.2	Assembly Report	602
18.7.3	Dimension	606
18.7.4	Forms	607
18.7.5	Inspect	610
18.7.6	Library	615
18.7.7	Material	635
18.7.8	Multibody To Assembly	638
18.7.9	Name Generator	640
18.7.10	Parameter	641
18.7.11	Quick Access	648
18.7.12	UDF Forms	652
18.7.13	Utilities	654
18.7.13.1	Configuration Options	654
18.7.13.2	Annotation Value Input	660
18.7.13.3	Command Control	683
18.7.13.4	Component Parameters	664
18.7.13.5	Create search.pro	668
18.7.13.6	Create Tolerance Table	673
18.7.13.7	CS Assembler	663
18.7.13.8	Edit Standard Texts	669
18.7.13.9	Extend Relations	661
18.7.13.10	Full Backup	682
18.7.13.11	Load Save Converter	664
18.7.13.12	Open Base Model	661
18.7.13.13	Open/Create Drawing	679
18.7.13.14	Points	664
18.7.13.15	Select Surfaces by Color	664

Table of Contents

18.7.13.16	Show Pitch	669
18.7.13.17	Table To CSV	670
18.7.13.18	Table To Excel	671
18.7.13.19	Work Dir Manager	683
19	Variables	685
19.1	User input	685
19.2	Date and time information	685
19.3	Object information	686
19.4	Drawing information	688
19.5	Parameter information	688
19.6	Text operations for variables	689
20	Regular Expressions	691
21	Available Arithmetic Operations	694
22	Frequently Asked Questions	695
23	Copyright	697

1 Product overview



GENIUS TOOLS for Creo are a collection of extension modules for Creo Parametric.






These extension modules are contained in the products GENIUS TOOLS® Library, GENIUS TOOLS® Parameter and Startup TOOLS and cover different application scenarios optimally. The product package Startup TOOLS comprises all GENIUS TOOLS for Creo modules.









Take a look at the following overview of the different GENIUS TOOLS products.

Extension modules (GENIUS TOOLS for Creo)	Startup TOOLS	GENIUS TOOLS Paramet er	GENIUS TOOLS Library
Parameter management ("Parameter")	X	X	
Creating bills of materials in assembly mode ("Assembly Report")	X	X	
Editing assembly parameters	X	X	
Library management ("Library")	X		X
Importing external model data ("Library Data Importer")	X		X
Form-driven models ("Forms")	X		X
Form-driven UDFs ("UDF Forms - Design Tools")	X		X
Multiple dimension editing („Dimension")	X	X	X
Material selection ("Material")	X	X	X
Ring menu and Mapkey management ("Quick Access")	X	X	X

Extension modules (GENIUS TOOLS for Creo)	Startup TOOLS	GENIUS TOOLS Paramet er	GENIUS TOOLS Library
Transferring model properties (<i>"Value Transfer"</i>)	X	X	X
Name Generator	X	X	X
Inspection and change symbols for drawings (<i>"Inspect"/"Inspect Revision"</i>)	X	X	X
Converting multibodies into assemblies (<i>"Multibody to Assembly"</i>)	X	X	X
Further useful tools (<i>"Utilities"</i>) (3D Note Form, Open/Create Drawings, Export Table to Excel, Configuration Tool, Javascript Editor, Load Save Converter, Default Text Editor and many more)	X	X	X






2 Full details of all functions








Icon	Component	Subscription license
	Parameter management ³²⁷ ("Parameter") Parameter creates uniform meta data for use in an automated generation of bills of materials, as well as for precalculations and integration to ERP systems.	
	Creating bills of materials in assembly mode ("Assembly Report") <i>Assembly Report</i> outputs freely definable bills of materials in assembly mode and generates position numbers as component parameters which can be further used in Windchill, Creo View and Creo drawing mode.	
	Editing assembly parameters ⁴⁷⁹ This function generates component parameters in assemblies. Different component parameter values can be assigned for component models with the same name.	
	Library management ¹⁸¹ ("Library") <i>Library</i> is used to make Creo objects available from a library and to allow certain actions for each object such as copy or copy-paste into a model.	
	Importing external model data ("Library Data Importer") This component imports external model data, e. g. from PTC Windchill, into a library for GENIUS TOOLS Library. It is a program which has to be installed separately.	
	Form-driven models ⁸⁵ ("Forms") <i>Forms</i> generates user-defined form masks that allow Creo users to more quickly customize the properties of parts and assemblies (PRT/ASM).	


Icon	Component	Subscription license
	Form-driven UDFs / Design TOOLS⁴³⁷ ("UDF Forms") <i>UDF Forms</i> allows you to define features once in accordance with standards and to then conveniently place them in the design process.	
	Multi-dimensional editing⁷⁴ ("Dimension") The component <i>Dimension</i> enables the simultaneous and fast editing of dimensional values and names of a feature, component, assembly or its subcomponents.	
	Material selection²⁶⁷ ("Material") With <i>Material</i> Creo users can select materials based on various properties and assign them to a model or body.	
	Ring menu and mapkey management⁴⁰⁵ ("Quick Access") <i>Quick Access</i> is a ring menu that contains shortcuts to various commands in different Creo modes as well as individually configurable mapkeys (macros).	
	Transferring model properties⁴³⁰ ("Value Transfer") With <i>Value Transfer</i> numerous values in dimensions and parameters can be edited in one step in assembly mode.	
	Name generator³⁰⁶ This component assigns consecutive numbers to file names of parts, sheet metal parts and assemblies.	
	Inspection and change symbols for drawings¹²⁹ ("Inspect") <i>Inspect</i> lets you place, number, and manage inspection and change symbols on drawings. <i>Inspect Revision</i> stores all versions of inspection symbols and can create a revision history of all symbols.	X
	Converting multibodies into assemblies²⁹⁴ ("Multibody to Assembly") <i>Multibody To Assembly</i> allows you to transfer parts that have been created with bodies into an assembly structure.	X

Icon	Component	Subscription license
------	-----------	----------------------

Further useful tools⁴⁶⁹ („Utilities“)

	3D Note Form ⁴⁷⁰ Enables quick modification of dimension and parameter values in the notes on the model via editable form masks.	X
	Close all other windows ⁴⁷⁷ Closes all Creo windows except the current and the main window.	X
	Command Control ⁴⁷⁸ Hides Creo Parametric ribbon menu commands (all commands).	X
	Copy Component Parameter To Substitution Component Parameter ⁴⁸⁵ Copies value to the component parameter for a simplified part.	X
	Create Search.pro ⁴⁸⁷ Creates a project or assembly specific search path file (search.pro) based on customizable configuration settings.	
	Create Tolerance Tables ⁴⁹¹ Creates a tolerance table at a freely selectable location on a drawing using pre-defined tolerances.	
	CS Assembler ⁴⁹³ Automates the assembly of components into an assembly using defined coordinate systems.	
	Configuration Utility Provides an interface for editing all configuration options and saving them to the correct locations.	
	Default Text Editor ⁴⁹⁵ Creates a database of multilingual descriptions that can be used in multiple components to fill dialog boxes.	
	Export Points ⁴⁹⁸ Outputs reference points (single points or point fields) or dynamically generated curve points (X-Y-Z values) to a PTS or DAT file.	

Icon	Component	Subscription license
	Export Table One-To-One to Excel ⁵⁰⁶ Creates an Excel file without using a template.	
	Export table to CSV ⁵⁰⁹ Exports an existing drawing table or fills a template table.	
	Export Table to Excel ⁵⁰⁹ Fills a file template with parameters and data from cells of tables, e. g. of a Creo drawing table.	
	Extend Relations ⁵¹⁹ Adds more functions to model relations that can be used to create parameters for models and bodies.	X
	Extend Dimension Functions ⁵²⁴ Opens a dialog to quickly increase or decrease a selected dimension by a defined value.	X
	Full Backup ⁵²⁷ Quickly saves the current model with all dependent data.	X
	JavaScript-Editor ⁵⁷⁷ Enables faster developing and testing of JavaScript code. Opens in the respective component.	
	Load Save Converter ⁵³³ Saves Creo objects from previous Creo-, Wildfire- or Pro/ENGINEER versions in the currently used version.	X
	Open Base Model ⁵³⁵ Opens geometric base models that are the reference source for a feature.	X
	Open / Create Drawing ⁵³⁶ Opens a drawing if a drawing with the name of the model already exists, or creates a drawing.	
	Select Surfaces by Color ⁵³⁷ For selecting surfaces of the same color or all uncolored surfaces.	X
	Show Information ⁵³⁸ Creates company-specific text copy in the Creo Parametric main window.	
	Show Pitch ⁵⁴² Places a text specified in the configuration on a non-cosmetic thread.	

Icon	Component	Subscription license
	Toggle Symbol Variants ⁵⁴⁵ Switches between these variants of grouped symbols more quickly.	X
	Work Dir Manager ⁵⁴⁶ Automatically collects all directories used during the work process and enables the current working directory to be changed quickly.	X

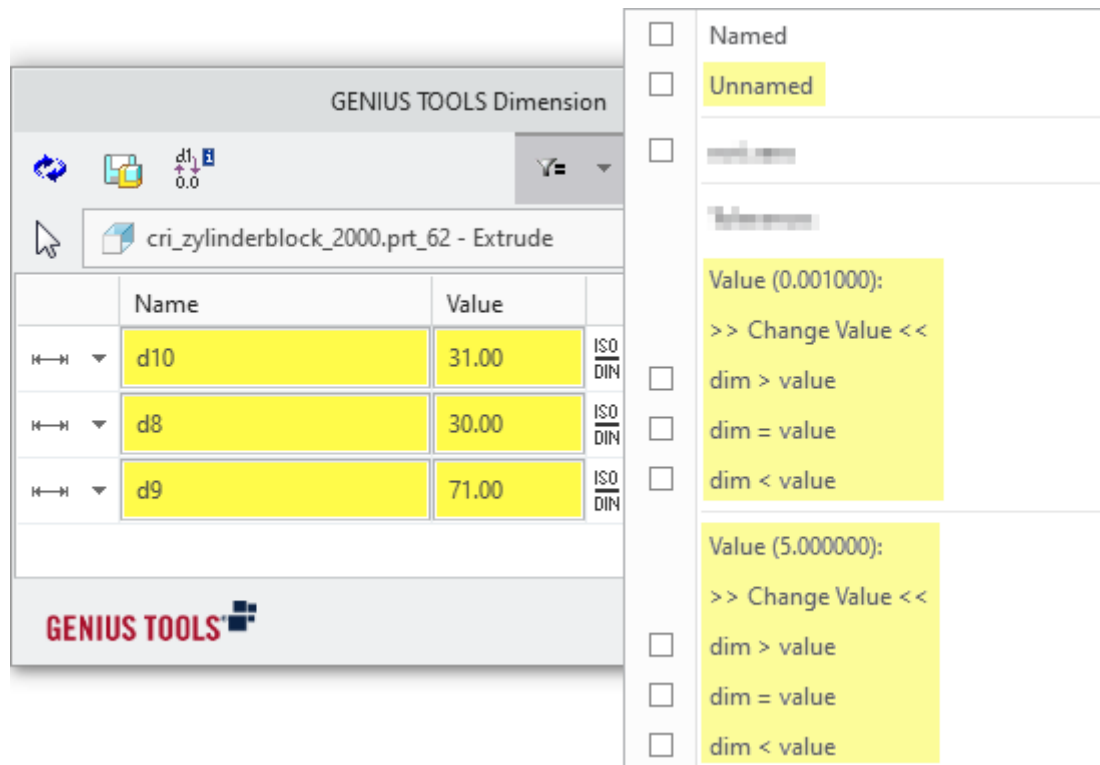
3 What's new in release 10

The GENIUS TOOLS add-on applications for Creo - included in the GENIUS TOOLS Library and GENIUS TOOLS Parameter products - support Creo Parametric version 10.

3.1 Dimension

Extended filter options (Version 10.0.1.0)

Filters have been extended to include *Unnamed* and the ability to search by value.



All dimensions are displayed that are equal to, smaller than or larger than the selected value. It is possible to search for two different values, e. g. to search for a range of values. The values that can be filtered by are adjustable via <<Change Value>> in the user interface. Alternatively, the values can be set with the configuration options `gtd_filter_value` and `gtd_filter_value2`.

The filter options are only active if the configuration option `gtd_filter_additive=0` is set (default).

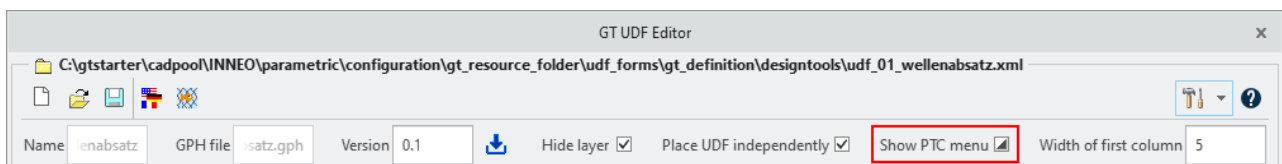
3.2 Parameter

New list „REST API“ (Version 10.0.1.0)

In GENIUS TOOLS Parameter Editor, you can set up a new value list for parameters: REST API. The REST API list function is available for PTC Windchill with 2.5 REST Services installed. BasicAuth authentication protocol is supported.


3.3 UDF Forms

UDF Forms: Checkbox for display of the PTC menu (Version 10.0.1.0)



A new checkbox *Show PTC menu* is available in the *GENIUS TOOLS UDF Forms Editor*. This checkbox is used to determine whether the PTC menu for selecting the assembly properties is displayed: *On* ☒ / *Off* ☐ / *Conditionally On* ☒. *Conditionally on* ☒ is selected by default. With this function, the menu for selecting assembly properties is displayed only if Creo Parametric decides that it is required for the particular UDF.

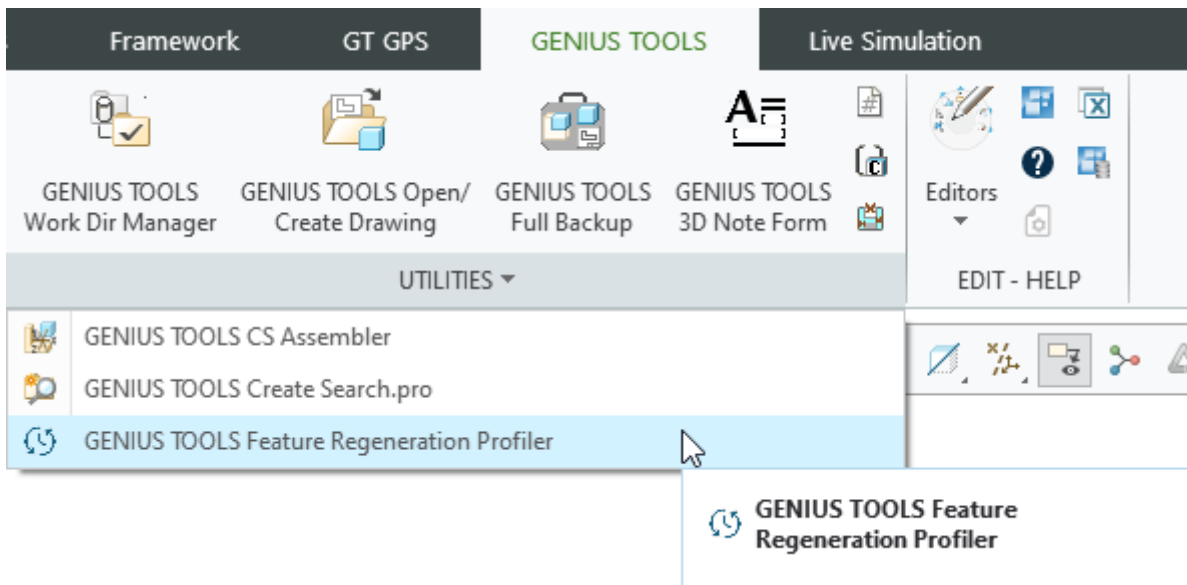
UDF Forms: Integration of Creo assembly functionality (Version 10.0.1.0)

The Creo function for standard assembly of a UDF is integrated in the user interface of *GENIUS TOOLS UDF Forms*. The function *Create new UDF group by Creo std. Dialog (values would not be transferred)*  is only available with the configuration option `gtuf_show_create_by_creo = 1` (Default: 0).

3.4 Utilities

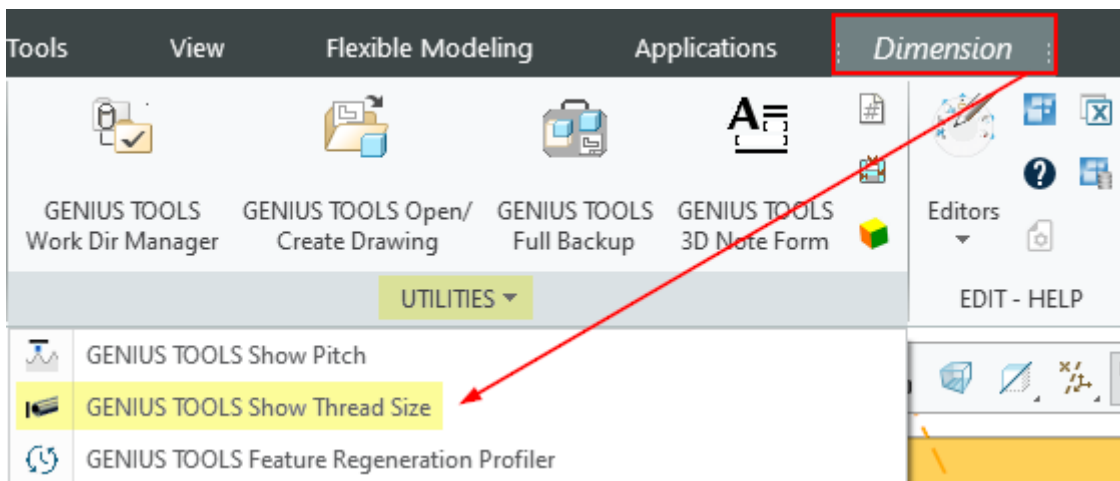
New function „Feature Regeneration Profiler“ (10.0.1.0)

With the *GENIUS TOOLS Feature Regeneration Profiler* you can regenerate your models and simultaneously display the times for the regeneration of each individual feature.



New function „Show Thread Size“ (10.0.1.0)

This function extends the displayed dimensions of a bore thread by the thread size.



3.5 Value Transfer

GENIUS TOOLS Value Transfer: Usage of boolean parameters (Version 10.0.1.0)

When using boolean parameters, the values *yes*, *true*, *1*, *ja* (engl. yes), and *wahr* (engl. true) are interpreted as *correct*. Any other input is interpreted as *false*.

3.6 Javascript

Excel support (Version 10.0.1.0)

GENIUS TOOLS Javascript Editor supports Excel with these new functions:

excelOpen

Opens a document from a path and allows access to the individual cells from this Excel table.

excelCellValueGet

Reads a value from a particular cell.

File selection and text query (Version 10.0.1.0)

There are three new functions in GENIUS TOOLS JavaScript Editor. In this context, the dropdown menu was adjusted to reorganize functions.

getStringUI("request", "default_value" (optional));

Queries a text.

openFileUI("*.extension", path (optional), title (optional), preselected_file (optional));

Opens a file.

saveFileUI("*.extension", path (optional), title (optional), preselected_file (optional));

Saves a file.

4 License-dependent functions

The following functions are limited to subscription licenses for either GENIUS TOOL® Library, GENIUS TOOLS® Parameter or Startup TOOLS.

Please note: The sum of all extension modules for Creo Parametric, which comprise these products, are called GENIUS TOOLS for Creo.



Extension module	Description	Release
Load Save Converter	Converts Creo objects from previous Creo-, Wildfire- or Pro/ENGINEER versions to the currently used version.	7.0.0.0
Inspect Revision	Stores all versions of inspection symbols on a drawing. Easily creates an overview of a revision history.	7.0.0.0
Open Base Model	Opens geometric base models that are reference sources for features with one click in the context menu of a feature.	7.0.0.0
Extend Relations	Adds functions to model relations that define parameters for models and for bodies.	7.0.0.0 - 8.0.0.0
Select Surfaces By Color	Surfaces of the same color can be selected with one click and be colored or otherwise modified thereafter.	7.0.1.0
CS Assembler	Automatically adds a number of components to an assembly by deploying a coordinate system (CS).	7.0.1.0
Extended Dimension Functions	Model dimensions can be increased or decreased very quickly with the mouse in a defined increment in the graphics window.	7.0.2.0

Extension module	Description	Release
Copy Component Parameter To Substitution Component Parameter	If a value has been assigned to a component parameter for a part in the master representation, this value can be copied to the component parameter for a simplified part.	7.0.2.0
Toggle Symbol Variants	If a grouped symbol has variants in the first level, you can switch between these variants more quickly.	7.0.2.0
Multibody to Assembly	Converts multiple bodies into assemblies (Creo Advanced Assembly Extension (AAX) is required.)	8.0.0.0
Work Dir Manager	Automatically collects all directories used during the work process and enables the current working directory to be changed quickly.	8.0.0.0
Full Backup	Quickly backs up the current model with all dependent data.	8.0.0.0
Command Control	With Command Control, Creo Parametric ribbon commands / commands can be hidden or deactivated.	8.0.1.0

Deactivating functions which require a subscription license

When using a permanent license, users will receive a warning that the functions listed above cannot be opened. You can avoid receiving such a warning by deactivating these functions, i. e. by setting the corresponding configuration option to 0.

Function	Configuration option
CS Assembler	gtu_start_csassembler
Copy Component Parameter To Substitution Component Parameter	gtu_start_copyCParamToSubsCParam
Extend relations	gtu_start_relationExtension
Extended Dimensions Function	gtu_start_extendedDimensionFunctions
Inspect Revision	gt_start_inspect_revision
Load Save Converter	gtu_start_loadSaveConverter
Open base model	gtu_start_openGeomOrigin
Select surfaces by color	gtu_start_selectSurfacesByColor
Toggle symbol variants	gtu_start_toggleSymbolGroups
Convert multiple bodies to assemblies	gt_start_multibody_to_assembly
Management of the used working directories	gtu_start_work_dir_manager
Model and data backup	gtu_start_fullbackup

Use **GENIUS TOOLS Configuration Utility**  and remember to save the changes in the main window with  and to click on *Reread Configuration* in the GENIUS TOOLS ribbon menu.

5 Assembly Report

GENIUS TOOLS Assembly Report supports you in creating reports, such as a bills of materials (BOM), in assembly mode. Reports are defined using a graphical editor. Any number of report definitions can be created as templates and applied to individual assemblies.

GENIUS TOOLS Assembly Report is available in assembly mode and for drawings of assemblies with the following features:

1. Table display with different display modes
 - single-level bill of materials
 - bulk bill of materials
 - multi-level bill of materials
 - multi-line cells
 - multiple parameters per cell
2. Usable column values
 - assembly and part parameters
 - assembly component parameters
 - report parameters (file name, assembly level, model type, quantity, mass etc.)
 - position number (as assembly component parameter)
3. Position number assignment (if used in the BOM)
 - general start and increment value
 - several number ranges in one report, definable with specifiable criteria (e. g. standard parts start at 500, foundation parts at 800)
 - manually editable
4. Multiple filtering according to all parameters
5. Multiple sorting according to all parameters
6. Export reports
 - to Microsoft Excel (with template and cell/column assignment)
 - to CSV files

5.1 Fundamentals

This section contains a glossary of terms and information on the standard behavior of GENIUS TOOLS Assembly Report.

5.1.1 Glossary

Single-level bill of materials

Single-level bill of materials that contains all elements of an assembly.

Head parameter

Information displayed above the report table in GENIUS TOOLS Assembly Report. Creo model parameters as well as general information such as number, file extension or level can be used as head parameters.

Bulk bill of materials

Unstructured bill of materials that lists each part of an assembly with the quantity with which it will be used in the final product.

Model parameter

Model parameters are parameters taken directly from the current Creo assembly.

Number range

A number range is a defined sequence in a group of elements to be ordered. Depending on defined filters in report definitions, number ranges are used to assign item numbers to assembly components.

Position parameter

The position parameter is created in assemblies and parts and contains a sequential number (position number).

Report

A report in GENIUS TOOLS Assembly Report is a bill of materials (BOM) displayed in the report table. Reports can be exported to various file formats.

Report definition

Template for creating a bill of materials that defines the numbering of items, the selection of parameters and the sorting. The information is stored in an XML data structure.

Report definition, external

Information for the report definition, which is stored in an XML file, usually in the resource directory.

Report definition, internal

Information for the report definition, which is stored in the data structure of an assembly as **embedded data** ^{s92}.

Multi-level bill of materials

Multi-level BOM that contains all elements of an assembly as well as all elements of sub-assemblies.

5.1.2 Standard behavior

GENIUS TOOLS Assembly Report supports you in creating various reports in Creo's assembly mode. The report definitions can be stored both internally (in the assembly data structure) and externally as XML files (e.g. in the resource directory of GENIUS TOOLS for Creo).


Whether internal or external report definitions are used depends on the configuration. External report definitions can be used for exchange and transfer to other assemblies.

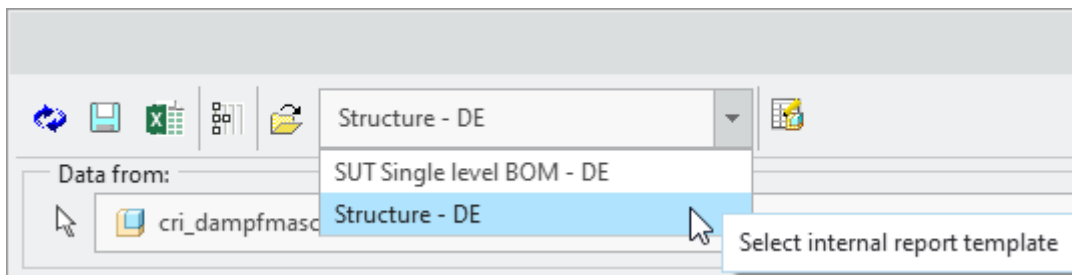
First use in an assembly

After starting GENIUS TOOLS Assembly Report the external standard report definition is used (defined in the configuration option `gta_default_file`). The same report definition is used for each assembly.

When saving a report, the report definition is saved in the assembly. Each time Assembly Report is opened with this assembly, the report definition saved in the model is used.

Subsequent use in an assembly

Each time Assembly Report is called, the assembly is searched for internally stored report definitions. You can select already loaded report templates via the drop-down menu. Additional report templates can be added via *Open external report template from file* . A selection of predefined templates is available there. You can use them or define your own template.



Changing the report definition

External usage

External report definitions are automatically saved in an assembly as soon as a report based on them is saved. Internal definitions of the same name are overwritten.

Warning: The defined name in the XML file is used, not the file name.

Internal usage

As soon as an internal report definition is applied, the display changes. If several internal report definitions exist, you can define a definition as the internal standard using GENIUS TOOLS Assembly Report .

Copy mechanism

If an external report definition is applied to an assembly, it is automatically copied and saved into the model. (See [Embedded data](#)⁵⁹².) This means that a model will be changed after changes to the report definition have been made, even without any changes on the model itself.

This behavior can be deactivated with the configuration option `gta_save_xml_in_md1`, i. e. report definitions remain available only in the external XML files. (Default: 1=activated)

Warning: The configuration option has no effect on report definitions already contained in the model.

Updating report definitions

If you make changes to external report definitions, GENIUS TOOLS Assembly Report can automatically apply these changes, see [Automatic update of report definitions](#).⁶²

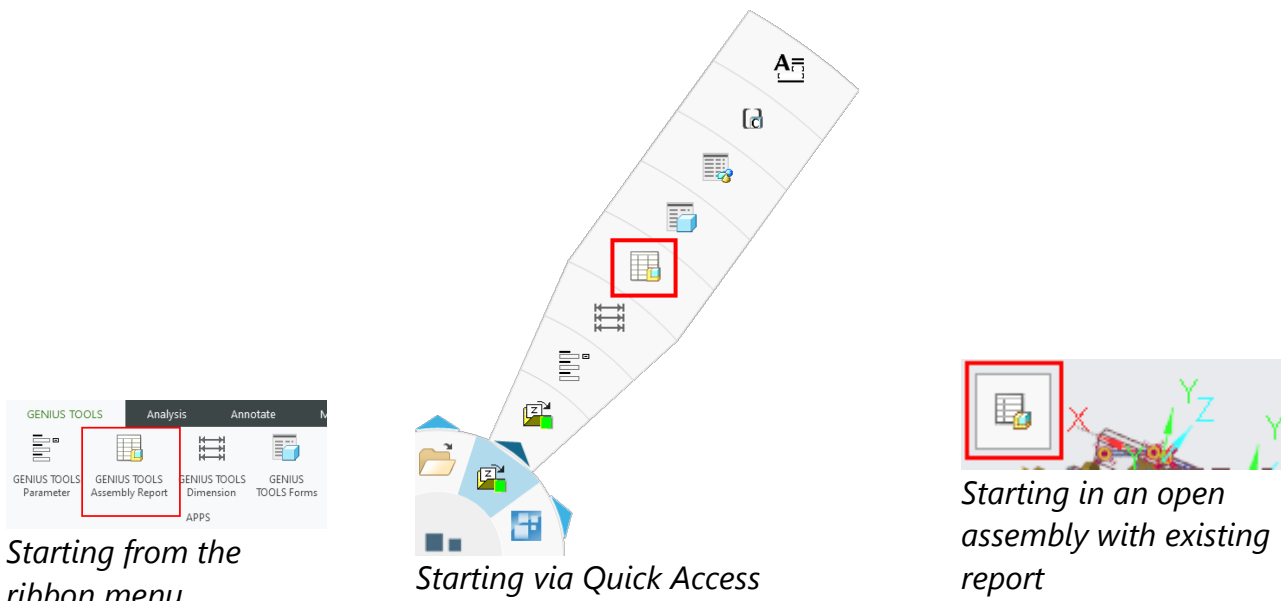
5.2 Usage

In this section you will find information about the use of GENIUS TOOLS Assembly Report. The general structure of the program is explained.

Starting the program

Start GENIUS TOOLS Assembly Report from the ribbon menu in the tab GENIUS TOOLS or by using GENIUS TOOLS Quick Access (key [`<`]).

Assembly Report is available in assembly mode and for drawings of assemblies.



Please note: The external standard report definition is always used first in an assembly.

(Configurable in the configuration option `gta_default_file`. Default: %

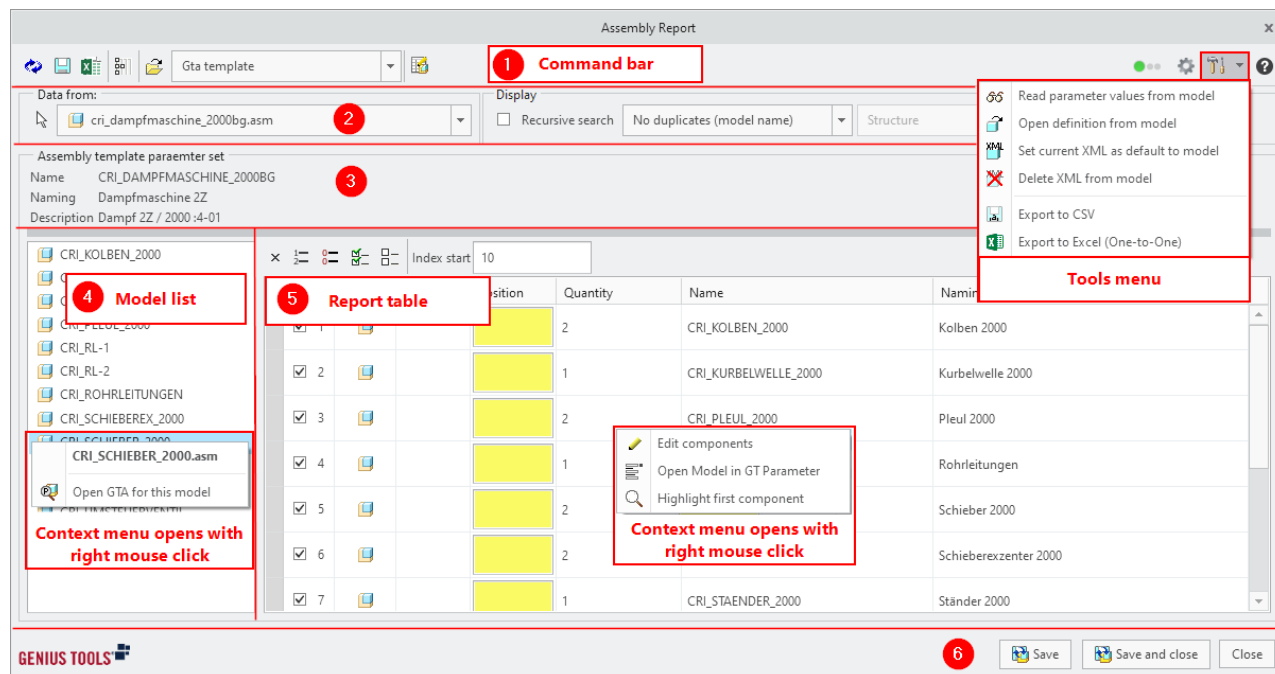
`GT_RESOURCE_FOLDER%\assembly\gt_assembly.xml`, Startup Tools: %`GT_RESOURCE_FOLDER%\assembly\gt_assembly_single_level.xml`.)

Warning: Changes to report definitions must be applied explicitly to already opened assemblies!

Report definitions are stored in assemblies (depending on the configuration option `gta_save_xml_in_md1`). Each time Assembly Report is started, the Assembly Report checks whether the current assembly contains a report definition. They are preferred for report generation.

5.2.1 User interface

The user interface of GENIUS TOOLS Assembly Report consists of the following elements:




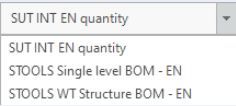







1. Command bar³⁰ with Report definition and Tools menu.³¹
2. Model selection and View³⁴
3. Head parameters: information of the current Creo model and the description of the configured report definition.
4. Model list³⁵: shows subassemblies
5. Report table³⁶
6. Save:³⁸ The Save button turns green to indicate change in the report table.

5.2.2 Command bar

The following buttons are included in the command bar:

Icon	Name	Description
	Discard changes and update	Discards all unsaved changes and reapplies the report definition.
	Save parameters	Saves changes and the report definition to the current assembly.

Icon	Name	Description
	Export to Excel ⁵⁰⁹	Opens <i>GENIUS TOOLS Copy Table to Excel</i> using a template. ⁶¹
	Create and display model tree	Regenerates the model tree and displays it in Creo Parametric. Parameters marked for the model tree in the editor are also displayed in the model tree.
	Open external report template from file ³⁴	Opens an external report definition (XML file) and applies it to the current assembly. If you select an external report template for an assembly, it is automatically copied to the model, i.e. it becomes the internal report template.
	Select an internal report template	Displays the internally stored report templates of the assembly. If no templates are displayed, check that the <code>gta_save_xml_in_md1</code> configuration option is set to 1.
	Open Editor ³⁹	Opens the Assembly Report Editor with the current report definition.
	Status indicator ³³⁸	Shows the current status for <i>loading</i> , <i>working</i> and <i>saving</i> with traffic light colors and opens the status dialog.
	Options ³²	Opens the options dialog for the current report definition.
	Tools menu ³¹	Contains various supporting functions.
	Help	Opens the help.

The **tools menu**  contains the following functions:

Symbol	Name	Description
	Read parameter values from model	Parameters are read again.
	Open definition from model	Opens and applies an internal report definition from the current assembly.
	Set current XML as default in model	Defines the current report definition as the default of the current assembly and saves it there (if not present).
	Delete XML from model	<p>Deletes a selectable report definition from the current assembly.</p> <hr/> <p>Please note: Report definitions are reapplied to an assembly when Assembly Report is reopened.</p> <hr/>
	Export to CSV	<p>Exports the current report to a CSV file.</p> <hr/> <p>Please note: An export to CSV uses all displayed rows.</p> <hr/>
	Export to Excel (One-To-One)	<p>Exports the current report without a template to an Excel file.</p> <hr/> <p>Warning: For Excel output an Excel version 2010 or higher must be installed! Make sure that no Excel instance is running when exporting a report!</p> <hr/>

The export settings to CSV and Excel can be defined as described in [Configuring export of reports](#).⁵⁹

5.2.3 Report options

In the options the settings of the selected parts list representation are displayed. These are set in the editor, see [General information](#).⁴¹

In the lower area export settings for this report definition can be changed quickly.

GT Assembly Report Options

Create component parameter

Change designation

Recursive

Duplictaes

No duplicates (model name)

Display

List

Start position

10

Position increment

5

Filter

Recursive filtering:

subType != 'PART_SKELETON_MODEL'

Sorting

POS: upwards

subType: upwards

DESCRIPTION_1_DE: upwards

Export

Export Rownumber

Export Modeltype

Export Creo position

Export position value

Open export csv-file

GENIUS TOOLS

Ok

Cancel

Export

Export line number, model type, position in Creo, position value

These four options define whether the line number, model type, Creo position or position parameter (POS) are output one-to-one when exporting to a CSV file or Excel spreadsheet. The default value for the four options is defined by configuration option.

Open CSV file after export

Defines whether CSV files are opened after an export (default program: Excel).

GENIUS TOOLS for Creo

© 2023 INNEO Solutions GmbH

33

5.2.4 Report definitions

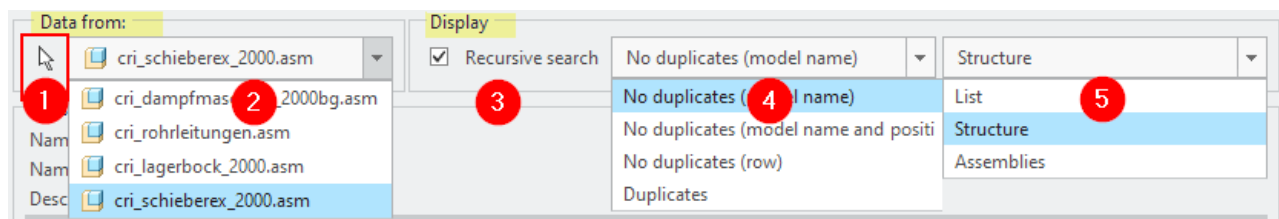
Note that you can select both external and internal report definitions (templates). If you select an external report template (XML file) for an assembly, it is automatically copied into the model, i.e. it becomes an internal report template.

You can customize any report definition supplied with GENIUS TOOLS for Creo or develop your own report templates, see [Use cases](#)⁵⁴.

5.2.5 Model selection and view

In this area you select a model and in which BOM type (single-level, bulk, multi-level) the components are displayed.

The **model selection** (Data from:) displays all models in a dropdown list that were selected by using the object selection.



Select model and how to display the report table

1. Object selection: Select subassemblies with the arrow function.

2. Report table: Shows the selected subassembly.

To get to the main assembly, the selected subassembly must be deselected by clicking again.

Tip: Assemblies selected in the model list are ignored. Only assemblies selected by object selection are displayed in the dropdown list.

In the **Display settings (BOM display)** you can order the contents of the report table by different criteria. The default settings are defined in [Assembly Report Editor](#)⁴¹.

The following settings correspond to the common parts lists. It is recommended to specify these settings in the report definition.

	Recursive search	Display mode	Duplicates
Bulk bill of materials	yes	List	No duplicates (model name)

	Recursive search	Display mode	Duplicates
Single-level bill of materials	no	List	No duplicates (model name)
Multi-level bill of materials	yes	Structure	No duplicates (model name)

3. Recursive search: The report table shows sub-assemblies and parts. If activated, you can select a display mode (5).

4. Duplicates: This setting determines how models with the same name and different item numbers are displayed.

- No duplicates (model name): Models with the same name are displayed in one row, one row is displayed per model name.
- No duplicates (model name and position): Models with the same name but different position numbers (index) are displayed in one row per position number.
- No duplicates (row): Only rows that have the same values in all columns are considered as duplicates and combined into one row.
- Duplicates: Always shows separate rows, regardless of whether rows share values.

5. Display mode: The report table can be displayed using different types of layout:

- List: flat list of all models
- Structure: Contained sub-assemblies and parts are displayed in indented groups under their parent assembly. The depth of the indentation per level is defined in the configuration option `gta_struct_insert_space` and can range from 0 to 10 space characters.
- Assemblies: Sort by assembly. The name of each assembly is displayed as a heading for the contained sub-assemblies and parts.

Please note: Changing the view can lead to a re-reading of the report table. Unsaved changes will be lost.

5.2.6 Model list

The model list displays subassemblies of the current assembly. If no subassemblies exist, it is hidden.







Click on an entry in the list to open a subassembly with the current report definition.

Use the **context menu**, which opens by right-clicking on a subassembly, to open it with an embodied report definition in a new Assembly Report window.

5.2.7 Report table

The report table displays a report on the current assembly of the model selection. This report depends on the current report definition and the display of it varies accordingly.

The following buttons are displayed in the **command bar** above the report table.

Symbol	Name	Description
	Deselect all lines	Deselects all models in the report table.
	Use position numbers from Windchill	Adopts existing position numbers from WTParts of the same name in the connected Windchill. Attention: Only available after activating the option in the editor and with connected Windchill. See Using position values from Windchill ⁶³ .
	Set position	Sets the position parameter according to the rules of the report definition for each model contained in the assembly. Only selected models are considered.
	Set position values to zero	Resets the position parameter for the models contained in the assembly. Only selected models are considered.
	Select All Components	Activates all rows in the report table.
	Deselect all components	Deactivates all rows in the report table.
	Starting position	Defines the initial value for numbering. This value is used when numbering the selected rows. It is not included in the definition.

Select the checkboxes in the *Row* column to select models for a report and automatic numbering.

Tip: Hold down the Shift or CTRL key and click in the gray area in front of the line number to select multiple lines. This allows values to be changed for several rows at the same time and the model activation can be changed too.

Row	Type	Creo Po...	Position	Quantity	Name
1		10	10	2	CRI_KOLBEN_2000
2		15	15	1	CRI_KURBELWELLE_2000
3		20	15	2	CRI_PLEUL_2000
4		25	25	2	CRI_SCHIEBER_2000

(1) Yellow input fields contain position numbers assigned several times for different components,

(2) Red input fields contain different position numbers of the same components.

If a parameter has a different data type than the one specified in the report definition, a message is written to the [status dialog](#)³³⁸. In some cases, the parameter value cannot be written to the model, e. g., if Assembly Report tries to write the item number to a string parameter.

Sorting the report table

Click on a column name to sort the report table in descending lexical order (1, 10, 11, 12, ..., a-z) by column. Another click sorts the table in ascending lexical order. A third click restores the original sorting.

Please note: If you want to use a different sort order, it must be configured in the editor.

Use Drag-and-Drop to rearrange individual lines. Click in the gray area left of the row number and drag the row to the desired position. Click the gray area left of the row number and press Shift or CTRL to select multiple rows and drag them to the desired position. Hold down Shift or CTRL while dragging.

Tip: Sort by row number to drag and drop reordered report tables to their initial state.

5.2.8 Saving

When saving, the POS parameter is written to the model.

It is possible to reset the POS parameter before saving, so that all components that were, for example, filtered manually from the parts list receive the default parameter value -1. To do this, activate the `gta_clear_pos_param` configuration option. The default setting is 0 (not active).

The Save button turns green when the report table has been changed.

5.3 Configuration

In this section you will find more detailed information about the structure of the Assembly Report Editor and how to create report definitions and Excel templates.

5.3.1 Assembly Report Editor

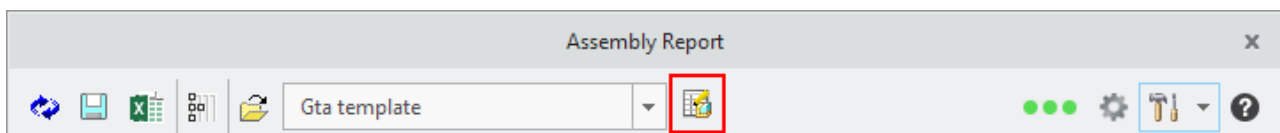
The editor is used to create and manage report definitions for GENIUS TOOLS Assembly Report. (See chapter [Use cases](#).⁵⁴)

Various parameters can be displayed in a report:

- [Model parameters](#)⁴⁴ are existing parameters and are read out.
- [Component parameters](#)⁴⁵ can be existing component parameters or can be newly created.
- [Report parameters](#)⁴⁶ are created solely for the assembly report.

Starting the program

Start the editor in the command bar of the GENIUS TOOLS Assembly Report dialog window.



Saving changes

The changes made in the editor are saved in the resource directory.

When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer (Caddepot). Path: `<operatingenvironment>/parametric/configuration/gt_resource_folder`.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

5.3.1.1 User interface

The user interface of GENIUS TOOLS Assembly Report Editor consists of the following elements:





1. Command bar⁴⁰
2. General Information⁴¹
3. Table parameters⁴²
4. Tabs: Position attributes⁴⁹ – Table view⁵⁰ – Table attributes⁵²
5. Save and close:

If this area is grayed out, changed data cannot be written into the model. Make sure that the configuration option `gta_save_xml_in_md1` is set to 1.

Warning: Take care not to open any more dialog boxes (Creo as well as GENIUS TOOLS), if GENIUS TOOLS Assembly Report Editor is open as it will be closed automatically.

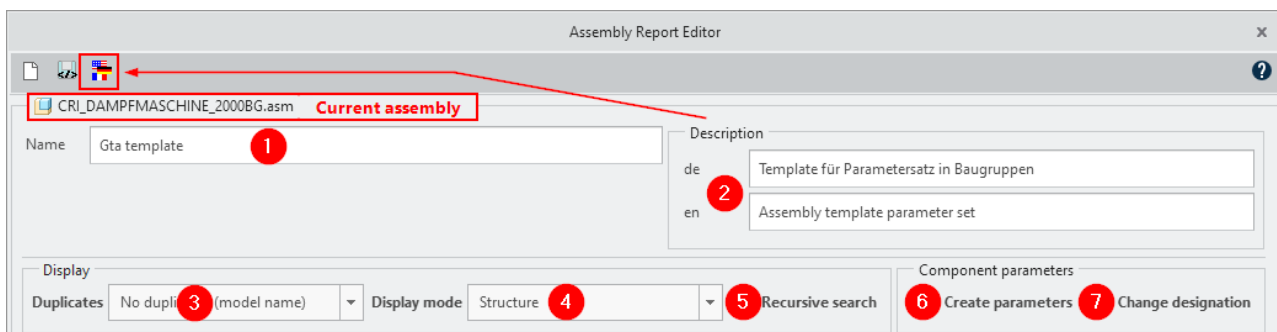
5.3.1.2 Command bar

The following buttons are included in the command bar:

Symbol	Name	Description
	Create a new empty report	Discards all entries and creates an empty report definition.
	Export report under a new name	Saves a report definition as an XML file under any name.
	Change used languages	Manages the available languages of a report definition.
	Help	Opens the help.

5.3.1.3 General information




The general information of the opened report definition is the following. The current assembly to which the report definition is applied is displayed above the name.

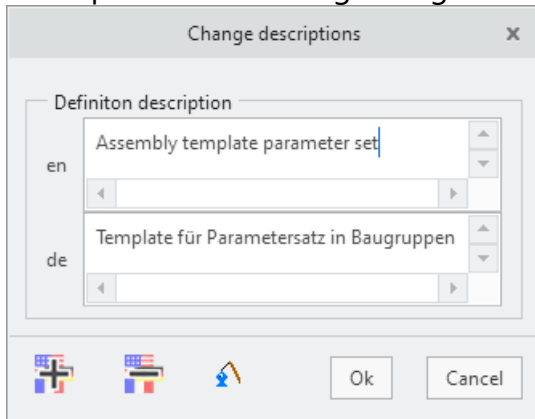


Origin of the report definition *Gta template from the assembly*
CRI_DAMPFMASCHINE_2000BG


1. Name: Specifies the name of the report definition as it will later be saved in assemblies.


2. Description: Specifies a description of the report definition in different languages. By default, a description is available in German and English.


The localized descriptions are managed via the languages button  in the command bar that opens the following dialog. Use the  symbol for default texts .



Display

These settings are default settings that can be changed in the Assembly Report dialog under **Display**. 

3. Duplicates: Defines how models with the same name but different position numbers are displayed, see **Assembly Report dialog**. 


4. Display mode: Defines the arrangement of the assembly components: List of all components, structure, division into the individual subassemblies, see **Assembly Report dialog**.  For selection, the checkbox *Recursive search* must be activated.

5. Recursive search: Sub-assemblies and parts are displayed in the report table. If activated, the selection of a display mode opens (4).

Component parameters

These settings apply to parameters of **assembly components**. 

6. Create parameters: Generates the assembly component parameters created for the report, if they do not already exist.

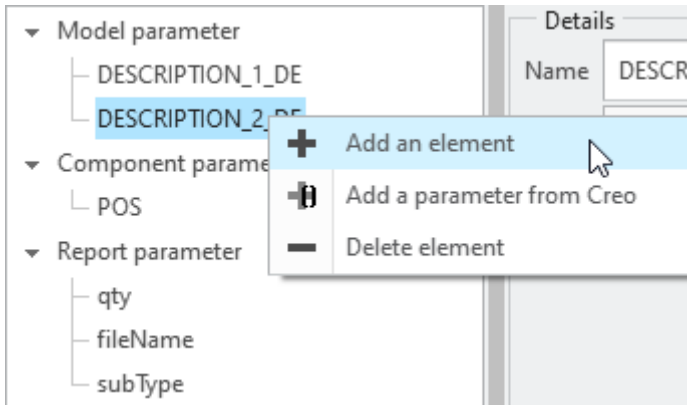
7. Change designation: Changes the reporting of existing assembly component parameters, according to the report definition that allows reporting for each **component parameter**. .

5.3.1.4 Creating table parameters

In the section *Table parameters*, you specify which parameters are displayed in the report table and how the parameter values are output.

Editing table parameters

The table view has its own context menu. Right-click on an item to open the context menu.



Add an element: Adds a parameter of the selected type with an arbitrary name. Parameters can be freely renamed.

Add parameters from Creo: Adds a parameter of the selected type from the Creo Parametric parameter selection.

Delete element: Deletes the selected parameter from the list.

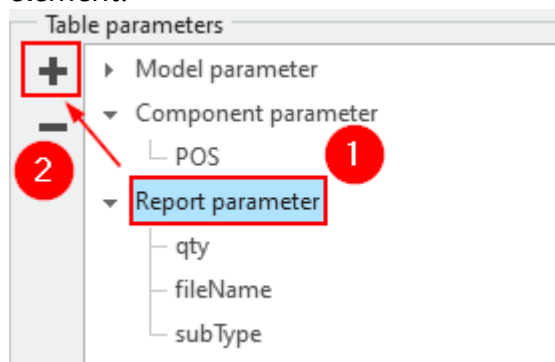
Please note: Be careful not to assign parameter names twice. Parameter names can be freely assigned in the table view.

Displaying table parameters in the report table

Add all parameters that you need for the report definition.

Procedure

1. Select one of the three types of table parameters (1): model parameter, component parameter or report parameter.
2. Add a parameter by using the Plus button (2) or the context menu entry *Add an element*.



3. Define how the parameter will be displayed, see [Defining table parameters](#)⁴⁴.

4. Move the created parameter to the list of Head parameters in the **Table view**⁵⁰ tab by clicking on it.
5. Add the Head parameter to a column in the **Table view**⁵¹ area.
6. Specify how the parameter values are to be filtered and sorted in the **Table attributes**⁵² tab.

A separate section of this chapter explains the procedure using an **example**⁶⁹⁴.

See the chapter *Use cases* for an **example**⁵⁴ of creating a report table with all table parameter types.

Defining table parameters

In the area *Details*, the selected table parameter is configured. The following information is required for all three parameter types:

The screenshot shows the 'Details' configuration window for a table parameter. It contains the following fields and elements:

- Name:** A text field containing 'DESCRIPTION_1_DE' with a red circle '1' next to it.
- Source:** A dropdown menu showing 'Model parameter' with a red circle '2' next to it.
- In model tree:** A checked checkbox.
- Width in tree and table:** A text field containing '20' with a red circle '3' next to it.
- Parameter title:** A section with two rows: 'de' with the value 'Benennung' and 'en' with the value 'Naming'. A red circle '4' is next to the 'de' row.

Name (1): The name of the parameter.

Source (2): Defines the type of parameter (**model parameter**⁴⁴, **component parameter**⁴⁵ or **report parameter**⁴⁶).

Width in tree and table (3): Defines the width of the parameter column in characters. Affects the Creo model tree and the column in the report table.

Parameter title (4): Defines the localized name of a parameter for the report table.


Defining types of table parameters

1. Model parameters

- are existing parameters from assemblies and contained models,
- are read.

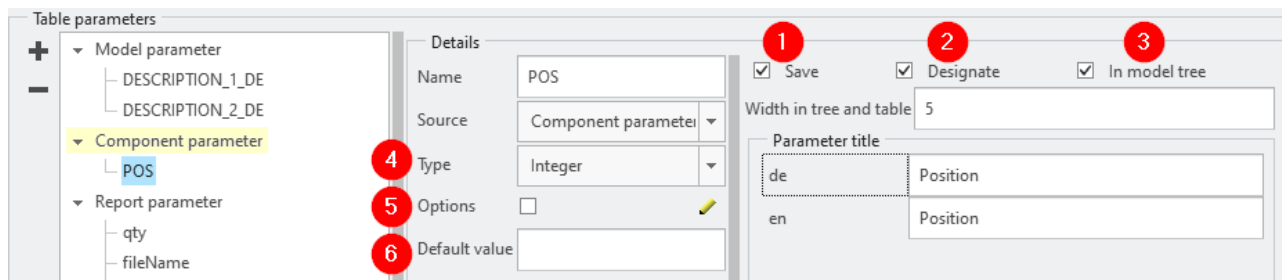
The screenshot shows the 'Table parameters' configuration window. It has two main sections:

- Table parameters:** A list on the left showing a tree structure with 'Model parameter' expanded, containing 'DESCRIPTION_1_DE' and 'DESCRIPTION_2_DE'. Below it are 'Component parameter' (with 'POS') and 'Report parameter' (with 'qty' and 'fileName').
- Details:** A configuration panel on the right for the selected parameter. It includes:
 - Name:** 'DESCRIPTION_1_DE'
 - Source:** 'Model parameter' (dropdown)
 - In model tree:** A checked checkbox with a red circle '1' next to it.
 - Width in tree and table:** '20'
 - Parameter title:** A section with 'de' (Benennung) and 'en' (Naming).

In model tree (1): Defines whether the parameter is also displayed in the model tree when *Create and show model tree*  is clicked.


2. Component parameters

- are parameters that are available in assembly components in assembly mode, e. g. position numbers,
- can be created or read out in the model if the component parameter already exists,
- are filled by manual input or predefined rules.



Save (1): Specifies whether to rewrite the value. If checked, the value can be edited and saved in *GENIUS TOOLS Assembly Report*.

Designate (2): Specifies whether to designate the value when creating it. If this option is checked, the information for designation is transferred to the parameter properties. Select the checkbox if you are working with Windchill.

In model tree (3): Defines whether the parameter is also displayed in the model tree when *Create and show model tree*  is clicked.

Type (4): Choose from the data types String, Integer, Double und Boolean.

The *Type* specification entered here replaces other specifications of the parameter. If this specification is not supposed to overwrite the properties of the parameter, set the configuration option `gta_update_component_parameter_type` to 0.

Type of parameter	Description
Boolean	Displays checkboxes.
Double, Integer	Allows the input of numbers.
String	Allows the input of text.

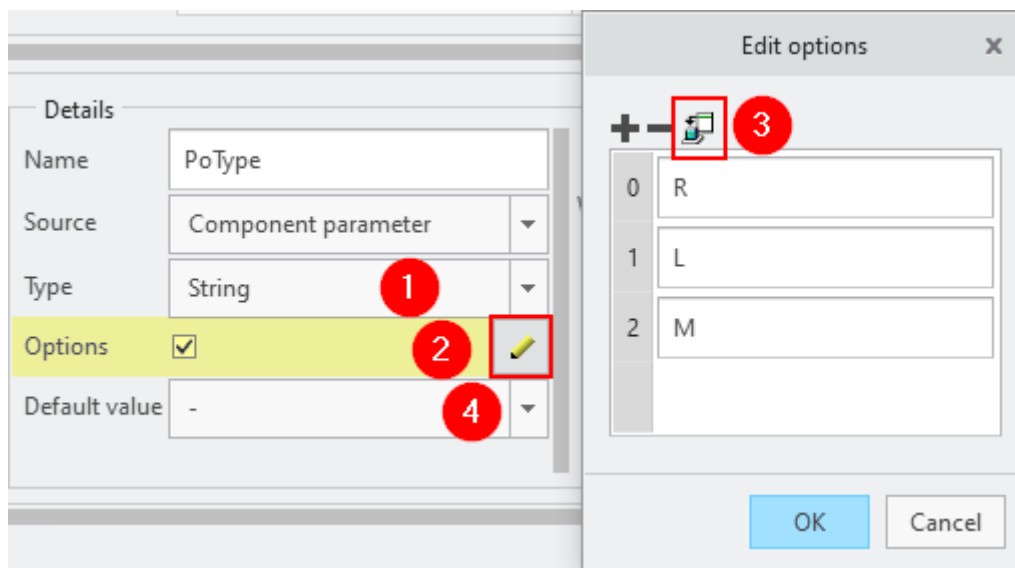
Changing the type of a component parameter

To change the data type of a component parameter, e. g. from integer to string, the configuration option `gta_update_component_parameter_type` must be set to 1 (default). This will update the data types of the component parameters as defined in the report definition.

Changing data types may be necessary, to unify component parameters that were created with different data types, for example when working with Windchill.

Please note: If you use the configuration option `gta_autoload_folder` to automatically update the report definition⁶², you must save changes in the report definition as an XML file. If this configuration option is set, changes that do not enter the XML file will not be transferred to the component parameter.

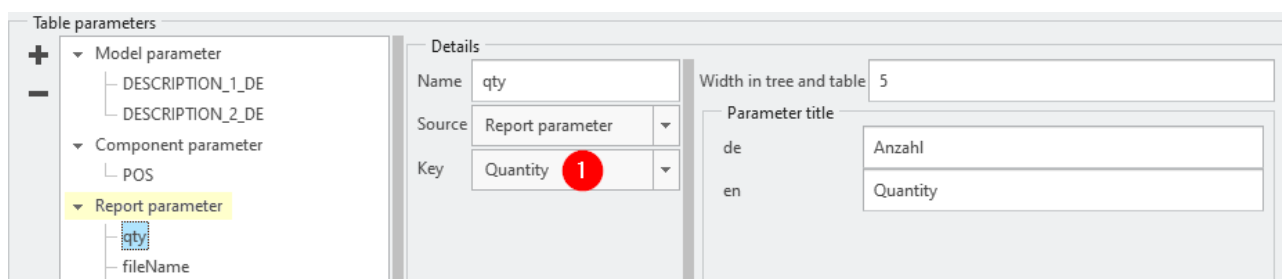
Options (5): Defines a selection list. Click the edit button (1) to define a list of possible parameter values available for users to choose from. The list can also be read in from a text file (2).



Default value (6): Defines a value with which a parameter is created in models. If the default value is empty, the parameter will be created empty.

3. Report parameters

- are properties of individual models such as number of assemblies or file name,
- are created for *GENIUS TOOLS Assembly Report*.



Key (1): Defines a property that is displayed in the parameter in *GENIUS TOOLS Assembly Report*. The following types of report parameters are available:

Key	Description
Quantity	Number of times the model is assembled.
File name	
Extension	
Subtype	
Mass	
Total mass	Mass of all components for the model as calculated by Creo
Sum	This parameter type for calculating values has been replaced by the type <i>Relation</i> . It is maintained only to ensure compatibility with existing report definitions.
Level	Assembly level
ASM skeleton	Skeleton: yes/no
Relation	Calculation of the value to be displayed. The calculation rule is entered in one line using JavaScript syntax. For this purpose, a separate input field <i>Rule</i> ⁴⁸ opens.
Generic	Name of the generic part
Feature-ID	ID number of the feature ID numbers are listed on one line separated by commas when duplicates are not displayed in the list, i. e. for the <i>View settings</i> ³⁴ <i>No duplicates (model name)</i> and <i>No duplicates (model name and position)</i> .
Parent model parameter	<i>Special case</i> ⁴⁸ : Reads model parameters from the parent model.
Is embedded	Checks if a model is embedded in an assembly.
Structure text	Numbered, hierarchical structure with indents. Relationships are immediately apparent.

Filling report parameters with a rule

The displayed value of a report parameter can be calculated with a JavaScript rule. To do this, specify the key *Relation* and enter a one-line JavaScript syntax.

- You can use arithmetic operations, the conditional operator (?) and logical operators.
- You can define whether the return value of the rule is a string, integer, number (Double) or Boolean value.
- You can use instructions in the notation `mdl.PARAMETER_NAME` to use parameters of the ROOT assembly (as in drawing tables with repeat regions).
- You can use instructions in the notation `asm.mbr.PARAMETER_NAME` to use component parameters (as in drawing tables with repeat regions).

Examples:

```
(rpt.qty > 2.0) ? 1.0 : 0.0
```

If the value for the quantity is larger than 2, the return value is 1, otherwise it is 0 (Boolean value).

```
asm.mbr.PTC_MATERIAL_NAME.indexOf("STEEL_COMMON") >= 0 ? "yes" : "no"
```

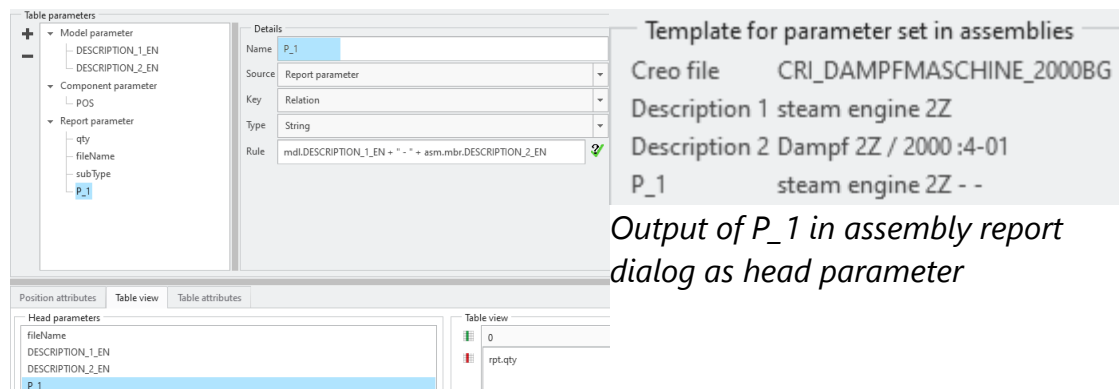
If the material *Steel common* is used, the rule returns *yes*, otherwise *no* (string value).

```
mdl.DESCRPTION_1_EN + " - " + mdl.DESCRPTION_2_EN
```

Outputs the descriptions 1 and 2 from the ROOT assembly.

```
mdl.DESCRPTION_1_EN + " - " + asm.mbr.DESCRPTION_2_EN
```

Outputs the description 1 from the ROOT assembly and 2 from the component, as shown in the example below:



Output of P_1 in assembly report dialog as head parameter

Mixed input for report parameter P_1 in the editor

Special case: Parent model parameter

When a report parameter of type Parent model parameter is created, a model parameter of the parent is read. The name specified in Details must be the model parameter name.

The Parent Model Parameter type (key) is useful to filter out subcomponents from the assembly report, for example, to display [purchased parts without subcomponents](#)⁶⁹.

5.3.1.5 Configuring report table

The table that makes up the assembly report is configured in the tabs **Position attributes**⁴⁹, **Table view**⁵⁰ and **Table attributes**⁵².

Start	Increment	End	Parameter

Assembly Report Editor: Lower segment with tabs

Tab: Position attributes

Define rules for assigning position numbers in the section *Position attributes*. The item number is assigned as a value to the position parameter in individual assemblies and parts.

Position parameter: In the dropdown list, select a parameter as the position parameter. Only parameters of type **component parameter**⁴⁵ are available.

Position PreSave rule: A Javascript PreSave function can be stored, the result of which is assigned to the position parameter. The function is executed for each line before the position parameter is written to the model. This rule is applied to each line in the report. The model remains unaffected.

This is opposed by the `gta_clear_pos_parameter` configuration option. If this configuration option is set to 1, the entire assembly is processed so that all components of the assembly are inheritantly set to position -1.

Example:

```
asm.mbr.cparam.POS <= -1 ? 0 : asm.mbr.cparam.POS
```


– Case 1 (POS <= -1):)

The position parameter receives the value 0.

– Case 2 (POS > -1):

The position parameter retains its value.

For a more detailed use case, see the **example**⁶⁵.

Use position numbers from Windchill: Shows an additional button  for the transfer of position numbers from WTParts of the same name directly from Windchill. See **Using position values from Windchill**⁶³.

Increment

Position start: Defines the start of the position numbers.

Position increment: Position numbers are incremented by the position increment.

Use of number ranges: Defines whether the rules defined under number ranges are to be used for item number assignment.

Number ranges

In the *Number ranges* area, rules for exceptions to position number assignment are defined. If the condition of a number range is fulfilled in a model, an item number is assigned according to the number range.

Each row of the number range table corresponds to an individual number range.

Use the plus button to add a row. Configure the number range afterwards.

Start: Defines the start of the position numbers in the number range.

Increment: Defines the increment of position numbers in the number range.

End: Defines the highest position number to be assigned.

Parameter: Displays selection parameters for a number range.

Create and configure one or more parameters in the Selection parameters table. A selection parameter is automatically added to the selected number range. The selection parameters and the stored conditions are used as filters for number ranges.

Parameter: Specifies a parameter (table parameter) to be used as a selection parameter.

Comparison: Select the comparison here (greater than, less than, equal to, not equal to).

Value: Defines the value with which the parameter is to be compared.

To remove a line, click on the line number and press the minus button.

Tab: Table view

Head parameters and report table are configured in the table view.

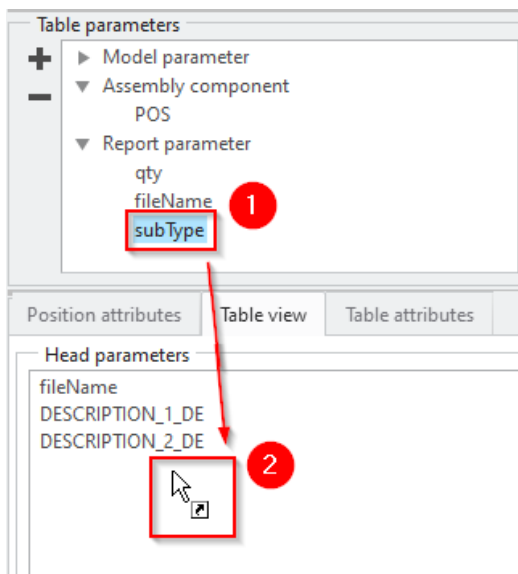
All parameters that are to be displayed in the report table must first be created in the section *Table parameters*⁴².

Head parameters

Parameters defined under Head parameters are displayed in the *report table*³⁶ area of Assembly Report.

Click on a table parameter (1) and drag it into the head parameters (2). To change the order of the parameters also use drag and drop.

Also use the context menu to manage the parameters.



Head parameters can be filled from report parameters.

Table view

The report table is configured to the right of the head parameters.

Configure the columns of the report table using the two buttons Add Column and Delete Column.

Click *Add Column* and drag and drop a table parameter into the column field. Alternatively, use the context menu to manage the parameters.

You can also display several parameters and static texts in a column. Drag several parameters one after the other into a column field or store static texts. Make sure that there is at least one space between parameters and texts!

Please note: As soon as a column for assembly components contains static text, a space or several parameters, no changes can be made in the column.

Table view

0	
rpt.qty/ rpt.Filename	1

Position	Quantity/ Name
	2/ CRI_KOLBEN_2000 2
	1/ CRI_KURBELWELLE_2000
	2/ CRI_PLEUL_2000
	2/ CRI_SCHIEBER_2000

Specify several parameters per column in the editor (1). They are evaluated in the Assembly Report table displayed (2).

Select a column header and click *Remove column* to remove unneeded parameters.

The check box *Add totals row* under the column configuration adds a row below the table that shows the sum of all values for each column that contains number parameters (Integer or Double). Each column for which you want to create a sum may contain only one Integer or Double parameter.

Please note: All displayed values in each column will be added up. In a structured list that can contain, for example, sub-assemblies and the parts used in them, parts may contribute multiple times to the value given in the totals row.

Tab: Table attributes

The *Table Attributes* section defines general report properties and filter and sort parameters.

Filter Parameters

The filter conditions of the report table are defined under *Filter parameters*. Only those parts and assemblies are displayed that fulfill these filter conditions.

1. Use the plus button to add a row. To delete a line, click on the line number and the minus button.

Inputting a filter and a condition

2. **Parameter:** Select a filter parameter that defines the filter condition.

Modellparameter

Komponentenparameter

Reportparameter

asm.mbr.PARAMETERNAME

asm.mbr.cparam.PARAMETE

prt.PARAMETERNAME
RNAME

3. **Comparison:** Select the comparison here (equal to, not equal to, greater than, less than).
4. **Value:** Defines the value with which the parameter value is to be compared.

Tip: Use Regular Expressions ³⁴⁹ in the value column!

5. **Recursive filtering:** includes subcomponents.

- Off (default): All components and assemblies that meet the filter conditions are filtered out from the Assembly Report.
- On: All components and assemblies that meet the filter conditions are filtered out with their subcomponents without checking the filter conditions on the subcomponents.

Please note: Recursive filtering is only relevant for the *display mode List*,³⁴ e. g. quantity BOMs. In the *Structure* display mode, subcomponents of assembly components that have been filtered out are not displayed.

A recursive search is useful, for example, to display purchased parts without their subcomponents, see [use case](#)⁶⁹.

Sorting Parameters

The table of sort parameters defines the parameter values according to which the displayed models in the report table are sorted.

The order of the parameters in this table determines the sort order. The above parameters are preferred.

Use the plus button to add a row. Then select a parameter from the drop-down list and select the sorting direction.

To remove a row, click on the row number and press the minus button.

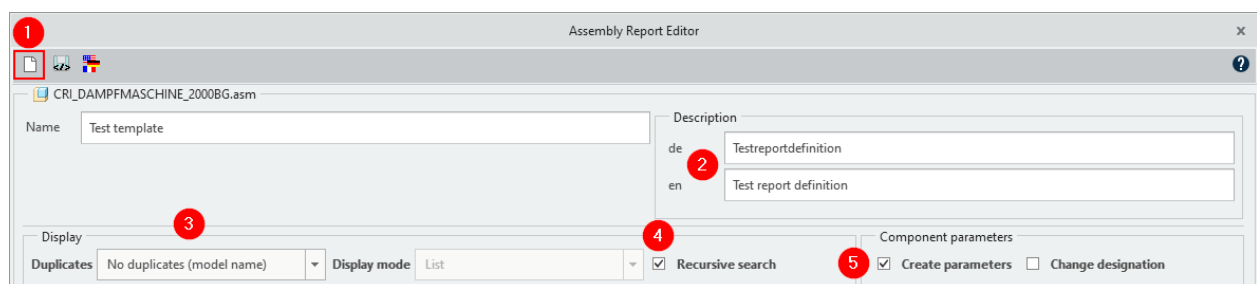
5.3.1.6 Use cases

GENIUS TOOLS Assembly Report reports can be customized for use within the company. Individual report definitions supplied with GENIUS TOOLS for Creo can be customized or custom report definitions can be developed. Creating your own report definitions using the GENIUS TOOLS Assembly Report Editor is explained in this example.

1. Open an assembly.
2. Start GENIUS TOOLS Assembly Report and open the Editor.

Creating a new report definition

1. Create a new, empty report definition.
2. Enter a name for the report definition.
3. Under *Duplicates*, select the option *No duplicates (model name)*.
This option ensures that item numbers of multiple assembled models are different.
4. Check the box *Recursive search*.
With this setting you ensure that all layers in a model are filtered and taken into account in the numbering.
5. Check the box *Create parameters*.
This setting ensures that the position parameter is created in each model.

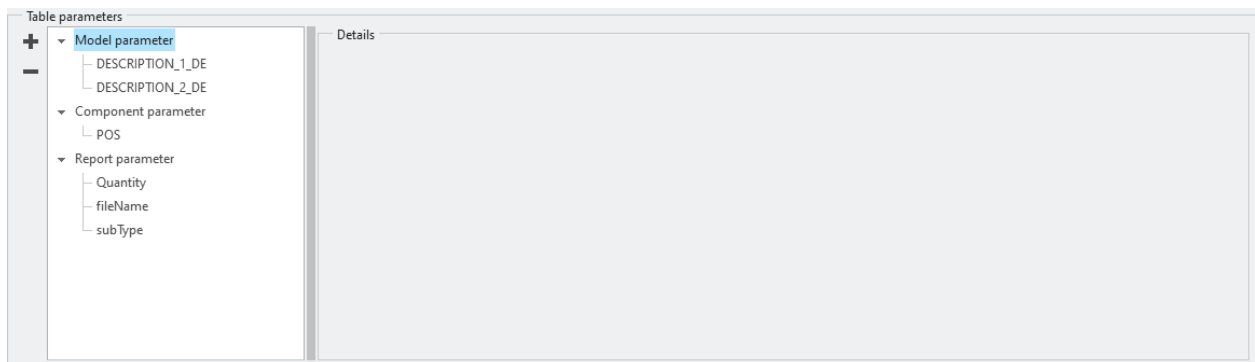


Creating table parameters

In the section *Table parameters* section of the report definition, add all the model, component, and report parameters you need for the report.

Please note: Any parameter to be used in other sections of the editor must be created in this section.

1. Add the model parameters::
 - DESCRIPTION_1_DE
 - DESCRIPTION_1_EN
 - DESCRIPTION_2_DE
 - DESCRIPTION_2_EN
2. Add the Component parameters::
 - POS
3. Add the Report parameters::
 - Quantity
 - fileNameDateiname
 - subType
4. After all parameters have been created, the section *Table parameters* in the GENIUS TOOLS Assembly Report Editor appears as follows:



Defining head parameters

Parameters in this list will be displayed later in the header of the report.

1. Select the model parameter DESCRIPTION_1_DE under *Table parameters*.
2. Drag and drop it into *Head parameters*.
3. Repeat the procedure with all model parameters.

Position attributes	Table view	Table attributes
Head parameters		
DESCRIPTION_1_EN		
DESCRIPTION_2_EN		
DESCRIPTION_1_DE		
DESCRIPTION_2_DE		

Tip: Parameters can be re-sorted with drag and drop.

Configuring the table view

In the next step, configure the table view of the report table. Use the two buttons on the left side of the *Table View* section to specify the number of columns displayed.

1. Create three columns. The first column is always counted as column 0.
2. Select the Report parameter *Quantity* under *Table parameters*. Drag and drop the parameter into column 0.

The screenshot shows the configuration interface for the table view. On the left, the 'Table parameters' section contains a tree view with 'Model parameter', 'Component parameter', and 'Report parameter'. The 'Report parameter' is expanded, showing 'Quantity', 'fileName', and 'subType'. A red circle with the number '1' is around 'Quantity'. A red arrow points from 'Quantity' to the first column (column 0) in the 'Table view' table. A red circle with the number '2' is around the arrow. The 'Table view' table has three columns labeled 0, 1, and 2. The 'Table parameters' section also includes a 'Details' section with 'Name: Quantity', 'Source: Report parameter', and 'Key: Quantity'. The 'Parameter title' section shows 'de: Anzahl' and 'en: Quantity'.

3. Repeat with the parameters *fileName*/column 1 and *DESCRIPTION_1_DE*/column 2. To make the name in reports bilingual, column 2 is now configured to display two parameters.
4. Drag and drop the parameter *DESCRIPTION_1_EN* into column 2. Separate the two parameters with a linebreak (Enter key).

2

asm.mbr.DESRIPTION_1_DE
asm.mbr.DESRIPTION_1_EN

Tip: You can also add normal text to parameters. Use a space between text and parameter!

2

DE: asm.mbr.DESRIPTION_1_DE
EN: asm.mbr.DESRIPTION_1_EN

Separate parameters from static text with a space character

Defining table attributes

Now, configure the table attributes. Table attributes determine which models are displayed in the report table.

1. Select the option *No duplicates* under *Duplicates*. This option ensures that position numbers of multiple built-in models differ.
2. Activate the options *Create parameters* and *Recursive search*. These settings ensure that the position parameter is created in each model and that models of deeper planes are filtered.
3. Add a filter parameter: Click on the Plus button and select the parameter *subType* from the dropdown list.
4. Select *diverse* under *Compare* and *PART_SKELETON_MODEL* under *Value*. This setting sorts out skeleton models.
5. Click on the plus button under *Range parameters* and select the *POS* parameter from the drop-down list. Select *Up* as the direction. Repeat this step with the parameter *subType*.

Position attributes | Table view | Table attributes

Filter parameters

Recursive filtering: ☐

	Parameter	Compare	Value
+			
-	0 rpt.subType	diverse	PART_SKELETON_MODEL

Sorting parameters

	Parameter	Direction
+		
-	0 asm.mbr.cparam.POS	Up
-	1 rpt.subType	Up

Configuring position attributes

Configure the position attributes:

1. Select the parameter *POS* as *Position parameter*.
2. Enter a start position of *100* and an position increment of *10* under *Increment*. Elements in the report table are now numbered consecutively in steps of ten starting at 100 (100, 110, ...).

Table parameters

- Model parameter
- Component parameter
 - POS**
- Report parameter

Detail

Name	Source	Type	Options	Default

Position attributes | Table view | Table attributes

Position parameter: **POS**

Position PreSave rule: R ? -1 : (asm.mbr.cparam.POS <= -1 ? 0

Increment

Position start	100
Position increment	10

Use number ranges ☐

Warning: Make sure that the correct position parameter is set! Position parameters must be of the type *Assembly Component parameter*. Saving must be activated for the parameter.

Defining number ranges

Create number ranges for exceptions to normal numbering:

1. Under *Position attributes* set the check mark at *Use number ranges*.

Increment

Position start: 10

Position increment: 5

Use number ranges ☒

2. For the definition of a number range, the tables *Number ranges* and *Range parameters* are available. With *Number ranges* you define the type of numbering. Use *Range parameters* to define the parameter you want to assign a number range to.

3. Add a new number range with the plus button under *Number ranges*.
4. Enter *Start*, *Increment* and *End*.
5. Add a comparison with the plus button under *Range parameters*.
6. Define the comparison.
7. After you have defined the range parameter, the defined parameter is automatically transferred to the table *Number ranges*.
8. Save the report definition.

Number ranges				Range parameters:		
Number ranges:				Range parameters:		
	Start	Increment	End	Parameter	Compare	Value
0	1000	1	2000	rpt.subType	equal	PART_SHEETMETAL

In this configuration, all sheet metal parts are numbered in steps of five from 1000 to 2000.

Result

Save the report definition and close the GENIUS TOOLS Assembly Report Editor. The created report definition generates the following report table:

Row	Type	Creo Pos.	Pos.	Quantity	Creo file	DE: Description 1 EN: DESCRIPTION_1_EN
✓ 1			10	1	CRI_RL-2	DE: Dampfleitung 2 EN: -
✓ 2			15	7	CRI_UEBERWURF_U_NIPPEL	DE: Anschlussverschraubung EN: compression fitting
✓ 3			20	1	CRI_RL-1	DE: Dampfleitung 1 EN: -
✓ 4			25	2	CRI_LAGERBOCK_2000	DE: Lagerbock 2000 EN: bearing block 2000
✓ 5			30	1	CRI_KURBELWANGE_MITTE_2000	DE: Kurbelwange mitte EN: crankweb center
✓ 6			35	1	RO-CRI_RL-2-0001	DE: - EN: -
✓ 7			40	2	CRI_GELENKSTUECK_SK_7	DE: Gelenkstück EN: joint piece
✓ 8			45	2	CRI_PLEULAUGE_O_2000	DE: Pleulauge oben EN: connecting-rod small end
✓ 9			50	2	CRI_PLEULAUGE_U_2000	DE: Pleulauge unten EN: connecting-rod big end
✓ 10			55	2	CRI_PLEULSTANGE_2000	DE: Pleulstange EN: conrod
✓ 11			60	2	CRI_PLEULVERBINDER_2000	DE: Pleulverbinder EN: piston rod connector
✓ 12			65	1	RO-CRI_RL-2-0002	DE: - EN: -
✓ 13			70	7	CRI_LOETNIPPEL	DE: Löt nipple EN: tab
						DE: Kurbelbolzen

5.3.2 Configuring export of reports

When reports are exported as a CSV file and as an Excel file via the [Tools menu](#)³¹, all data is exported as displayed in the user interface. All reports can be customized by the following configuration settings.

gta_export_file

Specifies the filename of the exported file. Take care to set the correct file name extension. If you want to export to different formats (e.g. CSV and XLSX), do not specify an extension in the configuration option. GENIUS TOOLS variables can be used.

gta_export_path

Specifies the directory where reports are saved. Default: working directory

CSV exports

gta_open_export_csv

Defines whether a CSV file is opened after export (1) or not (0). Default: 1

gtu_table_to_csv_export_sep

Defines the separating symbol between values in CSV files. Default: ; (semicolon)

You can define the way the CSV file will be encoded with:

gtu_table_to_csv_write_file_as_utf8

Defines whether the CSV file is written as ASCII (0) or UTF8 (1). Start value: 1

gtu_table_to_csv_write_file_as_utf8_with_bom

Defines whether a CSV file written with UTF8 is additionally encoded with Byte Order Mark (BOM) (1) or not (0). Initial value: 0. For this, above option `gtu_table_to_csv_write_file_as_utf8` must be set to 1.

Simple Excel exports

gtu_table_to_excel_open_export

Defines whether an Excel file is opened after export (1) or not (0). Default value: 1

Warning: For the output to Excel, an Excel version 2010 or higher must be installed on the executing computer!

If you want to use an Excel file with macros (XLSM), also adjust the security settings in Excel to allow macros to run.

Reports with Excel templates

The Copy to Excel function in the `command bar`³⁰ requires a previously configured Excel template⁶¹.

gta_export_template

Specifies the Excel template to be used by default for the export, i. e. entered in the export dialog. Default: `%GT_RESOURCE_FOLDER%\assembly\gt_assembly.xlsx`

5.3.3 Creating an Excel template

The function **GENIUS TOOLS Export table to Excel** in the command bar of GENIUS TOOLS Assembly Report requires a previously configured Excel template for the export process.

Warning: The use of export templates for GENIUS TOOLS Assembly Report has been changed in version 8.0.2. and only works with specific GTA abbreviations in the template parameters. For existing templates, change the parameters in the comment fields accordingly, e. g. from %head2% to gta:%head2%.

Step-by-step guide:

1. Use a separate Excel file for each report template.
2. Create a new template, following the steps in chapter **Create template** ⁵¹².

Example:

A	B	C	D	E	F	G	H
1	Datname / filename:						STK-Typ / BOM type:
2	Benennung / Description 1:						Baukasten / Single level
3	Bezeichnung / Description 2:						Datum / Date:
4	Artikelnummer / Part number:						
5							
6	Position	Qty.	Part number Part no. Semifinished	Rev.	Description 1	Description 2 Description semifinished	Material Coating / Coloring
7							Mass Mass Σ
8							
9							

3. Take care to use the component acronym *gta* for head parameter in the header.

A	B	C	D
Name			GTA:
Description 1			gta:%head1%
Article number			

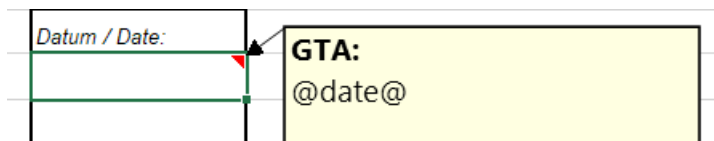
Head parameters with numeration

Please note: This function is called *Comment* until Excel 2019, from Microsoft 365 onwards it is called *Note*.

4. Take care to use the component acronym *gta* for modell parameter in the first row of the table columns.
Use *gta:%ruleParameter%* to output the position parameter. Enter *gta:%col*%* to output a column. Replace * with the column number.
Without the display of the position parameter, the counting of the columns starts at 0.

	Position	Quant	File name	Description
6				
7				GTA:
8				gta:%col3%

5. Use variables of GENIUS TOOLS for Creo⁵⁸⁵ if required, e. g. date and time information as well as Creo object information.



Please note: The variables `@feat_id@`, `@selmdl_n@` and `@selmdlpath@` are not supported.

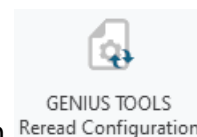
6. Save the template.

5.3.4 Automatic update of report definitions

Report definitions are stored in an XML file. If you make changes to it, GENIUS TOOLS Assembly Report will automatically apply the changes, if the configuration option `gta_autoload_folder` is set.

Procedure

1. Create a folder containing the current report definitions (XML files), e.g. folder update in the assembly folder of the resource folder.
2. Open GENIUS TOOLS Configuration Utility.⁵⁸⁸
3. Make sure, that the configuration option `gta_save_xml_in_mdl` is set to 1.
4. In the `gta_autoload_folder` configuration option, specify the path to the folder, for example, `%gt_resource_folder%\assembly\update`, and save the change.



5. In standby mode, read in the configuration using the button `Reread Configuration` in the GENIUS TOOLS menu ribbon.

Result

When GENIUS TOOLS Assembly Report is opened, the externally modified XML file is loaded. This saves the changes in the model, which can be read in GENIUS TOOLS Embedded Data.⁵⁹²

Tip: It is helpful to specify a version of the report definition. The version will be visible in the `embedded data`⁵⁹² in the model, but not in the Assembly Report dialog. Write the version in the XML file before the report definition name in the notation `gta version="version specification"`, for example:
`<gta version="0.2" name="Gta template">`

Example

Version 0.3 of report definition *Gta template* was changed and defined as version 0.4 (`gta version="0.4"`).

GENIUS TOOLS Display Embedded Data			
Name	Type	Version	Description
Gta template	GT Assembly	0.3	Single level BOM (ASM & PRT)
MECHANICA			
mfslot			
mppsot			
Delete Close			

GENIUS TOOLS Display Embedded Data			
Name	Type	Version	Description
Gta template	GT Assembly	0.4	Single level BOM (ASM & PRT)
MECHANICA			
mfslot			
mppsot			
Delete Close			

before starting Assembly Report

after starting Assembly Report

5.3.5 Using position values from Windchill

If you work with PDMLink and want to manage the information positions number (POS) there, connect to PDMLink as follows.

1. Creating the reusable attribute POS

To receive the information at check-in, the reusable attribute "POS" must be created in PDMLink. This attribute must be of the same type (INTEGER or STRING) as the component parameter used in BOM *FindNumber*.

2. Transferring POS to the EPMDocument level

In Windchill Type and Attribute Management the created attribute POS is assigned as a global attribute to the *CAD document usage link*

Aktionen ▾ Typ - CAD-Dokument-Verwendungs-Link

Eigenschaft	Wert
Interner Name	wt.epm.structure.EPMMemberLink
Anzeigename	CAD-Dokument-Verwendungs-Link
Beschreibung	EPM Document Uses Link

Attribute Layouts

Eigenschaften ▾

Name ↑	Interner Name
Name des Platzhalters für den Ident...	identifierSpaceName
OPTION1	OPTION1
OPTION2	OPTION2
OPTION3	OPTION3
OPTION4	OPTION4
OPTION5	OPTION5
t/f placed	placed
10 Pos	Pos
t/f required	required

The information POS is thus also available on the EPM level, and can be shown there as a column in all usage tables.

3. Forwarding POS to the WTPart level

The correct target position for the information position number on WTPart level is the column *LineNumber* or *FindNumber*.

From PDMLink version 10.2 M010 PDMLink can be configured in a way that the content from POS is also transferred directly to the desired target position on WTPart level:

3.1 Transfer FindNumber or LineNumber

The information can be transferred in two different columns. While *LineNumber* must be unique on each structure level, in *FindNumber* the same position number can be assigned to different components. We recommend to use *LineNumber*.

3.2 Passing POS to the WTPart level

To transfer the content of POS to *LineNumber* or *FindNumber*, the following must be configured in the settings manager:

- Instead of FINDNUMBER, use the name of the component parameter you are using, such as POS.

EPM Service Preferences			EPM service preferences
Build Service Preferences			Preferences used by build service
	Attributes Delimiter	,	This preference defines the delimiter character that separates the attributes to be published.
	Attributes to be published on Link		Attributes to be published on Link
	Attributes to be published on Master	*	Attributes to be published on Master
	Attributes to be published on Occur...		Attributes to be published on Occurrence
	Attributes to be published on Part	*	Attributes to be published on Part
	Build Image Association by Default	Yes	Build Image Association by Default
	Contributing Content Attributes	*	Attributes to be published on Part by Contributing Content relationship.
	Contributing Image Attributes	*	Attributes to be published on Part by Contributing Image relationship.
	Find Number Attribute	FINDNUMBER	Name of an attribute on CAD Document Uses Link that should be copied to the Part Usage's Find Number.
	Line Number Attribute		Name of an attribute on CAD Document Uses Link that should be copied to the Part Usage's Line Number.

- During check-in, the defined item numbers are then transferred to the desired target item. It is then equally available at the WTPart level, if applicable, and the position numbers are displayed in the defined columns.

4. Define usage in Assembly Report Editor

In the *Assembly Report Editor*³⁹, go to the tab *Position attributes* and check the box *Use position numbers from Windchill*.

The screenshot shows the 'Position attributes' tab in the Assembly Report Editor. It has two sub-tabs: 'Table view' and 'Table attributes'. The 'Table attributes' sub-tab is active. Below the sub-tabs, there is a 'Position parameter' dropdown menu set to 'POS'. To the right, there is a checkbox labeled 'Use position numbers from windchill' which is currently unchecked. A red arrow points from the 'Table attributes' sub-tab to this checkbox.

5.3.6 Use cases

This chapter collects some application examples for GENIUS TOOLS Assembly Report.

5.3.6.1 Removing single lines from parts list

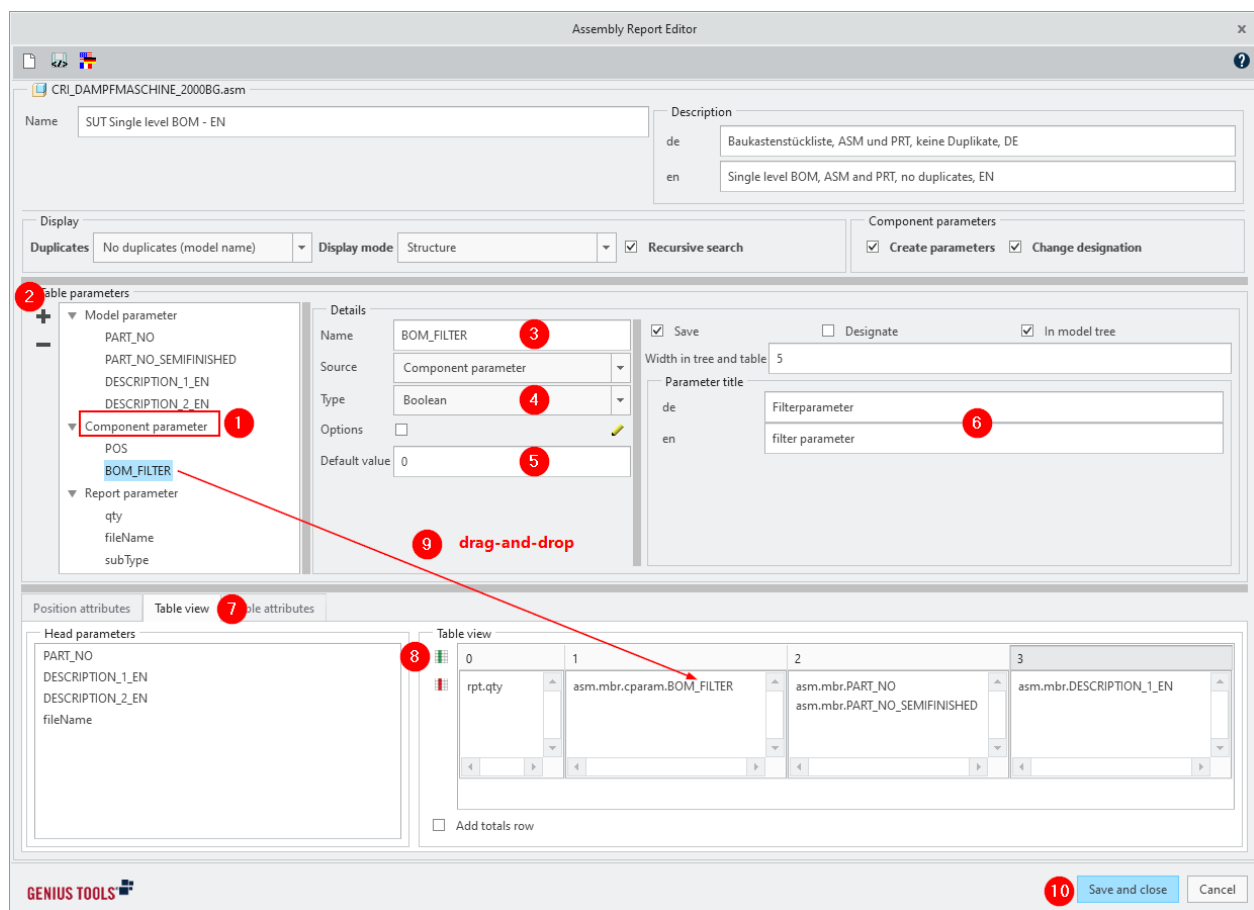
Using the component parameter BOM_FILTER, you can see an example of how a parameter is created and formatted. This example parameter adds another column to the assembly report. This column contains a checkbox in each row that is checked by default. If this checkbox is now unchecked in a row and this operation is saved, this table row is deleted. The deleted values are assigned the position number -1 in the background, see *Position PreSave rule*⁴⁹. An entry removed in this way can be *retrieved via component parameters*⁴⁸¹.

The following steps are implemented in the GENIUS TOOLS Assembly Report Editor. The procedure is adapted to the operating principle in Windchill.

Creating BOM_FILTER

1. In the area *Table parameters*, click *Component parameter*.
2. Click **+** to add a new parameter to this category.
3. Enter the desired name for this new component parameter.
4. Select *Boolean* as the type. This will generate a checkbox in the user interface.

5. Set the *Default value* to 0 so that the checkbox is ticked.
6. Assign a title for this component parameter. This title will appear in the Assembly Report in the selected language.
7. The newly created component parameter must be added to the table. To do this, click in the tab *Table view*.
8. To add an empty column, click in the column that you want to be to the right of the new column and click the *Add a Column* button.
9. Click the newly created component parameter and drag and drop it into the new, empty table column.
10. Confirm your settings by clicking *Save and close*. The GENIUS TOOLS Assembly Report Editor closes and you are back on the start page of the Assembly Report user interface.



Creating a component parameter and adding it to the table

Saving the preview of the table to validate it

1. You will now see a preview of the table as it will be output in the parts list. The orange column indicates that changes have been made. Confirm these changes with *Save*.

Assembly Report

Data from: cri_dampfmaschine_2000bg.asm

Display: ☒ Recursive search No duplicates (model name) Structure

Single level BOM, ASM and PRT, no duplicates, DE

Part no. 002021270
Description 1 steam engine 2Z
Description 2 Dampf 2Z / 2000 :4-01
Creo file CRI_DAMPFMASCHINE_2000BG

Index start 10

Row	Type	Creo Pos.	Pos.	Quantity	filter parameter	Part no. Semi Part no.	Description 1
<input checked="" type="checkbox"/> 1		10	10	2	<input checked="" type="checkbox"/>	002021269	piston 2000
<input checked="" type="checkbox"/> 2			15	1	<input checked="" type="checkbox"/>	002021215	cardan joint
<input checked="" type="checkbox"/> 3			20	1	<input checked="" type="checkbox"/>	002021269	piston 2000
<input checked="" type="checkbox"/> 4			25	1	<input checked="" type="checkbox"/>	002021237	piston rod
<input checked="" type="checkbox"/> 5			30	3	<input checked="" type="checkbox"/>	002021209	sealing ring
<input checked="" type="checkbox"/> 6		15	35	1	<input checked="" type="checkbox"/>	002021267	crankshaft 2000
<input checked="" type="checkbox"/> 7			40	2	<input checked="" type="checkbox"/>	002021225	cam
<input checked="" type="checkbox"/> 8					<input checked="" type="checkbox"/>	002021213	crank bolt

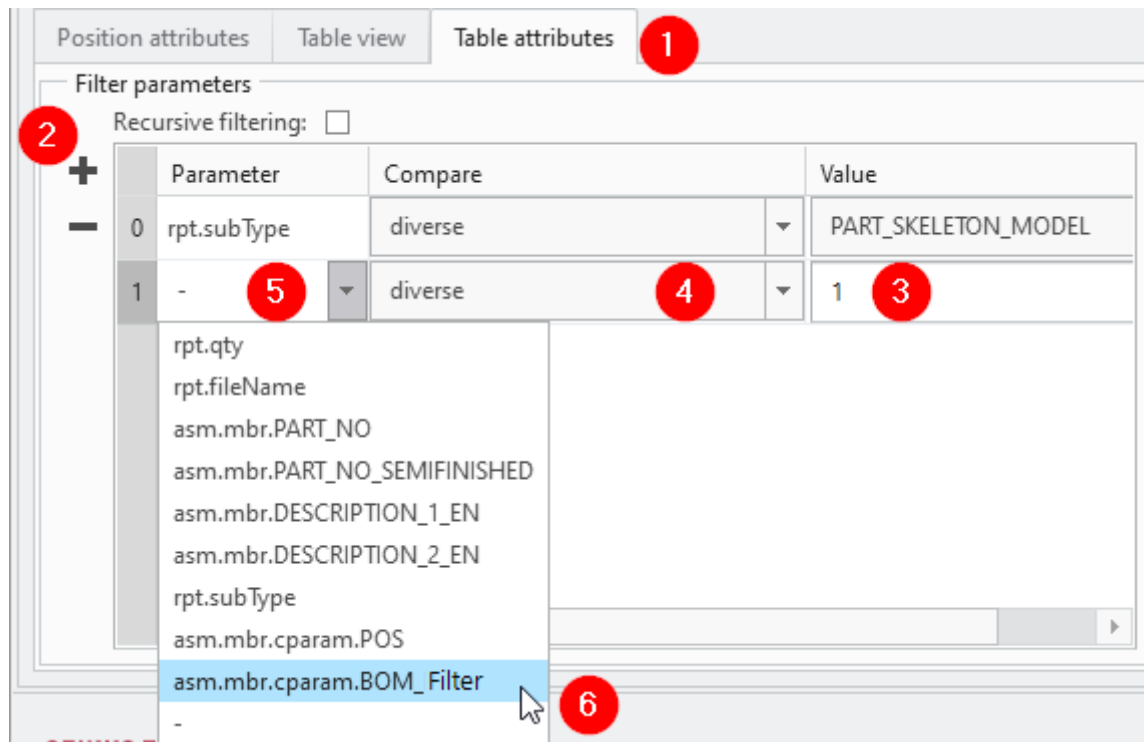
GENIUS TOOLS 1 Save Save and close Close

Preview of the parts list. The appearance of the table must be confirmed via "Save".

Adding filter parameters

Then continue working in the GENIUS TOOLS Assembly Report Editor.

1. Under *Table attributes*, define the criteria according to which the components of the model tree are filtered. All entries that match these filter rules are displayed in the parts list.
2. Use **+** to add further criteria.
3. Enter a value according to which filtering is performed.
4. Determine the comparison operation for this value.
5. Click in the cell to see a drop-down menu. In it you can select the component parameter you just created.



Defining a filter parameter

Applying the Position PreSave rule

1. Click *Position attributes*.
2. Enter this rule:

```
asm.mbr.cparam.BOM_FILTER ? -1 : (asm.mbr.cparam.POS <= -1 ? 0 :
asm.mbr.cparam.POS)
```

The default value for the position parameter of non-displayed components is -1. The configuration option `gta_clear_pos_parameter` defines that all components of the model are assigned -1. The Position PreSave rule entered here offers you the possibility to set -1 more selectively. For this, `gta_clear_pos_parameter` must be set to 1.



Set Position PreSave rule

Effects of the rule

This rule controls the position parameter depending on the component parameter BOM_FILTER (Boolean 0/1).

```
asm.mbr.cparam.BOM_FILTER ? -1 : (asm.mbr.cparam.POS <= -1 ? 0 :
asm.mbr.cparam.POS)
```

`BOM_FILTER` is the check that determines whether -1 or `(POS <= -1 ? 0 : POS)` is executed.

This results in the following possibilities how the value is set:

- **Case 1:** `BOM_FILTER` equals 0 (false) and `POS <= -1`
The position parameter receives the value 0.
- **Case 2:** `BOM_FILTER` equals 0 (false) and `POS > -1`
The position parameter keeps its value.
- **Case 3:** `BOM_FILTER` equals 1 (true)
The position parameter receives the value `-1`. This concerns entries that are not displayed.

5.3.6.2 Filtering out subcomponents of purchased parts

The following application example illustrates how purchased parts are displayed in a BOM with submodels (recursive BOM), but the submodels are sorted out.

Starting state

You see the model tree for the opened model and the Assembly Report. In both user interfaces, the parts that belong together are highlighted. These parts are considered in the example. The following steps are performed in the GENIUS TOOLS Assembly Report Editor.

1. Open the GENIUS TOOLS Assembly Report Editor.

The screenshot shows the GENIUS TOOLS Assembly Report Editor interface. On the left is the 'Model Tree' with columns 'POS' and 'Klassifikation'. On the right is the 'Assembly Report' table with columns 'Row', 'Type', 'Creo Posit', 'Position', 'Quantity', 'Name', and 'Classification (Parent model)'. Red boxes highlight specific parts in both views, connected by red lines.

Row	Type	Creo Posit	Position	Quantity	Name	Classification (Parent model)
1		0	0	1	RAHMEN	Fertigung
2		5	5	1	OBERROHR	Fertigung
3		10	10	1	UNTERROHR	Fertigung
4		15	15	1	SITZROHR	Fertigung
5		20	20	1	SITZSTREBE	Fertigung
6		25	25	1	KETTENSTREBE	Fertigung
7		30	30	2	RAD	Fertigung
8		35	35	2	REIFEN	Kaufteil
9		40	40	2	LAUFRAD	Kaufteil
10		45	45	32	FELGE	Fertigung
11		50	50	2	NABE	Fertigung
12		55	55	2	VENTIL	Fertigung

Bicycle with specification of position number and classification in the model tree and the corresponding lines in the GENIUS TOOLS Assembly Report

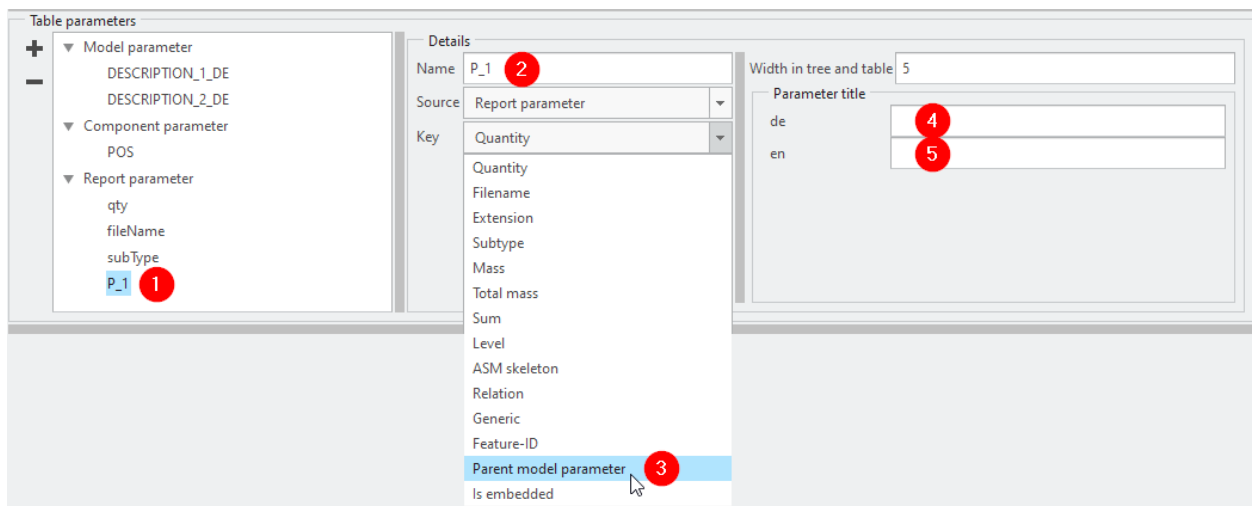
Creating a report parameter

1. Click *Report parameter*.
2. Create a new parameter by clicking *Add a parameter* +.



Defining a new report parameter as a parent model parameter

1. Select the created report parameter with a click.
2. Set a name.
3. Use the drop-down menu to select a key, in this case *Parent model parameter*.
4. Specify under which column title the new report parameter will be listed in the parts list - in German
5. and English.



Creating a report parameter in the Assembly Report Editor, with intermediate steps.

Report parameter filled in completely

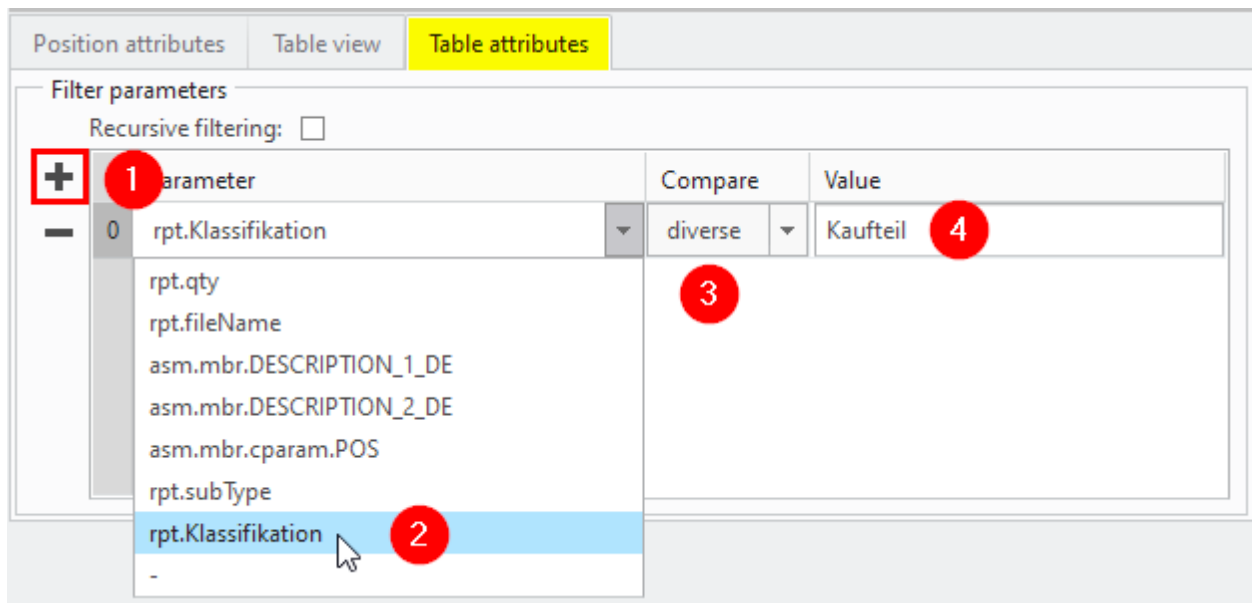
Adding a new report parameter to the report table

1. To add a new empty column, click in the column you want to be to the right of the new column and click the *Add a column* button.
2. Click the newly created component parameter and drag and drop it into the new empty table column.

Adjusting the filter

1. Under *Table attributes*, use **+** to add a new filter.
2. to 4. Define the filter that will be applied for classification.

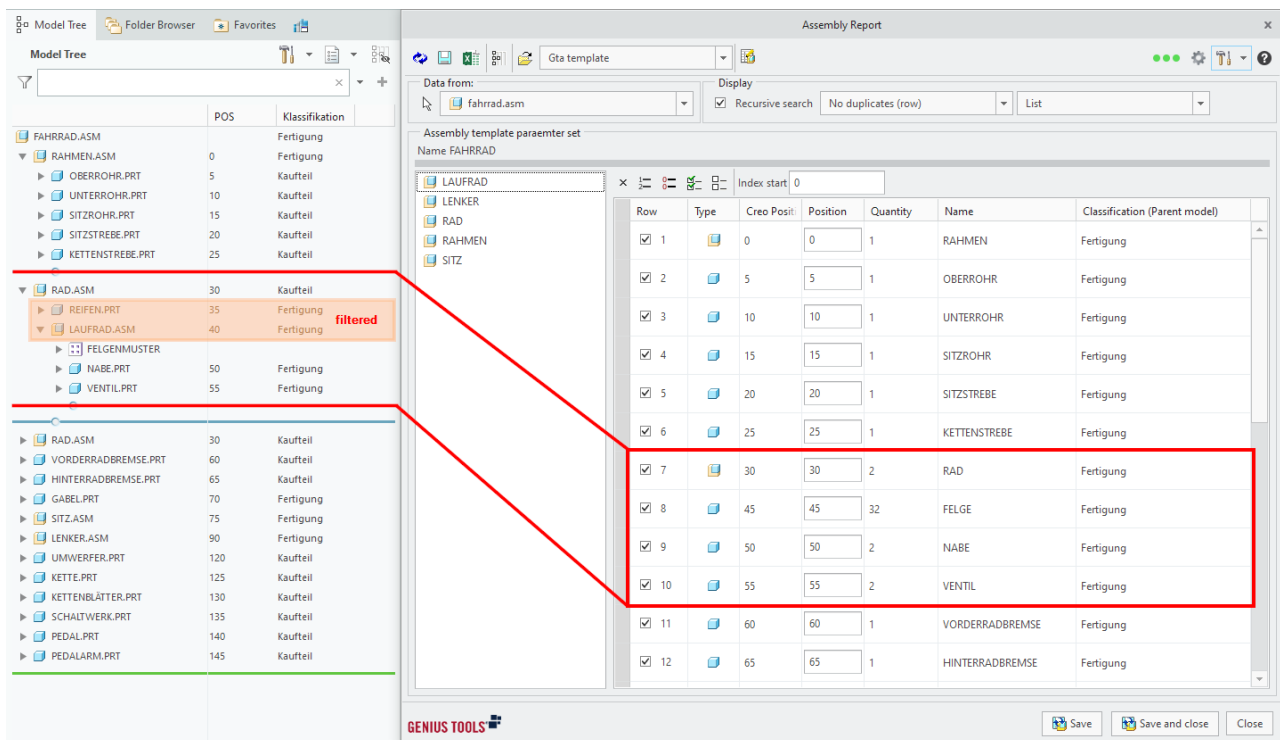
The filter defined in this case specifies that all models whose parent models are not purchased parts are included in the parts list.



Creating a filter rule for classification

Interim result

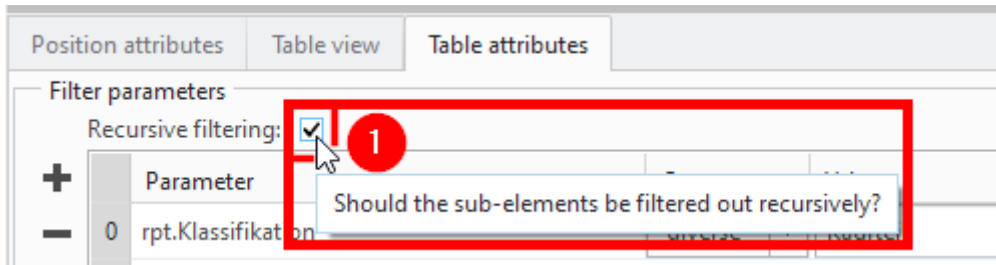
Comparing Model Tree and the table in the Assembly Report, you can see that two entries have been filtered out. However, the models on the subsequent lower layers were not filtered out, although NABE.PRT and VENTIL.PART are not purchased parts either. Therefore, the filter must be adjusted.



Interim result of filtering via classification

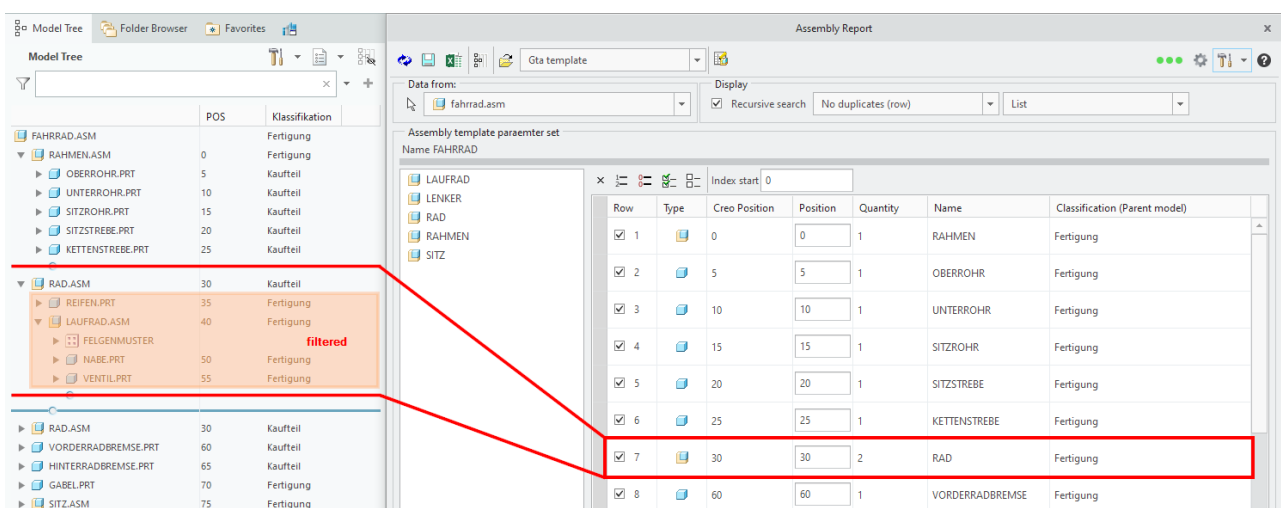
Setting filter to Recursive

In the Assembly Report Editor, under *Table attributes*, activate the *Recursive filtering* function by ticking the check box. This will apply the filter rules to all sub-elements as well.



Result

Now compare the model tree and the assembly report. You will now see that all models in the orange area have been filtered out. Note that *Classification* and *Classification (parent model)* are different parameters. *Classification (parent model)* is a report parameter that is useful for filtering.



Final result with all filtered out models

6 Dimension

With GENIUS TOOLS Dimension you can quickly, clearly and comfortably edit dimension values and names of parts and assemblies, features, feature groups and variable UDF dimensions.

GENIUS TOOLS Dimension is available in assembly, part, sheetmetal and drawing mode, providing the following features:

1. Display dimensions with following properties:
 - Dimension type (linear, angle, diameter, radius)
 - Dimension name
 - Dimension status (in relations, family tables etc)
 - Dimension value
 - Tolerance type
 - Boundaries
2. Filter displayed dimensions by name, dimension type and tolerance type
3. Dynamic search for dimensions
4. Highlighting dimensions in the graphics window when selecting a value in the GENIUS TOOLS Dimension window
5. Easy renaming of dimensions
6. Assigning new dimension values (including mathematical expressions)
7. Quickly displaying the original Creo dialog "Dimension properties"
8. Quick call-up of the relations dialog (for relation-driven dimensions)
9. Quickly assigning dimensions to family tables
10. Saving the values as CSV-file

6.1 Starting the program

Start GENIUS TOOLS Dimension from the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).

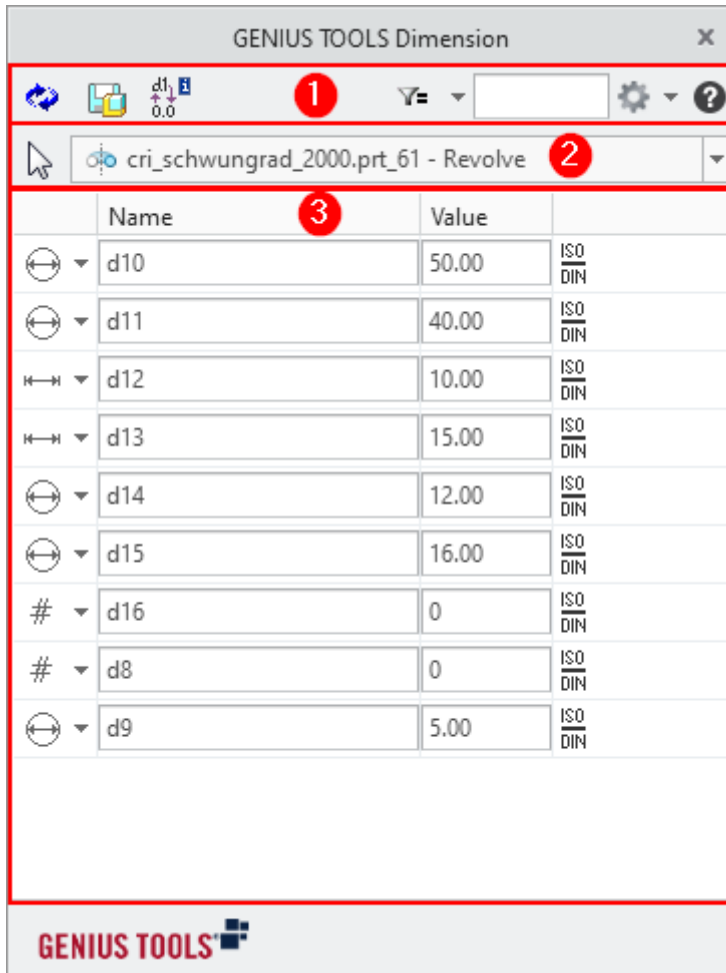


GENIUS TOOLS Dimension starts only after selecting an element in the Creo main window or in the model tree. Alternatively, an element can be selected previously.

Please note: When exiting Dimension, only changes with valid inputs will be saved.

6.2 User interface

The user interface of GENIUS TOOLS Dimension consists of the following elements:

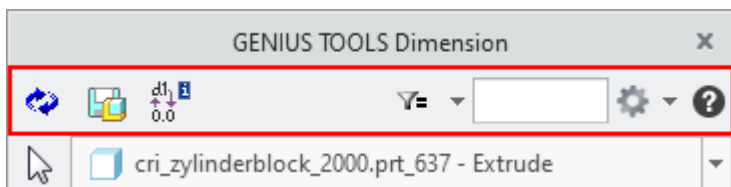


1. Command bar with filter
2. Model selection
3. Dimensions table








The user interface height is calculated by the amount of shown dimensions. Maximum 1/2 of screen size. Minimum 4 lines. The width is calculated by the combined width of the shown columns.

6.2.1 Command bar

The command bar consists of general control elements and the filter area with search field.



The following elements are available in the command bar:

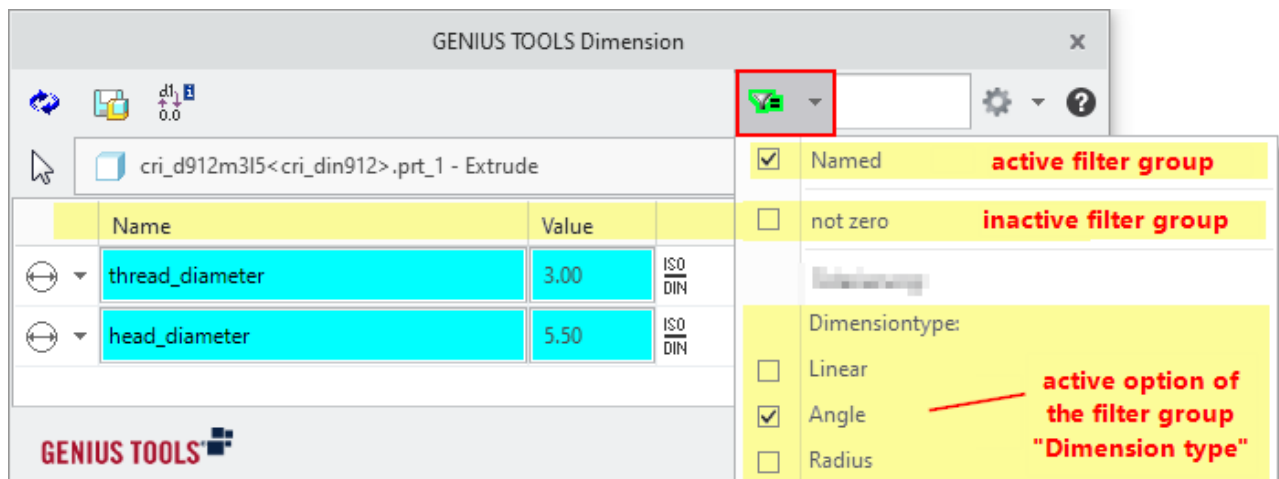
Icon	Name	Description
	Reload	Reloads the currently selected model or feature. Unsaved changes will be discarded.
	Save and regenerate	Saves all changes made and regenerates the whole model. Note: It is not possible to save with invalid (highlighted red) input fields or not performed calculations.
	Change dimensioning display	Changes the display style of dimensions in Creo Parametrics between dimension value and dimension name.
	Filter inactive/active	Opens the list of available filter options. If filter options are already active, the icon is displayed green.
	Search	Search field to search for names and values.
	Tool menu	Selection to open a family table from current model or save as CSV file with comma or semicolon.
	Open Help	Opens the Help.

6.2.2 Filter and search

The filter controls the display of dimensions with predefined options. Click on the filter icon and select the appropriate filter options.

Filter options

The filter controls the display of dimensions with predefined options. Click the filter icon and select the appropriate filter options.



Filter behavior

The selected filter options are linked with each other ("AND operation"), i.e. in the example all named measures are displayed that correspond to the measure type Angle.

Within a group, all dimensions that match the active filter options are displayed.

The configuration option `gtd_filter_additive=1` can be used to change the filter behavior so that the results of all active filters are displayed individually. That means, in the example above, all named dimensions as well as all angle dimensions would be displayed ("OR-operation").

Filter groups

- Named dimensions / Unnamed dimensions / name filter: Shows all dimensions with modified names.

d30 Unchanged name of a dimension

WIDTH Modified name of a dimension, a named dimension

- not zero: Shows all dimensions that have a value disparate from zero.
- Tolerance: Shows all dimensions that correspond to the selected tolerances.
- Dimension type: Shows all dimensions that correspond to the selected dimension types.
- Boundaries: Shows all dimensions that correspond to the selected limits.
- Value (0.001) and Value (5.000): All dimensions are displayed that are equal to, smaller than or larger than the selected value. It is possible to search for two different values, e. g. to search for a range of values. The values that can be filtered by are adjustable via <<Change Value>> in the user interface. Alternatively, the values can be set with the

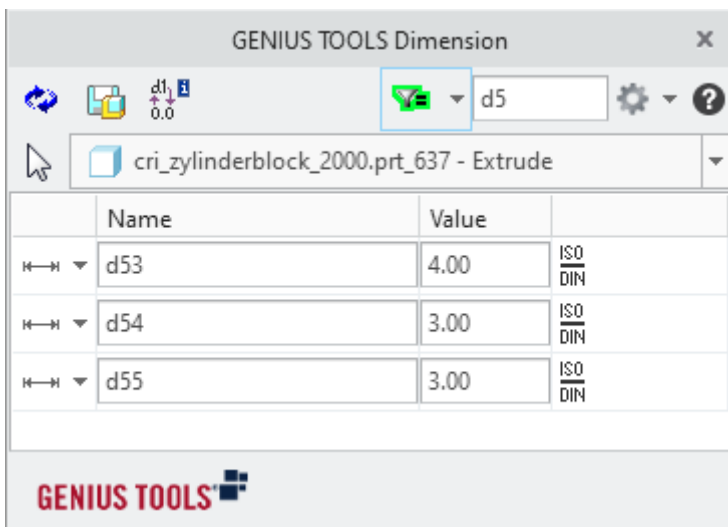
configuration options `gtd_filter_value` and `gtd_filter_value2`. The filter options are only active if the configuration option `gtd_filter_additive=0` is set (default).

Default filter selection for named dimensions

With the configuration option `gtd_filter=1` you can set that the name filter (*Named dimensions*) is activated when GENIUS TOOLS Dimension is started.

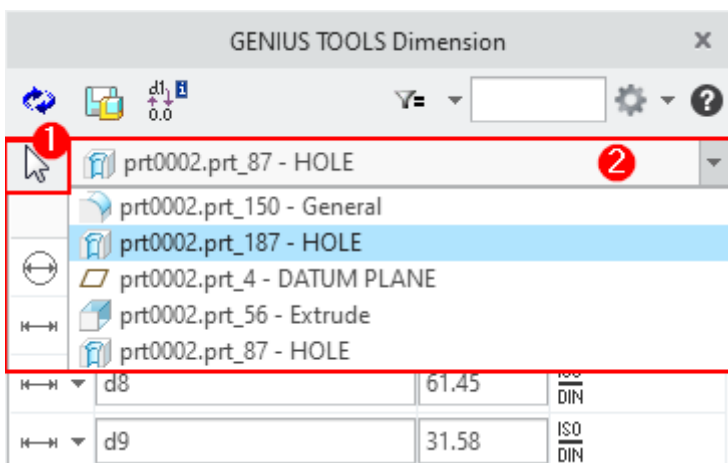
Search field

Use the search to search for names and values of dimensions. Only the currently displayed dimensions will be searched.



6.2.3 Model selection

The model selection consists of the two elements object selection and drop-down list.

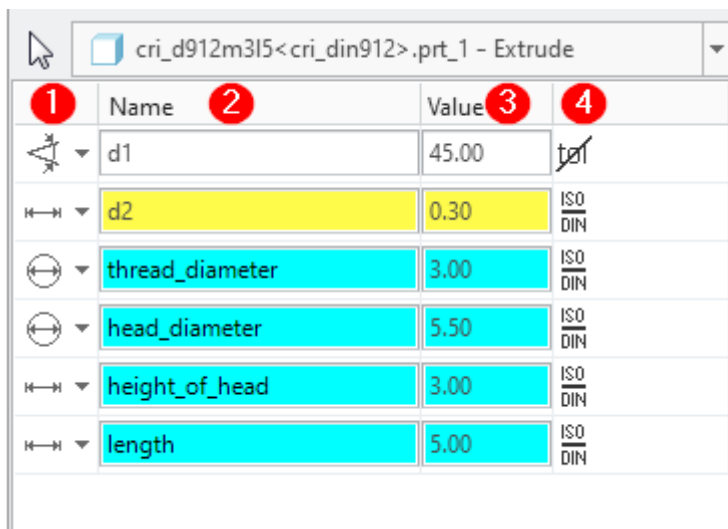


Select models, features, groups and UDFs in the model or in the model tree using the object selection (1).

The drop-down list (2) shows the current selection. Open the list and select from the recently selected objects.

6.3 Dimensions table

The dimensions table displays each dimension in a separate row:



1	2	3	4
	Name	Value	
	d1	45.00	tol
	d2	0.30	ISO DIN
	thread_diameter	3.00	ISO DIN
	head_diameter	5.50	ISO DIN
	height_of_head	3.00	ISO DIN
	length	5.00	ISO DIN

1. Dimension symbol with context menu
2. Dimension name
3. Dimension value
4. Tolerance type

Before each dimension, the symbol corresponding to the dimension type is displayed (1). Click on the symbol to open the context menu for the dimension. The menu shows suitable options for each dimension type.





Click into input fields (columns *Name* (2) and *Value* (3)) to highlight the selected dimension in the Creo model.

Please note: In Drawing mode only displayed dimensions will be highlighted.

Behind each dimension, the tolerance type (4) is displayed. Click on the icon to open the Creo dialog Dimension properties (Creo Parametric 3.0) or the Dimension ribbon (Creo Parametric 4.0 and 5.0) for the dimension.

6.3.1 Dimension symbols and context menu

The following symbols are displayed before dimensions:

Icon	Description
	Linear dimensioning
	Radial dimensioning
	Diameter dimensioning
	Angular dimensioning
#	Pattern count
	Thread

Click on a dimension symbol to open the context menu for the dimension.

Context menu

The following options are available in the context menu:

Copy dimension value

Adopts the value of a dimension that has to be selected in the model into the current dimension.

Open relations manager

Opens the Creo relations dialog.

Dimension properties

Opens the Creo dimensions dialog.

Add to current family table

The selected dimension will be added to the family table of the current model.

Add to parent family table

The selected dimension will be added to the family table of the generic part.

Warning: The two commands **Add to current family table** and **Add to parent family table** will be executed immediately; no saving or regenerating is required.

Specific filter options depending on the dimension type can be found below the general options.

6.3.2 Input fields

In input fields, dimensions are displayed with names and associated values. They are sorted alphabetically by name.

Click into an input field to modify the name or the value of a dimension. Dimensions are highlighted in the Creo main window when you click in an input field.

Warning: Do not rename dimensions according to the scheme ad[Number] (e.g. ad34)! ad dimensions are used internally by Creo Parametric and are filtered out automatically.

Changes can be made in white input fields. The meaning of colored input fields can be found in the table in section [Colors in input fields](#)⁸³.

Dimensions without a specific dimension type are treated as count and will be automatically rounded to integers.

The width of the table columns **Name** and **Value** are controlled with the configuration options `gtd_label_col_size` and `gtd_value_col_size`. The number of decimal places in the **Value** column is controlled with the configuration option `gtd_dec_places`.

Warning: Check the Creo configuration option `show_dim_sign`, as GENIUS TOOLS Dimension respects it.

`show_dim_sign=no`: Entering a negative value changes the direction vector of a dimension. The value will be positive.

`show_dim_sign=yes`: When a negative value is entered, the value remains negative.

Calculations

Input fields for values allow calculations as in Creo. Input up to 250 characters is supported.

Enter a mathematical function into a value field and press Enter. The result is copied into the field. After saving and regenerating, the result will be applied to the model.

Please note: When calculations create unusable values (e.g. -1 in a count field), the last valid value will reentered. An information will be prompted in the Creo Parametric message area.

Also use more complex mathematical operations such as root, powers, or rounding in value fields.



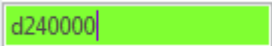

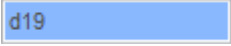


Examples:

`sqrt(d27)` square root of the dimension d27

`pow(d23,3)` third power of the dimension d23



6.3.3 Colors in input fields








Input fields in the dimensions table have a color coding to show how the dimension is controlled. Move the mouse over an input field, a tooltip shows the meaning.

Field color	Description
 White	Unchanged name or value.
 Red	Invalid content or the dimension name already exists in the model; it is not possible to save.
 Green	Modified content, the dimensions table needs to be saved.
 Yellow	Locked dimension, e.g. by a relation. Value and Name cannot be modified. Hint: The dimension can be edited in the Creo relations dialog/ribbon (accessible via the Dimension context menu).
 Blue	The dimension is determined by the family table of the current model.
 Cyan	The dimension is determined by the family table of the parent model. Value and name cannot be modified using the dimension table.
 Magenta	The dimension is determined by a pattern table. Value and name cannot be modified using the dimension table.

6.3.4 Tolerance types





The tolerance type is displayed behind dimensions. Click the icon to open the Creo dimension properties dialog (Creo Parametric 3.0) or the Dimension ribbon (Creo Parametric 4.0 and 5.0).

Icon	Description
	No tolerance table.
	Tolerance table "General according to DIN/ISO".

Icon	Description
	Tolerance table "Shaft with tolerance class".
	Tolerance table "Hole with tolerance class".
	Tolerance table "Breaking edge".
	Tolerance mode: "Plus-Minus" (The symbol does not display the values.) No tolerance table, but free values.
	Tolerance mode: "Limits". (The symbol does not display the values.) No tolerance table, but free values.
	Tolerance mode: "+-symmetrical" (The symbol does not display the values.) No tolerance table, but free values.
	Tolerance mode: "+- symmetrical (superscript)" (The symbol does not display the values.) No tolerance table, but free values.

Please note: Tolerances are not displayed for threads. The entire dimension text is displayed instead.

Tolerance symbols change color (blue or cyan) when tolerances come from a family table.

	d44	8.00	ISO DIN
	d46	140.00	
	d47	0	ISO DIN

7 Forms

With *GENIUS TOOLS Forms* you create forms and apply them. These are user-defined form masks with which you can assign different properties to models. Fill these form masks with various information. You can select and edit dimensions and parameters of parts, features and assemblies, suppress features or components and replace components with others. Depending on which properties are relevant for a model, select the corresponding information in the form. Use this module to call different variants of a model or to configure frequently used start objects.

You can save related properties as a template in a value table so that you can quickly switch between different configurations. Self-defined mapkeys can also be stored in form elements, which are then executed by clicking on the associated buttons in *GENIUS TOOLS Forms*.

GENIUS TOOLS Forms is available in assembly, parts, sheet metal, skeleton, layout and drawing mode with the following features:

1. Systematically observe and manipulate model properties in form dialogs:
 - Dimension values
 - Dimension tolerance
 - Model-parameter values
 - Feature-parameter values
 - Suppress features
 - Material
 - Replace components
 - Variant dimensions tables
2. Executing saved mapkeys
3. Usage of form- / auxiliary parameters
4. The forms are generated via the graphical editor.
5. The forms are saved directly in the models.
6. Rules between properties can be defined with JavaScript.
7. Loading of external data into the mask is possible (e. g. EXCEL, CSV).

7.1 Fundamentals

With *GENIUS TOOLS Forms* you can use forms to control properties and metadata of Creo parts and assemblies.

GENIUS TOOLS Forms is also integrated into *GENIUS TOOLS Library*. If a library model contains a form, it will be displayed directly in the Library details window and can be used for editing the model.

Glossary

Form

A form is a collection of model properties. A form is either stored directly in the model (default) or as an XML definition outside a model.

Form definition

A form definition contains the configuration of a form. This form definition is stored either externally (as an XML file) or internally (at the Creo model).

Form element

A form element describes a model property that is stored in a form with different values.

Form element list

A form element list is the overview for managing the form elements in *GENIUS TOOLS Forms Editor*.

Webcode

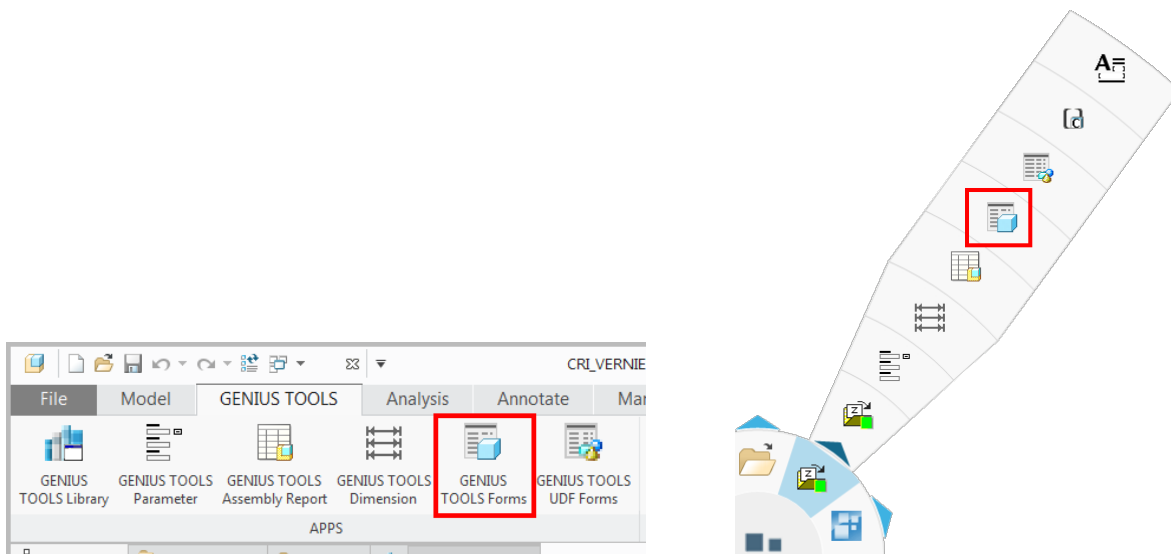
The webcode is a linking property between model and form. The webcode is required by *GENIUS TOOLS Library*¹⁸¹ to establish a connection between an external form and a model through a library object.

7.2 Usage

This section contains information on using *GENIUS TOOLS Forms*. It describes the general structure of the program.

Starting the program

Start *GENIUS TOOLS Forms* from the ribbon menu in the *GENIUS TOOLS* tab or via *GENIUS TOOLS Quick Access* ([<] key).



Forms and UDF Forms in the model

If a model contains a form or UDF form, the corresponding forms icon is displayed in the Creo Parametric main window. Click on the icon to open the form.



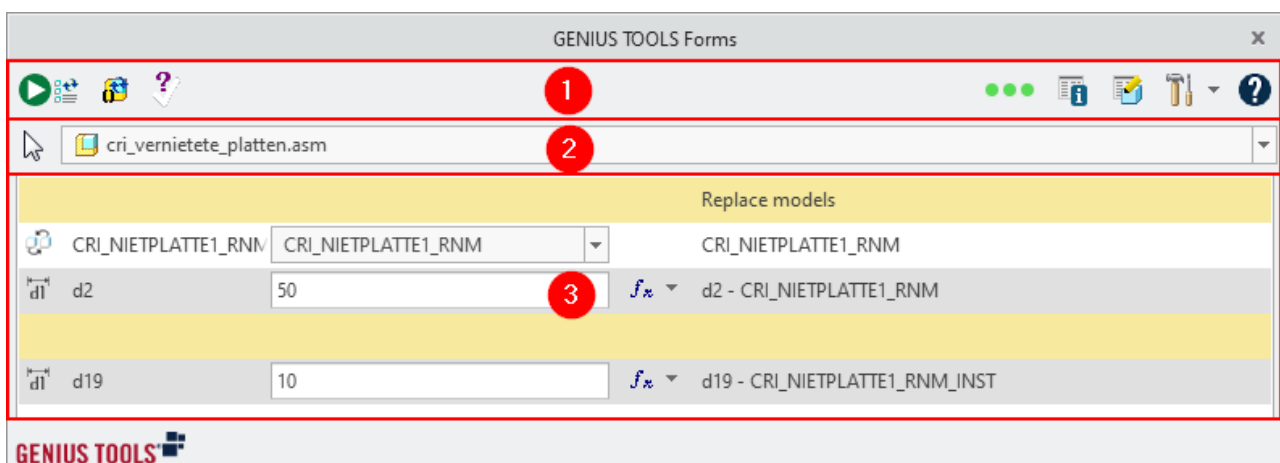
Form in the model



UDF Form in the model

7.2.1 User interface

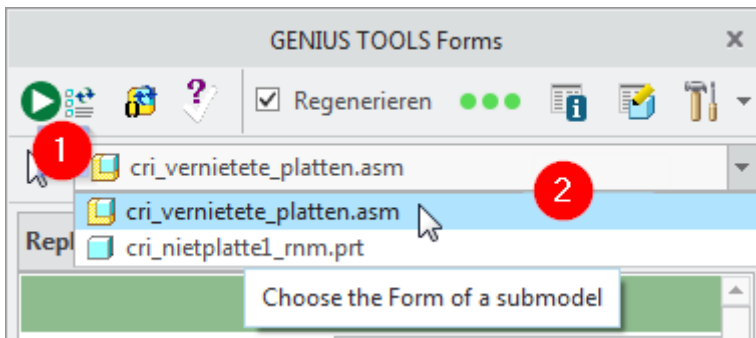
The user interface consists of the following elements:



1. Command bar⁹⁸ with tools menu
2. Model selection⁸⁸
3. Form section⁹¹

Model selection in assembly mode

Use the model selection to switch between forms of different assemblies and parts in the current assembly. The model selection is only displayed in assembly mode.







The model selection with object selection (1) and drop-down list (2).


Select the models in an assembly using the object selection (1) directly in Creo or in the model tree. The drop-down list (2) shows the current selection. Open the list and select from the models. Models that have already been selected with the object selection are additionally displayed in the list.

7.2.2 Command bar

The Command bar displays general controls. The following controls are available:

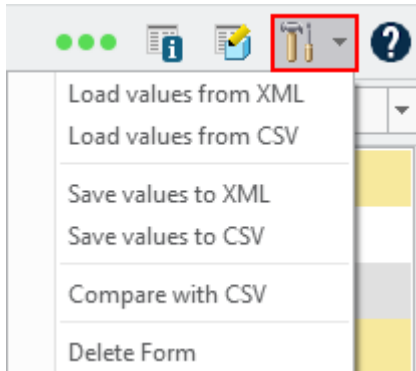
Icon	Name	Description
	Apply changes and regenerate	Applies changes in the form to the model and regenerates it. The quantity of regenerations can be defined with the configuration option <code>gt_regen_times</code> . This can be useful for complex models. If the form includes a JavaScript <i>PreSave</i> function, this function is run before the values are applied. If there is an error, you will be asked whether you want to save anyway. Click <i>No</i> to open the status dialog with the error message.
	Open GENIUS TOOLS Value Transfer ⁴³⁰	Opens the module <i>GENIUS TOOLS Value Transfer</i> . The values are taken from the form. Note: The function is only available in assembly mode.

Icon	Name	Description
	Execute Check function	Starts the JavaScript function <code>CheckUI</code> . Note: The function must be saved in the Form.
	Regeneration	If the checkmark is set, the model is regenerated after the values are transferred to the model. By default, the checkbox is set, but the function is not displayed. The configuration option <code>gtf_show_regen_btn</code> controls whether the checkbox is visible. Set the checkbox to visible and clear the checkbox if you want the function to <i>Apply changes</i> but not to <i>regenerate</i> , in order to avoid long regeneration times for large components and to be able to continue working directly without waiting. In this case, note that values and image do not match.
	Status indicator 	Shows the current status for <i>loading</i> , <i>working</i> and <i>saving</i> with traffic light colors and opens the status dialog.
	Forms information	The button <i>Forms information</i> displays an info area with text and a preview image. Click on the preview image to enlarge it. Text and image are stored in the <i>GENIUS TOOLS Forms Editor</i>  .
	Open GENIUS TOOLS Forms Editor 	Opens the <i>GENIUS TOOLS Forms Editor</i> to edit a form.
	Tools 	The Tools menu contains various supporting functions.
	Help	Opens the help.

Warning: To apply changes, use the button *Apply changes and regenerate* ! Modifications in a form will not be adopted by models on closing.

Tools menu

The following functions are called up via the Tools menu.



Function	Description
Load values from CSV/XML file	<p>Values from a CSV or XML file are read in and applied to the form.</p> <p>When reading from an XML file, the values are assigned to the form elements using IDs. If no IDs are available, the assignment is made using the names of the form elements.</p> <p>When reading from a CSV file, the values are always assigned using the name of the form element.</p> <p>Values of dimensions or parameters without correspondence in the current form are ignored when loading.</p> <p>Line names of value tables may be included. These must be entered manually, since the tag is not created when saving. The ID attribute will be omitted.</p> <p>No JavaScript OnChange functions are executed when reading from a CSV or XML file.</p> <p>Example: <code><element name="ValueTable" value="Square" /></code></p>
Save values to CSV/XML file	<p>Saves the current values in the form to a CSV or XML file.</p> <p>If you want to save a CSV file, make sure that all form elements have unique names. If multiple form elements have the same name, only the first of these elements is written to the CSV file.</p> <p>Units of real parameters are written to the file as well.</p> <p>Value table settings are not saved.</p>
Compare with CSV file	<p>Compares the values of the chosen form with the values from a CSV file.</p>


Function	Description
	Select the desired CSV file. The values from the current form and from the CSV file are displayed in a Compare dialog box. Changes are highlighted in color.
Delete Form	Deletes the form from the current model.

The default folder for CSV and XML imports and exports is defined in the configuration option `gtf_default_folder`.

Please note: Make sure that all form elements have unique names.

With the configuration option `gtf_import_xml_use_name_if_no_id_defined = 0` all entries without ID will be ignored.

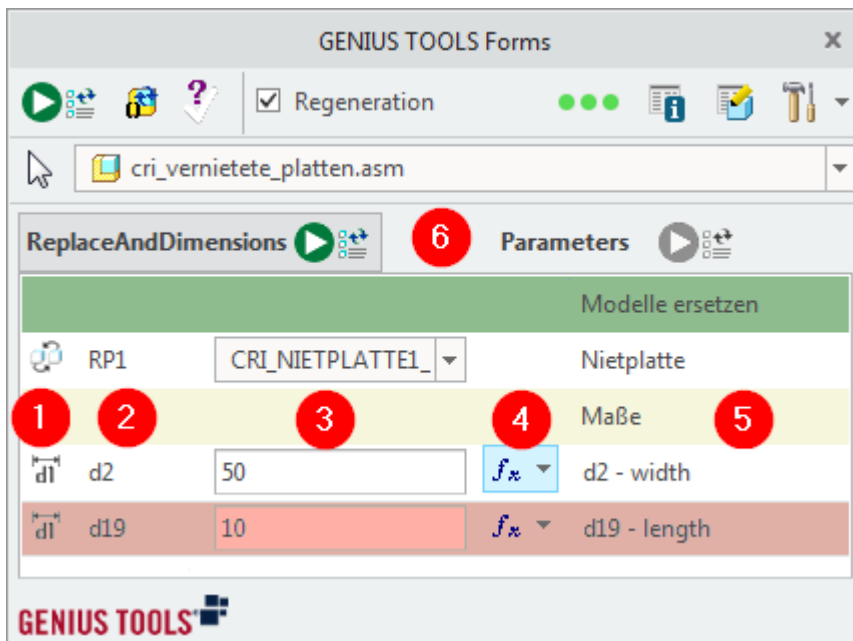
7.2.3 Form display

The form area displays the form of the selected model. In each line a property is listed, whose value can be changed in the third column **Values with input fields**. Click on *Apply changes and regenerate*  to apply the changed values to the model.

Different values are stored in the value tables for several properties, so that when a value table is selected, the values of several rows are set simultaneously. This reduces the need to set individual properties per line and offers the possibility to set the model to a predefined state.

The content of the form is customized in *GENIUS TOOLS Forms Editor*⁹⁹. Visibility and order of the individual form columns are adjusted in *GENIUS TOOLS Forms Configuration*¹¹⁴.

Tip: After opening *GENIUS TOOLS Forms*, the selection that corresponds to the current model is displayed. If none of the selection options meet the condition, the element icon turns yellow. The first stored option is displayed.



Form area with highlighting of a form element (red marked line) and division into five columns

1. Type

The type of a form element is displayed as an icon in the first column. Click an icon to highlight the associated model element in Creo.

Color highlighting indicates whether a form element was changed:

Icons of type column



Meaning

Change to element value



The current model value is not intended as an element value. Outside the *GENIUS TOOLS Forms* user interface, a value was set that is not stored in *GENIUS TOOLS Forms*.



The element addresses a model property that does not exist, e. g. if a parameter was deleted.

2. Name / ID

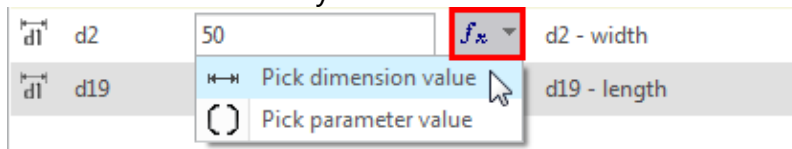
3. Values with input fields

Depending on how the form is defined, there are differently limited choices for [setting properties](#)⁹³.

4. Function

Behind input fields that provide for direct input, there is an additional button for the following functions:

- *Pick dimension value:*
Allows the transfer of a value from any dimension into the input field.
- *Pick parameter value:*
Allows a value from a parameter with the same name to be transferred to the field. If there is no parameter with the same name, a dimension with the same name is searched for and its value is taken over. The parameter value transfer is only available in assembly mode.



Transfer values from dimensions and parameters into a Form

5. Description

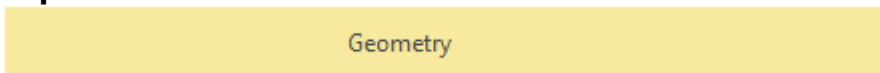
The fifth column displays a language and configuration dependent description of the form elements.

6. Tabs (groups in the editor)

Logical sectioning of larger forms. Click on a tab to open that section of the form.

Structuring a form

Separator



Separators provide a logical structure for form elements.

7.2.4 Using a form

Changes to the values are applied to the model and visible after clicking the button *Apply changes and regenerate* . Below you will find an overview of the input types used to set properties. Depending on the settings in the [GENIUS TOOLS Forms Editor](#)⁹⁷, you have different and possibly limited options for setting a value in the individual lines. You can change the input types and values that can be set in the *GENIUS TOOLS Forms Editor*.

Warning: Check the Creo configuration option `show_dim_sign`, as *GENIUS TOOLS Forms* respects it.

`show_dim_sign=no`: Entering a negative value changes the direction vector of a dimension. The value becomes positive.

`show_dim_sign=yes`: If a negative value is entered, the value remains negative.

Filling input and selection fields

Input field

Length

Input fields accept any string for input.

Binary checkbox

Length ☒ y: 600; n: 300

Checkboxes allow to choose from two options. This can be Yes/No decisions, for example. Always two values are deposited in the configuration for checkboxes. With the checkmark set, the first value is used. Checkmark not set applies the second value.

Radio button

Length ☒ 600 mm
☐ 300 mm
☐ 350 mm

Radio buttons display predefined selectable values. Only one of the values can be selected.

Dropdownbox

Length
Width
300 mm

Dropdownboxes contain a selection of possible inputs.


Extended dropdownbox

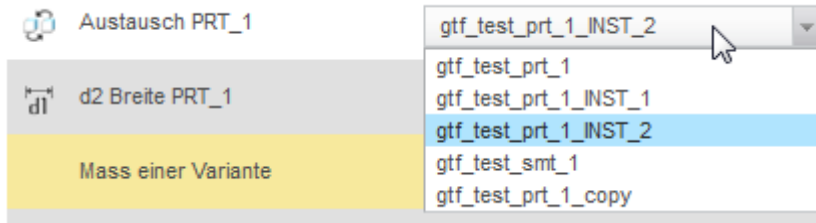
Length
Width
350 mm

Extended dropdownboxes accept free input of values as well as selection from a list. Each input filters the displayed list items.

Modifying components

Replacing components

The function *Replace components*  exchanges predefined components for one another using selection fields.



Replacements are selected via lists


Select a model from the drop-down list to replace a component. Click the icon *Replace component* in the first column to highlight the component to be replaced in Creo.

Please note: Replacements fail when assemblies have missing components.

Warning: Using the *GENIUS TOOLS for Creo* together with the *Startup TOOLS*: When replacing with an interchange assembly (instead of using a family table), a new feature ID is assigned to the part. Check the Creo configuration option `remember_replaced_components`. This option is set in Creo Parametric, but is disabled by *Startup TOOLS*.

If new feature IDs are created by an interchange assembly, the new IDs are automatically written to the model.

Suppress components / features

The function *Feature suppression*  works using checkboxes. If a checkbox is active (checkmark is set), the component / feature is displayed. With a deactivated checkbox, the component / feature will be suppressed after regenerating.

Please note: On resuming, dependent suppressed elements are not resumed.


Click on the *Suppress components / features* icon to highlight the associated component / feature in Creo.

Automatic setting of properties

Run Mapkey / JavaScript




Mapkey with configuration-dependent "Start" label

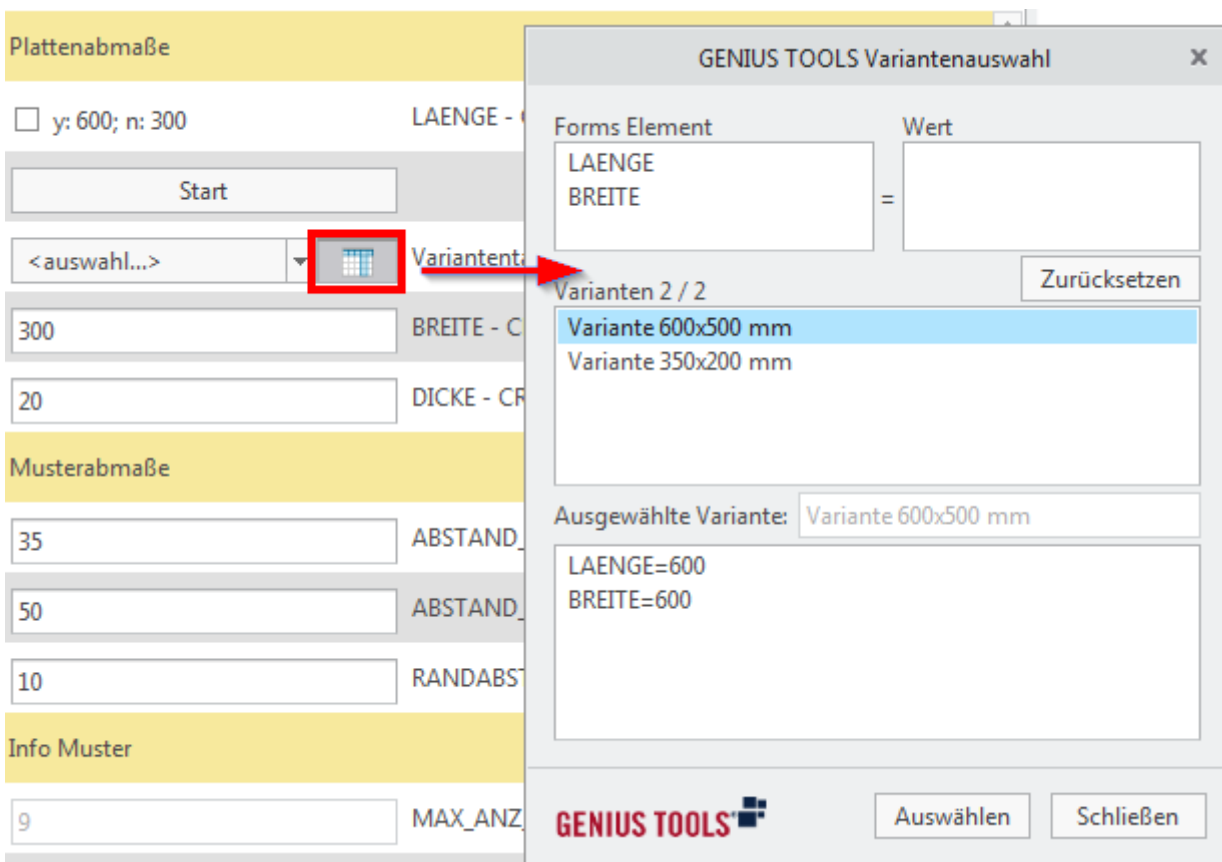
Mapkeys / JavaScript elements  stored in the form are started by clicking the corresponding button.

Setting multiple properties

Value table

Varianten	<auswahl...>	
BREITE	<auswahl...>	
DICKE	Variante 600x500 mm Variante 350x200 mm	

Variant tables  display preconfigured variants of a model. In most cases, each variant affects multiple properties of a model at the same time. Select the variants from the list or alternatively use the button *Variant selection* to filter variants.



Variant selection for filters by individual properties

Please note: If the value table symbol turns yellow or red, please check whether the externally read value tables are correct.

Warning: Value tables cannot affect each other!

7.3 Configuration

In this section you will find more information about the *GENIUS TOOLS Forms Editor*, which allows you to create and manage forms. Suggestions for creating and configuring

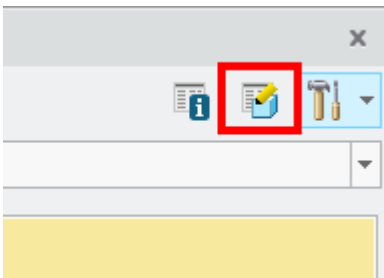
forms can be found in Use cases¹¹⁸.

7.3.1 Forms Editor

You create forms with the *GENIUS TOOLS Forms Editor*.

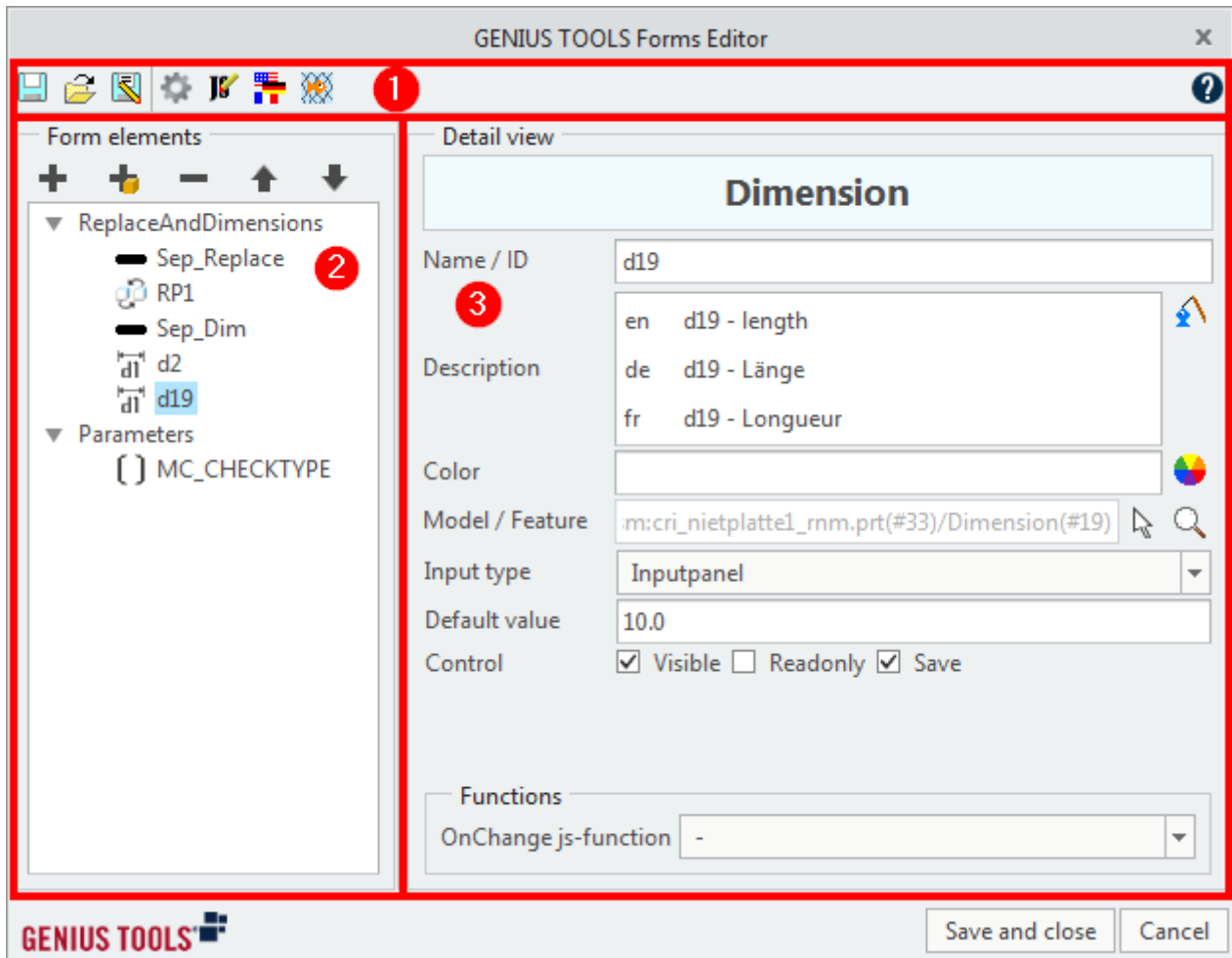
Starting the program

Start the Editor via the command bar in the *GENIUS TOOLS Forms* user interface.



7.3.1.1 User interface



The user interface of *GENIUS TOOLS Forms Editor* consists of the following elements:









1. Command bar ⁹⁸
2. Form elements list ⁹⁹
3. Detail view ¹⁰³

7.3.1.2 Command bar





The following buttons are contained in the command bar:

Icon	Name	Description
	Save	Saves the current form and closes <i>GENIUS TOOLS Forms Editor</i> .
	Open	Opens a forms definition (XML file).

Icon	Name	Description
	Save as	Saves the current form as a forms definition (XML file).
	Edit Forms configuration ¹¹⁴	Opens the dialog for configuring the <i>GENIUS TOOLS Forms</i> user interface.
	Edit JavaScript ⁵⁷⁷	Opens the <i>JavaScript Editor</i> .
	Manage languages	Opens the dialog for editing the included languages.
	Automatic default texts	Opens the dialog for automatically setting standard texts for all used form elements. The names of the form elements are used as keys for the translation, see Set standard texts for multiple elements ⁴⁹⁷ .
	Help	Opens the Help.

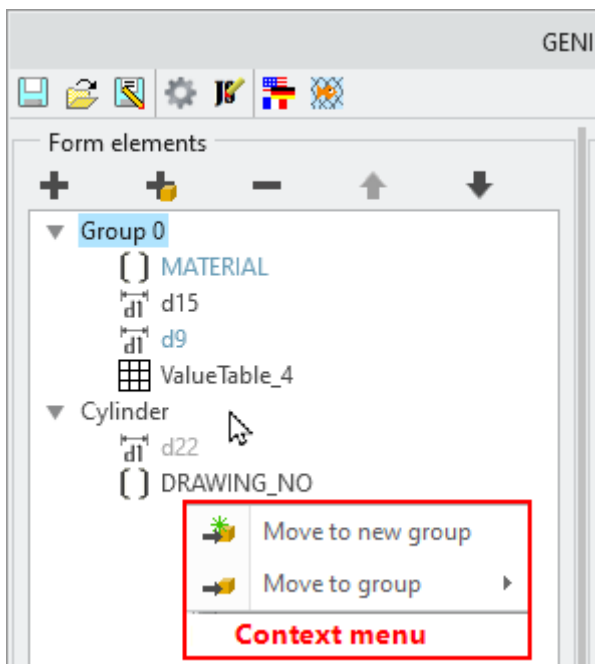
7.3.1.3 Managing form elements

The list control is used to manage form elements. The following buttons are contained in the list control:

Icon	Name	Description
	Add new item to list ¹⁰⁰	Opens selection dialog for creating a new form element.
	Add new group	Creates a new group, displayed as tab in <i>GENIUS TOOLS Forms</i> . The groups serve as separators in the table.
	Delete selected item from list	Deletes the selected form element or group. Warning: Groups are deleted along with their contents. Move any form elements you still need before deleting a group.
	Move selected item	Moves the selected form element or group up or down in the list.

The form element list is displayed under the list control. All created form elements are displayed here in the order in which they are displayed in the *GENIUS TOOLS Forms* panel. Read-only elements are displayed in blue font. Non-visible, read-only elements are

displayed in gray font. A value with a blue background is currently selected and opened in *Detail view*.



To move a group or a form element to another group, right-click to open the context menu and select *Move to new group* or *Move to group* (with selection of an existing group).

You can also select multiple elements and move them using the *Move* buttons or drag-and-drop. Right-click on an item to move the items within the created groups. Select an item to view its details and make changes.

Adding form elements

Use the button *Add new item to list* **+** in the list control to add new form elements. Click *Next* to open the selection dialog.

Add an Element [X]

Type of controlled element:

- ☒ Dimension
- ☐ Renamed Dimensions
- ☐ Dimension Tolerance
- ☐ Model Parameter
- ☐ Component Parameter
- ☐ Feature Parameter
- ☐ Feature/Component suppression
- ☐ Component replace
- ☐ Separator
- ☐ Value Table
- ☐ Mapkey / Javascript
- ☐ Material
- ☐ Supplementary Parameter

☒ Add multiple elements of a type until cancelled

Next

This control button allows you to select multiple elements, here using the example of the type "Dimension"

You can select multiple elements by ticking the checkbox *Add multiple elements of a type until cancelled*. The function is available for selection until you click *Cancel*.

Select [X]

Select 1 item.

OK Cancel

Creo button for item selection

After selecting one or more elements in the model, fill in the corresponding *Detail View*¹⁰³ dialog box.

Types of form elements

Form element	Description	Select element
Dimension	The <i>input type</i> ¹⁰⁵ can be an input field, a radio button, a dropdown box, or an extended dropdown box.	X

Form element	Description	Select element
Renamed dimensions	All dimensions whose names have been edited are automatically transferred as dimension form elements from the selected model or feature to the current form definition.	X
Dimension tolerance	The input type ¹⁰⁵ can be an input field, a radio button, a dropdown box, or an extended dropdown box.	X
Model parameter	Different input types ¹⁰⁵ are possible.	X
Component parameter	Different input types ¹⁰⁵ are possible.	X
Feature parameter	Different input types ¹⁰⁵ are possible.	X
Suppress feature/component suppression ¹⁰⁸	Separate configuration options.	X
Replace component ¹⁰⁸	Separate configuration options.	X
Separator ¹⁰⁹	A separator used to structure a form.	
Value table ¹¹¹	Sets values for multiple existing form elements.	
Mapkey / Javascript ¹¹⁰	Mapkey and JavaScript elements are displayed as buttons.	
Material ¹¹⁰	Separate configuration options.	X
Supplementary Parameter	Do not exist in the model and cannot be saved. Used for JavaScript functions only. Accept any of the four available data types and can be configured with different input options depending on their data type, see input types ¹⁰⁷ .	

Please note: If you have added a read-only dimension, the *Read-only* property is automatically checked and the *Save* property is unchecked in the form element created with that dimension.

Configuration of the form element list

The `gtf_editor_multiselect` configuration option allows you to set the checkbox *Add multiple elements of a type until cancelled* to be checked when starting *GENIUS TOOLS Forms* (default: 0 = off).

7.3.1.4 Describing a form

The *Detail View* is used to configure individual form elements.. Click an element in the element list to display its *Detail view*.

General settings

All form elements have the following entries.

The screenshot shows a 'Detail view' window for a 'Dimension' element. The 'Name / ID' field contains 'd2'. The 'Description' field shows a list of entries: 'en d2 - width', 'de d2 - Breite', and 'fr d2 - width'. The 'Color' field is set to 'Coral'. There are also icons for undo/redo and a color picker.

Detail view with general options of an element of the type "dimension"


Name / ID

The displayed name of a form element (second column in the form).

Title

For groups only: The title is used as the tab caption for the group and can be maintained language-dependent. If there is no title, the name of the group appears as the tab caption.


Description

The localized description of a form element (fourth column in the form section). The displayed description depends on the configuration options `gtf_lang` and `gtf_def_lang`. Default texts can be added using the button  (Description of the standard text selection dialog ⁴⁹⁷).

Use the languages dialog (flag icon  in the command bar) to enter descriptions in multiple languages.

Please note: Description texts are limited to 80 characters.

Color

The color of the row of the form element. Enter a color directly from the RGB color space (comma separated) or click on the color circle  to select a color (Description of the color dialog ⁵⁹⁰). The brightness of the background color determines whether the foreground color of the element is white or black.

Display settings

The following specifications depend on the form element.

Visible

Defines whether an element is displayed in the form area (e. g. for elements influenced by value tables).

Input type

- Input panel: input fields accept different inputs depending on the form element.
- Radio button, dropdown box, extended dropdown box: the section *Input values* opens.

Default value

Specifies the preset default value of a form element.

Please note: The default value is displayed only if it is a valid input. If not, the current value from the model is displayed.

Read only

Defines whether the field is read-only.

Save

Defines whether the element value can be stored in the model.

Tip: Use the *Read-only* option for input fields in combination with *value tables* ¹¹¹ or *JavaScript* ¹¹⁶. In this way, changes are visible without the individual values being edited by the user.

Input values

Enter values for [selection fields](#)¹⁰⁵.

Functions

You can assign an *OnChange* function to the element, see [JavaScript in Forms](#).¹¹⁶

The *OnChange* function must be defined in the respective configuration via the [JavaScript editor](#)⁵⁷⁷. *OnChange* is executed when the value of a form element is changed automatically (e. g. by value tables) or manually by exiting a field or by pressing Enter.

Please note: Avoid creating infinite loops with your functions.

7.3.1.5 Creating form elements

This section contains information about defining different input types. To set several multiple properties at once, first define all the form elements you need, and then maintain them in [value tables](#)¹¹¹.

Input field

If you select *Input field* as the input type, you can enter any string into the form. You can use a [JavaScript function](#)⁵⁷⁷ to limit the values that can be entered, for example, to a range of values.

Creating (extended) selection fields and radio buttons

In *GENIUS TOOLS Forms*, different types of input can be used. Selection fields (dropdown), extended selection fields (writable) and option fields are controlled by a table consisting of values and labels.

These value tables provide combinations of existing form elements (dimensions, parameters, substitutions and suppressions).

Tip: If you do not want a form element to be set in a particular variant, enter * in the cell for that element. In this case, the form element retains its previous value.

The screenshot shows the configuration window for a 'Dropdownbox' input type. The 'Default value' is set to '10.0'. Under the 'Control' section, 'Visible' and 'Save' are checked, while 'Readonly' is unchecked. The 'Input values' section contains an 'External file' field with a file explorer icon. Below this is a table with three columns: 'Value', 'en', 'de', and 'fr'. The table contains three rows of data. To the right of the table are buttons for adding (+), removing (-), moving up (↑), and moving down (↓) rows.

Value	en	de	fr
5	thin	dünn	mince
10	medium	mittel	moyen
15	thick	dick	épais

Defined selection fields

Creating a table

Each table row describes one variant.

- Use the plus button **+** to add a row. Select a row and click the minus button **–** to remove a row.
- Use the arrow buttons **↑↓** to move rows up and down.
- Enter values and the associated labels in the table. The label is displayed in the form area instead of the value in the field and is language-dependent.

Using values from external file

You can also store the values for the selection in an external CSV file.

- The values from the CSV file are saved in the form.
- When the form is called, it checks to see if the linked CSV file exists and applies the changes.
- Select the required file under *External file* with the button The file explorer opens by default in the folder `%GT_RESOURCE_FOLDER%forms` starting from version 8.0.2 (Prior versions open the working directory.). You can define a different folder with the option `gtf_external_data_folder`.
- Click *Update* to re-read the displayed values from the CSV file.

Linking an external file

Starting with version 8.0.2, there are three ways to specify an external CSV file. In older versions, the CSV file is searched in the working directory.

1. Filename with extension, no path: the file must be located in the folder defined by the `gtf_external_data_folder` configuration option. The default is `%GT_RESOURCE_FOLDER%forms`.

2. A relative specification to the folder defined in the `gtf_external_data_folder` configuration option. Use this if you want the CSV file to be in a different folder.
3. An absolute path specification: Avoid using this if possible, as changes cannot be committed.





Creating an external file


The CSV file must contain the selection values in the first column and the labels in the other columns. The header cell of the first column is ignored. The header cells of the other columns must contain the two-character abbreviations of the corresponding languages.

	A	B	C	D
1		en	de	fr
2	5	thin	dünn	mince
3	10	medium	mittel	moyen
4	15	thick	dick	épais
5				

Editing parameters

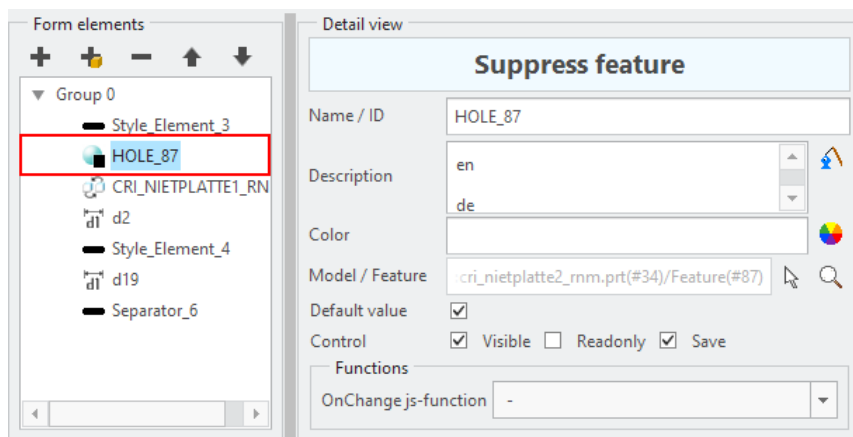
Four parameter types are available for editing model, component and feature parameters. The parameter types are editable with these field types:

Icon	Name	Editable by
	Boolean parameter (Yes/No parameter)	– Radiobox
	Integer parameter This parameter type accepts only integer input. All inputs following a decimal point will be ignored.	– Radiobox – Inputpanel – Dropdownbox – Extended dropdownbox
	Real parameters Real parameters accept integer input as well as point-separated values.	– Radiobox – Inputpanel – Dropdownbox – Extended dropdownbox
	String parameters String parameters accept any character string.	– Radiobox – Inputpanel – Dropdownbox – Extended dropdownbox

The asterisk  indicates help parameters. Help parameters do not exist in the model and cannot be saved. Their only purpose is to be used in JavaScript functions¹¹⁶. Help parameters can have any of the four available data types.

Feature / component suppression

Form elements for suppress actions define the suppress state of a model or feature.



Use the Selection button to select the model / feature to be suppressed from the model tree or directly from the model. Under *Scope*, specify whether to replace the selected model instance or all assembled models with the same name.

Suppress actions are displayed as a checkbox¹¹¹ in the form. Select the default state of the check box under *View*.


Please note: Suppressing all parts only affects parts in the same assembly.

Component replace

Replacement form elements are used to replace models or features with other models or features. Component replacements can be managed using any type of input field, but the use of drop down lists is recommended.

Use the Creo Selection button to define the model or feature to be replaced using the model tree or a model. Under *Scope*, specify whether to replace the selected model instance or all component models with the same name.

To name instances, use the following format: Instance<Generic>.prt.

If you are using a value list, you can import the instances from the family table of the selected model. To do this, click *Import instances* .

Warning: Take care to use the correct spelling for instances.

You can use models or features from the current work directory, from the directory of the assembly or from family tables.

Detail view

Replace

Name / ID: CRI_NIETPLATTE1_RNM

Description: en CRI_NIETPLATTE1_RNM, de CRI_NIETPLATTE1_RNM

Color: [Color Picker]

Model / Feature: rnietete_platten.asm:cri_nietplatte1_rnm.prt(#33)

Scope: ☒ All in form model, ☐ Only selected, ☐ All in level, ☐ In all subassemblies

Input type: Dropdownbox

Default value: CRI_NIETPLATTE1_RNM.PRT

Control: ☒ Visible, ☐ Readonly, ☒ Save

Input values

External file: [Text Box]

Value	en	de	fr
:1_RNM.PRT	ATTE1_RNM	ATTE1_RNM	ATTE1_RNM
_RNM>.PRT	_RNM_INST	_RNM_INST	_RNM_INST
/INST1.PRT	_RNM_INST1	_RNM_INST1	_RNM_INST1
TE_SMT.PRT	PLATTE_SMT	PLATTE_SMT	PLATTE_SMT

Functions

OnChange js-function: -

Configured replacement element using a dropdownbox

Please note: Forms are processed from the top-down. Everything before a replace action is applied to the old model. Everything that comes after a replacement action is applied to the new model.

Replacement actions are only supported for interchange assemblies and instances from family tables.

Replacing all models with same name only works within the same assembly.

Separator



Separators are used to organize a form and group its elements. To configure a separator, enter a name and localized descriptions for it.

Use the *Visible* option to use a separator only in the Editor.

Tip: The name of a separator can be edited using [JavaScript](#)⁵⁷⁷.

Mapkey / JavaScript

Elements that execute a mapkey command or a Javascript function, are displayed as buttons on the form and require special configuration.




1. Define the label for the button under *Button text*.
2. In the *Mapkey/Javascript* section, enter the code to execute
 - a Mapkey command or
 - a [Javascript function](#)⁵⁷⁷ (Javascript functions must begin with "js:")
3. Use the *Shorten Mapkey*  button to display long Mapkeys in a more compact format.
4. The button  launches the [JavaScript Editor](#)⁵⁷⁷ to help you write JavaScript code.

Material

Material input fields are used to assign materials to individual models. Enter a name and a localized description to configure a material input field. Enter a default value. The default value is descriptive. A matching material must be entered by the user.

Detail view

Material

Name / ID	CRI_NIETPLATTE1_RNM		
Description	en		
	de		
	fr		
Color	DarkSeaGreen		
Model / Feature	cri_vernietete_platten.asm:cri_nietplatte1_rnm.prt(#33) 		
Default value	STAHL-ALLGEMEIN		
Control	<input checked="" type="checkbox"/> Visible	<input type="checkbox"/> Readonly	<input checked="" type="checkbox"/> Save
Functions			
OnChange js-function	-		

Creating a binary checkbox

To use this input type, you must have a Boolean parameter in your model. Create this Boolean parameter as a [form element](#)⁹⁹ to create the *Binary Checkbox* input type. The input type is then uneditable.

You can specify whether the value is enabled or disabled by default.

Creating value tables

This section provides instructions for defining value tables that allow you to set multiple model properties simultaneously. Value tables control variants for already created form elements (dimensions, parameters, replacements and suppressions). Each table row is a single variant. Use the arrow buttons on the right to move rows or columns.

Tip: If you do not want a form element to be addressed in a particular variant, enter * in the cell for that element.

ValueTable_6	d2	d19
V1	10	20
V2	20	40
V3	40	80

Value table

Creating an external file as value table

You can also store the values for the selection in an external CSV file. The first two rows of the CSV file link the values given below to the required form elements.

There are two options for creating tables:

- Rows 1 and 2 are empty: the columns are assigned according to their order. To force this behavior, the 2nd header row must be empty.
- Specifying the table header in lines 1 and 2 (see figure). Based on this information, the header is created in the value table and the table is filled in. The columns are assigned according to their ID in the 2nd header row.

To create the header rows for a CSV file, first create the required columns in the value table. Then, export the table as a CSV file . The exported file contains the required table header.

	A	B	C
1	variant_table Dimensions	d2	d19
2	variant_table Dimensions	dim 33 2	dim 33 19
3	V1	10	20
4	V2	20	40
5	V3	40	80

Enter the desired values into the CSV file. Then select the CSV file under *External file*. The values from the CSV file are saved in the form. When the form is called up, it checks whether the linked CSV file exists, and changes are applied.

Linking external files

There are three ways of specifying a CSV file. They are the same as for [Selection and option fields](#). ¹⁰⁶

You can click *Update* to re-read the displayed values from the CSV file.

Editing value tables

The following control buttons are available for editing value tables.

Icon	Name	Description
	Import value table from file	Imports a value table from the following file formats: <ul style="list-style-type: none"> – xls (Excel 97-2003) – xlsx (Excel 2003-2016) – csv (Comma-separated values) – txt (text files with UTF-16LE/Unicode formatting)
	Export value table to file	Exports a value table to the formats: <ul style="list-style-type: none"> – xls (Excel 97-2003) – xlsx (Excel 2003-2016) – csv (Comma-separated values)
	Edit value table in spreadsheet software	The value table is opened in a spreadsheet program (depending on the client computer) and can be edited. Then it is reimported.
	Sort selected column	Sorts the rows by the active column. Three sorting orders are available: <ul style="list-style-type: none"> – 0-Z – Z-0 – initial
	Add/Remove row	Adds a new line under the current cursor position. To remove, select a row and click <i>Remove row</i> .
	Add/Remove column	Adds a new column to the right of the current cursor position. To remove, select a column and click <i>Remove column</i> .

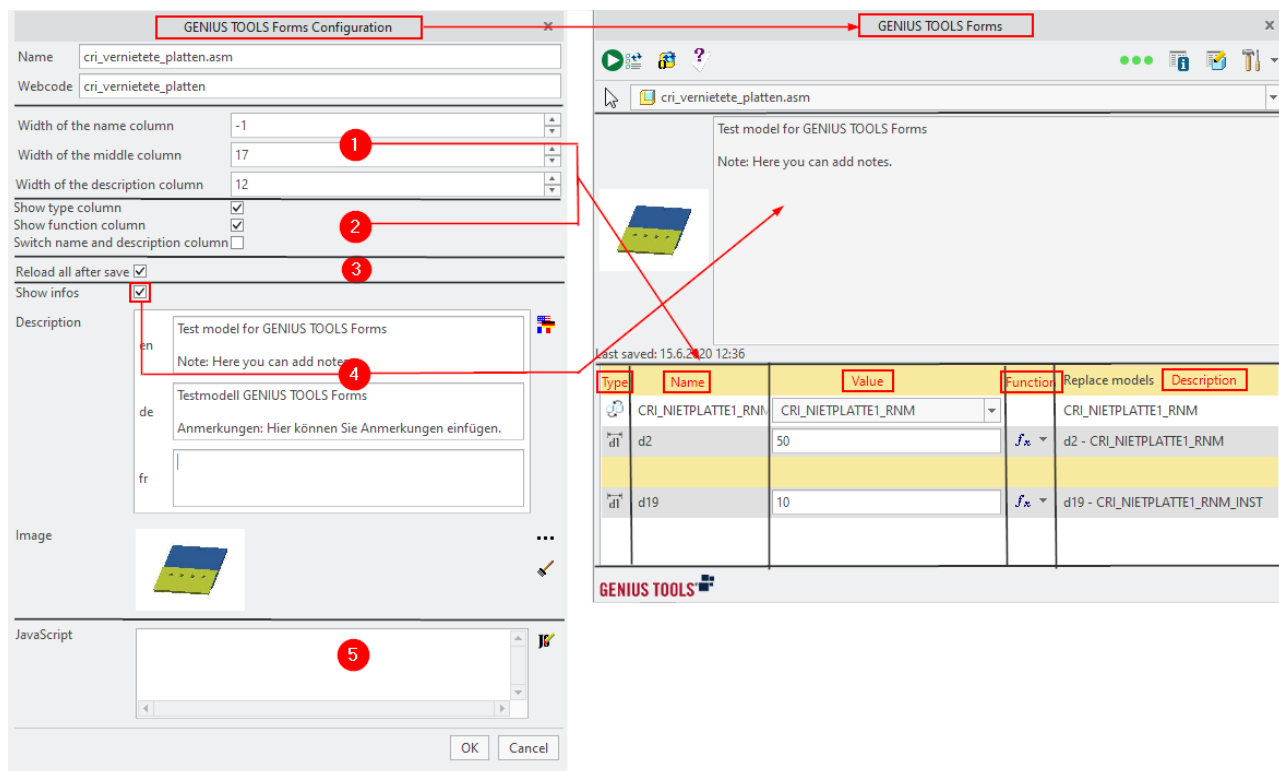
Icon	Name	Description
	Remove all rows and columns	Deletes all rows and columns from the table.

Please note: All changes made to a value table during editing in a spreadsheet program (delete, move or change rows/columns) are applied by *GENIUS TOOLS Forms*.

Find an example on how to **create a value table** in the section *Use cases*.¹¹⁸

7.3.1.6 Editing form display

The button *Opens Forms configuration* opens the Forms Configuration dialog which allows you to specify the user interface of *GENIUS TOOLS Forms*⁸⁷ as follows:



Structure of GENIUS TOOLS Forms Configuration and impact on GENIUS TOOLS Forms user interface

1. defining column widths of a form¹¹⁵
2. displaying and arranging columns¹¹⁵
3. renewing Creo values¹¹⁵
4. providing description and sample image¹¹⁶
5. entering JavaScript code¹¹⁶ to be used in the form

Column widths

Width of the name / value / description columns: Specifies the width of the respective column in characters.

- Value "0": hides the column
- Value "-1": applies the default values of the configuration options to the column

Please note: Do not hide the value column if you want users to fill out the form.

Displaying and arranging columns

In this segment you can:

- **Show Type column:** removes or displays type column (first column)
To apply this setting, save it in the *Forms Editor*.
- **Show Function column:** removes or displays function column (fourth column)
To apply this setting, save it in the *Forms Editor*.
- **Switch Name and Description column:** If you switch the Name and Description columns, the description will appear before the value column, which can be used if you prefer to write your own descriptions.
To apply this setting, save it in the *Forms Editor* and restart *GENIUS TOOLS Forms*.
You can hide the Name, Value and Description columns in the previous segment by entering "0".

Renewing Creo-values


Display Creo values after saving: Re-read all values of the model into the form after saving.


- | | |
|-----|---|
| On | <ul style="list-style-type: none">- displays the current values of the model (calculated values e. g. from relations / dependencies) in the form.- in case of an error message - i. e. if the input value could not be transferred to the model, the previous value is displayed again in the form |
| Off | <ul style="list-style-type: none">- all entered values are kept in the form, including those values that cause errors (i. e. that cannot be applied to the model)- does not re-read the dependent values from the model- useful for not losing incorrect entries |

Please note: Before disabling this feature, test to determine how dependencies are affected.


Providing description and sample image

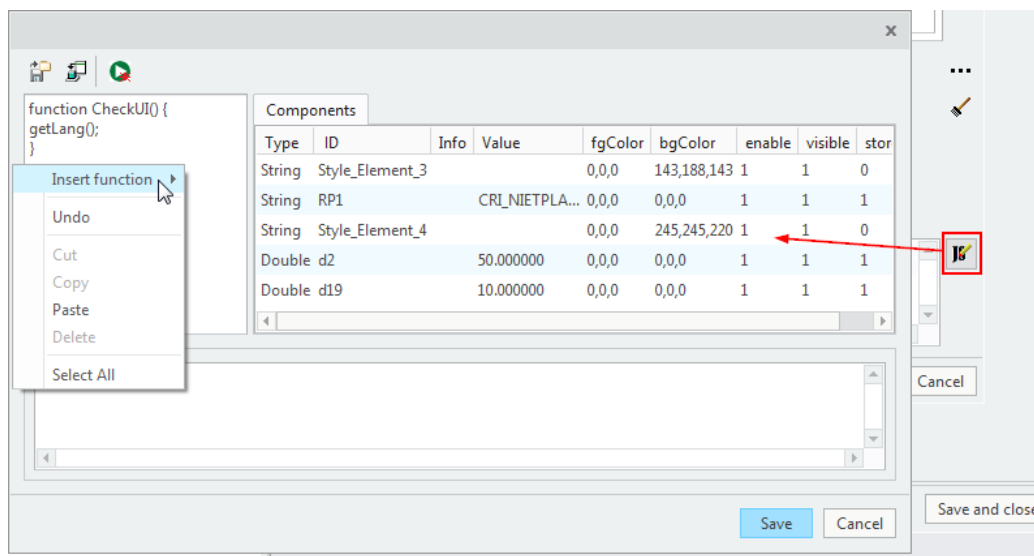
Display information: Automatically displays the information area when a form is opened.

Description: Enter a descriptive text for the form in the fields. Use the button *Open the multilanguage description window*  to manage the languages.

Preview image: Enter any image here. Images are saved directly in the forms definition. The size of the saved image depends on the configuration options `gt_image_width` and `gt_image_height`. An image can be deleted by clicking the broom icon .

7.3.1.7 JavaScript in Forms

Executable JavaScript code is stored in the JavaScript field. Enter the code directly into the input field or click on the JavaScript icon  to open the [JavaScript Editor](#)⁵⁷⁷. Right-click to open the context menu for quick function entry (Insert function). For more information, consult the chapter [JavaScript Editor](#)⁵⁷⁷ and the application example for using JavaScript with a GENIUS TOOLS module³⁹⁶.



JavaScript Editor in GENIUS TOOLS Forms

Time of execution

JavaScript can be executed at different times. You can add as many functions as required to your JavaScript code.

Time of execution	Function
After loading a form	PostLoad
After loading of a value table (CSV/XML)	PostLoadFromFile

Time of execution	Function
Before saving values in a form	PreSave
After saving values in a form	PostSave
After clicking the button <i>Check values</i> in the <i>Forms</i> dialog	CheckUI
After editing a <i>form</i> value or after clicking Enter	OnChange, is a function activated in the Functions ¹⁰⁵ section of the <i>Detail view</i> .

The names of *PostLoad*, *PostLoadFromFile*, *PreSave*, *PostSave* and *CheckUI* are fixed. Functions of the type *OnChange* can have any name.

Loading values from a CSV or XML file does not trigger any JavaScript OnChange functions.

Please note: Use the JavaScript function *creoMapkeyAddToStack* only as a PostSave function, because it executes mapkeys and mapkeys usually close windows.

7.3.1.8 Automatic load of external data

The automatic reloading of external forms can be used, for example, to transfer legacy data.

If the selected or current model does not contain a form, but a webcode, a check is made in a defined directory to see if a form definition (XML) with the parameter name **WEBCODE** is available. If so, it is loaded into the model and the form is opened. Other parameter names can be specified in the `gtf_autoload_parameter` configuration option to reload externally stored forms. You may need to do this, for example, when importing models created with MUI - a component of Startup TOOLS up to version 2018.

If you want to add forms to models created with MUI, you must first load, update, and validate the forms in the template model. Then they are saved as XML files in a specified folder.

1. Specify a folder with form configurations (XML) under `gtf_autoload_folder`.
2. A parameter must now be found in the model for automatic loading. Specify the parameter name in `gtf_autoload_parameter`.
3. If the parameter name exists, but there is no form in the model yet, an XML file with the same name will be searched and loaded. The `gtf_autoload_overwrite` configuration option can be used to specify whether this operation should also be performed for models that already contain a form. In this case, the existing form will be overwritten.

7.3.2 Use cases

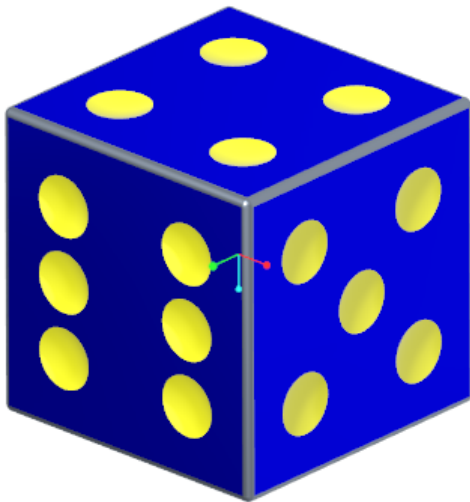
This section provides an example of how to create and use a form.

Requirements

The example model is a cube. The size of the cube, the radius of the individual numbers of pips and the depth of the numbers of pips are supposed to be controllable. The positions of the numbers of pips in relation to each other and on the cube are fixed. They are not changed by the form.

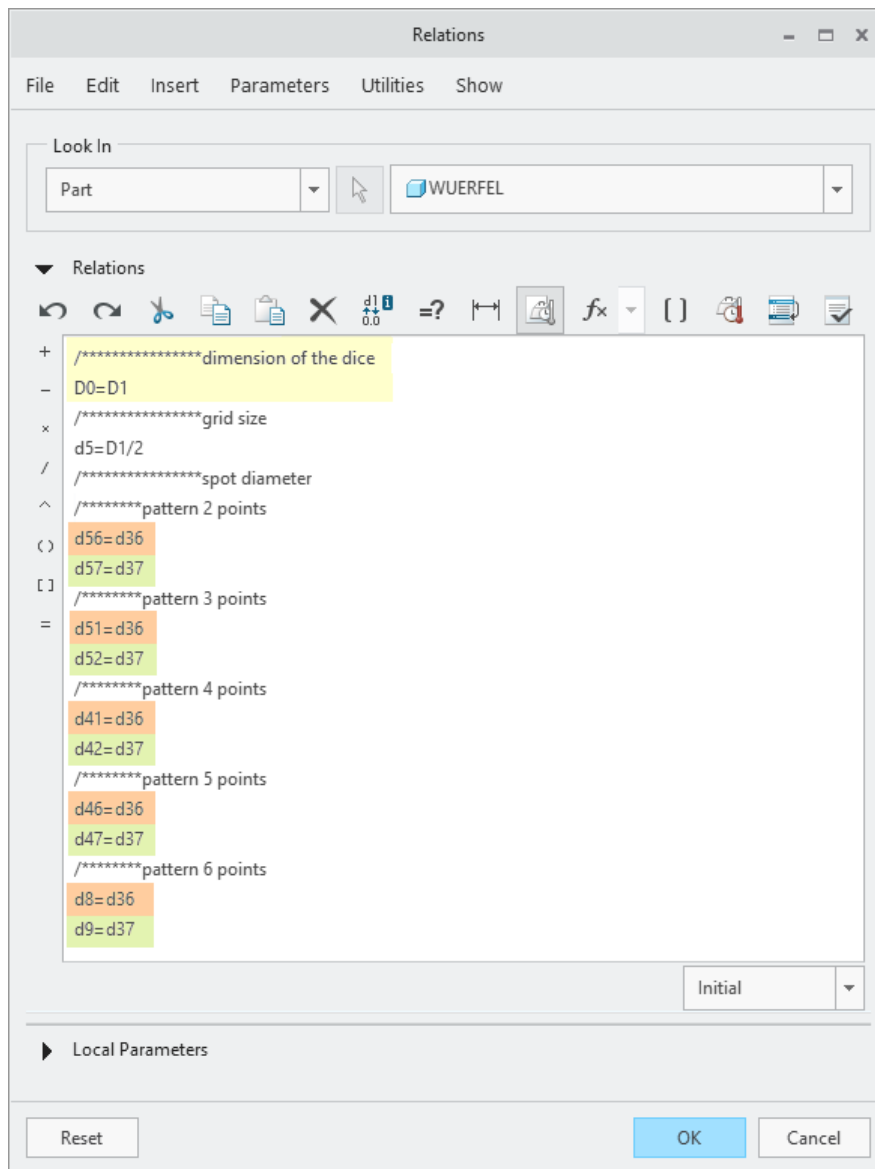
Creating a form

1. Open the model in which you want to create the form.






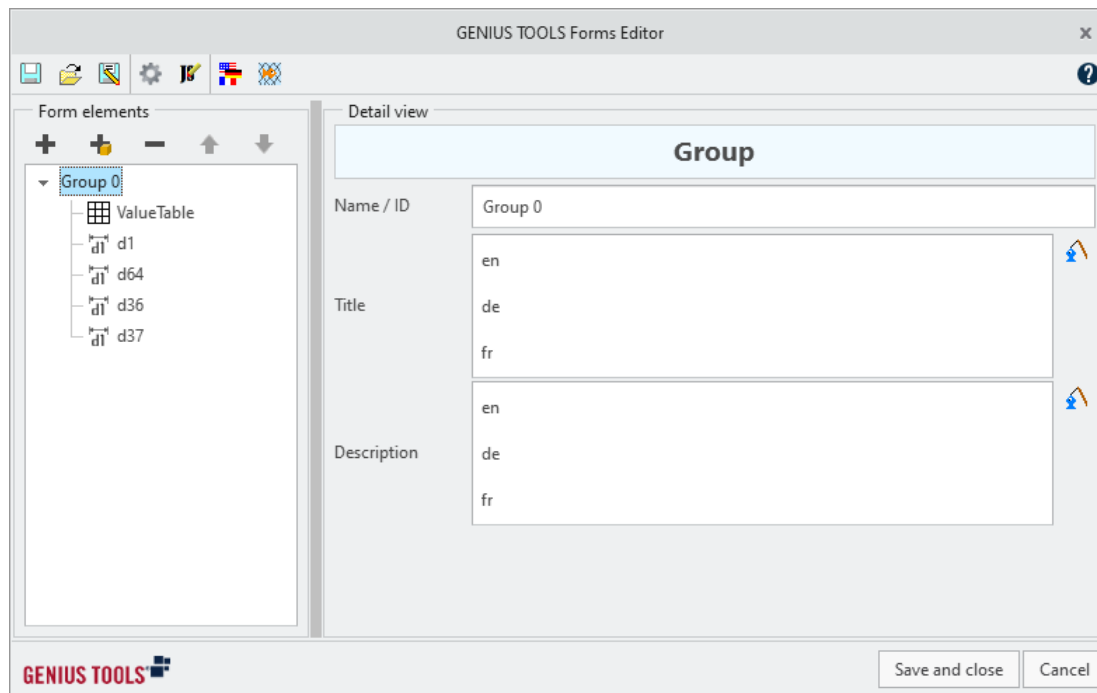
Initial model of a dice

2. Open the Creo window *Relations*.
3. Check the stored relations for compatibility and delete existing redundancies, if necessary.
4. Store comments in *Relations* to record the purpose of the relations. You can use this information when you create the form.

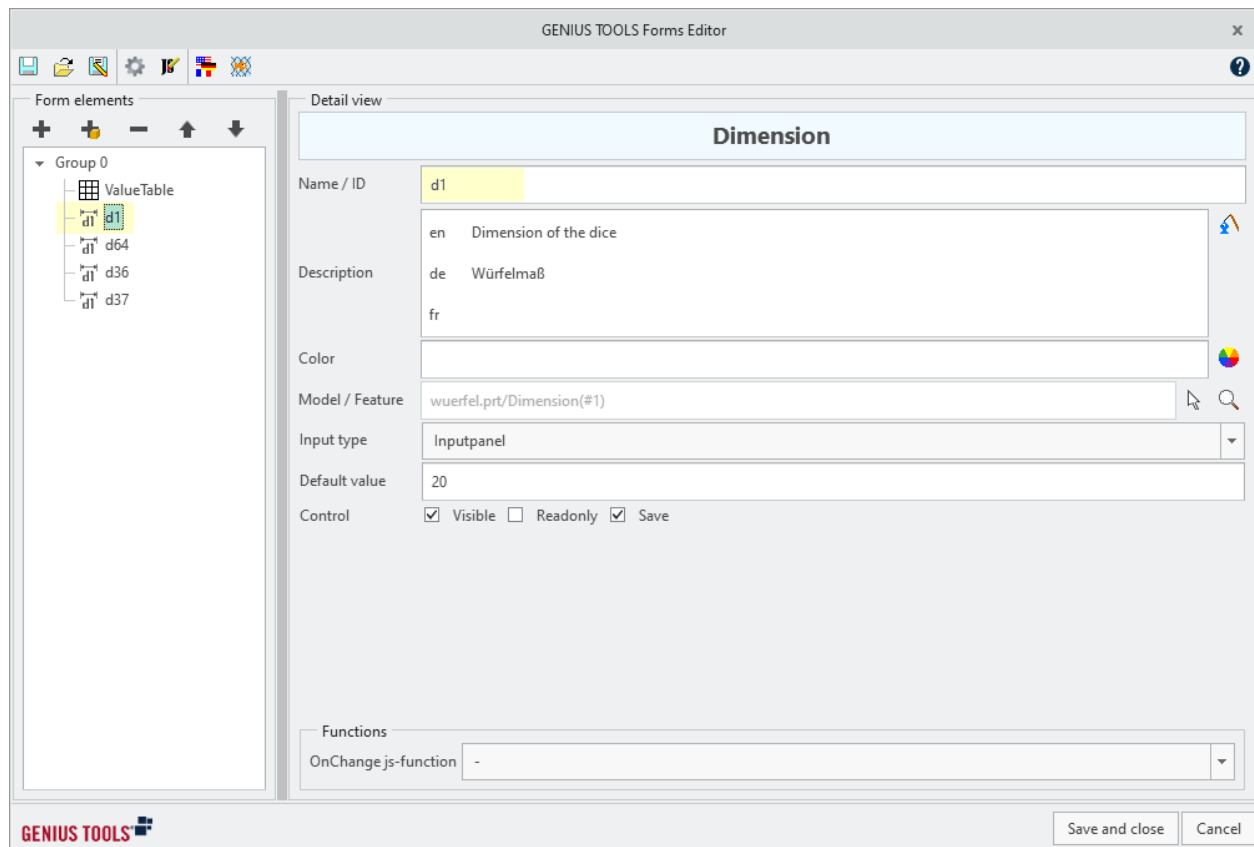



Clearly defined relation overview for sample model

5. Click OK to close the Creo window *Relations*.
6. Go to the *GENIUS TOOLS* tab and open the *GENIUS TOOLS Forms*  module.
7. Open the *GENIUS TOOLS Forms Editor*  to create the desired form.
8. Add a new group  to which the form elements will be attached.
9. Define properties of the group:



10. Add new items **+** to this group.
11. Select the *type of controlled element: Dimension*.
12. Select one side of the cube: *d1*. Since the size of the cube is defined so that all side edges have the same length, it does not matter which side you select at this point. All edges of the cube in the form to be created are controlled by the relation defined above.
13. Fill in the detail view:
 - a. Name / ID *d1* is filled in automatically. This name clearly indicates the relation.
 - b. The *Descriptions* are used to store the descriptions that will be displayed in the form for this form element in both English and German.
 - c. The input type *Inputpanel* allows to enter any values in the form.
 - d. An unchecked *Readonly* allows to assign different values to the input field in the form.
 - e. When the *Visible* and *Save* checkboxes are selected, the form element is visible and editable.



14. The form element for the dimension of the dice is now created. Repeat steps 10 and 11 for the edge radius, point diameter, and point depth. *Add new items*  to this group.
15. Select the *type of controlled element: Dimension*.
16. Select one side of the dice: *d64*. The round that defines the edge radius was inserted into the model last. Thus, the edge radius is the same for all cube edges and does not need to be defined via relations.

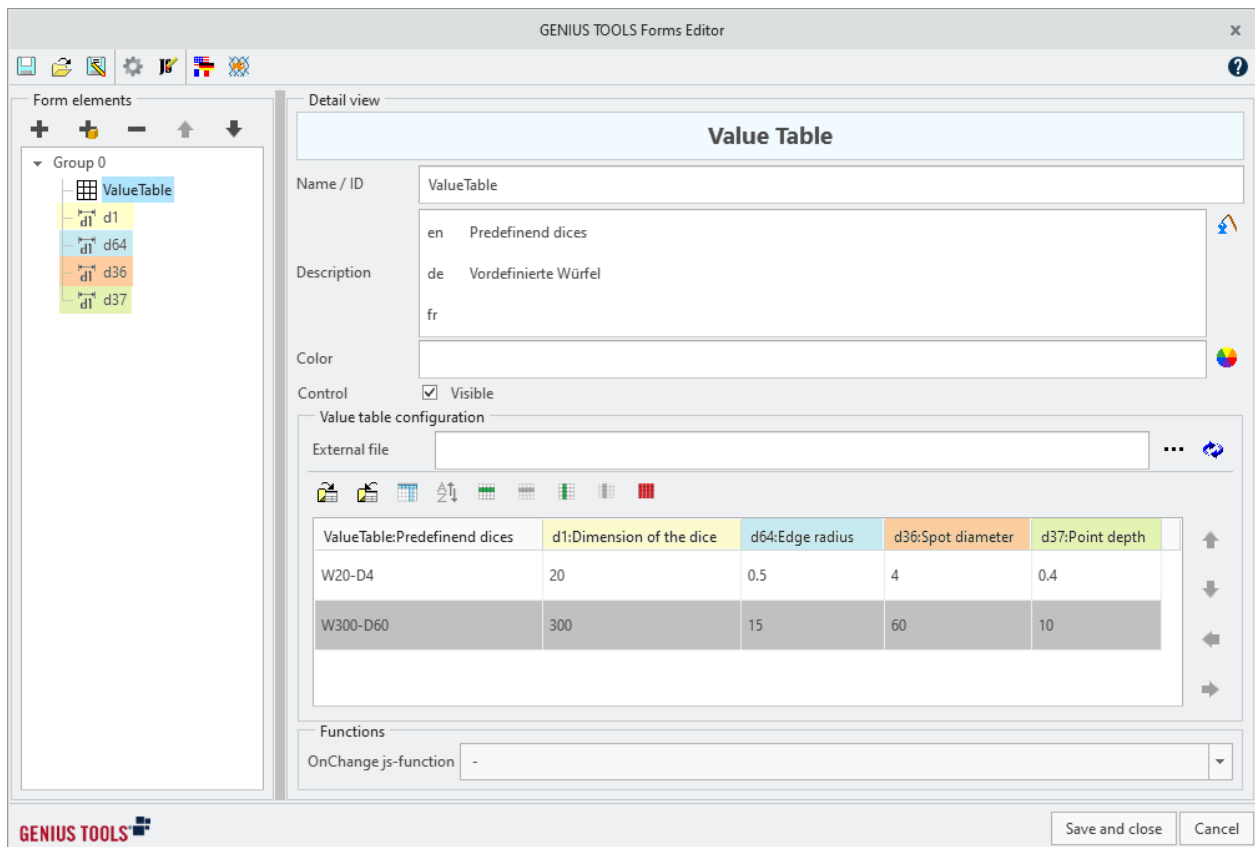
17. Add new items **+** to this group.
18. Select the *type of controlled element: Dimension*.
19. Select the point diameter of a number of pips of the cube: *d36*. The point diameter is defined the same for all numbers of pips.
20. Complete the detail view in the same way as the *d1* view.

21. Add new items **+** to this group.
22. Select the *type of controlled element: Dimension*.
23. Select the point depth of a number of pips on the dice: *d37*. The depth of the point is defined in the same way as the diameter of the point for all numbers of pips.
24. Complete the detail view in the same way as the *d1* view.

25. Create a value table. Add new items to this group.
26. Select the *type of controlled element: Value Table*.
27. Use to add a column to the value table for each of the form elements you created.

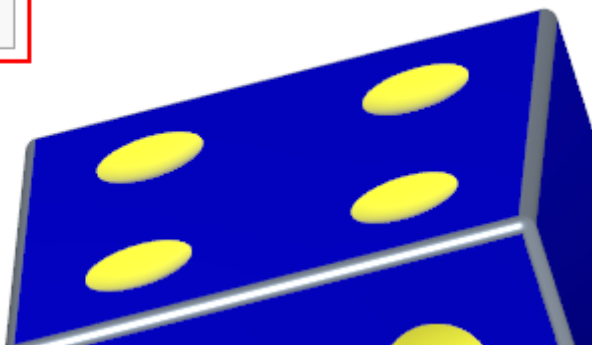
Please note: Value tables can only be created for existing form elements.


28. Add two rows to the value table using .
29. Fill in the rows with predefined cube sizes.
In the first row of the table, the predefined cube W20-D4 describes the initial model of the cube. With this predefinition, the initial state of the cube can be restored at any time.
The second row of the table describes a variance of all form elements.



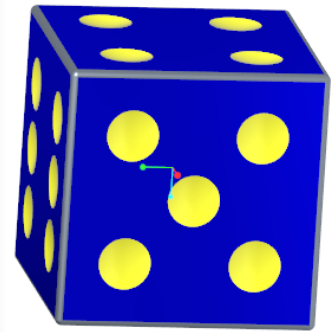
30. Click *Save and close*.

31. When you open *GENIUS TOOLS Forms*, you will see the created form directly. When you close *GENIUS TOOLS Forms*, you can also access the form directly in the open model.



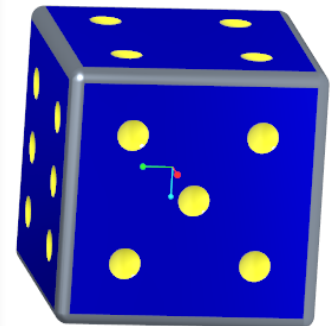
32. The following result is obtained by setting the predefined cube W300-D60 and then applying the value table by clicking the button .

ValueTable	<select...>	Predefinend dices
d1	20	Dimension of the dice
d64	0.5	Edge radius
d36	4	Spot diameter
d37	0.4	Point depth



33. The form elements can be set individually and the settings can be applied by clicking the button . For example, it looks like this

ValueTable	<select...>	Predefinend dices
d1	100	Dimension of the dice
d64	5	Edge radius
d36	12	Spot diameter
d37	2	Point depth



Setting tolerance from a value table automatically

Depending on the value of a dimension, the tolerance is to be set automatically. In a form with three elements (dimension, tolerance, value table), a JavaScript function⁵⁷⁷ is stored that is called when the dimension is changed. The JavaScript function sets the value of the value table from which the tolerance is set.

1. Create a new component with a cuboid.
2. Edit the dimensions of the cuboid.
name, tolerance table, display of the tolerance (Symmetric), tolerance value

3. Create a form with three elements:

4. Fill in the tolerance value table:

ValueTable_3	tpm2
value < 100	0.05
100 <= value < 200	0.1
200 <= value < 300	0.2
300 <= value < 400	0.3

5. Create the following JavaScript function:

```
function OnChangeLaenge() {
  l=getValue("laenge");
  setInputValue("ValueTable_3", l);
}
```

6. Specify the time at which the JavaScript function is to be executed. The function should be applied when there is a change in length.

Detail view

Dimension

Name / ID: laenge

Description: en, de, fr

Color: [Color Picker]

Model / Feature: quader.prt/Dimension(#2)

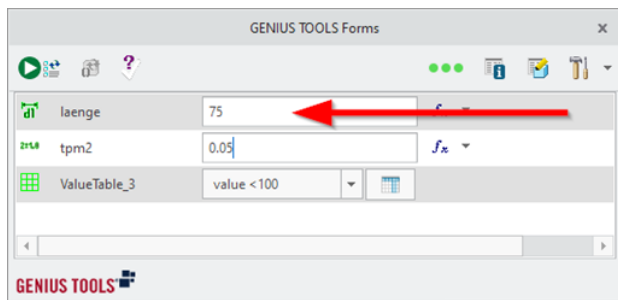
Input type: Inputpanel

Default value: 200

Control: ☒ Visible ☐ Readonly ☒ Save

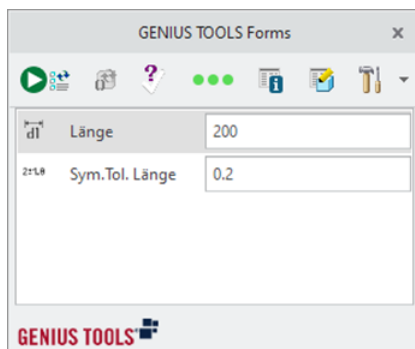
Functions: OnChange-js-function: OnChangeLaenge

7. Test that the JavaScript code works as expected. Change the length in the form. The value table and tolerance are automatically adjusted. The value table and tolerance change automatically.



Icon	Label	Value	Unit
dl	laenge	75	f _z
2+L8	tpm2	0.05	f _z
ValueTable_3	value < 100		

8. Optionally, you can optimize the display of the *GENIUS TOOLS Forms* dialog, e. g. by hiding the value table and configuring the column display.



Icon	Label	Value
dl	Länge	200
2+L8	Sym.Tol. Länge	0.2

8 Inspect

Insert inspection symbols and inspection symbol tables on drawings in Creo Parametric with GENIUS TOOLS Inspect.

Inspect is available in drawing mode with the following features.

1. free placement of inspection symbols
2. placement of inspection symbols linked to
 - dimensions
 - shape and position tolerances
 - surface quality symbols
 - notes
 - symbols
3. free placement of inspection symbol tables
 - numbering tables
 - report tables
4. numbering of inspection symbols by height, by symbol type, similar to DIN 6770 (numbers are not assigned anew)
5. data export to Excel

The component **GENIUS TOOLS Inspect Revision**¹⁶⁹ creates a snapshot of all inspection symbols on a drawing at a point in time as well as a history of all snapshots.

You can also use symbols in GENIUS TOOLS Inspect to mark **changes on a drawing**¹⁶⁵.

8.1 Fundamentals

Glossar

Inspection symbols

Numbered symbols marking characteristics that have to be considered in quality control because they are critical to the quality of function of the finished part or product.

Inspection symbols table

Overview of all inspection symbols used in a drawing.

Inspect configuration

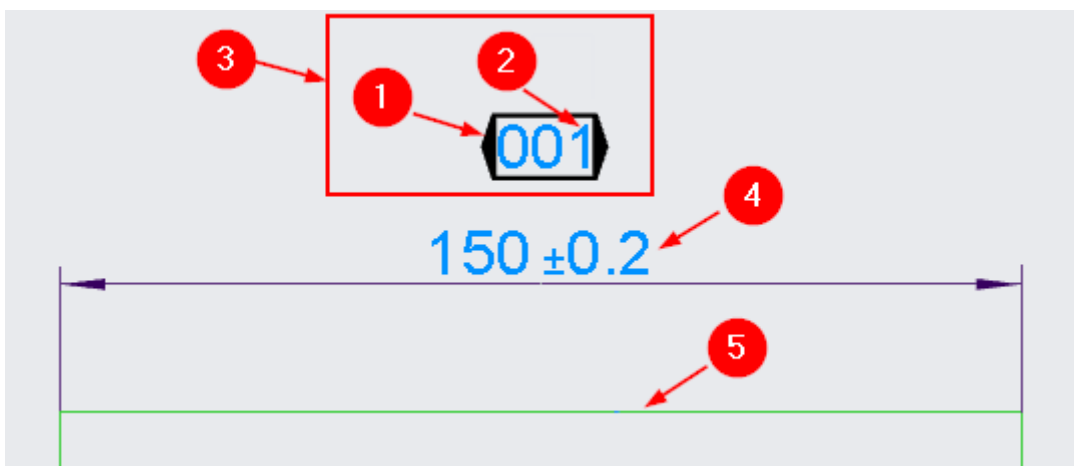
XML file that contains definitions for symbols, tables and display settings. (See also [Inspect configuration](#) ¹⁴⁹.)

Structure of an inspection symbol

The figure below shows an inspection symbol generated with GENIUS TOOLS Inspect with its individual components.

Each inspection symbol (numbered symbol) marks a characteristic in a technical drawing.

- 1) symbol
- 2) number of the characteristic
- 3) inspection symbol, consisting of the symbol (1) and the number of the characteristic (2)
- 4) value of the characteristic
- 5) characteristic in the drawing, e.g., length of a geometry element



Font used for out-of-the-box symbols

The symbols delivered with Inspect use the font *ISONORM LT Regular* based on ISO 3098-5, which is part of Creo in version 4 and later.


If you experience issues with displaying the out-of-the-box symbols, please check whether this font is available in your system.

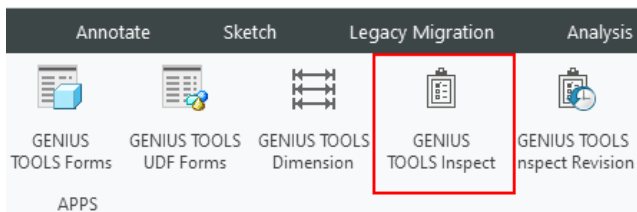
You can create your own symbol files to use as inspection symbols. For more information on how to do this, please refer to [Creating a Creo symbol for Inspect](#) ¹⁵⁸.

8.2 Usage

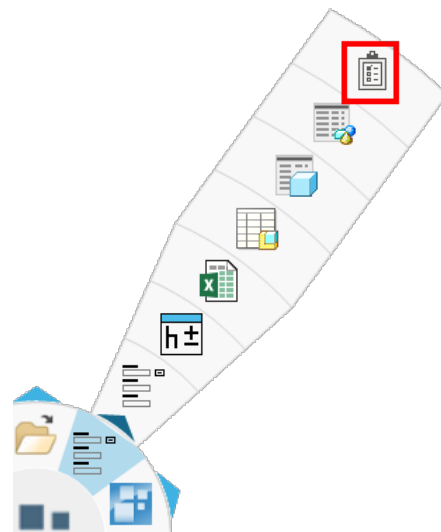
This section contains information on using GENIUS TOOLS Inspect. It describes the general structure of the program.

Starting the program: in drawing mode

Start GENIUS TOOLS Inspect  in drawing mode from the GENIUS TOOLS ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).



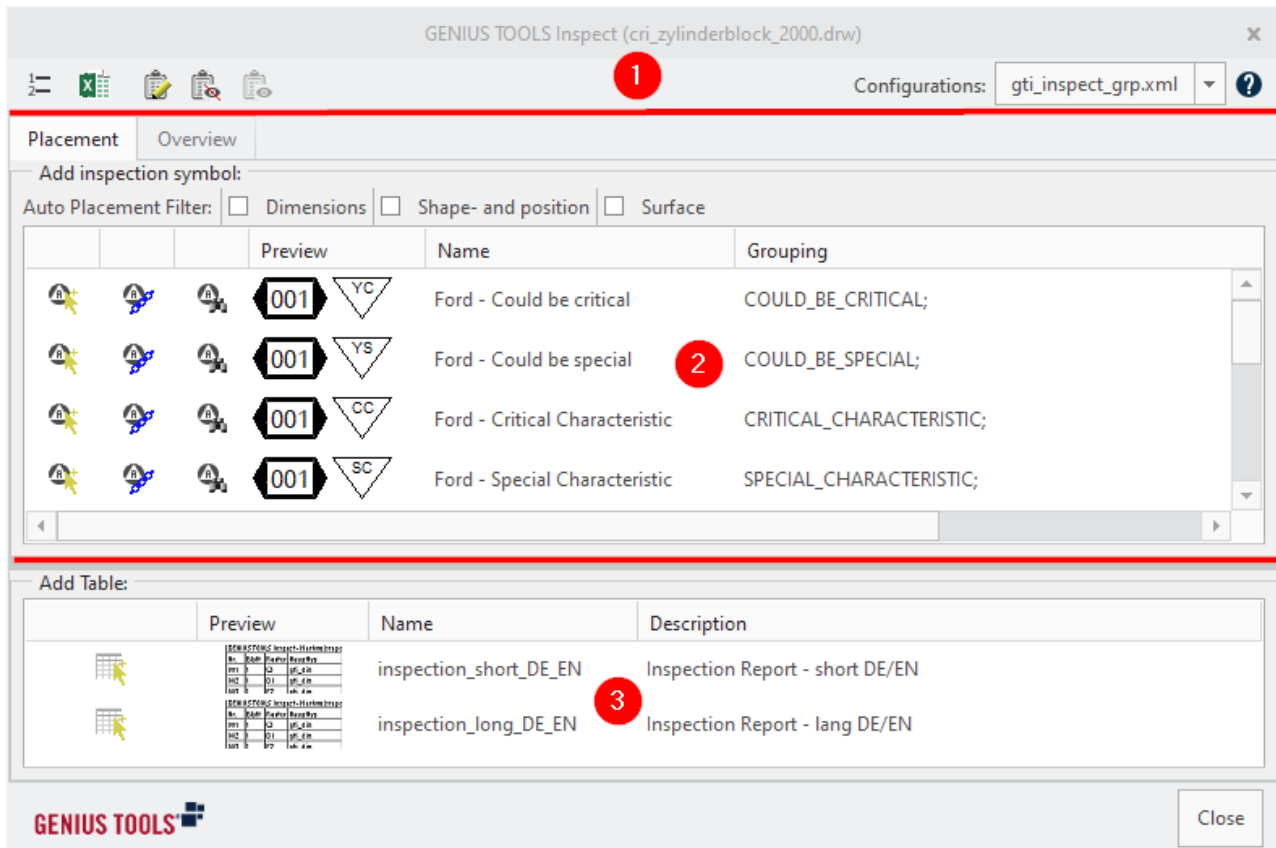
Starting from GENIUS TOOLS ribbon menu



Starting via Quick Access

8.2.1 User interface

The user interface consists of the following elements:




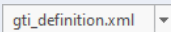



1. Command bar
2. Symbol overview
3. Table overview

8.2.2 Command bar

The command bar displays general control elements. The following buttons are included:

Icon	Name	Description
	Number symbols	Numbers all placed symbols. The numbering is determined by configuration options.
	Export overview to Excel	Opens the export dialog for the symbol table.

Icon	Name	Description
	Hide Inspect Elements	Hide all Inspect elements. WARNING: Linked symbols are unlinked from their targets. They do not become free symbols, see also Hide and unhide ¹⁴³ .
	Unhide Inspect Elements	Unhide all Inspect elements. WARNING: Linked symbols are re-linked to their targets. If the target for a symbol cannot be found in the drawing, the symbol is deleted.
	Open Inspect Editor	Opens the editor.
	Configuration selection	Switches between multiple Inspect configurations that can contain different symbol and table definitions as well as view settings.
	Open Help	Opens the help.

Always place all symbols before numbering them.

The numbering of symbols is influenced by the [configuration options](#)⁶¹⁰ `gti_number_sort_at_height`, `gti_number_sort_at_type`, `gti_start_number` and `gti_din_compliant`. For more information, please refer to [Numbering](#)¹³⁷.

Warning: Do not use the same symbol in different Inspect configurations if you are using numbering similar to DIN 6770. Otherwise, you may experience numbering conflicts.

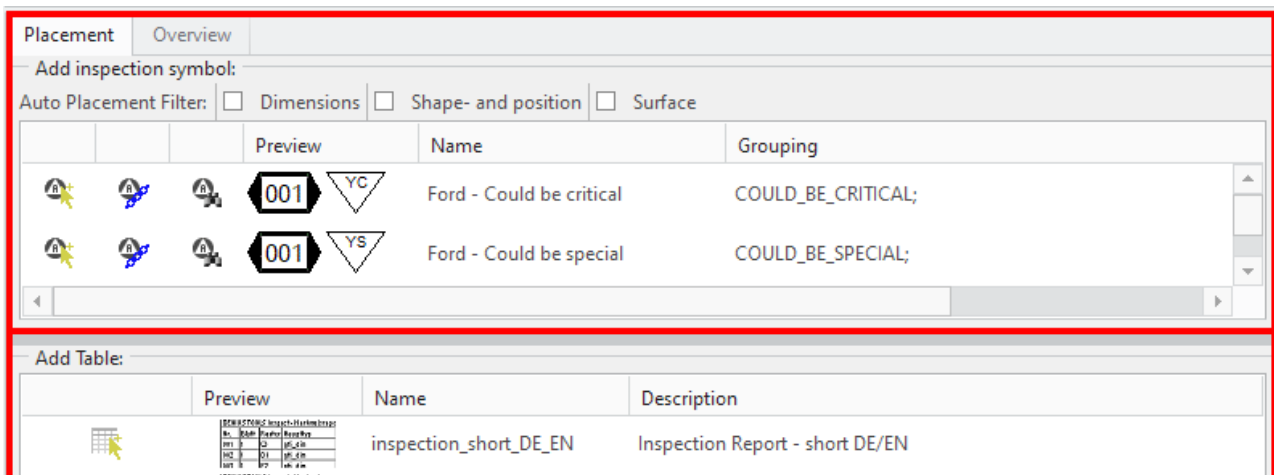
8.2.3 Positioning and Overview tabs

The user interface is divided into two tabs: *Placement* and *Overview*.

The columns displayed, the order of the columns, and the order of the inspection symbols are defined in the [Editor](#)¹⁴⁷.

1. Placement

The *Placement* tab is used for placing inspection symbols and tables on drawings.



Available inspection symbols are displayed under *Add inspection symbol*. Each inspection symbol can be added to a drawing as an independent element or linked to an object. Click on the respective icon to add inspection symbols to a drawing.

Inspection symbols

Free insertion : Use the left mouse button to place inspection symbols on the drawing as independent elements. Use the middle mouse button to cancel the placement.

Linked insertion : Place inspection symbols in two steps. First, select a drawing item with the left mouse button. Second, place an inspection symbol on the drawing using the left mouse button. Use the middle mouse button to cancel the placement.

Inspection symbols can be linked to dimensions, shape and position tolerance symbols, surface quality symbols, notes and drawing symbols displayed on drawings. These inspection symbols will be attached to the drawing item with equal alignment.

Linked insertion by autoplacement : Inspection symbols are automatically placed on all selected drawing items (Dimensions, Shape and position, Surface). Items which are already linked to the selected inspection symbol are not re-linked. GENIUS TOOLS Inspect checks whether the SYM file of the selected inspection symbol matches the existing inspection symbol. These inspection symbols will be attached to the drawing item with equal alignment.

Tables

Available symbol tables are displayed under *Add table* . Click on the placement icon and place the table freely on the drawing. There are two types of tables:

Numbering table: Overview of the types of inspection symbols and the numbers used.

Report table: Overview of all properties of the inspection symbols.

Tables are created in the [Editor](#)¹⁴⁷.

2. Overview

The *Overview* tab shows all inspection symbols that have already been placed. Additional information (e. g. tolerance information) is displayed for linked inspection symbols.

GENIUS TOOLS Inspect								
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <input checked="" type="checkbox"/> Dimensions <input checked="" type="checkbox"/> Shape- and position <input checked="" type="checkbox"/> Surface <input checked="" type="checkbox"/> Notes <input checked="" type="checkbox"/> Symbols </div> <div>Configurations: gti_inspect.xml</div> <div>?</div> </div>								
<div> <div>Placement</div> <div>Overview</div> </div>								
Linked inspection symbols:								
Characteristic	Sheet	Grid	Main type	Subtype	Nominal dimension	Off-size	Minimal dimension	Maximal dimension
001	2	B5	Dimension	↔ (Linear)	60			
002	2	B5	Dimension	↔ (Linear)	44	-0.3 / 0.3	43.7	44.3
003	2	B4	Dimension	↔ (Linear)	30			
004	2	B4	Dimension	↔ (Linear)	15	-0.2 / 0.2	14.8	15.2
005	2	C4	Dimension	↔ (Linear)	7.75	-0.2 / 0.2	7.55	7.95
006	2	C2	Dimension	↔ (Linear)	50	-0.3 / 0.3	49.7	50.3
007	2	C3	Dimension	↔ (Linear)	30	-0.2 / 0.2	29.8	30.2
Free inspection symbols:								
Characteristic number	Sheet	Grid	Creo symbol	Description	Symbol			
000	2	B7	-		DIN 6770			
000	2	A8	-		DIN 6770			
000	2	C8	-		DIN 6770			

The overview is controlled by the Editor.

Linked inspection symbols

Use the filter in the top bar to show or hide inspection symbols linked to dimensions, shape and position tolerances, surface qualities, notes or symbols.

The out-of-the-box inspection symbols include a description parameter. Its content is displayed in the *Description* column and can be changed.

You can add information (e. g. from a revision parameter) by creating a new column. In GENIUS TOOLS Inspect Editor go to the tab [Inspection symbol > Parameters](#)¹⁴⁹.

Linking free inspection symbols

Right-click on an independently-placed inspection symbol. Then select the context menu item *Link element* and select a drawing item to link to.

Please note: Using the configuration option `gti_lang` you define - not depending on the system language - the language used for the contents of the overview table.

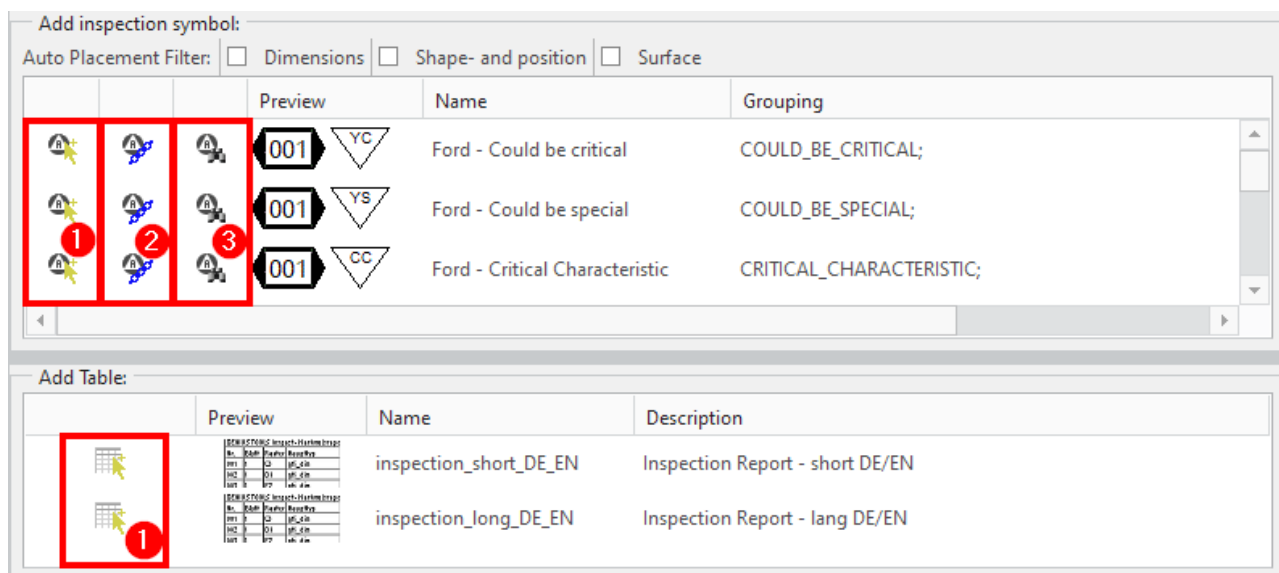
8.2.4 Usage examples

In this section, the usage of GENIUS TOOLS Inspect is explained using examples.

1. Positioning¹³⁶: Ways to place and link inspection symbols and tables on a drawing.
2. Numbering¹³⁷: Numbering options and their effect on numbering.
3. Overview¹⁴⁰: Overview tab functionality.
4. Hide and unhide¹⁴³: Showing and hiding inspection symbols and tables.
5. Export¹⁴⁴: Excel export of inspection symbol tables.

8.2.4.1 Positioning

In addition to the free placement of elements, there are additional options for attaching inspection symbols to elements.



User interface for placing inspections symbols and tables on engineering drawings

Free symbols: Free placement of inspection symbols and tables (1)

Press the button for free placement of an inspection symbol or a table and then select its position on the drawing with the left mouse button.

You can position several inspection symbols and then press the middle mouse button to cancel the operation.

Linked placement of inspection symbols (2)


Simple linked placement

Click the button , then select a drawing item using the left mouse button.


Now place the inspection symbol on the drawing using the left mouse button. You can position several symbols, then stop the process by clicking the middle mouse button.

Linked placement by previous selection (single elements)

Select one or more elements in the drawing.


You can then automatically insert inspection symbols by clicking the button . These inspection symbols will be attached to the drawing item with equal alignment.

Linked placement by previous selection (multiple elements)

Select multiple elements or an area in the drawing. Then you can automatically insert inspection symbols by pressing the button .

Inspection symbols will be linked to all dimensions in the selection. These inspection symbols will be attached to the drawing item with equal alignment.

Linked-Automatic placement (3)

Click the button for automatic insertion  to place inspection symbols on unlinked drawing elements. These inspection symbols will be attached to the drawing item with equal alignment.

You can control the drawing elements to be considered (Dimensions, Shape and position, Surface) by filter (Auto Placement Filter).

8.2.4.2 Numbering

The numbering of the inspection symbols is influenced by the following configuration options: `gti_number_sort_at_height`, `gti_number_sort_at_type`, `gti_start_number`, `gti_din_compliant`, `gti_numbering_all_sheets` and `gti_number_range_per_sheet`.

Configuration option	Input value		
<code>gti_number_sort_at_height</code>	-1 (descending)	0 (no sorting)	1 (ascending)
<code>gti_number_sort_at_type</code>	-1 (descending)	0 (no sorting)	1 (ascending)
<code>gti_start_number</code>	Numerical value for numbering start		
<code>gti_din_compliant</code>	0 (off)	1 (on)	

Configuration option	Input value	
gti_numbering_all_sheets	0 (off)	1 (on)
gti_number_range_per_sheet	0 (off)	1 (on)

Numbering per sheet or drawing

With the configuration option `gti_numbering_all_sheets` you can control whether symbols are sorted and numbered per sheet (`gti_numbering_all_sheets=0`) or globally over the entire drawing (`gti_numbering_all_sheets=1`). Whether in the case of `gti_numbering_all_sheets = 0` a separate number range or a consecutive numbering should be used is controlled with the configuration option `gti_number_range_per_sheet`.

Number ranges

With the configuration option `gti_number_range_per_sheet` you can control whether each sheet (`gti_number_range_per_sheet=1`) has its own number range. Otherwise (`gti_number_range_per_sheet=0`) one number range is used for the whole drawing.

Warning: This configuration option will only be read if the options `gti_numbering_all_sheets` is set to 0 or `gti_din_compliant` is set to 1.

Numbering similar to DIN 6770

The configuration option `gti_din_compliant` determines whether numbering is similar to DIN 6770.

To be able to determine the next number, the highest number assigned is stored in the parameter `GTI_LAST_SYM_<configuration name><sheet index>`. If `gti_number_range_per_sheet` is set to 1, the highest number per sheet is remembered.

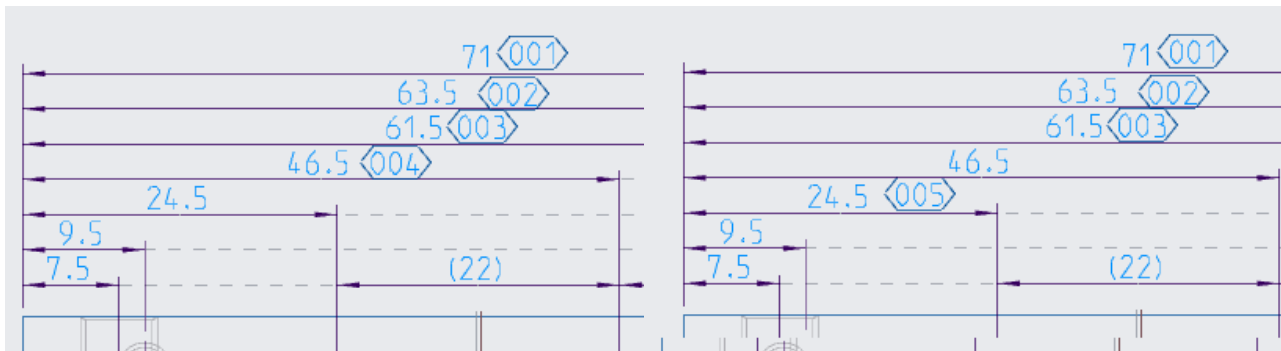
Only inspection features with the number 000 are re-numbered. 000 is the default value for newly placed inspection symbols.

The configurations `gti_numbering_all_sheets`, `gti_number_sort_at_height` and `gti_number_sort_at_type` are not read out when using DIN 6770 (`gti_din_compliant=1`) numbering. It is automatically sorted by height *from top to bottom* (`gti_number_sort_at_height=-1`) and over each sheet (`gti_numbering_all_sheets=0`).

If you do not set the option, i. e. if `gti_din_compliant=0`, values from `gti_numbering_all_sheets`, `gti_number_sort_at_height` and `gti_number_sort_at_type` are read.

Example

In the left figure, four new inspection symbols have been placed and numbered consecutively. If the inspection symbol with the number 004 is deleted, the number 004 remains stored in the parameter `GTI_LAST_SYMBOL`. In the right figure, a new inspection symbol has been placed. When the inspection symbols are re-numbered, the new symbol is numbered 005, not 004.



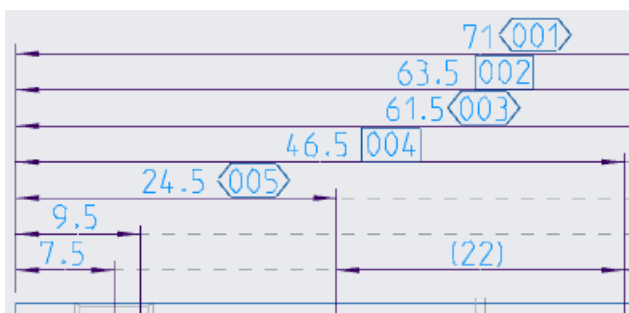
Placement of the inspection features and subsequent numbering with 001 - 004.

Placement of a new inspection feature and subsequent numbering with 005.

Numbering order

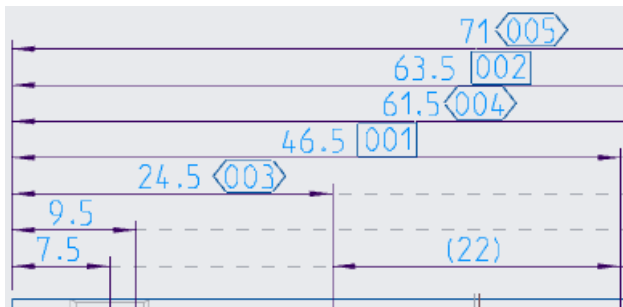
Numbering order is determined by the configuration options `gti_number_sort_at_height` and `gti_number_sort_at_type`.

The option `gti_number_sort_at_height` specifies whether the inspection symbols should be numbered by height, from *top to bottom* (`gti_number_sort_at_height=-1`) or from *bottom to top* (`gti_number_sort_at_height=1`).



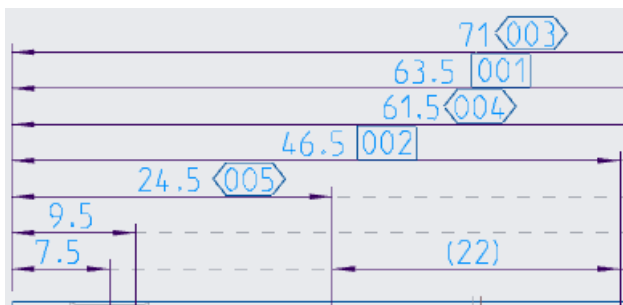
Numbering by height.

The option `gti_number_sort_at_type` specifies whether the inspections symbols are to be numbered by inspection symbol type, should be numbered alphabetically *descending* (`gti_number_sort_at_type=-1`) or *ascending* (`gti_number_sort_at_type=1`).



Numbering by type.

If both numbering options are active at the same time, numbering is first by inspection symbol type and then by height within a type.



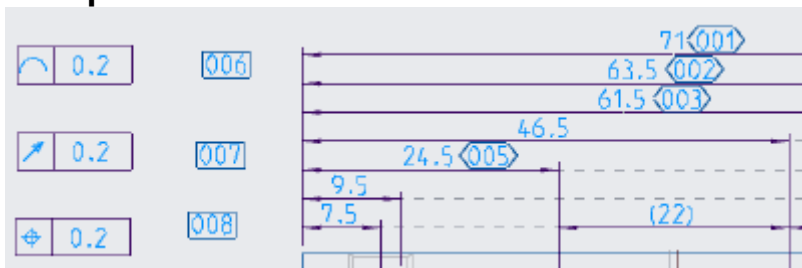
Numbering by type and height.

8.2.4.3 Overview

In the Overview tab, all relevant data on the inspection symbols is displayed in a table.

Use the Editor to define the displayed columns and the column order.

Example



Placement

Overview

Linked Symbols:

Symbolnum...	▲	Main type	Subtype	Nominal dimens...	Minimal dimensi...	Maximal dimens...
001		Dimension	↔ (Linear)	71	70.7	71.3
002		Dimension	↔ (Linear)	63.5	63.2	63.8
003		Dimension	↔ (Linear)	61.5	61.2	61.8
005		Dimension	↔ (Linear)	24.5	24.3	24.7
006		Shape and position	∩ (Profile of a line)			
007		Shape and position	↗ (Run-out)			

Free Symbols:

Symbolnum...	▲	Description	Sheet	Grid	Name of symbol	Creo symbol
008		-	1	D2	Main	

GENIUS TOOLS

Close

Filtering

If you hide inspections symbols that are linked to a defined type of drawing element (e.g., dimensions or tolerances), columns that are only relevant for this type will also be hidden. Which columns are relevant to which type is defined in the Editor.

GENIUS TOOLS Inspect (cri_zylinderblock_2000.drw)

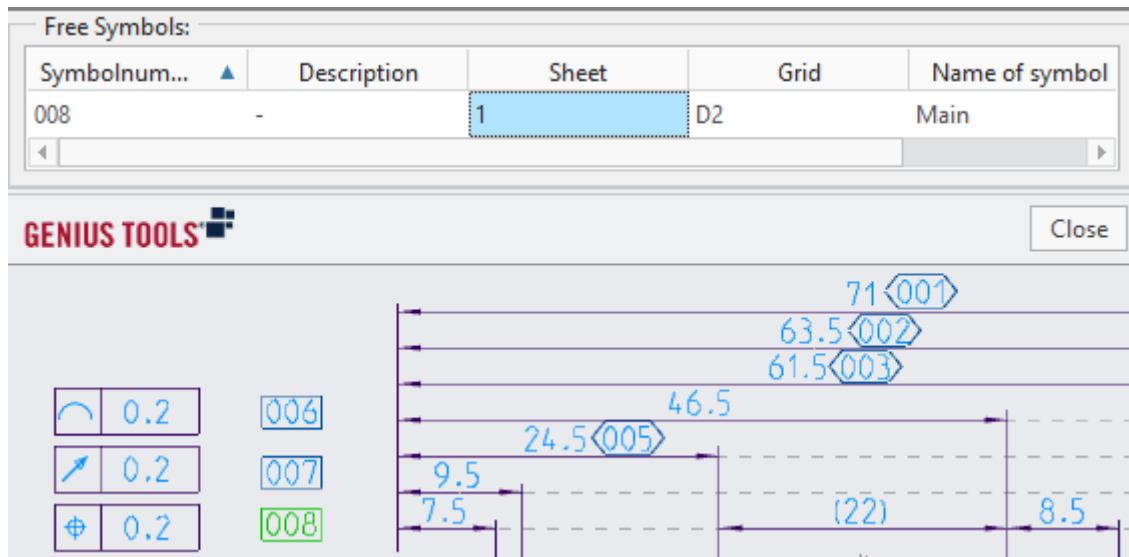
</

Sorting

By clicking on a column header, the inspection symbols are sorted according to this column. By default, the symbols are sorted by inspection symbol number (ascending).

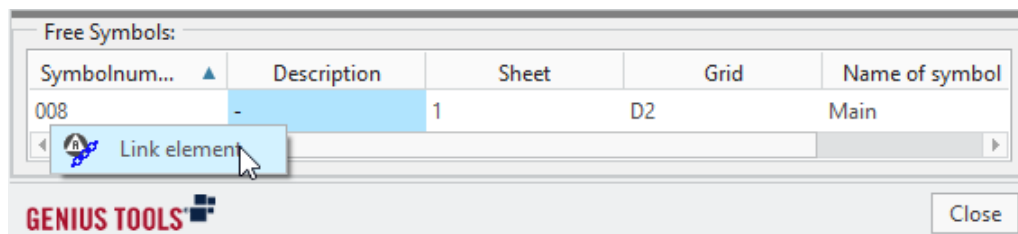
Highlighting

A click on any cell in an inspection symbol's row highlights the inspection symbol in the drawing.



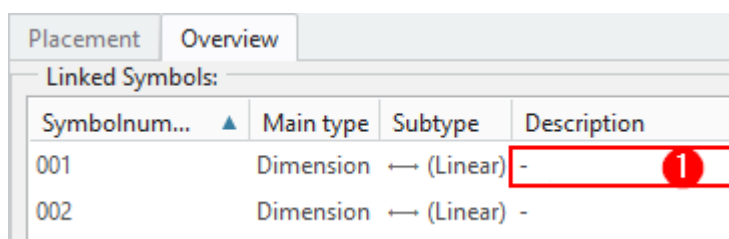
Linking free symbols

You can link free inspection symbols to drawing elements at a later date. To do so, right-click the inspection symbol to be linked in the *Free symbols* table. Now click *Link element* and select the corresponding drawing element by left-clicking.



Insert description

A left click in the description column opens an input dialog for entering a description. This description is saved in the inspection symbol and will be available the next time Inspect is started.



The description field is variable text¹⁵⁹, which is saved in a Creo symbol.

Tip: When you are in an input, you can jump to the next input line with *Tab*.

8.2.4.4 Hide and unhide

Hiding Inspect elements



To hide all Inspect elements, click the button shown above (1).

All Inspect tables and Inspect inspection symbols that contain the variable text `LinkedTo` are hidden.

Symbols without the variable text `LinkedTo` are not hidden. This can be the case for inspection symbols created with version of Inspect prior to 6.0. The variable text has to be added to these symbols if you want to use the hide functionality.

When hiding Inspect elements, linked inspection symbols are unlinked from their target elements. This does not turn them to free symbols – GENIUS TOOLS Inspect saves the attributes of the target element to the variable text `LinkedTo` to be able to re-link the inspections symbols when they are unhidden.

Unhiding Inspect elements



To unhide all Inspect elements, click the button shown above (2).

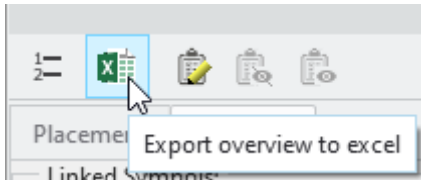
All inspection symbols which were linked before hiding are re-linked to their target elements.

Warning: Inspection symbols whose target elements have been deleted are deleted on un hiding!

8.2.4.5 Export to Excel

Exporting inspection symbols to Excel

To export your inspection symbols to Excel, click the button shown below. This opens the dialog [Export table to Excel](#)⁵¹⁰, in which you select the Excel template, export file and configuration.



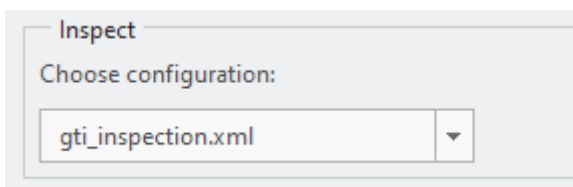
Starting the Excel export can take some time because the Excel template has to be checked.

Warning: Close all Excel windows bevor starting the export.

Export dialog

By default, the out-of-the-box template is selected and a file name is generated from the name of the drawing. You can also design an [individual Excel template](#)⁵¹².

The selected configuration determines which inspection symbols should be exported. Inspection symbols will be exported if the selected configuration refers to their symbol file, i. e., if the configuration contains their inspection symbol type.



Configure path to templates

The default selection of a template can be changed by using these options:

gti_excel_template

Defines the Excel template.

gti_folder

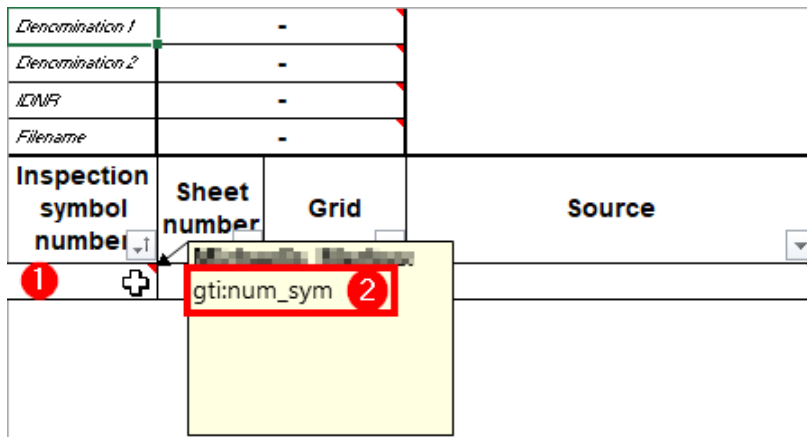
Defines the directory, which contains the Excel template. (Default: `<gt_resource_folder>`)

Customizing the Excel template

If you want to create your own template, you can use the supplied template `gti_inspection_template_de_en.xlsx` as a basis and add a comment (2) to the first value cell

(1) of the desired column. The chapter [Create template](#)⁵¹² describes how to set up an export template step by step.

Please note: This function is called *Comment* until Excel 2019, from Microsoft 365 onwards it is called *Note*.



Comment used for column assignment.

The text of the comment determines which values are to be copied into the column. For the head parameters variables without component acronyms are possible, e. g. %DRAWING_NO%, %curmod:DESCRIPTION_1_DE%.

A	B	C	D	E	F	G
Zeichnung-Nr. / Drawing no.:	-					
Benennung / Description 1:	-					
Bezeichnung / Description 2:	-					

GTI:
%curmod:DESCRIPTION_1_DE%

During table export, all tags in a comment are searched and replaced with the appropriate value. The remaining text in the comment is preserved.

EN	Tol. Standard	Tol. Tabe
Dimensions		

Min. dim: gti:min_dim
Max. dim: gti:max_dim

Comment on the filling of the column with tags and additional text to be displayed.

EN
Dimensions
Min. dim: 50.15
Max. dim: 50.2

Display in the exported table

For report parameters, the text in the comment must consist of the component abbreviation *gti:* and the fill command:

Comment text	Column name
gti:num_sym	Symbol number
gti:tpe_main	Main type
gti:tpe_sub	Subtype
gti:cls_tol	Tolerance class
gti:bse_dim	Nominal dimension
gti:min_dim	Minimal dimension
gti:max_dim	Maximal dimension
gti:tpd_dim	Theoretically precise dimension
gti:ipc_dim	Inspection dimension
gti:val_tol	Off-Size
gti:descr	Description
gti:gti_param	Parameter
gti:gti_note	Note
gti:tpe_tol	Tolerance standard
gti:num_sheet	Sheet
gti:grd	Grid
gti:src	Source
gti:mod	Tolerance table
gti:nme_sym	Name of symbol
gti:tpe_sym	Creo symbol
gti:<columnName>	Output of additional user-defined parameters

Creating a template with multiple spreadsheets

You can use other module acronyms besides *gti*. Thus you can export data from *GENIUS TOOLS Inspect* and *GENIUS TOOLS Inspect Revision* together, see chapter [Export data from several GENIUS-TOOLS-components](#)⁵¹⁷ for an example.

8.3 Configuration

This section provides further information on the configuration and structure of the Inspect Editor.

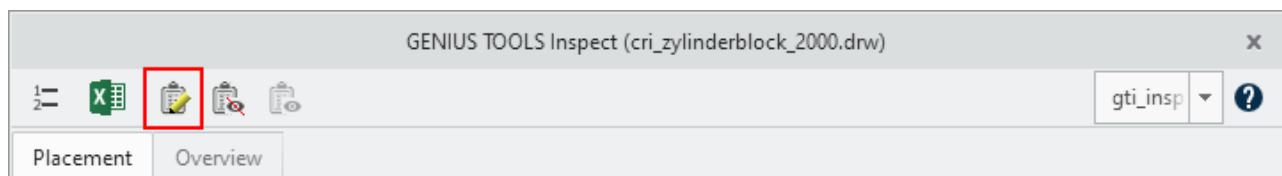
8.3.1 Inspect Editor

The Inspect Editor is used to create and manage settings for GENIUS TOOLS Inspect. You can define several sets of settings, consisting of inspection symbols, tables and view settings. Each configuration is stored as an XML file in the *gti_folder*.

Users can switch between the different configurations in the UI.

Starting the program

Start the editor  from the command bar of GENIUS TOOLS Inspect.



Saving changes

The changes made in the editor are saved in the resource directory.

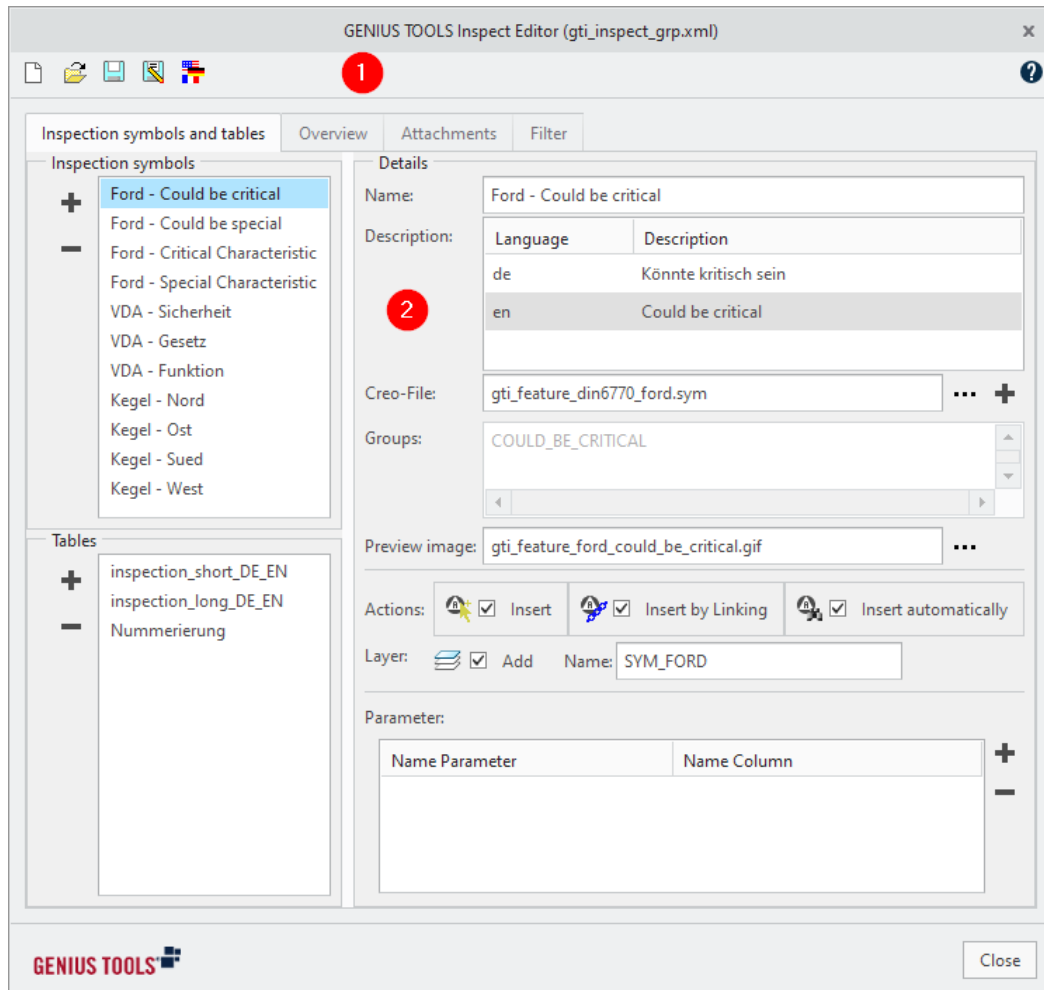
When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer (Caddepot). Path: `<operatingenvironment>/parametric/configuration/gt_resource_folder`.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

8.3.1.1 User interface

The user interface of Inspect Editor consists of the following elements:









1. Command bar
2. Detail view of the tab Inspection symbols and tables

The editor is divided into four tabs: *Inspection symbols and tables*¹⁴⁹, *Overview*¹⁵⁶, *Attachments*¹⁵⁶ and *Filter*¹⁵⁷.

8.3.1.2 Command bar

In der Befehlsleiste sind die folgenden Schaltflächen enthalten:

Symbol	Name	Description
	New configuration	Creates a new <i>Inspect configuration</i> ¹⁴⁹ (XML). Each configuration can contain different symbol and table definitions as well as view settings.

Symbol	Name	Description
	Open configuration	Opens an existing configuration from an XML file.
	Save	Saves the current configuration to an XML file.
	Save as	Saves the current configuration under a new name.
	Edit localization	Adds or deletes a language from a configuration.
	Open help	Opens the help.

8.3.1.3 Inspect configuration

A configuration in GENIUS TOOLS Inspect is an XML file that contains definitions for symbols, tables and display settings. New configurations are created by the editor.

Inspect directory: Configurations are by default saved in the inspect directory in the resource folder (`%gt_resource_folder%inspect`).

Find inspect directory: The path to the directory can be edited by the option `gti_folder`.

Save configuration: The name of a configuration file must not exceed 18 characters.

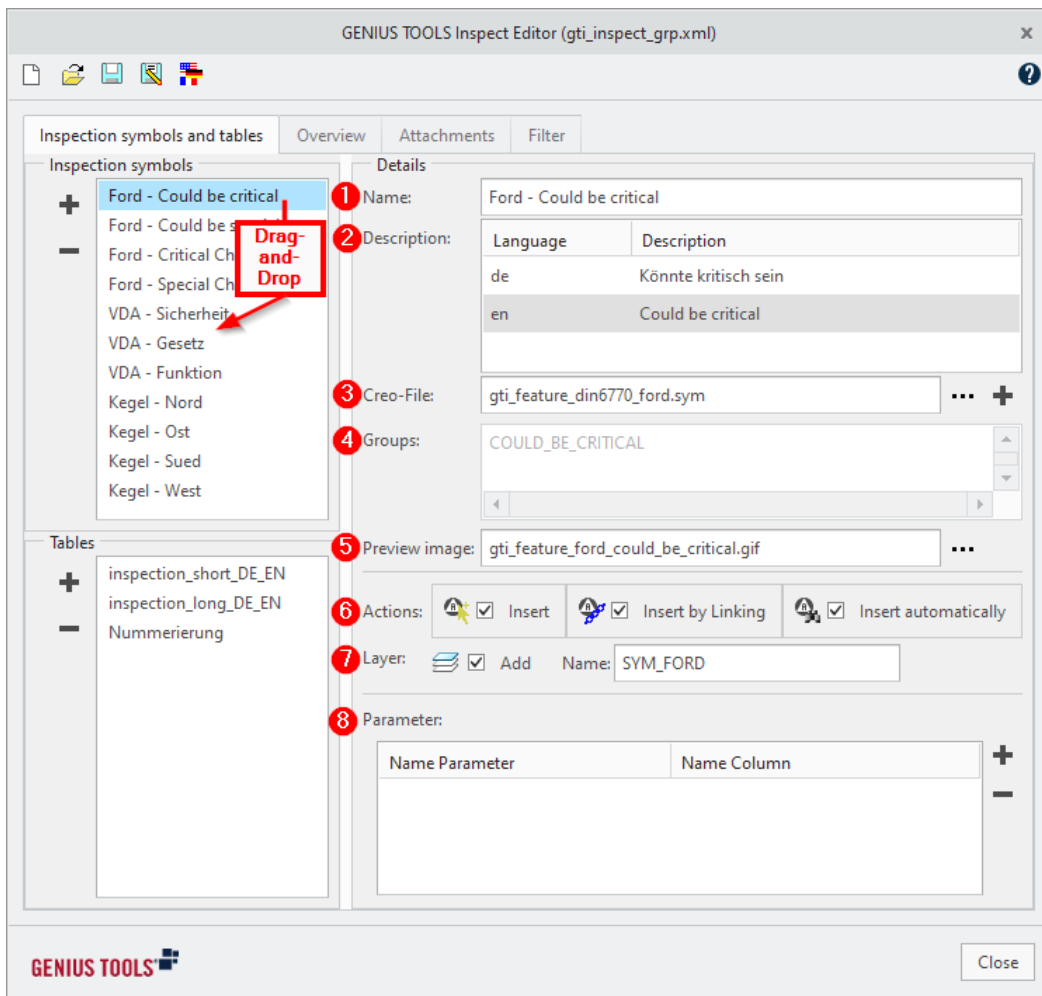
Warning: Configuration files that have been manually created with names longer than 18 characters cannot be opened by the editor. They must be edited manually.

8.3.1.4 Symbols and tables tab

In the tab *Inspection symbols and Tables*, the different sets of settings for inspection symbols and tables are configured. Use the (+/-) buttons to add or remove elements.

Click an inspection symbol or table in the element selection to view the details.

The order of the inspection symbols and tables determines their order in the user interface of GENIUS TOOLS Inspect. An elements position can be changed by drag-and-drop, dropping the element below the target element.



Click on a name of an inspection symbol or table to see the according details on the right of dialog.

Inspection symbols

- 1. Name:** Specifies the displayed name of the inspection symbol.
- 2. Description:** A localized description of the inspection symbol. Use the localization button in the command bar to manage the languages.
- 3. Creo-File:** Specifies the path to a SYM file. Default is `gt_resource_folder`.

Warning: Do not use the same symbol in different Inspect configurations if you are using numbering similar to DIN 6770. Otherwise, you may experience numbering conflicts.

- 4. Grouping:** If you are using a symbol that contains groupings, create a separate inspection symbol for each symbol variant that you want to place using Inspect. If you select a symbol file that contains groupings, the first grouping found is selected for the inspection symbol. To select the required symbol variant for the inspection symbol, proceed as follows.

- Open a drawing that contains the required symbol variant, or place the required symbol variant on a drawing.
- Open Inspect Editor and create the new inspection symbol. Select the symbol file under Creo-File using the Browse button (...).
- Click on the plus symbol next to the Creo-File input field. You are asked to select an element.
- Select the required symbol in the drawing. The grouping settings of this symbol are written to the inspection symbol definition.

5. Preview image: Specifies the path to a preview file of an inspection symbol.

6. Actions: Defines the actions that can be used on an inspection symbol.

7. Layer: Defines the layer on which the symbols are placed. Enter the layer under *Name*.

Please note: When you hide the layer, only inspection symbols that have been freely placed will be hidden with it. Linked inspection symbols can only be hidden using the Inspect hide and unhide functionality, see [Hide and unhide](#)¹⁴³.

8. Parameter: You can add additional information from drawing or model parameters to an inspection symbol. In order to do so, a symbol must have variable text, in which the necessary drawing or model parameter is defined. See [Creating variable text](#)¹⁵⁹ and [Adding parameter values to a symbol](#)¹⁶³ for more information.

Under *Name Parameter*, enter the name of the variable text. Under *Name Column*, enter a name for the table column that displays the values from the variable text. This column is added to the symbol, which you can check in the [Overview](#)¹³⁵ tab in GENIUS TOOLS Inspect.

Warning: The following terms are used by GENIUS TOOLS Inspect and cannot be used for naming parameters and their columns , both upper and lower case.

bse_dim	gti_param	max_dim	tpe_sub
cls_tol	linkedto	min_dim	tpe_tol
creo_id	nme_sym	mod	val_tol
descr	num	revision	src
grd	num_sheet	separator	tpe_sym
gti_note	num_sym	tpe_main	tpe_tol
ipc_dim			

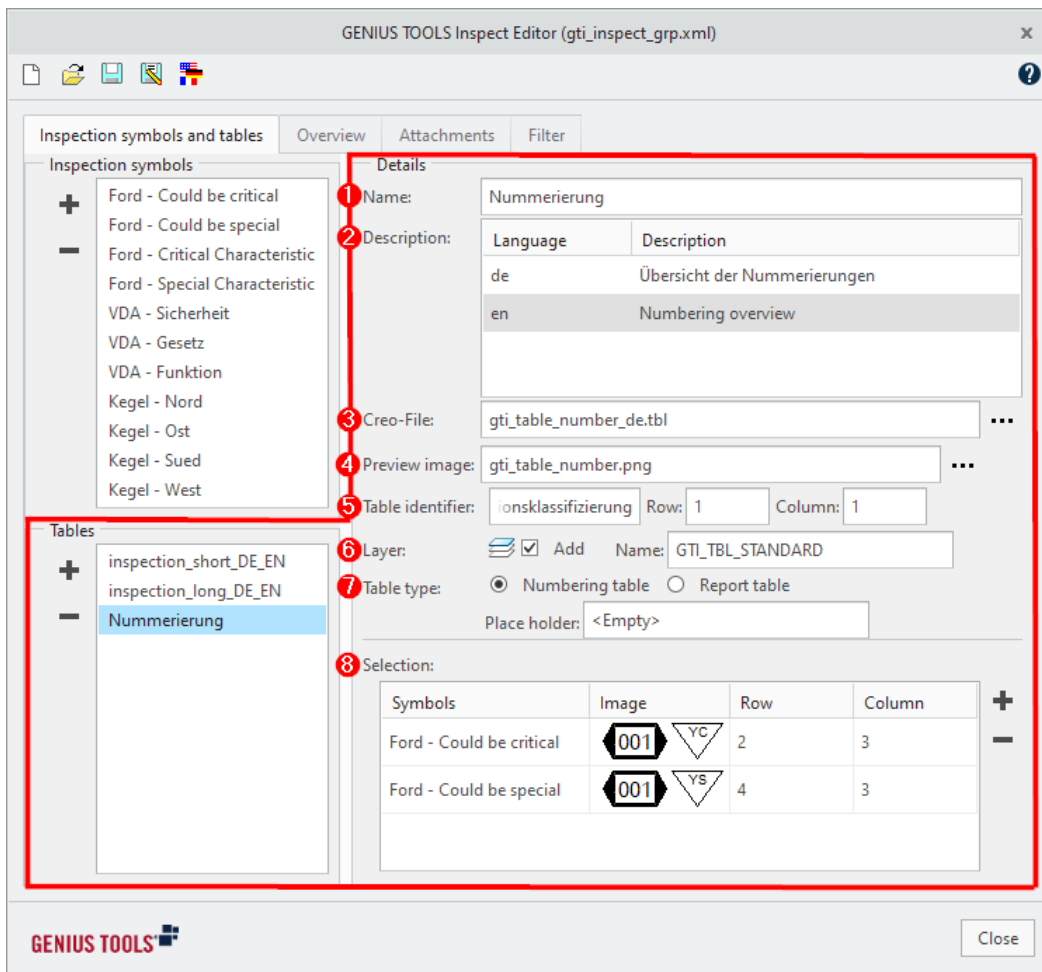
To output the additional parameters in the Excel export as well, you have to assign a column in the Excel template to each parameter. In the Excel template, specify the column name (**Name Column**) in the format *gti:<columnName>*, see also [Export](#)¹⁴⁴.

Tables

GENIUS TOOLS Inspect comes with two out-of-the-box tables, one numbering table and one report table. The configuration for the settings *Table identifier*, *Table type* and *Selection* differs according to the table type. The Creo table used also has some different characteristics.

- **Numbering table:** lists inspection symbols with the symbol image, a description and a list of numbers.
- **Report table:** displays detailed information on individual inspection symbols.

For both table types, you can customize the standard template or create your own table templates. Please refer to [Configuration for numbering tables](#)¹⁵⁴ and [Configuration for report tables](#)¹⁵⁵.



1. Name: Specifies the displayed name of the table.

2. Description: A localized description of the table. Use the localization button in the command bar to manage the languages.

3. Creo-File: Specifies the path to a TBL file. The default path is `%gt_resource_folder%\inspect`. Numbering tables¹⁵⁴ and report tables¹⁵⁵ differ.

4. Preview image: Specifies the path to a preview file of a table picture. The tooltip displays the picture in the original size so that large picture can improve legibility of the preview

Preview	Name	Description
	inspect_long_DE	Inspection Report - long DE
	inspect_long_EN	Inspection Report - long EN

GENIUS TOOLS Inspect - Characteristic Report (lang)														
No.	Sheet	Grid	Main Type	Sub Type	Nom. Dim.	Min. Dim.	Max. Dim.	Tol-Standard	Tol-Table	Tolerance	Parameter	Note	Description	Name of Inspection Feature
002	1	03	Dimension	Ø/Diameter	45	44.7	45.3	ISO/DIN	Shaft	h7			-	DIN 6770
003	1	01	Dimension	Ø/Diameter	80	79.97	80	ISO/DIN					-	DIN 6770
006	1	04	Note										break edge	DIN 6770
007	1	01	Shape and position	Concentricity							0.2 = 0.2 (C) = 0.2		before coloring	DIN 6770

5. Table identifier: Needed as a unique ID for a table. Numbering tables¹⁵⁴ and report tables¹⁵⁵ differ.

6. Layer: Defines the layer on which the table is placed. Enter the layer under *Name*.



7. Table type: The table type is either *Numbering table* or *Report table*. In addition, a placeholder for empty cells can be entered for a *Numbering table*.

8. Selection: This section only opens when a *numbering table*¹⁵⁴ is selected as table type.

Configuring numbering tables

Creo-File: Specifies the path to a TBL file. The default path is the `gt_resource_folder`. The Creo table has to contain the unique ID of the table in the cell specified under *Table identifier*, by default in the first cell of the first row. The table identifier is *Function related classification* for the example numbering table. If you want to define your own tables, use a unique table identifier.

In the TBL file for a numbering table, symbols and descriptions are given line by line. For each symbol, one cell is used for listing the symbol numbers. The table cell that should contain the list of numbers is specified under *Selection by Row and Column*.

Function related classification		Inspection No.
Symbol	Critical Feature	1
	Hazardous to human beings	
Symbol	Main feature	<Empty>
	Restricted functionality, reduced output/performance	
Symbol	Standard feature	all other quality parameter
	no direct effect on functionality	

Creo table for numbering table

Table identifier/Row/Column: The table identifier is a unique ID for a table. The table identifier is *Function related classification* for the example numbering table. If you want to define your own tables, use a unique table identifier.

Enter a row and column to specify where the table identifier is located in the table. By default, the table identifier is located in the first cell of the first row.

Selection: This section is only displayed for the numbering table.

The settings under *Selection* define the position of the inspection symbols. Click on the displayed name of an inspection symbol to display a drop-down list showing symbols that have not yet been added. Use the (+/-) buttons to add or remove rows.

Use the *Row* and *Column* settings to specify the cell which should contain the list of symbol numbers. For merged cells, enter the number of the first row.

Please note: Numbering tables do not work with multiple symbols that use the same Creo symbol file. Only the last row defined for each symbol file will be stored in the table definition.

Configuring report tables

Using the configuration option `gti_lang` you define - not depending on the system language - the language with which the contents of the tables are assigned.

Creo-File: Specifies the path to a TBL file. The default path is the `gt_resource_folder`. The Creo table has to contain the unique ID of the table in the cell specified under *Table identifier*, by default in the first cell of the first row. The table identifier is `gti_tbl_report_long` for the example report table. If you want to define your own tables, use a unique table identifier.

For report tables, the second row of the table specifies which data is to be written into each column. Enter the column code for the desired column into each cell of the second row. The available column codes are listed in the section on adapting the Excel export template under [Export](#)¹⁴⁴.

The first cell of the area to be filled with data is marked by the text *Wertezeile*.

If you insert a report table with Inspect, the table is filled automatically and the first two rows are hidden, making the table identifier and column types invisible.

Please note: There is no way to display the top two rows of the table once you have inserted it using Inspect. If you want to make changes to the TBL template, you have to edit it in Creo without using Inspect.

gti_tbl_gti		
num_sym	tpe_main	tpe_sub
GENIUS TOOLS Inspect Tabelle		
Number	Main type	Subtype
Wertezeile		

Creo table for report table

Table identifier/Row/Column: The table identifier is a unique ID for a table. The table identifier is `gti_tbl_report_long` for the example report table. If you want to define your own tables, use a unique table identifier.

Enter a row and column to specify where the table identifier is located in the table. By default, the table identifier is located in the first cell of the first row. If you insert a report table with Inspect, the table is filled automatically and the first two rows are hidden, making the table identifier and column types invisible.

Please note: Using the configuration option `gti_lang` you define - not depending on the system language - the language used for the contents of the report table.

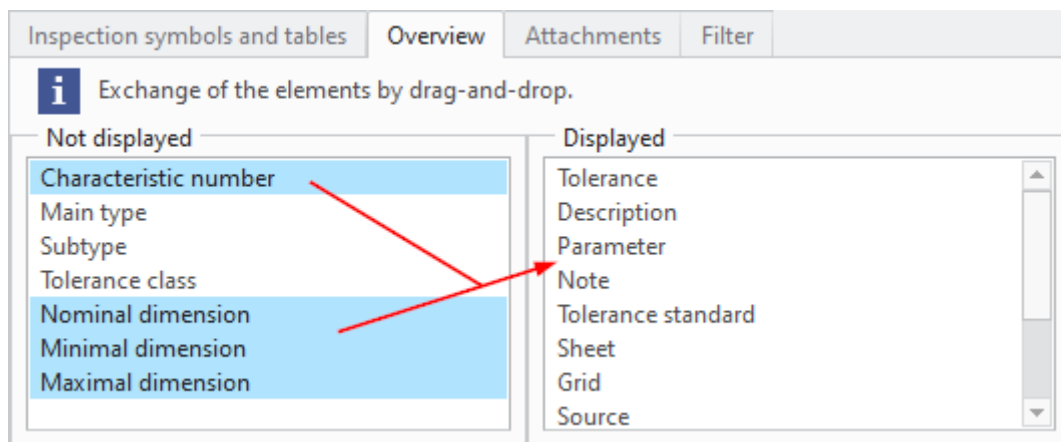
Customizing tables

For instructions on creating a customized report table see the chapter [Customize report tables](#). ¹⁶⁸

8.3.1.5 Overview tab

The *Overview* tab is used to manage the display of the *Linked Symbols* table.

Drag and drop elements to be displayed in the table into the desired position. Remove items that you do not want to display in the same way.

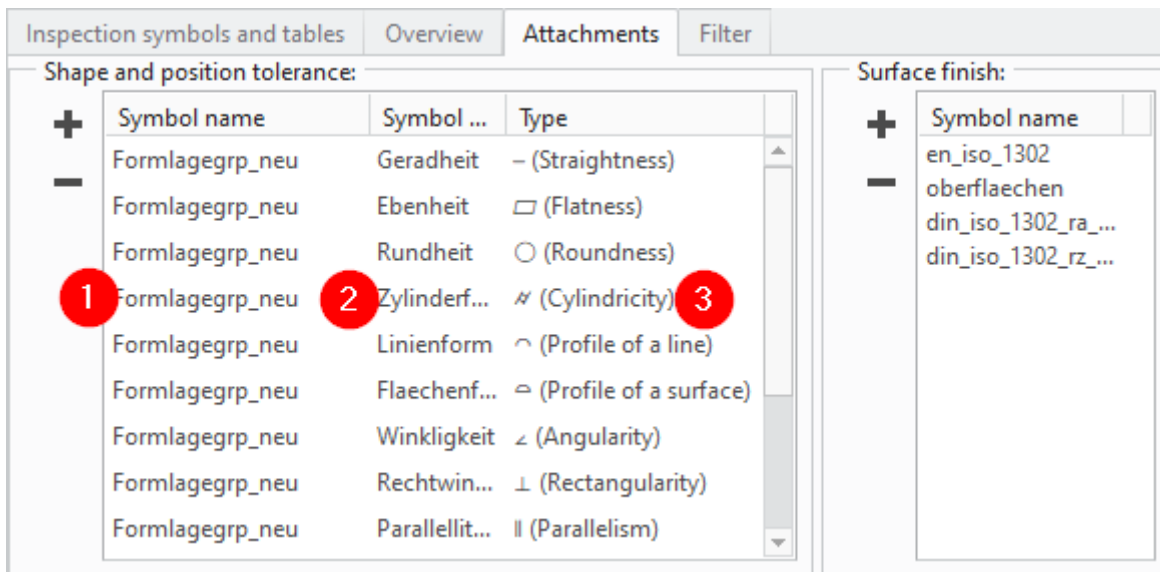


Use Drag and Drop to put the overview table together

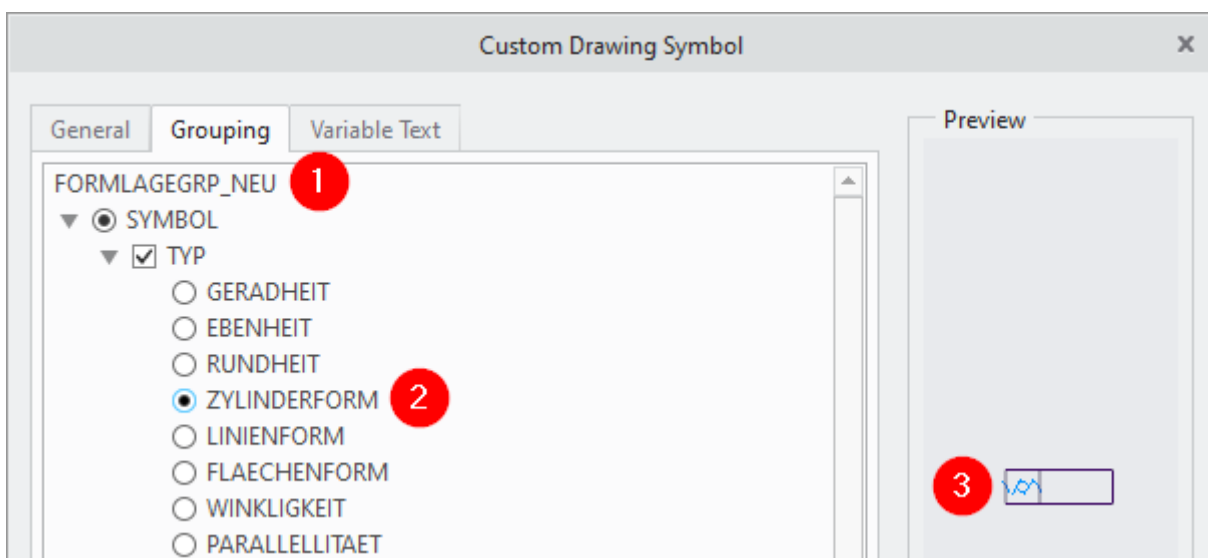
8.3.1.6 Attachments tab

The *Attachments* tab manages the assignment of various shape and position tolerances or surface quality symbols to user-defined drawing symbols.

This makes it possible to filter for inspection symbols linked to defined types of drawing elements.



Entries in the shape and position tolerance table or surface quality table correspond to properties of symbols in the symbol library.



Creo dialog for groupings in drawing symbols

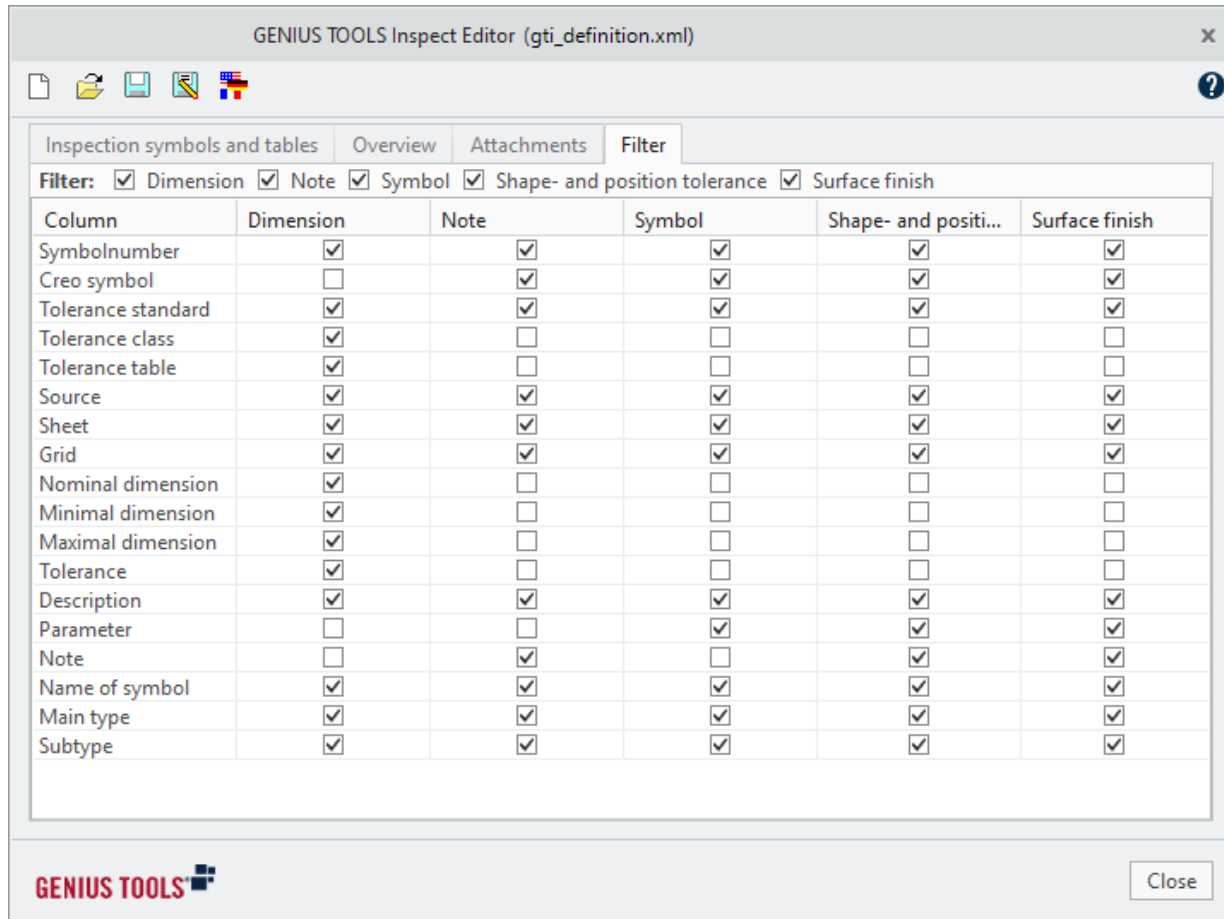
Enter the file names of the custom drawing symbol (as defined in the Creo dialog box) in the corresponding tables to assign them to the correct main types. In the case of shape and position tolerances, the main type (1) and subtype (2 and 3) can be determined.

8.3.1.7 Filter tab

The *Filter* tab defines the view of the three usable filters for the *Linked Symbols* overview. The configuration is stored per definition.

First, specify which filters (dimension, note, symbol, shape and position, surface) are to be activated by default.

In the second step, configure the columns to be displayed. Then save the configuration. The overview table *Linked Symbols* will always use the filters configured here when using the definition.



In this example, the *Tolerance class* column will only be displayed, if the *Dimension* filter is active, i.e., if inspection symbols linked to dimensions are displayed.

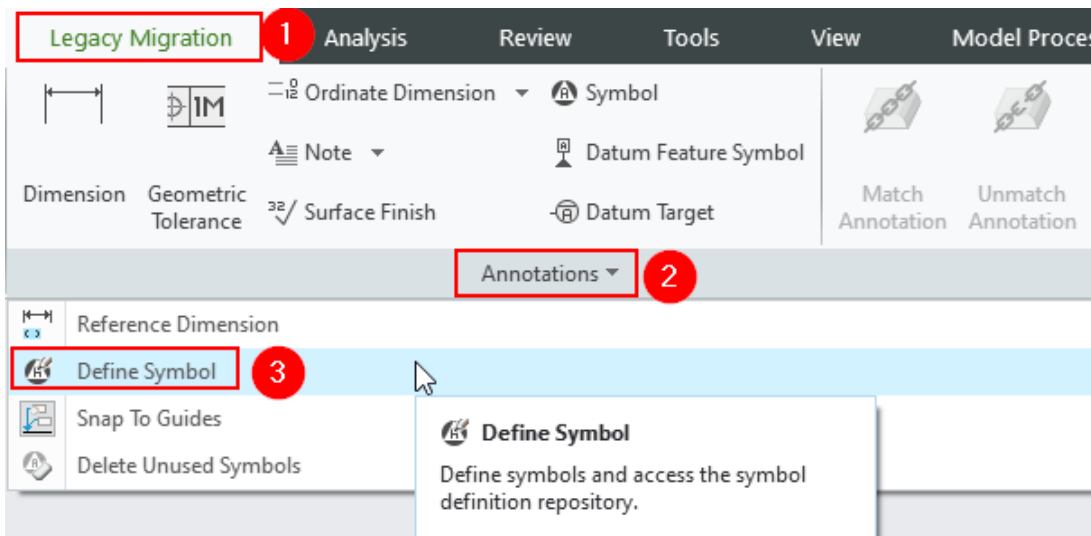
8.3.2 Creating a Creo symbol for Inspect

This section explains how to create a new symbol for use as an inspection symbol.

Please note: The symbols delivered with Inspect use the font *ISONORM LT Regular* based on ISO 3098-5, which is part of Creo in version 4 and later. If you experience issues with displaying the out-of-the-box symbols, please check whether this font is available in your system.

1. Sketching the symbol

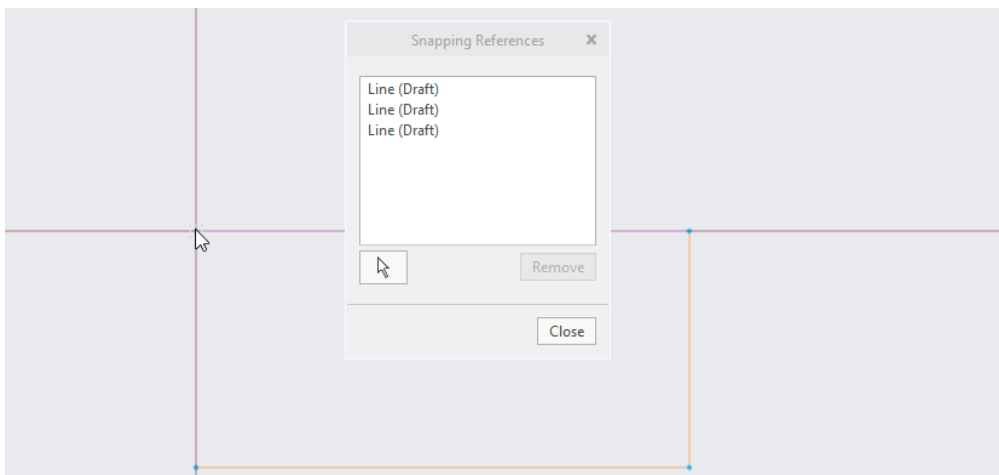
You are in drawing mode. In the Creo menu ribbon in the tab *Legacy Migration (1)*, you can access via *Annotations (2)* the command *Define Symbol (3)*.



Opening the symbol gallery.

In the Menu Manager in the section SYM DEFINITION click *Define* and enter a name for the symbol.

To define your own symbol, create a Creo Parametric symbol in the sketcher.



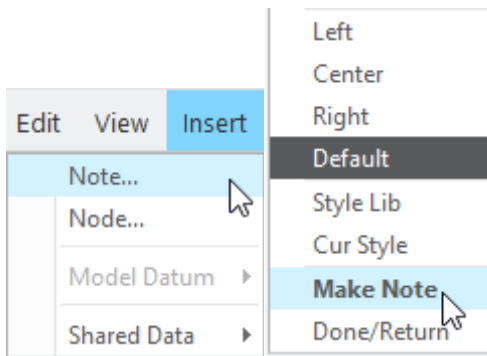
Sketching a symbol by lines.

2. Creating variable text

Variable text is data that is saved on a symbol as a note.

2.1. Create a note

Click *Insert* > *Note* and in the Menu Manager that opens click *Make Note* to create and place a new note.



Next you will be asked for the note name.

To ensure compatibility with GENIUS TOOLS Inspect, you have to create three notes. Enter the following names:

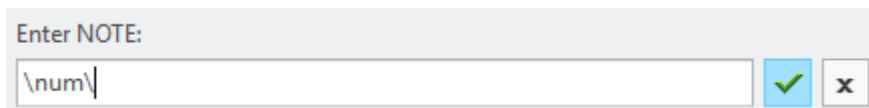
- \num\ to save the number of the inspection symbol
- \descr\ to save the description
- \LinkedTo\ to save the linked drawing element for the symbol

Warning: The following names must not be given to a note – both in upper and lower case –, as they are in use by the program.

bse_dim	gti_param	min_dim	tpe_sub
cls_tol	nme_sym	mod	tpe_tol
creo_id	num_sheet	revision	val_tol
grd	num_sym	separator	src
gti_note	max_dim	tpe_main	tpe_sym

Enter the note name, confirm by clicking on the green arrow, then exit the input dialog by clicking on the X button.

The next notes can be created in the same way.



2.2. Change the size of the notes

Since the notes \descr\ and \LinkedTo\ should not appear when the drawing is printed, their size has to be reduced. You can change it via the context menu of the note by right clicking on it. Select *Properties* > *Text Style* to change the font and size.

Text Style

Copy from

Style name: Default

Existing text: Select Text...

Character

Font: isofont

Height: 0.000010

Slant angle: 0.000000

Thickness: 0.000000

Width factor: 0.700000

Default (unselected)

Default (checked)

Default (checked)

Underline (unselected)

Kerning (unselected)

Text style of a note

- Remove the checkmark from the properties for *Height* and set a very small value, e. g. 0.00001.
- Remove the checkmark from *Font*.

3. Define attributes for the symbol

To finish creating the symbol, you need to define the placement, symbol origin and predefined values of the variable texts. The dialog for these settings opens when you click *Done* in the *Menu manager*.

Menu Manager

SYM GALLERY

Define

SYMBOL EDIT

Attributes

Copy Drawing

Copy Symbol

Parameters

Groups

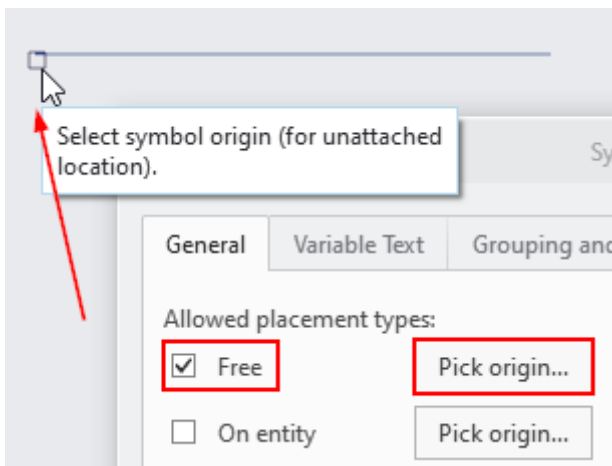
Note Rotate

Done

Quit

3.1. General attributes

Check *Free* and select the lower left corner as origin.

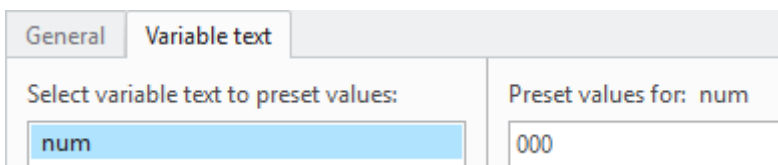


Choosing the placement and origin

3.2. Variable text

In the second tab *Variable text* enter in the section *Preset values* for the following characters:

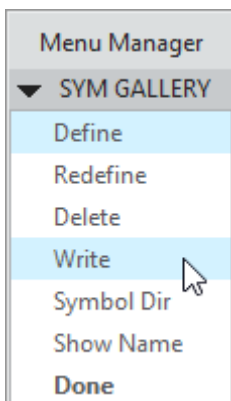
Name	Value
num	000
descr	-
LinkedTo	-1,-,-1,-



Predefined value for /num/.

5. Saving the symbol

As the last step, the symbol has to be saved. You can do this by clicking *Write* in the *Menu manager*. Enter the path, or leave it empty to use the current symbol directory for saving.



8.3.3 Adding parameter values to a symbol

You can assign values of drawing or model parameters to an inspection symbol. Parameter values can be either copied or linked to the symbol.

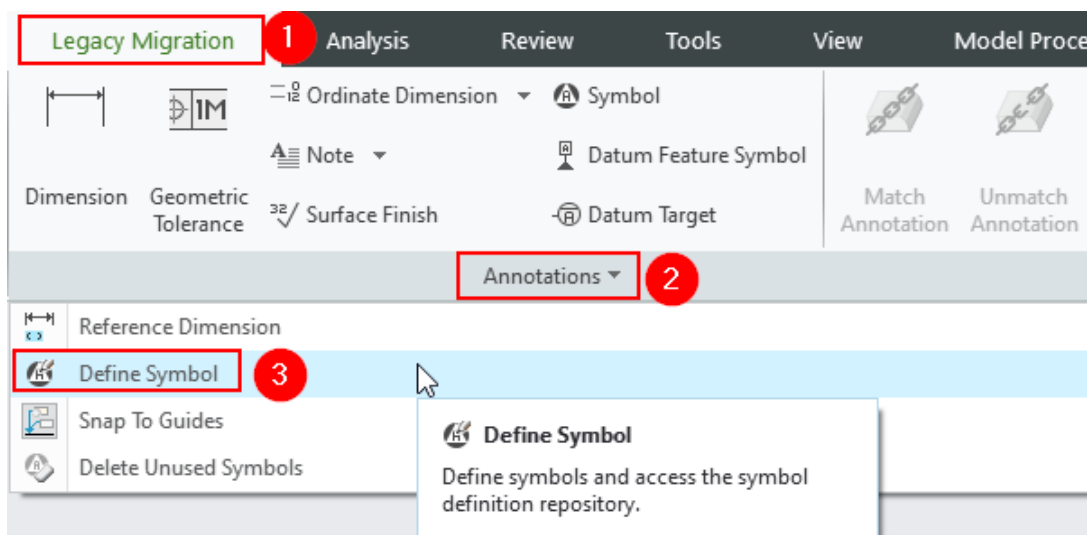
- Copy: The parameter value is copied to the inspection symbol when it is placed on the drawing and is static thereafter. (See [Copying parameter values](#)¹⁶⁴.)
- Link: The parameter value is updated on the inspection symbol when the drawing is opened. (See [Linking parameter values](#)¹⁶⁴.)

For general information on parameters consult the chapter [Variables](#).⁶⁸⁸

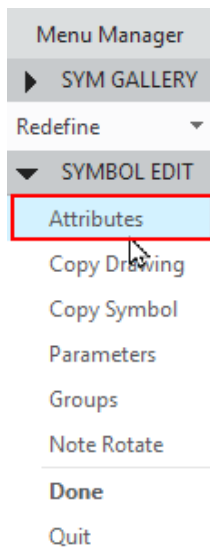
Parameters are written into the variable text in the dialog *Symbol Definition Attribute*.

Dialog Symbol Definition Attributes

1. In the Creo ribbon menu in the tab *Legacy Migration* (1), click *Symbol* (2) > *Symbol Gallery* (3).



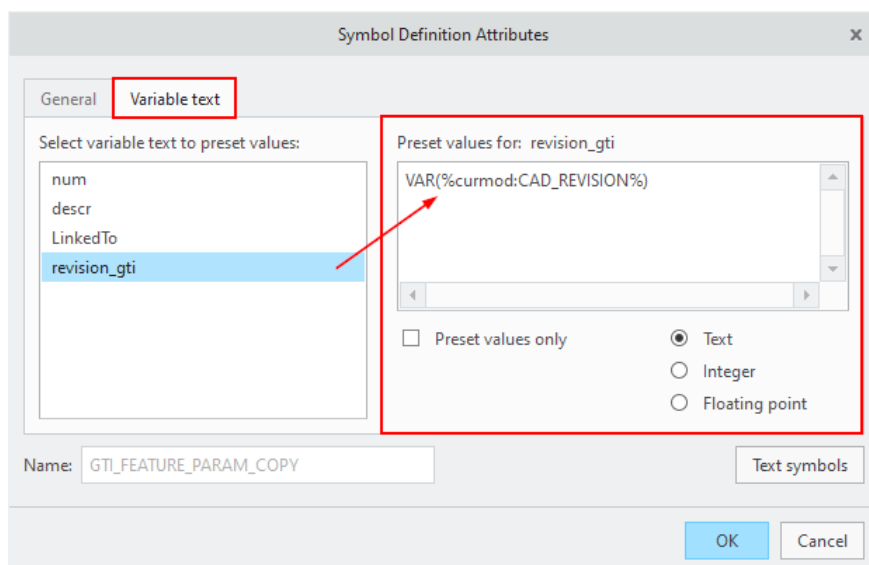
2. In the Menu Manager in the section SYM GALLERY click *Redefine*.
3. Click on the symbol in the drawing. The Creo symbol editor is opened.
4. In the Menu Manager click on *Attributes*.



Copying parameter values

Enter the parameter name in the variable text on the symbol in the **Symbol Definition Attributes** ¹⁶³ dialog. In the tab *Variable text* > field *Preset values for type*:

- for a revision parameter from the model: `VAR(%curmod:Parameter%)`
- for a revision parameter from the drawing: `VAR(%Parameter%)`



Preset value: Copy value from parameter CAD_REVISION

For a step-by-step description of how to copy parameter values into symbols, see the example **Creating Change Symbols** ¹⁶⁵.

Linking parameter values (Referencing)

The parameter value is updated at the inspection symbol when the drawing is opened. In the **Symbol Definition Attributes** ¹⁶³ dialog > *Variable text* tab > *Preset values for* field, type the parameter name in the following format:

- for a parameter from a model: *&Parameter*
- for a parameter from a drawing: *&Parameter:D*

Please note: The revision parameter CAD_REVISION should not be used for change symbols because the revision information on the drawing would change with each new revision.

Redefine parameters

Existing parameters can be redefined in variable text in the drawing mode. Enter the changes for the variable text in the [Symbol Definition Attributes](#) ¹⁶³ dialog.

8.3.4 Creating change symbols and tables

A change symbol indicates changes to the model (DRAWING) and not, like inspection characteristics, the dimensions, materials, etc. to be inspected.

Changes can be displayed from revision to revision as well as within a revision. The change symbol counts up the iteration

To display changes in a drawing, the revision must be copied to the change icon at the time of placement.

With the *revision_gti* variable, GENIUS TOOLS Inspect provides an interface that copies values of drawing or model parameters to an inspection symbol. It can also be used in an Inspect table.

For a revision values to be copied into a change symbol, the following default values have to be assigned in the variable text *revision_gti*:

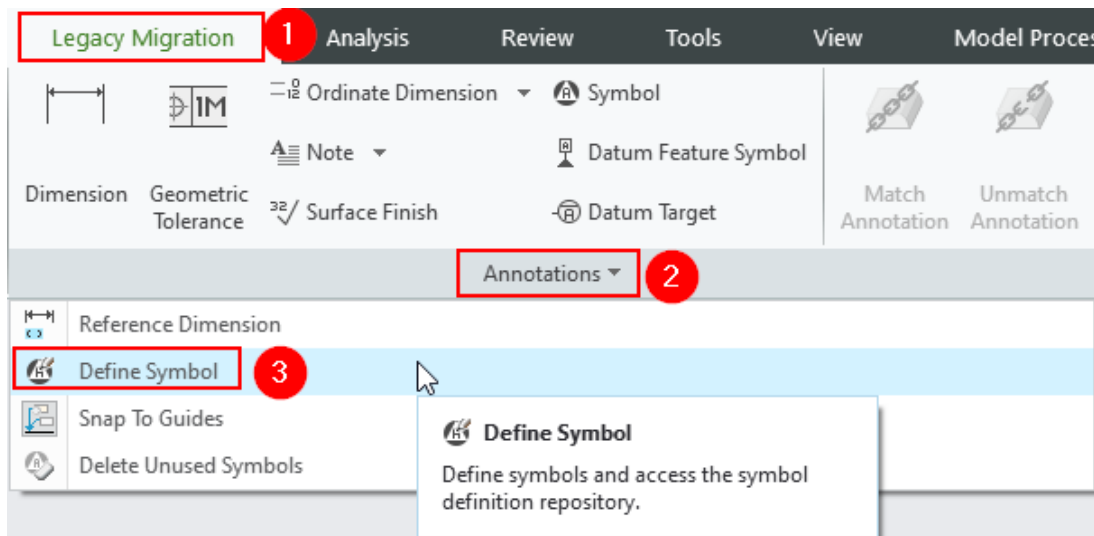
- for a revision parameter from the model: *VAR(%curmod:Parameter%)*
- z. B.: *VAR(%curmod:CAD_REVISION%)*
- for a revision parameter from the drawing: *VAR(%Parameter%)*
- e.g.: *VAR(%CAD_REVISION%)*

The step-by-step procedure is described below.

Creating change symbols

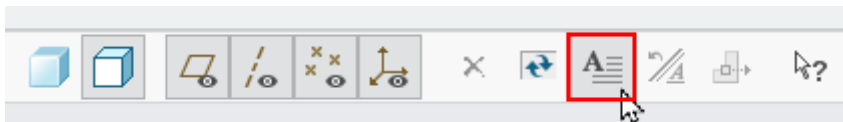
Procedure for copying the values of a revision parameter to a symbol

1. Open a drawing which includes the symbol in question or [create a new symbol](#) ¹⁵⁸ first.
2. In the *Legacy Migration* tab, go to *Symbol Gallery*:

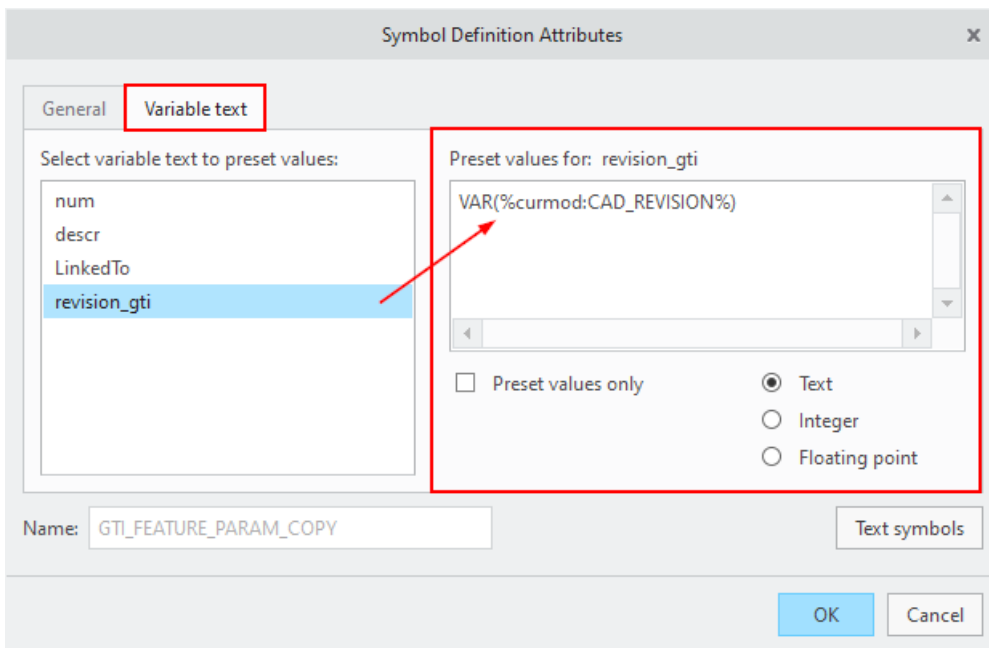


Opening the symbol gallery.

3. In the Menu Manager/SYM-GALLERY, go to *Redefine* and *Click Vari*.
4. Click on the symbol in the drawing.
5. The Creo symbol editor opens. Go to the *Create Note* icon in the menu bar:



6. In the Menu Manager, go to *Notes > Create Note*.
7. The *Select item* dialog opens. Select a point on the sketch where the note should be located.
8. Enter this name for the note: `\revision_gti\` and click on the green arrow and then on the cross beside it.
9. Click *Done* in the Menu Manager.
10. In the Menu Manager, go to *SYMBOL EDIT > Attributes*
11. This opens the *Symbol Definition Attributes* dialog. Specify the revision parameter for `revision_gti` on the *Variable Text* tab:
 - for a revision parameter from the model: `VAR(%curmod:Parameter%)`
 - for a revision parameter from the drawing: `VAR(%Parameter%)`



Copy parameter values for the variable text *revision_gti*

Creating change tables

Changes can be displayed on a drawing by inserting a table with a column that shows the copied value of the revision parameter from the icon. You can either use the file *gti_table_param_en.tbl* or create the required column yourself in a table as follows.

1. The Creo symbol must have a variable text – here: *revision_gti* – and the revision parameter must be copied into it. (See step 11 above.)
2. This variable text must be defined for the symbol in GENIUS TOOLS Inspect Editor. To do this, go to the *Inspect Characteristics and Tables > Parameters*¹⁴⁹ tab and type the name of the variable text under *Name Parameter* – here: *revision_gti* – and the column name under *Name Column* – here: *Revision*.

This column is now added to the symbol, as you can check in GENIUS TOOLS Inspect on the *Overview* tab.

3. Now create your own report table as in the example in the *Customize report tables*¹⁶⁸ chapter and perform step 9 and 10 (naming new column) as follows.
 - write in the second line the same name as you have defined in step 2 under *Name column* – here: *Revision* (this line will become invisible when the table is placed using GENIUS TOOLS Inspect)
 - write in the fourth line *Revision* (this is the displayed line in the table)
4. Save the table (TBL file) in the Caddepot directory of the synchronization server in the *Inspect* folder.

8.3.5 Customize report tables

This example shows how to create your own report table from a template from the Startup-TOOLS. (See also differences between [numbering and report tables](#)¹⁴⁹).

Step-by-step guide

The template is created in the resource folder of the caddepot.

Pause synchronization

1. In GENIUS TOOLS Starter App, go to the user menu and select *Pause synchronization*.

Open template

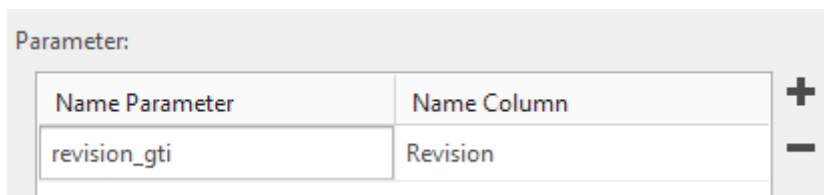
2. Open a drawing.
3. In the Creo ribbon, go to the tab *Table > Table from File*.
4. Open a report table from the *inspect* directory in the folder *gt_resource_folder* (path: %gt_resource_folder%\inspect) the TBL file *gti_table_inspect_l_en.tb*.
5. Place the table on the drawing.

Specify new table identifier

6. Go to the first row, first column for table identification. The standard report table has the table identifier: *gti_tbl_inspection_long_en*
7. Overwrite this identifier with your own identifier.

Insert and name columns

8. In the *Table* tab, go to *Add Column* and click with the mouse on the line of between two columns. The new column is created to the right of it.
9. Write the desired variable in the second line of the new column. This line becomes invisible when the table is placed by using GENIUS TOOLS Inspect. You can either
 - use variables that are predefined by GENIUS TOOLS Inspect (see [table](#)¹⁵¹) or
 - use your own variables which have been defined in the [variable text](#)¹⁵⁹ of the symbol. In this case enter here the entry in *Name column* as specified in the Inspect Editor under *Inspection Characteristics and Tables > Details > Parameters > Name Column*¹⁴⁹.



Variable (here: Revision) in "Name Column" in the settings for parameter in Inspect Editor

10. Write in the fourth line the name that the column shall display in the table.

11. Add further columns.

Deleting Columns

12. Delete existing columns by selecting them (left-click and drag the mouse over them) and press the *Delete* key.

Note: Make sure that your table does not contain empty columns. GENIUS TOOLS Inspect is unable to read tables with empty columns.

Save changed table

13. Mark the entire finished customized table by left-clicking and dragging the mouse over it.
14. In the *Table* tab, click on *Save as Table* and save the table under a new name. It is recommended to use the table identification. Creo creates a TBL file.

Storage location

15. To make this TBL file permanently available, you must store the file in the Caddepot of the synchronization server in the inspect folder. (The inspect folder in the Cadpool will be overwritten by Caddepot after the synchronization pause.)
16. Start the synchronization again in the GENIUS TOOLS Starter App in the user menu with *Synchronize Now*.

8.4 Inspect Revision

GENIUS TOOLS Inspect Revision allows you to create a snapshot of all inspection symbols on a drawing at a given time. Using a drawing revision parameter you can define the revision level of a drawing and generate a history of all revisions. Revision histories can be exported to Excel.

Please note: GENIUS TOOLS Inspect Revision is only available with a subscription license for GENIUS TOOLS for Creo.

8.4.1 Fundamentals

Glossary

Revision

Reviewed state of a technical document.


Revision status

Change of a revision identified by a number, letters and/or date.

Drawing revision parameter

Parameter indicating the revision status of a drawing.

Snapshot

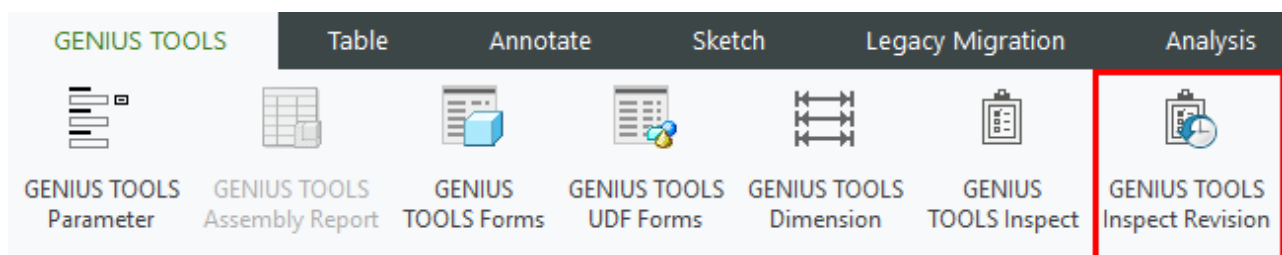
In GENIUS TOOLS Inspect Revision, the snapshot command  maps all inspection characteristics on a drawing and generates a time stamp. This will correspond to the revision status, if the drawing revision parameter has been changed for the snapshot.

Snapshot history

GENIUS TOOLS Inspect Revision displays the totality of all snapshots. These can be exported.

8.4.2 Starting the program

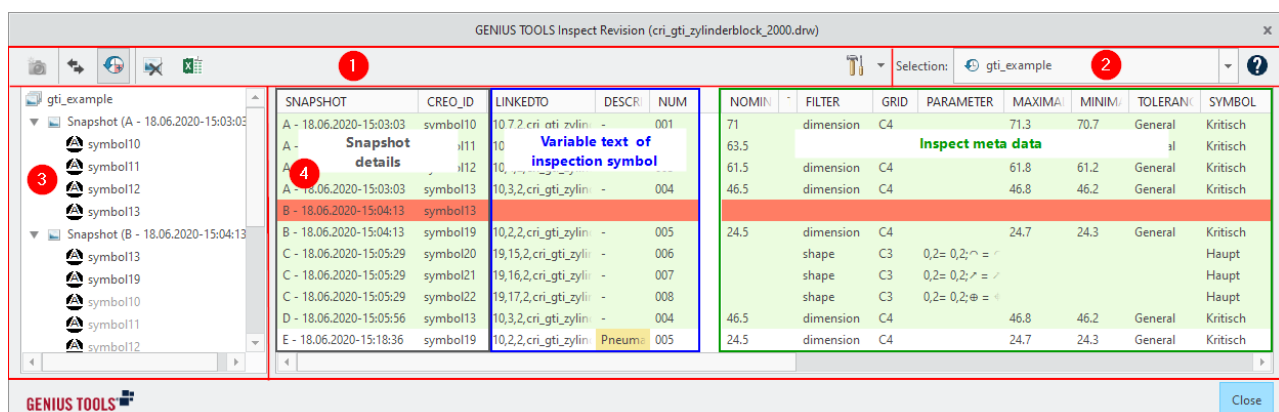
Start GENIUS TOOLS Inspect Revision in drawing mode from the ribbon menu in the GENIUS TOOLS tab.



Starting the program in GENIUS TOOLS ribbon menu

8.4.3 User interface

The user interface contains the following elements:



User interface of GENIUS TOOLS Inspect Revision

1. Command bar¹⁷¹
2. Select configuration file¹⁷²

3. Symbol / snapshot tree¹⁷³

4. Overview of snapshots with

– Snapshot details:

Drawing revision parameter¹⁷⁶,

Time of creation of snapshot and

Creo-ID of modified symbol. (Creo-ID is the identity number assigned by Creo.)

– Variable text of the inspection symbol.







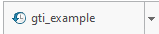
Variable text¹⁵⁹ is data stored in the SYM file of the symbol.

– Inspect metadata:

All data extracted by GENIUS TOOLS Inspect.

8.4.4 Command bar

The command bar contains these control elements:

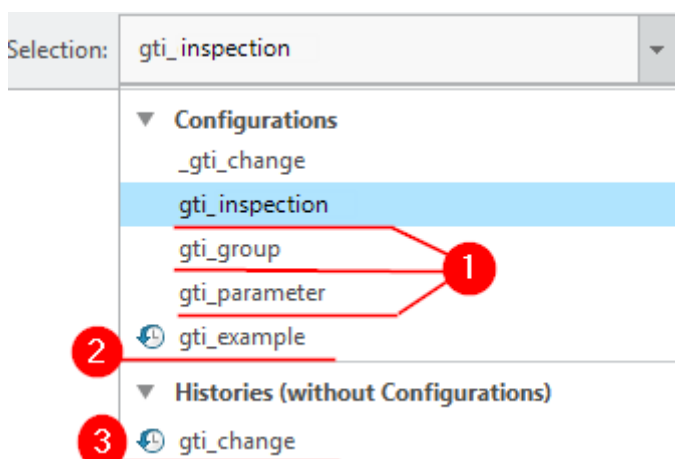
Icon	Name	Description
	Create snapshot ¹⁷³	Displays a snapshot of all inspection symbols on a drawing
	Switch tree display ¹⁷³	Switches display between symbol tree and snapshot tree
	Show complete state	Shows all existing inspection characteristics at the time of the respective snapshot
	Delete latest snapshot	Deletes the latest snapshot
	Export history to excel ¹⁷⁷	Exports snapshot history as XLSX file
	Tools	Contains supportive functions: – Save history as XML file
	Configuration / history ¹⁷²	Select configuration file and snapshot history

Icon	Name	Description
?	Help	Opens help page for GENIUS TOOLS Inspect Revision

8.4.5 Select configuration file

The command bar contains a field for selecting a configuration. A configuration is an XML-file that contains symbols, table definitions and display settings and that is created in GENIUS TOOLS Inspect Editor¹⁴⁷. The clock symbol next to the name indicates whether the configuration file contains snapshot data, i. e. a history.


Snapshot data is data stored in a drawing (DRW file). They can be exported from the XML file but also as stand-alone data without data of the XML file. Thus you may send a drawing with an inspection history without revealing the configuration settings.



Various configurations in the selection of GENIUS TOOLS Inspect Revision

- | | | |
|---|-------------------------------|---|
| 1 | Configuration without history | No data in Inspect Revision dialog |
| 2 | Configuration without history | Snapshot history displayed in Inspect Revision dialog |
| 3 | History without configuration | Snapshot cannot be created |



Save history as XML file

A history of snapshots can be exported without the corresponding configuration settings by saving a separate XML file in *Tools*  > *Save history as XML file*.

Configuring the selection field

The configuration file is by default set to *gti_inspection*. The default file can be changed in the configuration option `gti_start_file`.

8.4.6 Select tree view

You can expand the snapshot / symbol tree by clicking on the arrow symbol  and switch between the two tree displays by clicking on the symbol .


1. Snapshot view: Lists all snapshots and the corresponding inspection symbols

2. Symbol view: Lists all inspection symbols and the corresponding snapshots

The choice in the snapshot / symbol tree display area determines how snapshot data is displayed. These are the possibilities:

- Select the configuration file: displays complete history
- Select a snapshot (e. g. Revision A): displays a single snapshot
 - in snapshot tree: with all modified inspection symbols
 - in symbol tree: of selected inspection symbol
- Select an inspection symbol (e. g. symbol 24): displays a single inspection symbol
 - in snapshot tree: of selected snapshot
 - in symbol tree: with all snapshots


8.4.7 Creating snapshots

The snapshot function  maps all inspection symbols on a drawing and creates a time stamp. In addition, GENIUS TOOLS Inspect Revision can extract values from a [drawing revision parameter](#) ¹⁷⁶.

Tip: A snapshot corresponds to the revision status, if the drawing revision parameter has been changed for the snapshot.

When creating a snapshot all inspection symbols that are saved in the configuration file are compared with the previous snapshot. A snapshot can be created when at least one inspection symbol has been altered.

The following modifications are included in a snapshot:

Modification	Color/ example
1. A new inspection symbol has been placed (symbol does not yet exist in any snapshot)	green row A - 18.06.2020-15:04:13 symbol12 10,4,2,cri_gti_zyylinderblock_
2. A new value has been added to an inspection symbol	green cell 61.8 61.2 General
3. A value has been edited in the model	yellow cell in Inspect metadata columns (3) cri_gti_zyylinderblo Dimension ISO/DIN
4. A value on an inspection symbol has been altered	yellow cell in Variable text columns (2) 10,3,2,cri_gti_zylin Pneumatic length measuring 004
5. An inspection symbol was deleted (symbol is in snapshot but not in drawing)	red row B - 18.06.2020-15:04:13 symbol13
6. An unchanged symbol from an older snapshot 	greyed out text A - 09.02.2021-07:04:13 symbol12

Recorded inspection symbols

A snapshot unhides all hidden inspection symbols. This means that symbols without a target will be deleted in the process, see also [Hide und unhide](#) ¹⁴³.


A snapshot does not capture all inspection symbols and will blend in all hidden symbols as follow:

State of inspection symbol	Captured by snapshot?
Unnumbered inspection symbols (000 symbols)	no

State of inspection symbol	Captured by snapshot?
Inspection symbols whose target has been deleted	no Warning: Inspection symbols, whose target has been deleted, will be deleted.
Hidden symbols	
– with a linked target	yes, will be unhidden
– without a target	no Warning: Inspection symbols, whose target has been deleted, will be deleted.

Example: Creating a snapshot of a revision level



Steps:

1. Place the needed inspection symbols on the drawing and number them. (See chapters [Positioning](#) ¹³⁶ and [Numbering](#) ¹³⁷.)
2. Assign a new value to the drawing revision parameter in *Tools > Parameter*. In the example: CAD_REVISION = D
3. In the command bar click the *Create snapshot* button .

Result:


The snapshot will be displayed in the left section of the dialog (symbol- / revision tree). In the example: Snapshot (D - 15.06.2020 - 13:37:41)

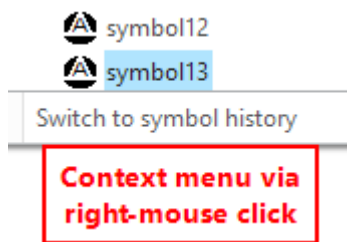
8.4.8 History of snapshots and symbols

After the first snapshot has been taken a clock symbol appears next to the configuration file.  gti_example  Configurations with a clock have a snapshot history, which is the total of all snapshots. This history can be [exported](#). ¹⁷⁷

Tip: You can get a history of drawing revisions by creating a snapshot after completing a new drawing revision.

You can view the history of a snapshot or a symbol by clicking on it.

Click on the  button to switch to the symbol tree or directly on a symbol to use the context menu to switch to its history.



GENIUS TOOLS Inspect Revision (cri_gti_zylinderblock_2000.drw)

Selection: gti_example

gti_example	SNAPSHOT	CREO_ID	LINKEDTO	DESCRIF	NUM	NOMIN	T	FILTER	GRID	MAXIMAL	MINIM
Snapshot (A - 18.06.2020-15:03:03)	B - 18.06.2020-15:04:13	symbol13									
Snapshot (B - 18.06.2020-15:04:13)	B - 18.06.2020-15:04:13	symbol19	10,2,2,cri_gti_zyl	-	005	24.5		dimension	C4	24.7	24.3
Snapshot (C - 18.06.2020-15:05:29)	A - 18.06.2020-15:03:03	symbol10	10,7,2,cri_gti_zyl	-	001	71		dimension	C4	71.3	70.7
Snapshot (D - 18.06.2020-15:05:56)	A - 18.06.2020-15:03:03	symbol11	10,6,2,cri_gti_zyl	-	002	63.5		dimension	C4	63.8	63.2
Snapshot (E - 18.06.2020-15:18:36)	A - 18.06.2020-15:03:03	symbol12	10,4,2,cri_gti_zyl	-	003	61.5		dimension	C4	61.8	61.2

History of snapshot of revision B

Selection: gti_example

gti_example	SNAPSHOT	CREO_ID	LINKEDTO	DESCRIF	NUM	NOMIN	T	FILTER	GRID	MAXIMAL	MINIM
symbol10	A - 18.06.2020-15:03:03	symbol13	10,3,2,cri_gti_zyl	-	004	46.5		dimension	C4	46.8	46.2
symbol11	B - 18.06.2020-15:04:13	symbol13									
symbol12	D - 18.06.2020-15:05:56	symbol13	10,3,2,cri_gti_zyl	-	004	46.5		dimension	C4	46.8	46.2
symbol13											
Snapshot (A - 18.06.2020-15:03:03)											
Snapshot (B - 18.06.2020-15:04:13)											
Snapshot (D - 18.06.2020-15:05:56)											
symbol19											

History of symbol 13

8.4.9 Drawing revision parameter

The drawing revision parameter informs about the revision level of a drawing. A new value should be assigned to the parameter after making modifications that are relevant to an inspection to inspection symbols or their values. (In the example below: A - D.) If you do not work with a revision parameter, you can still distinguish snapshots by their time stamp.

gti_example

gti_example	SNAPSHOT
Snapshot(A - 15.06.2020-13:29:37)	A - 15.06.2020-13:29:37
Snapshot(B - 15.06.2020-13:31:17)	A - 15.06.2020-13:29:37
Snapshot(C - 15.06.2020-13:34:52)	A - 15.06.2020-13:29:37
Snapshot(D - 15.06.2020-13:37:41)	A - 15.06.2020-13:29:37
	B - 15.06.2020-13:31:17
	B - 15.06.2020-13:31:17
	C - 15.06.2020-13:34:52
	C - 15.06.2020-13:34:52
	D - 15.06.2020-13:37:41

Parameter Time stamp

Details of snapshots in the snapshot tree display

There are three ways to use a drawing revision parameter in GENIUS TOOLS Inspect:

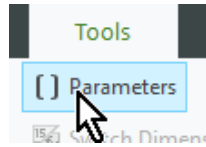
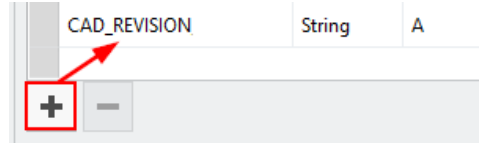

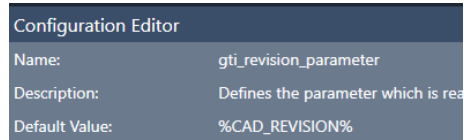
1. Use drawing revision parameter from Startup TOOLS

If you have Startup TOOLS installed your start part has been configured to include the drawing revision parameter CAD_REVISION and the necessary configuration option `gti_revision_parameter` is properly set. After the first placement of an inspection symbol on a drawing, the value of CAD_REVISION is automatically set to A.

2. Use drawing revision parameter from Windchill

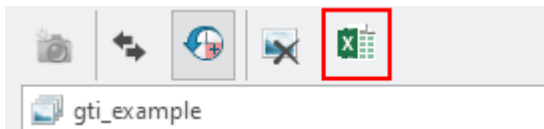
A drawing revision parameter is provided by Windchill and has to be specified in the configuration option `gti_revision_parameter`.

3. Create a drawing revision parameter yourself


1. Open Creo Parameter dialog: Tools tab > Parameters	<p>In Creo ribbon menu:</p> 
2. Add new parameter: <i>Plus</i> button (Add new parameter) > Enter parameter name (e.g. CAD_REVISION), Type (String) and Value (A)	<p>In Creo Parameter dialog:</p> 
3. In the configuration option <code>gti_revision_parameter</code> , define the new parameter name – you may use GENIUS TOOLS Configuration Utility  and the button <i>Reread Configuration</i>	<p>In GENIUS TOOLS Configuration Editor:</p> 

8.4.10 Export snapshot history

The history of snapshots can be saved to an XLSX file using a template.



Excel button in GENIUS TOOLS Inspect Revision

The button  opens the dialog [Export table to Excel](#)^{s10} which allows you to select the Excel template, the file to be exported and the snapshot history.

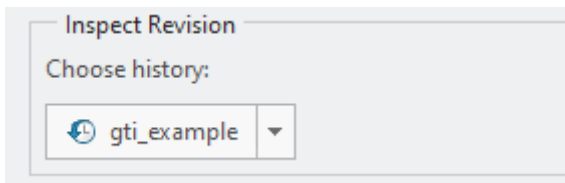
The Excel template is checked when starting the Excel export and may take some time.

Warning: Close all Excel windows before starting the export function.

Export dialog

The template file `gti_revision_template_de_en.xlsx` (in the directory `gt_resource_folder\inspect`) is entered by default as is the name of the drawing file.

Select a configuration – this defines the inspection symbols whose snapshot history is to be exported.



Configuring export

The excel export is set by using the following configuration options:

`gti_revision_excel_template` defines the name of the default Excel template for export.
(Default: `gti_inspection_template_de_en.xlsx`)

`gti_revision_folder` defines the folder in which the default Excel template is searched for.
(Default: `%gt_resource_folder%/inspect`)

`gti_revision_excel_coloring` defines whether data is exported with the coloring from the snapshot data segment of the UI.

Customizing an export template

You can customize a template by taking the template file `gti_revision_template_de_en.xlsx` from the directory `gt_resource_folder%/inspect` as a basis and adding a comment (2) to the first line (1) of the column you wish to adapt. The chapter [Create template](#)^{s12} describes how to set up an export template step by step.

Denomination 1	-
Denomination 2	-
IDNR	-
Filename	-

Revision data		Variable Text	
Revision	Creo_ID	Description	Linked to
1	Michaelis, Markus: gti_rev:rev_revision		

Acronym *gti_rev*: in the comment that assigns a column

Please note: This function is called *Comment* until Excel 2019, from Microsoft 365 onwards it is called *Note*.

The code in the comment defines the values to be copied into a column. It consists of the component acronym *gti_rev*: and the fill command. Use the following codes:

Command code	Column name
gti_rev:rev_revision	Revision
gti_rev:rev_id	Creo_ID
gti_rev:var_num	Number
gti_rev:var_descr	Description
gti_rev:var_LinkTo	Linked to
gti_rev:num_sym	Number of inspection symbol
gti_rev:num_sym	Symbol number
gti_rev:tpe_main	Main type
gti_rev:tpe_sub	Subtype
gti_rev:cls_tol	Tolerance class
gti_rev:bse_dim	Nominal dimension
gti_rev:min_dim	Minimal dimension
gti_rev:max_dim	Maximal dimension
gti_rev:val_tol	Tolerance
gti_rev:descr	Description

Command code	Column name
gti_rev:gti_param	Parameter
gti_rev:gti_note	Note
gti_rev:tpe_tol	Tolerance standard
gti_rev:num_sheet	Sheet
gti_rev:grd	Grid
gti_rev:src	Source
gti_rev:mod	Tolerance table
gti_rev:nme_sym	Name of symbol
gti_rev:tpe_sym	Creo symbol
gti_rev:var_<parame tername>	Output of additional user- defined parameters

Creating a template with multiple spreadsheets

You can use other module acronyms besides *gti_rev*:. Thus you can export data from GENIUS TOOLS Inspect and GENIUS TOOLS Inspect Revision together, see chapter [Export data from several GENIUS-TOOLS-components](#)⁵¹⁷ for an example.

9 Library

GENIUS TOOLS Library provides a convenient way to open, copy and insert objects from a library and to customize actions – such as opening, copying or inserting to a model – for each object individually.

Supported Creo object types are:

- Part, assembly, drawing and sketch
- UDF (user-defined feature)
- Drawing table, drawing frame, drawing symbol, drawing text

GENIUS TOOLS Library is available in different languages with the following functions:

1. Searching for Creo objects

- independently from object storage
- with visibility control: visible, invisible, preferred use

2. Fast search of complete library

- intuitively
- advanced search: type, status, parameter, dimensions

3. Configuration options

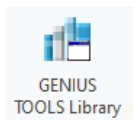
- for selection tables
- for defining actions (copy and insert by copying)
- for object creation with GENIUS TOOLS Forms
- for using UDFs from GENIUS TOOLS UDF Forms

4. graphical interface to manage libraries: GENIUS TOOLS Library Editor

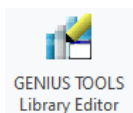
5. automatic check with Windchill with GENIUS TOOLS Library Data Importer

The library objects can be uploaded from hard drive or Windchill. Other PDM/PLM systems can be integrated on request.

Components of GENIUS TOOLS Library



GENIUS TOOLS Library (Library browser) for finding Creo objects in a library



GENIUS TOOLS Library Editor

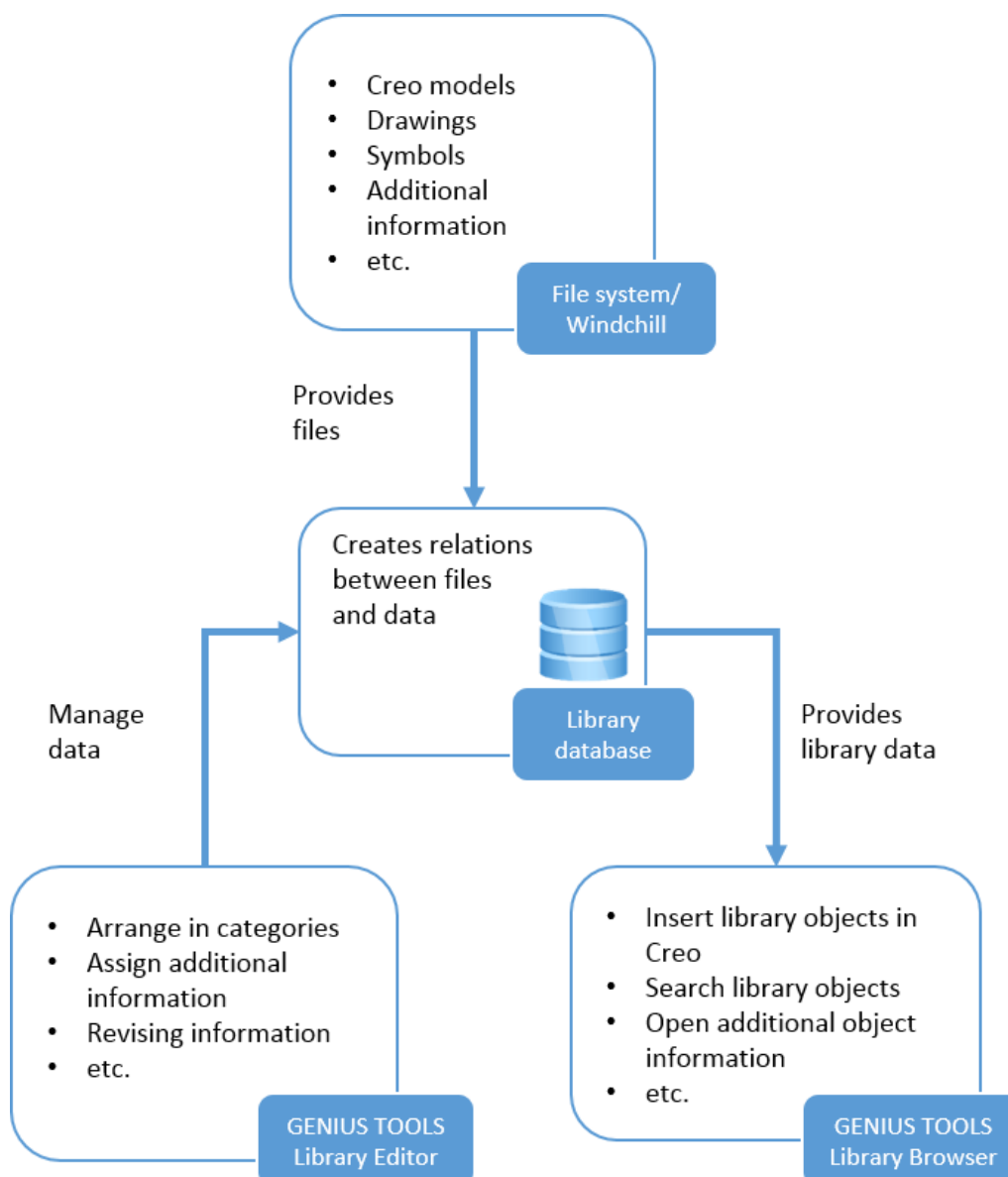
for managing libraries and their contents

9.1 Fundamentals

This introductory section gives you an insight into the operating principles of GENIUS TOOLS Library and deals with possible use cases. In addition, you will find a glossary of important terms here for a better understanding.

Summary

Libraries play an important role for the efficiency of a complex application like Creo Parametric. GENIUS TOOLS Library provides you with database-supported library management systems.



GENIUS TOOLS Library consists of the editor (Library Editor) for library database management and Library Browser for productive use in Creo Parametric

Library databases

contain additional information on the managed Creo models, and being centrally available, enable all users have the same information and to work on the same data.

The actual Creo models are not stored in the databases. When importing Creo model information or manually entering library objects, only the required information such as type or location of the models is captured.

Library objects

can be enhanced with metadata (additional information). This could be descriptive parameters of a Creo model, the object type (prt, asm, sym, etc.), or the current status of a library object, for instance. This metadata allows easy finding of library objects in the Library Browser via search or a status filter.

But also other additional information such as selection tables, locations of info documents, a list of models to be copied as well, or how a model should be opened or assembled, can be deposited for library objects. This way, error-prone operations can be automated or handled more easily through easy retrievability of required information.

Library categories

are available in a library database. Categories can be used for a simple logical structure of the library objects. Categories can also be expanded with additional metadata. As an example, library categories will allow the adding of a path name. This way it represents real directories and can be used to quickly switch the working directory.

9.1.1 Glossary

Category tree

The category tree in the editor and in the library area of GENIUS TOOLS Library contains all categories and library objects included in a library database. The visibility of the library objects in Library depends on the status set.

Library actions

Library actions are actions that can be executed on library objects e.g. "Open" or "Insert into assembly". The availability of actions in GENIUS TOOLS Library depends on the object type and the configuration in the GENIUS TOOLS Library Editor.

Library browser

The Library browser is the user interface of GENIUS TOOLS Library.

Library categories

Library categories are virtual organizational elements in the libraries of GENIUS TOOLS Library. Categories can be named as desired and can contain any objects. They are used to structure a library logically. Library categories can also represent real directories and can be used to quickly switch the working directory.

Library objects

Library objects are virtual representations of Creo files (e.g. parts, assemblies, drawings) or structural ideas (e.g. selection tables) in a library of GENIUS TOOLS Library. They are differentiated by their type and can be linked in categories.

Object types

Each library object has a type. It specifies possible properties and actions of the object.

Object links

Library objects can be linked within a library. This way, no copy of the object will be created in the database. Any changes to the object will affect all links of the object. This way, the same object can be available via different paths of the category tree at the same time.

Selection table

A selection table is a table that consists of several library objects. In a selection table, these objects can be grouped by parameters and dimensions. A quick selection can be made on the basis of the defined criteria via the detailed view in GENIUS TOOLS Library.

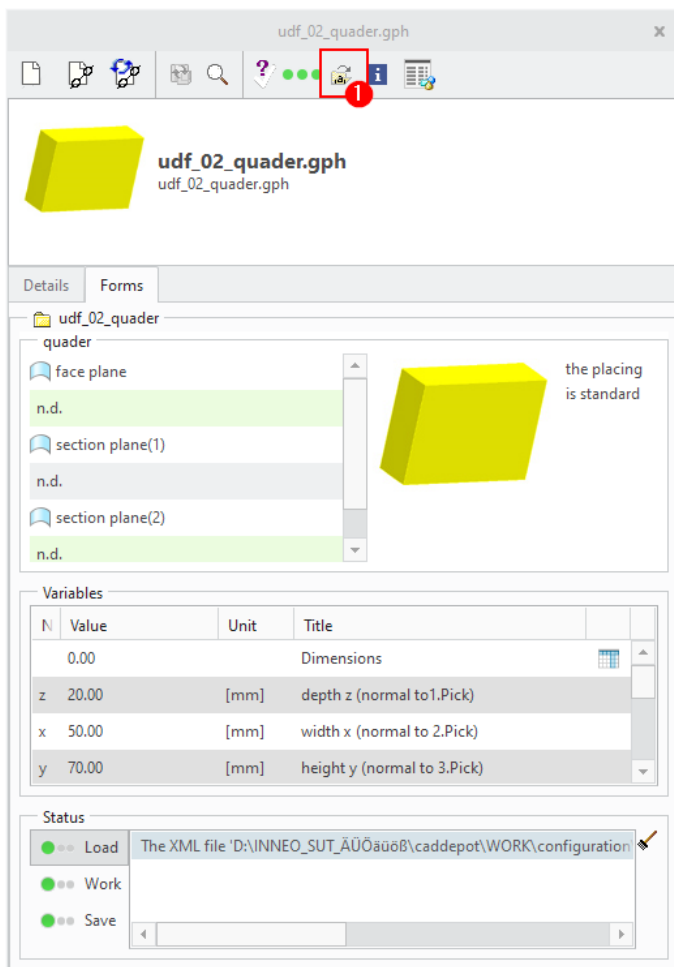
9.1.2 Use cases

- Find your library objects via the full-text search
- Call up library objects any time, independent of their type
- Call up detail information for any object
- Use single parts and assemblies as templates

Part modeling – sheetmetal and design part mode

Create new parts from existing library objects with GENIUS TOOLS Library. Use GENIUS TOOLS Forms for this. These objects can be assembled with Creo standard capabilities.

Alternatively, define UDF Forms. Assembling user defined features are easier with UDF Forms. The assembly operation can be made easier with additional information and descriptions. UDF Forms also allow later editing of the UDFs after assembly.



A UDF Form in the details window

Use library objects like sketches to create elements in Creo Parametric using the sketch mode.

Completing assemblies – design assembly

Complete your assemblies with parts and subassemblies stored in your libraries. There are two methods for this:

1. Assemble models such as standard or purchased parts without changes
2. Assemble other models using the template method
3. Create new models and drawings, with new rule-based filenames, with minimal clicks

Completing drawings – drawing mode

Use library objects such as drawing frames, symbols, tables and notes to complete drawings with little effort.

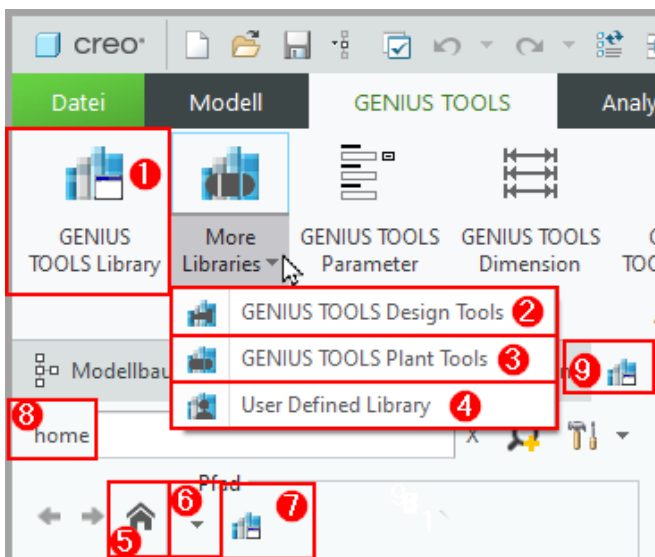
Using templates

When utilizing single parts and assemblies with copy rules, you will be able to specify which component, drawings and additional object should be copied. In addition, you can define name rules that are applied automatically to copied objects. This way, assembly templates can be set up faster, ready for use in new projects, including the required manufacturing documents.

9.2 Usage

This section contains information on using GENIUS TOOLS Library. It describes the general structure of the program. In the example below (Use cases²⁰⁷) you can find short step-by-step instructions to speed up your everyday work with GENIUS TOOLS Library.

Starting the library: in all Creo modes



1. Library: `gtl_favorite_button1_liblink` The stored default name for this library is company. In the SUT: `sut_int_de_creo`
2. Library: `gtl_favorite_button2_liblink` The stored default name for this library is designtools. In the abrufbarSUT: `designtools_tbx`
3. Library: `gtl_favorite_button3_liblink` The stored default name for this library is planttools.
4. Library: `gtl_favorite_button4_liblink` The stored default name for this library is user.
5. House-icon visible, if database `gtl_home_db` is defined.
6. List of all retrievable libraries and `gt_resource_folder\library`.
7. Displays the start category of the current library.

8. Name of the current Library.
9. This function activates the library guide and retrieves the Library viewer with the last configured library. If no start-library (gtl_start_db) is configured, the library chosen under 6) shows up from the database in alphabetical order.

With the button *More Libraries* icons can be combined with up to 4 libraries in the configuration settings. These Libraries can be started directly.

Configuring display window

GENIUS TOOLS Library can be configured to be displayed in a separate window. In this case, GENIUS TOOLS Library opens in a Creo window that cannot be closed.

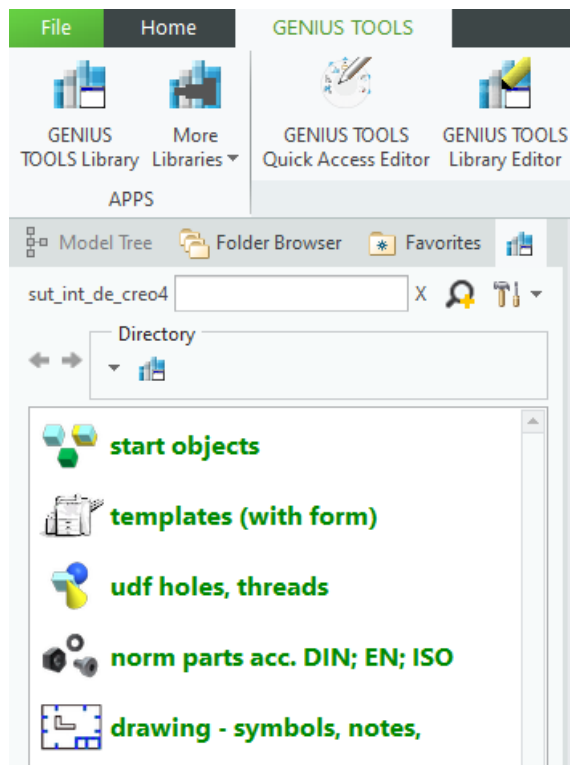
The display of the Library browser depends on the configuration option `gtl_run_mode`. This option determines whether the library browser is displayed in the Creo navigation area or as separate window.

If the Library browser is displayed in the Creo navigation area, each opened Creo window has its own library browser.

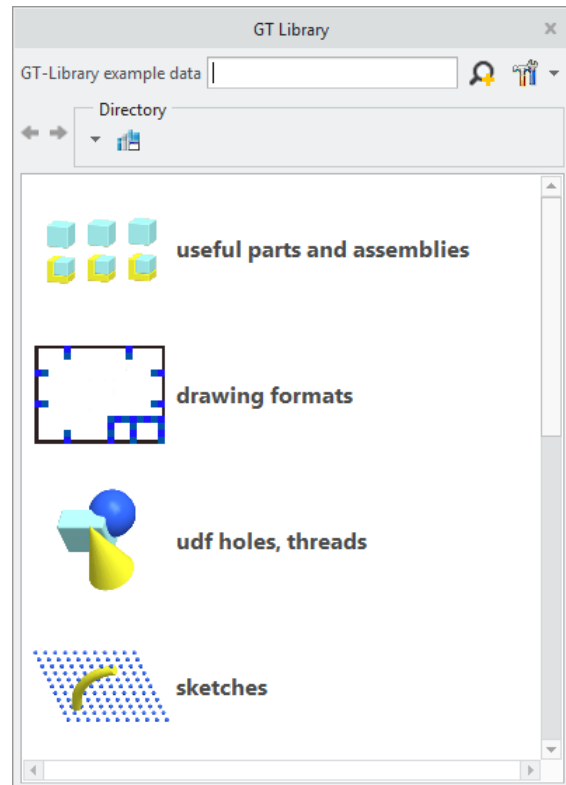
If the Creo model tree is detached and the Library browser is not configured to display in an extra window, the Library browser is displayed in the same window as the model tree.

The library browser as an independent Creo window is a library for all Creo windows and can be positioned freely. This setting is a useful choice for multi-monitor setups.

Tip: Use the pin button to attach the standalone window to its position.



GENIUS TOOLS Library in the Creo navigation area



GENIUS TOOLS Library as separate Creo window

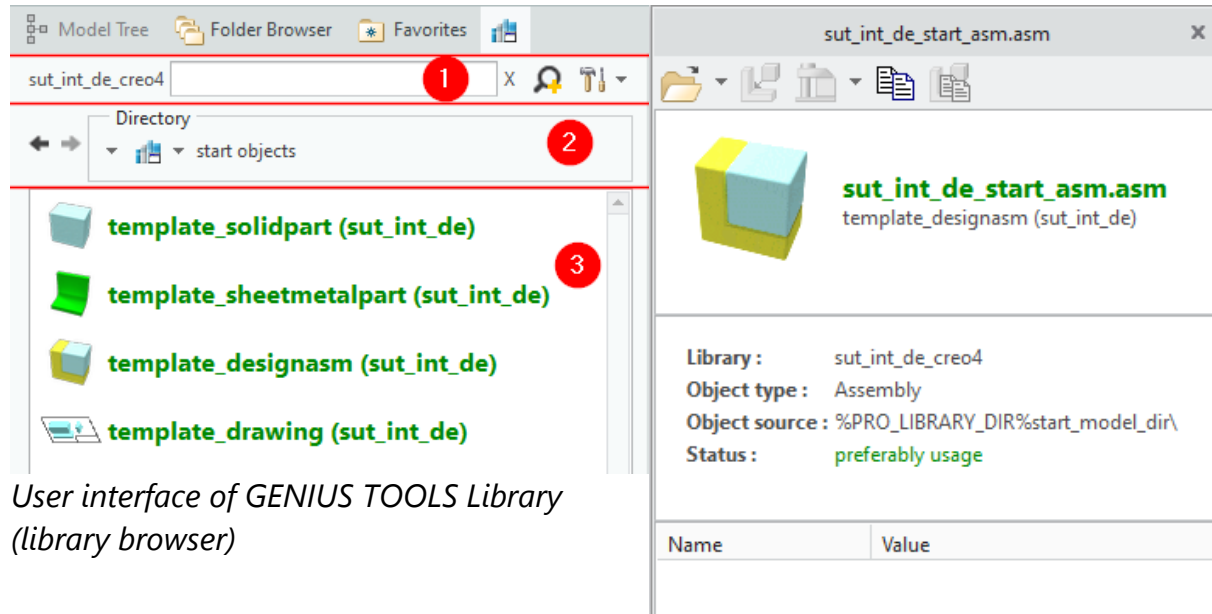
Open library from another library

An object of the link type is created in a library. In the object property Link the opening link is entered. This opening link can be copied to the clipboard in the target library in the RMB menu of the category tree.

Example: sut_int_de_creo | (Opens the sut_int_de_creo library on the top level)

9.2.1 User interface

The user interface of GENIUS TOOLS Library consists of the following elements:



User interface of GENIUS TOOLS Library
(library browser)

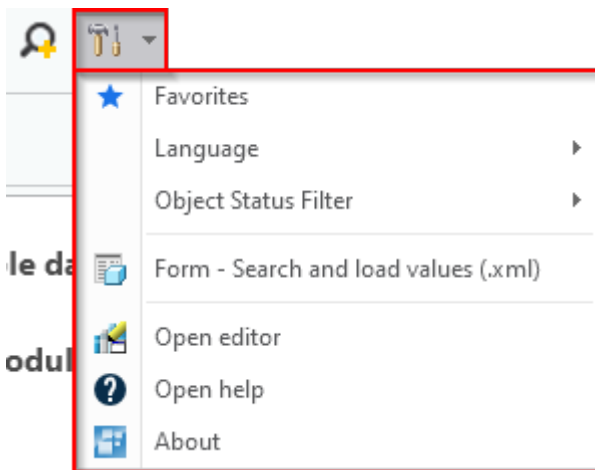
Details window

1. Search and filter¹⁹⁰ with tool menu
2. Navigation bar and library selection¹⁹⁸
3. Object display

The display of library objects is controlled through the search- and filter area or via the path option. Click on a library object (here: template_designasm) to open the [detail window](#)²⁰¹ with information about and actions on the library object.

9.2.2 Tool menu

You find the tool menu  in the Library browser directly next to *Object search*.



To hide or show the Favorites bar, chose *Favorites* in the tool menu.

Change the *Language* for the shown Library objects.

*Object Status Filter*¹⁹⁰ filters Library objects by means of their status. See also chapter *Search and filter*¹⁹⁰.

Form - Search and Load values (.xml) - Further explanation can be found in chapter *Load Form values from XML*²⁰⁶.

Open editor opens the GENIUS TOOLS Library Editor to work directly in the current library.

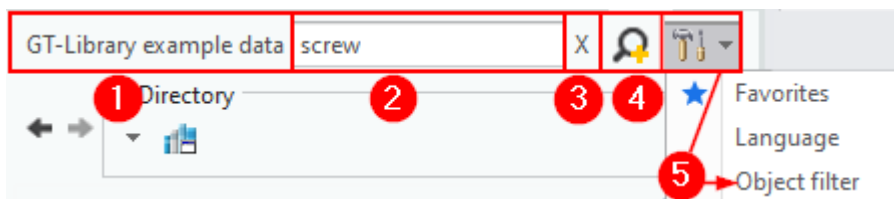
Open help opens the GENIUS TOOLS for Creo help.

About opens the GENIUS TOOLS for Creo settings.

9.2.3 Search and filter

Search the current library by keywords or use additional search criteria in the advanced search.

The search and filter area consists of the following elements:



1. Selected database
2. Direct search¹⁹¹
3. Clear search field
4. Advanced search¹⁹² (Object type and Parameters/Dimensions)
5. Tool menu with Object status filter¹⁹³

1. Selected database

This area displays the name of the library which you have selected in the navigation bar.

2. Direct search / search field

2.1. Search in current category and whole library

Enter a search term in the search field and confirm with Enter. The currently selected category is searched through and results are displayed in the object display.

If you press Enter a second time, the search is conducted across all library objects.

If you want the search to be conducted immediately through the whole library, i.e. without searching first the current category, set the configuration option *gtl_search_tree_dependent* to 0. (Default is 1, enabled)

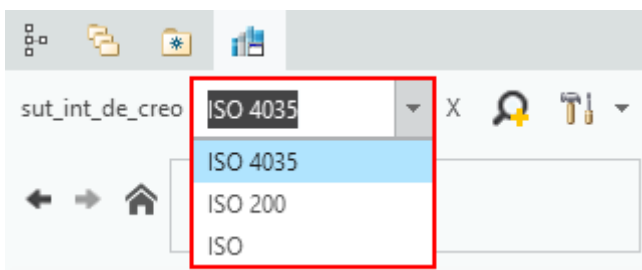
When changing the search term, the direct search function starts again in the selected category.

2.2. Search terms

You can enter multiple search terms separated by space character. This finds objects that contain each term, i.e. a Boolean AND is used. You can use further search operations. (See: [List of search operations](#)¹⁹⁵)

Please note: Entries in the direct search field will overwrite any filter in the advanced search, if the two entries conflict. If you have, for example, clicked on assembly in the advanced search and then enter *type:drw* in the direct search, you will receive all hits for drawings.

Additionally, all searched terms of a Creo session are now visible in a dropdown list.



Search history of current session

2.3. Exclude from search

Be aware that all object data, including parameter values, are searched through. You can exclude individual parameters and dimension from the search index, see [Customizing the search function](#).²⁴⁷

3. Empty search field

Deletes all inputs in the search field of the library browser.

4. Advanced search

The magnifying glass button (4) opens the advanced search dialog. With the advanced search, a search can be narrowed down by object types, parameters and dimensions.

4.1 Information for direct search inputs

The dialog lists further search operations that help you receive better search results. See [List of search operators](#)¹⁹⁵.

Use the *Clear Search* button in this area to delete all inputs in the search field of the library browser.

4.2 Object type

In this segment you can filter by object type. Activate the checkbox in front of an object type and click *Change Filter and Search*. The filter is immediately transferred to the search field – in addition to search terms that may have been entered previously – and the search is executed.

Filtering by object types in the Advanced Search dialog box

Object types are combined with OR. Exceptions are the object types *is Instance* and *is Generic*: they are combined with AND. For example:

- *Part* and *Assembly* are activated in filter: A library object must be a part or an assembly.
- *Part* and *is Instance* are activated in filter: A library object must be a part and an instance.

Please note: The object type filter is added to any input in the direct search field, i. e. all search items are AND-associated.

4.3 Parameters and Dimensions

In this area, filters are created based on parameter and dimension values. Parameters and dimensions must be added to a library database to be accessed here.

Select a parameter or dimension from the list. Then select a value from the value list or enter a value freely. Refresh the search using the *Add Filter and Search* button.

With free input, the values in the table are filtered automatically.

"Parameters and Dimensions" area in the advanced search

4.4. 3D Model Compare

The section for searching for 3D-like objects only appears if the selected library contains 3D-indexed objects. Objects are indexed in the [Batch Mode](#)²⁴² dialog.

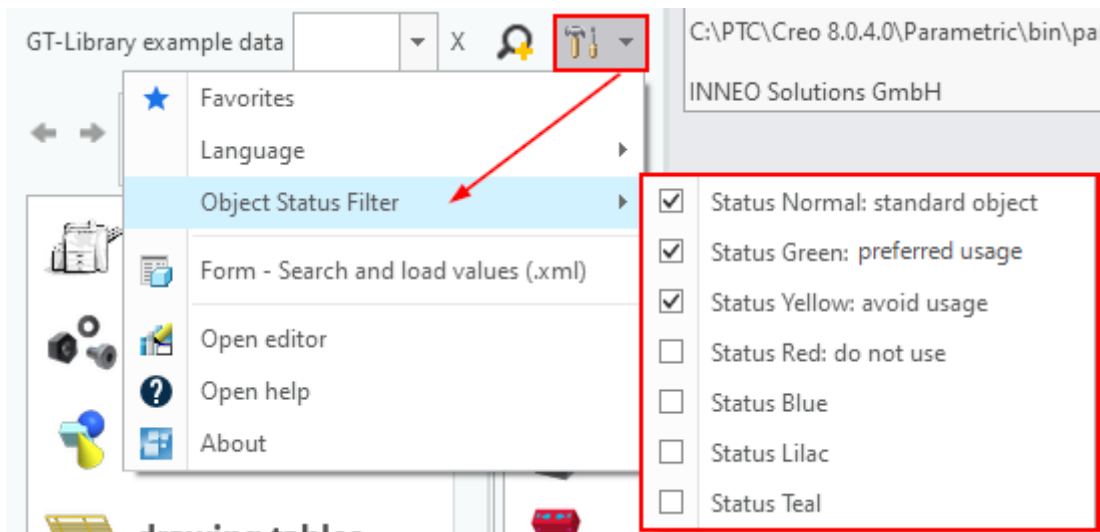
Section "3D Model Comparison" in the Advanced Search dialog

The search buttons correspond to the search operators *mdl3d*, *mdl3d_bb* or *mdl3d_vo* which you can alternatively enter directly in the search field ([Direct search](#)¹⁹¹).


Object status filter

The Object status filter can be found in the [tool menu](#)¹⁸⁹. The object filter allows you to filter library objects by their status. For an exact definition of the status values refer to your working instructions.

The status of a library element is defined in [GENIUS TOOLS Library Editor](#)²¹⁵. Colors can be changed by setting the configuration options `gtl_*_color`.



Filter library objects according to their status

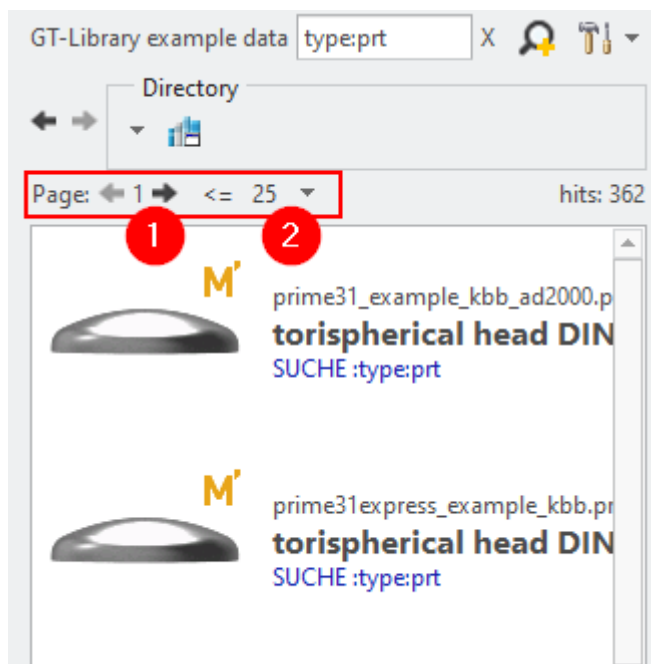
Open the **tool menu**  and select the statuses to be displayed in the *Object filter*. All library elements with the selected status are displayed in the object display.

Note that the object filter is not applied to categories, i. e. a „green“ object will be displayed even if it is included in a „red“ category.

Please note: The object status filter is added to any item in the direct search. If the two clash, the input in the direct search field will replace the object status filter. If you specify, for example, in the advanced search dialog the object status to be green – by typing `status:2` – then an object status filter that has *Status green* unchecked, will be disregarded.

Navigating search results

After a search item has been entered an additional bar appears which displays information on pages (1) and hits (2).



9.2.4 List of search operations

You can enter the following search operations in the direct search field. Delete previously entered search items.

Search operation	Term	Explanation	Input example
Object type*	type:	Finds objects of a specific category, file extension (.prt, .asm usw.) as well as instances and generic parts Note: Don't insert a space character between multiple terms	type:prt type:prtm,asm
Object status	status:	1=normal, 2=green, 3=yellow, 4=red, 5=blue, 6=lilac, 7=teal (turquoise)	status:2
Parameters and value	parameter-name=value	Finds objects with the specified parameter and its value	material=wood
Dimensions and value	maß=wert	Finds objects with the specified dimension and its value	d19=10

Search operation	Term	Explanation	Input example
Literal search	quotation mark: " "	Finds objects that contain the exact phrase with all words. There is no case sensitivity.	"Hexagon regular nut DIN EN ISO 4032"
BOOLEAN AND	AND	Finds objects that contain each search term	Hexagon regular nut AND Plate
BOOLEAN OR	OR	Finds objects that contain one of the search terms	Hexagon regular nut OR Plate
Parenthesize search terms	Brackets: ()	Joins two or more search items Note: You must type a space character between a value and a bracket	(m6 OR m10) type:prt
Exempt from search	exclamation mark: !	Does not display the objects that are searched for after the exclamation mark (e.g.: Find all objects except of the type part) Note: You must type a space character after the exclamation mark	! type:prt
Hidden objects	!tree	Displays objects that have not been added to the model tree	!tree
Hidden objects	!selection	Displays objects that have not been added to the selection	!selection
Similar objects **	mdl3d	Displays objects similar to the one in use	mdl3d
Similar objects – bb**	mdl3d_bb	Displays objects with similar bounding borders	mdl3d_bb
Similar objects – vo**	mdl3d_vo	Displays objects with similar voxels (grid points)	mdl3d_vo

*this search item can also be generated by the object type filter in the dialog box Advanced Search

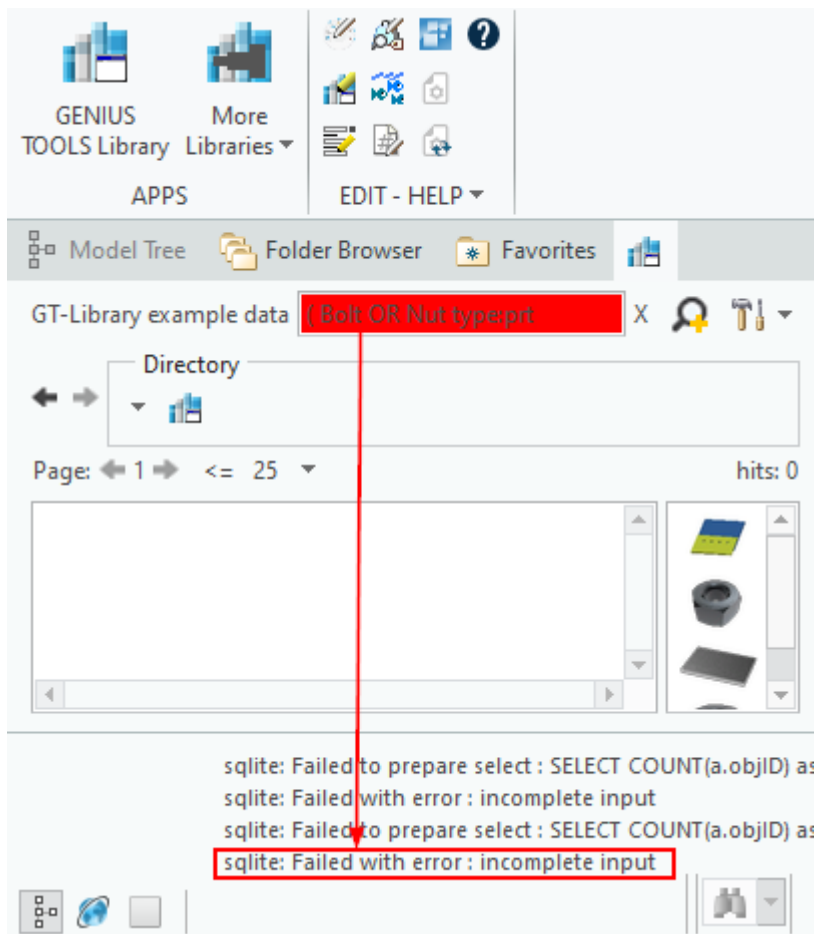
**for this search an object needs to be active (opened)

Examples

1. The search entry `type:prt,asm,drw` searches for all parts (.prt), assemblies (.asm) and drawings (.drw)
2. The search entry `"material=gold"` searches for all objects with a parameter "Material" and the parameter value "Gold".
3. The search entry `(status:2 OR status:3) material=gold ! type:asm` searches for all objects with the status Green and status Yellow and with a parameter "Material" and which are not assemblies. Take care to set a space character after and in front of the brackets.
4. The search entry `! status:3 ! status:4` searches for all objects that neither have the status Green nor Yellow.
5. The search entry `drill bush ! "DIN 179"` searches for all objects which contain the term "drill bush" but not the term "DIN 179".

Examples for incorrect search entries

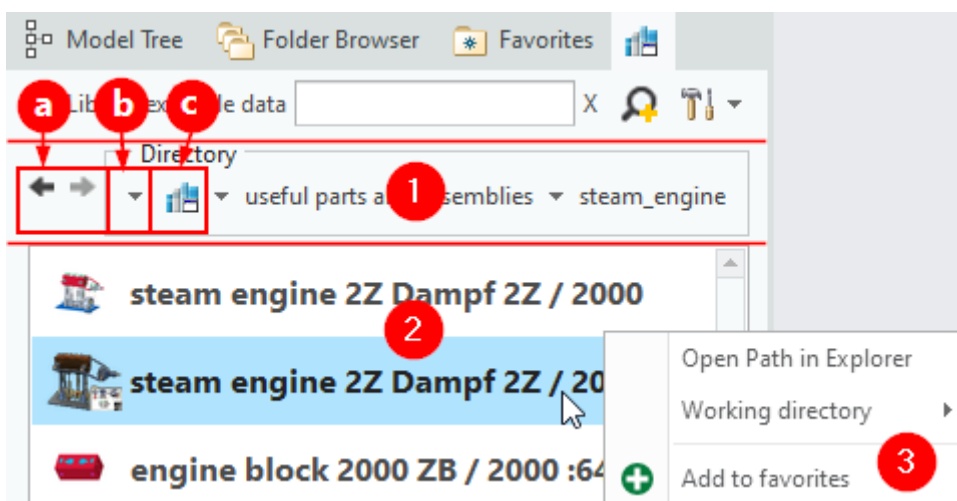
1. The search entry `status:2 status:3` is a Boolean AND and does not lead to any hits since objects cannot hold two status. Use a Boolean OR if you wish to search for all objects that either have status 2 or status 3.
2. The search entry `(Mutter OR Nut type:prt` generates an error message because a bracket is missing:



Error message when completing a search (8)

9.2.5 Navigation and object display

Below the search and filter area is the navigation area and the object display. The display of information in this area depends on the configuration.^[212]



1. Navigation area

a) Back and Next buttons

- b) Select library
- c) first level of selected library

2. Object display

3. Context menu (open with right-click)

Configuration options

If you want to open library objects and categories with a double click instead of a single click, set the configuration option `gtl_list_use_with_double_click` to 1.

1. Navigation area

The navigation area shows the path within the currently loaded library. Individual elements of a path (categories) can be directly selected by clicking on them. Their contents are displayed in the object display.

Use the Back and Next buttons (a) to navigate through your history (navigation and search history). Click on the library icon (c) to return to the first level of a library.

To switch libraries, click the arrow icon (b) in front of the library icon. Then select the new library from the list. When switching libraries the history is lost. Use the entry Home database to jump directly to your home database (depending on the configuration option `gtl_home_db`).

Please note: The home button will only be displayed if a home database has been specified.

2. Object display

The object display shows library objects and categories with preview and description. The status is indicated by the text color. (See also chapter [Object details](#).²²²)

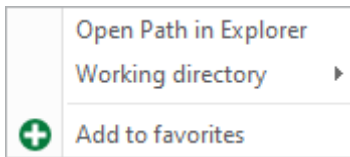
Library objects differ from categories by the additionally displayed filename which can be configured. You may also configure the display of a colored triangle next to a category.

Click entries in the Library browser to navigate through the category tree. The detail window is displayed as soon as the selected object is a library object.

Tip: The size of displayed preview images (40 or 100 pixels) is influenced by the configuration option `gtl_img_size`.

3. Context menu

Right-click on an object to open the context menu.



Context menu for a category

3.1. Context menu for categories

Info: Opens a linked information document. The function is only displayed when a document has been linked to a category.

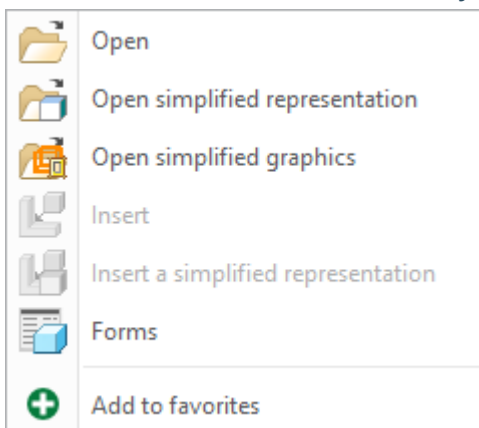
Open Path in Explorer: Opens a deposited path in Windows Explorer.

Change working directory to selected folder: Sets the Creo working directory to the deposited path.

Please note: *Open Path in Explorer* and *Change working directory to selected folder* are displayed only when library categories correspond to a folder path. Usually, there is no object source specified when using library categories only for logical structuring, and the functions cannot be used.

The structure of a context menu for categories depends on the category's configuration.

3.2. Context menu for library objects



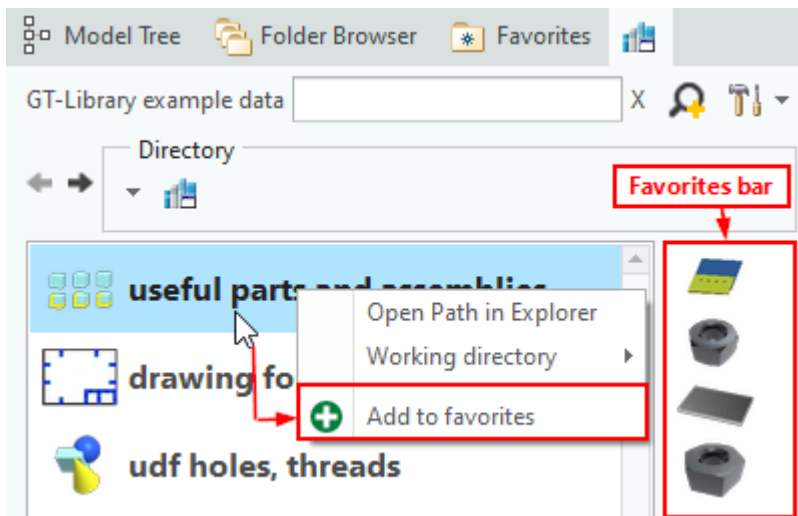
Context menu for a library object

The context menu for library objects depends on the object type and configured actions. Also refer to Actions on library objects²³⁶!

9.2.6 Favorites

You can display library objects that you frequently need in a favorites bar in the library browser. You can mark both individual library objects and categories as favorites. A favorites bar is available for each library.

Adding and removing favorites

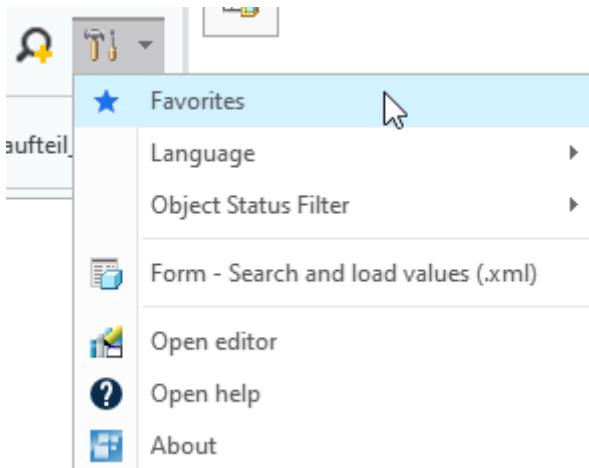


To add an object to the favorites bar, right-click it to open the context menu and select *Add to Favorites*. The selected object is displayed in the favorites bar with a small preview image.

To remove an object from the favorites bar, select the object in the favorites bar, then open the context menu and select *Delete from favorites*.

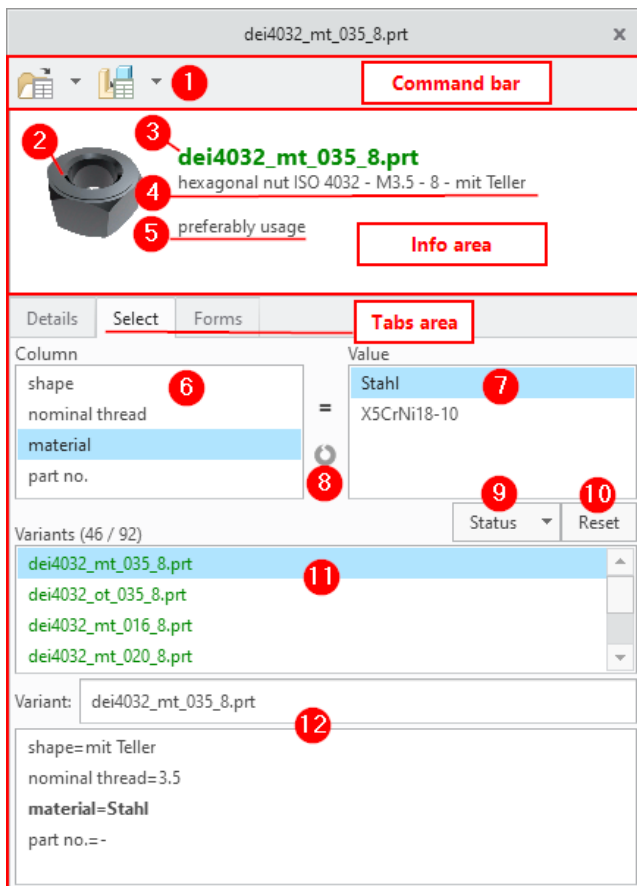
Showing and hiding favorites bar

To show or hide the favorites bar, go to the tools menu and select *Favorites*.



9.2.7 Detail window

The details window displays information about the currently selected library object. It is displayed when you click on a library object.



Detail window with active Selection tab

Command bar

Displays all actions that are available for the selected library object. The active tab can also affect the displayed actions.

1. Actions buttons: in the command bar depend on object type, configuration of a library object, and on the current Creo mode, see also [Actions on library objects](#)²³⁶.

Information area

This area displays a preview image and information on the object. The object selected in the Select tab affects the displayed information.

The displayed information can be configured.

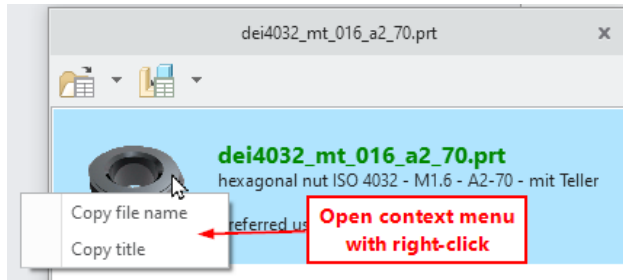
2. Detail image: Is the same as the preview image, but can be replaced by a more detailed, larger image, see chapter [Working with images](#)²³⁴.

If a separate [detail image](#)²³³ has been created, you can hide the display of the file name and object title with the configuration option `gtl_detail_window_image_show_title=0`.

3. File name

4. Object title: Can be hidden with the configuration option `gtl_detail_window_hide_details`. (Default: 0/active)
5. Status information: Can be displayed with the configuration option `gtl_detail_window_show_status_in_head_area`. (Default: 0/deactivated)

Right-click to open the context menu for copying the file name and title to the clipboard.



Tabs

The tabs displayed below the information area, depending on the selected library object.

By default, the Forms tab is active.

- for parts, if a form is available, otherwise the Selection tab is active: The active tab can be changed with the `gtl_detail_window_preselected_tab` configuration option.
- for UDF-Forms: The active tab can be changed with the `gtl_detail_window_preselected_tab_gph` configuration option.

Details tab

Displays general information on the library object: library, object type, object source and object status²²⁹.

- This tab is always displayed.
- If there are no other tabs in the Details window, this tab covers the entire area.

Select tab

This tab displays a selection table of the library object and enables filtering by following criteria.

With the help of the selection window criteria for the selected library object can be specified.

5. Feature selection: Chose variants of the library objects by their properties.
6. Property value: Chose values to the corresponding property.
7. Cursor skip: Automatically skips to the next property.

8. Status filter: Filters the instances by [object status](#)¹²⁴. The preselection of the checkboxes depends on the selection in the [object status filter](#)¹⁹³ in the tools menu. However, the selection set here does not affect the object status filter.
9. Reset: Deletes the chosen selection.
10. Variant selection: Variant of the available library object by means of the selected properties.
11. Variant display: The above chosen variants of the selected library object are displayed in bold type.


The content depends on the configuration in the GENIUS TOOLS Library Editor.

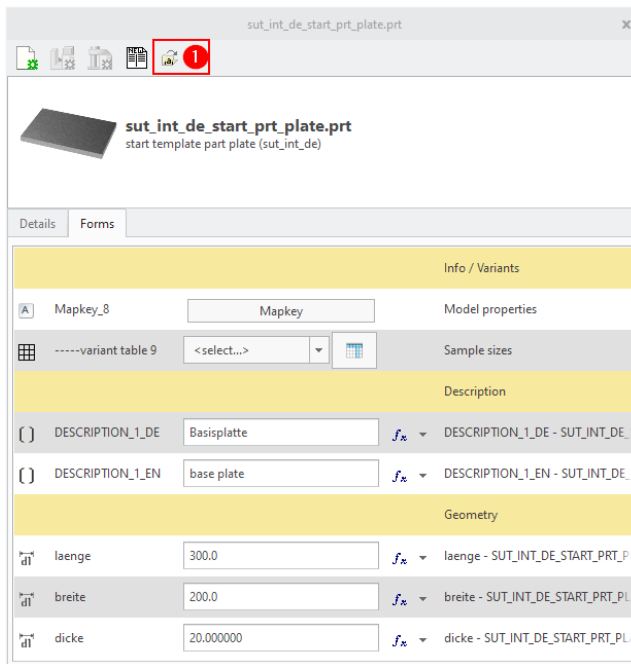
Forms tab

The Forms tab in the Details window displays Forms and UDF Forms for library objects.

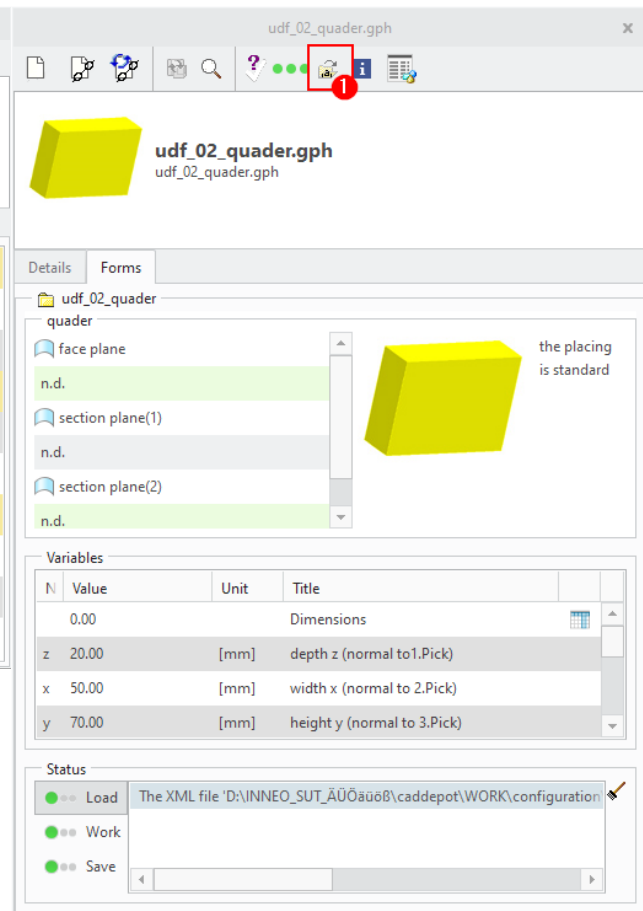
- Displays a Form or UDF Form (created with GENIUS TOOLS Forms or UDF Forms) from the library database matching the library object.
- If a selection table exists for a library object, the Forms tab is always displayed. Individual variants can contain Forms.

Tip: Use [GENIUS TOOLS Forms](#)⁸⁵ and [GENIUS TOOLS UDF Forms](#)⁴³⁷ for editing models that have already been created and UDFs that have already been inserted.

Use the [tool menu](#)¹⁸⁹  in the Library browser to access the function *Form - Load values (.xml)* if your company uses the XML interface for filling in Forms automatically. For information on how to work with the XML interface, see [Forms search and read values from XML](#)²⁰⁶.



Form in the GENIUS TOOLS Library Details window



UDF Form in the GENIUS TOOLS Library Details window

1) The function Read values from CSV searches mask values and reads these values from a CSV-file.

Forms

Models such as assemblies or parts can contain a Form. If the Form has also been imported into a library, it is displayed in the Forms tab in the Details window.

Use Forms to customize dimensions and parameters prior to assembling or opening a model, or to control it via value tables.

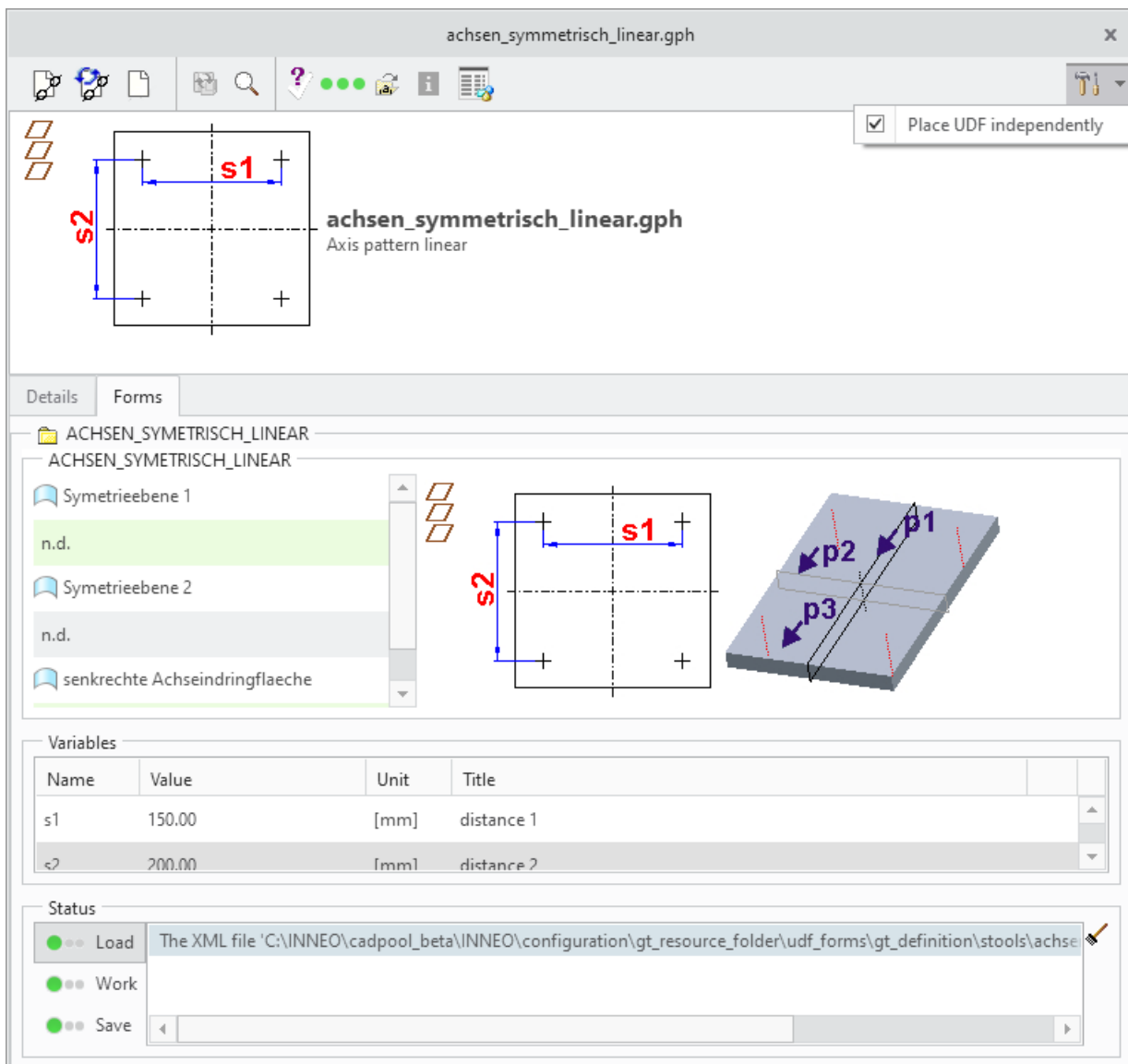
Using Forms in the GENIUS TOOLS Library Details window is similar to operating GENIUS TOOLS Forms.

UDF forms

If UDFs have been included in a library, they can obtain a UDF form. This will be displayed in the Forms tab in the Details window.

The command bar differs greatly for a UDF Form (i. e. a GPH file). Check the usage in the chapter [GENIUS TOOLS Forms](#)⁴³⁷.

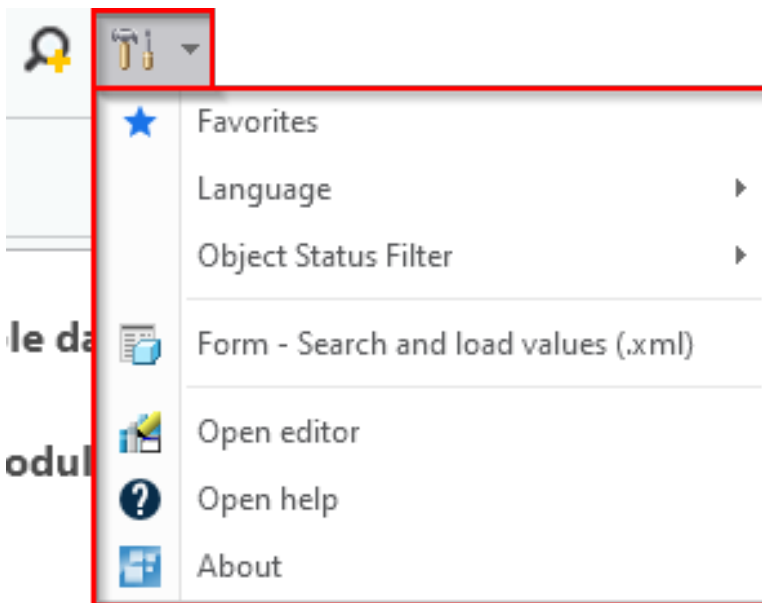
Use the UDF form to define placement references and to customize properties prior to assembly, e. g. by placing it dependently²⁴⁴ or independently.






Detail window for a UDF form

9.2.8 Forms search and read values from XML

GENIUS TOOLS Library comes with an XML interface for importing pre-calculated values for Forms. Your company has to define whether this interface is used and for which Forms to create XML files. (See chapter XML interface for Form values²⁴⁸.)



To load Form values from an XML file, proceed as follows.

1. Navigate to the Library tab in the navigation area or switch to the Library window.
2. In the **tool menu** , select *Form - Load values (.xml)*. A file selection dialog is displayed.
3. Select the XML file that contains the required values and click *Open*. The list in the library browser is filtered to the object or objects that contain the Form specified in the XML file. The Form is filled in the background.
4. Click the library object with the Form to open the detail window. The values from the XML file have been set in the Form.
5. Verify the Form values. You can copy the library object now  or verify the names for the file copies . The names for the file copies are specified in the XML file together with the Form values.

9.2.9 Use cases

This section contains step-by-step examples for using GENIUS TOOLS Library.

9.2.9.1 Using library objects

Assembling a standard part with selection table

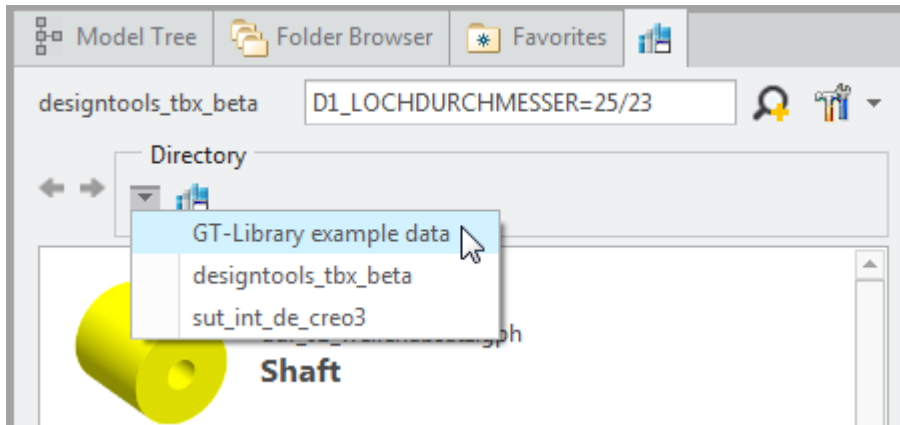
Prerequisites:

- an assembly to insert a standard part into
- a properly configured library database (actions, status)

– standard parts with selection tables in the database

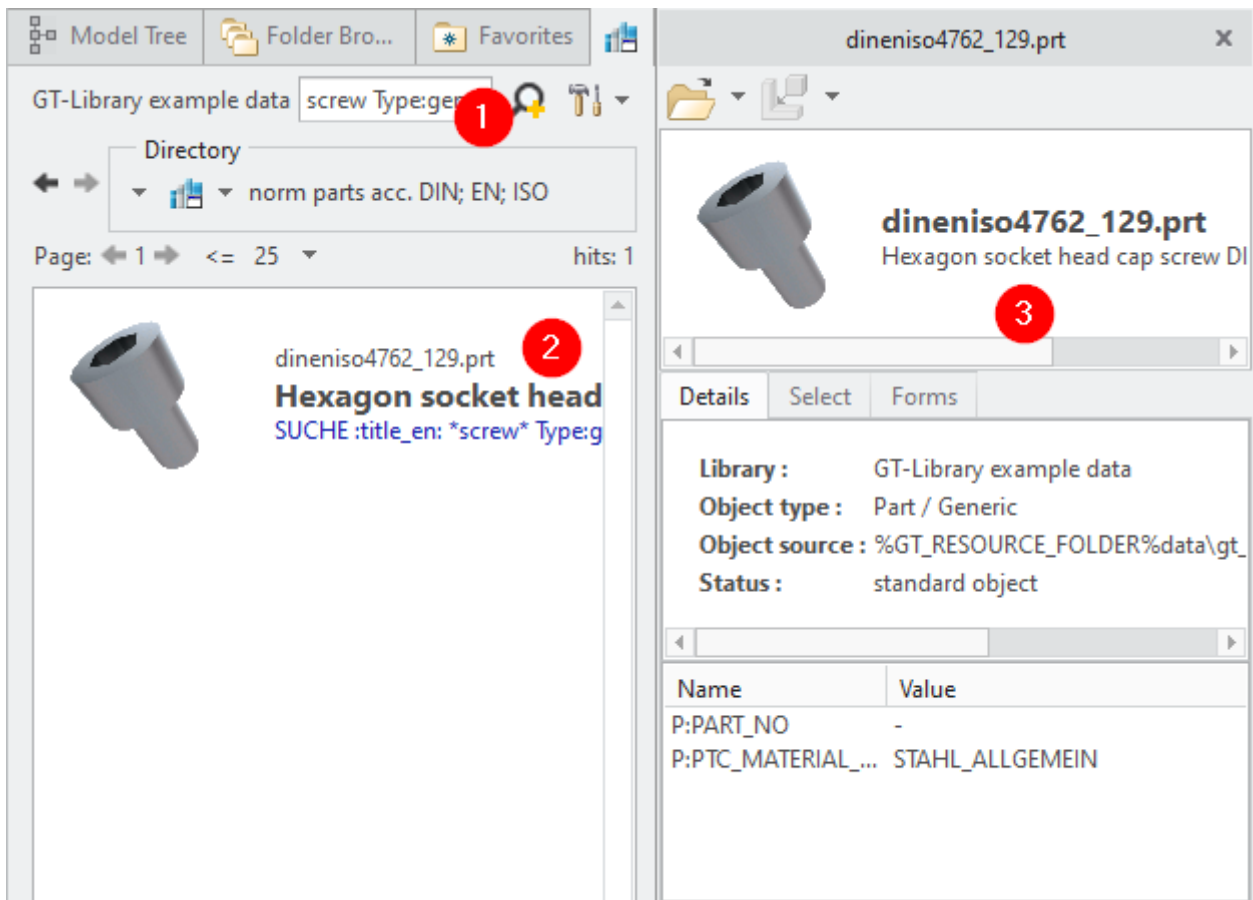
Steps: 1 – Library browser

1. In Creo Parametric, switch to the assembly to insert the standard part. Make sure not to have a mode such as component placement or sketch mode activated.
2. Navigate to the Library tab in the navigation area or switch to the Library window.
3. Open the library containing the standard parts needed.



Opening the library that contains the standard parts needed

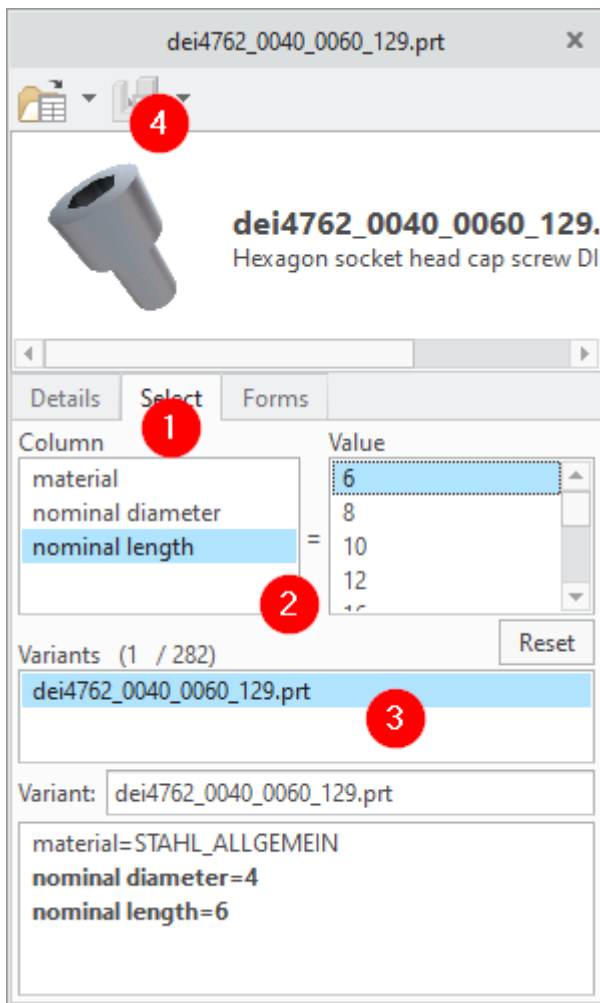
4. Use the search field (2) to find the standard part or navigate the library structure.
5. Click the standard part in the object display (2) and switch to the Details window (3).



Standard part in library browser (left) and in details window (right)

Steps: 2 – Details window

6. Open the *Select* tab (1).
7. Select the standard part using the selection table (2).
8. Select the desired variant (3).
9. Click the *Insert selected variant* button (4).
10. Insert the part into the assembly as usual in Creo.



Details window

Assembling a new part with a Form

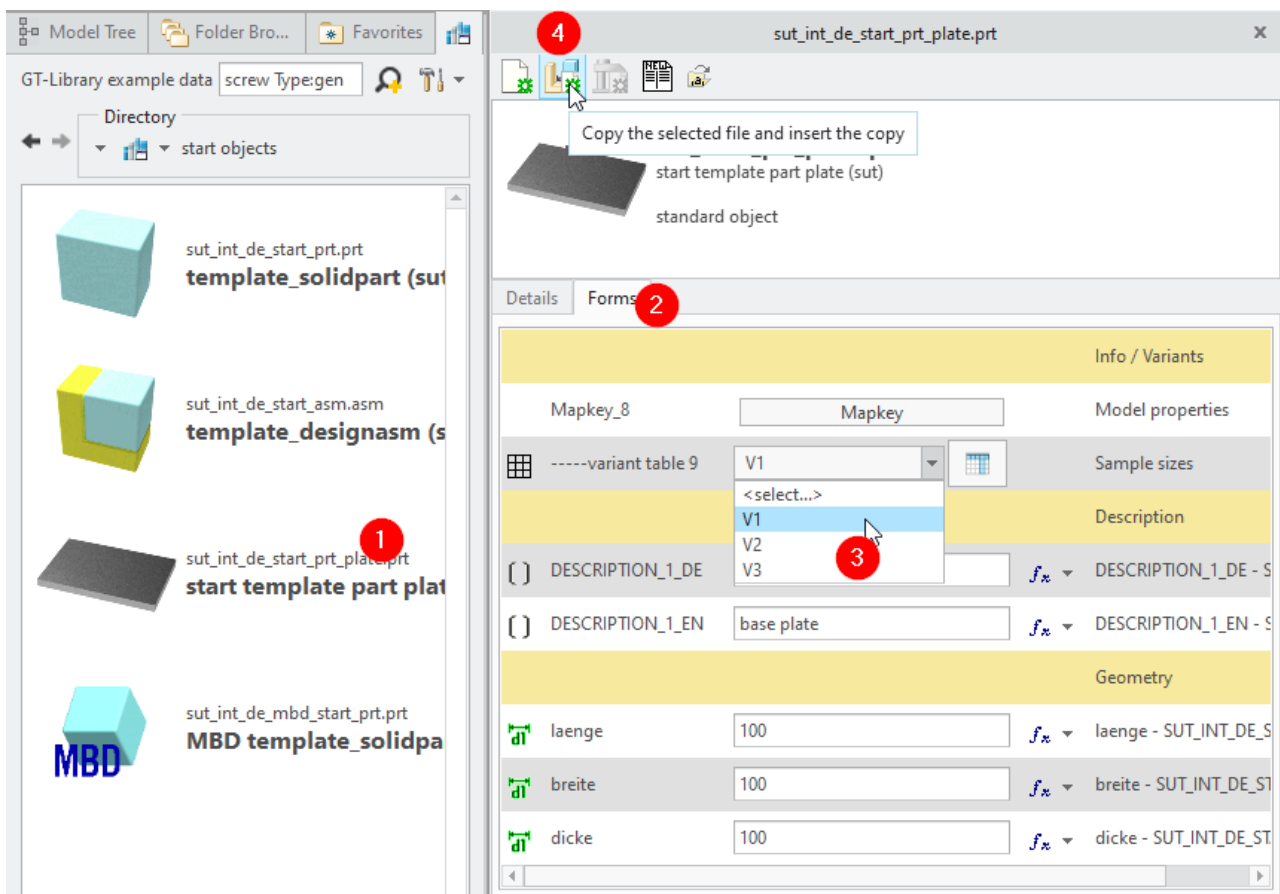
Prerequisites:

- an assembly to insert the part into
- a properly configured library database (actions, status)
- a part with an external form in the database

Steps:

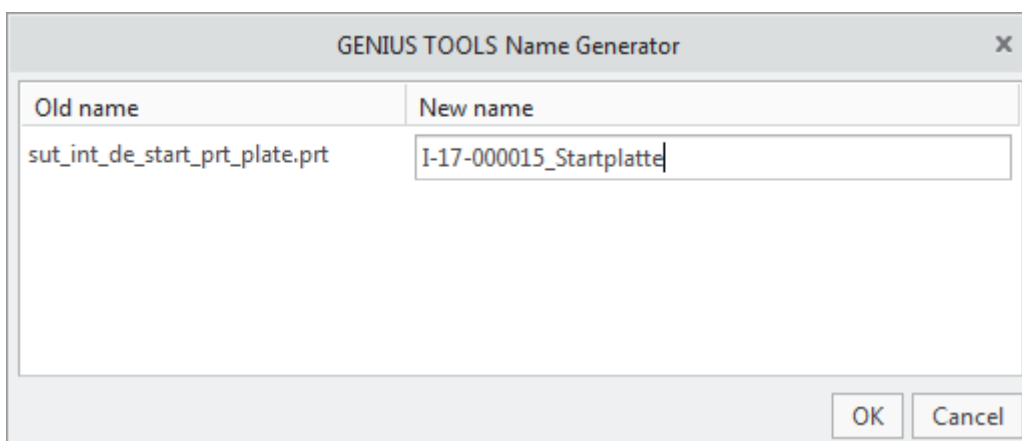
1. Switch to the assembly to insert the part in Creo. Make sure not to have a mode such as component placement or sketch mode activated.
2. Navigate to the Library tab in the navigation area or switch to the Library window.
3. Open the library containing the part with the form.
4. Use the search to find the desired part or navigate the library structure.
5. Click the part in the object display (1) and switch to the Details window.
6. Open the *Forms* tab (2).

7. Configure the part in the form (3).
8. Click the button *Copy the selected file and insert copy* (4).



Object display in library browser (left) and in details window (right)

9. Select a Name Generator in the following dialog and confirm your selection with *Next*.
10. Complete the new part name in the last dialog and click *OK*.



Dialog box of GENIUS TOOLS Name Generator

11. Insert the part into the assembly as usual in Creo.

9.3 Configuration

In this section, you will find further information on the folders used in the resources directory. In addition, you will find information on configuring libraries for GENIUS TOOLS Library in the GENIUS TOOLS Library Editor tool, and on using multi-lingual libraries.

9.3.1 Areas in the resource directory

GENIUS TOOLS Library uses the resources directory of GENIUS TOOLS for Creo. In the ... `\gt_resource_folder\library` folder, you can find the Library-database (file extension: .db) and a folder with same name for each database.

Resource directory in a Startup TOOLS installation: `%GTS_ROOT_DIR%\configuration\gt_resource_folder\`

Resource directory in a GENIUS TOOLS for Creo installation:
`<GTfCDirectory>\gt_resource_folder\`

The path to the resource diorectory can be edited with the configuration option `gt_resource_folder`.

These folders contain the preview images for the library objects. The original preview images for library objects are located in the *img* subfolder. The *img_w40* and *img_w100* subfolders contain automatically imagestyleclass="Default" scaled preview images. The optional *img_detail* and *img_tooltip* subfolders contain images for the detail window and images to display as tooltips for library objects.

In addition, you can find backup copies of your databases in the *library* folder. As soon as a database is opened via the Library Editor, a backup copy is created. However, only one backup copy per day is created.

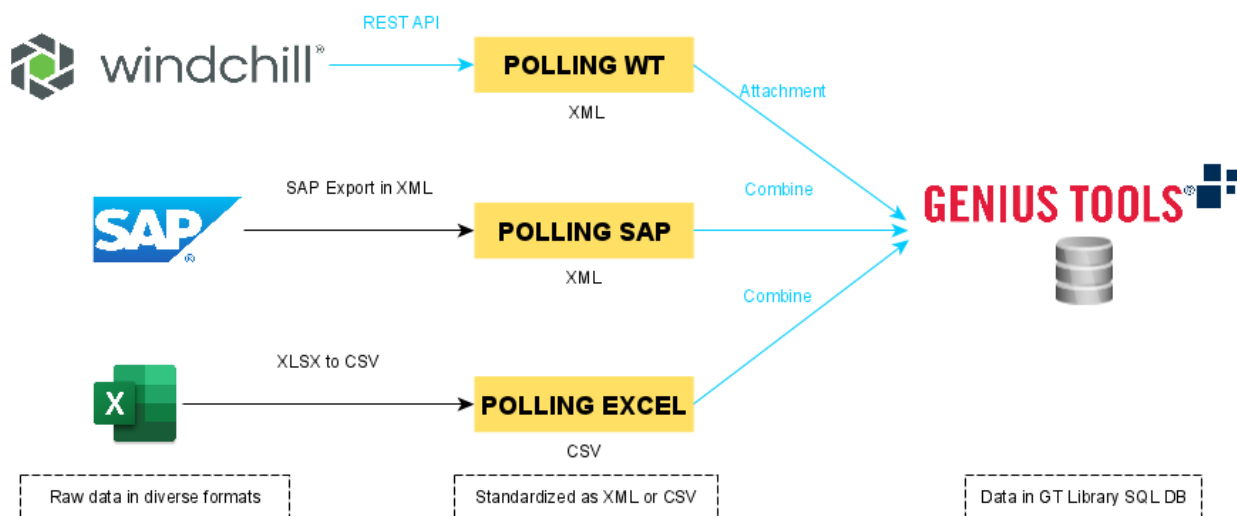
To restore the initial state of a database, unload it from the editor and the Library browser. Then, delete the database. Rename the desired backup copy with the name of the database. The database is now ready for use again.

9.3.2 Import data from PDM/PLM

You can import Creo model and meta data from an existing library or other directory structure, e. g. from a PDM, PLM or ERP system, into a library for GENIUS TOOLS Library. This process is carried out by our module GENIUS TOOLS Library Data Importer.

In addition, information from other systems can be transferred. The combined information are updated and can be searched.

All processes that import, convert or map can be started in a time-controlled way. Source formats can be CSV, XML (with XSLT) and Windchill REST.



For further information consult the manual *GENIUS TOOLS Library Data Importer.pdf* which is located in the *help* directory in your operating environment.

9.3.3 Multilingual use

For distributed work at multiple cross-country and cross-language sites, GENIUS TOOLS Library supports a multi-lingual user interface as well as multi-lingual libraries.

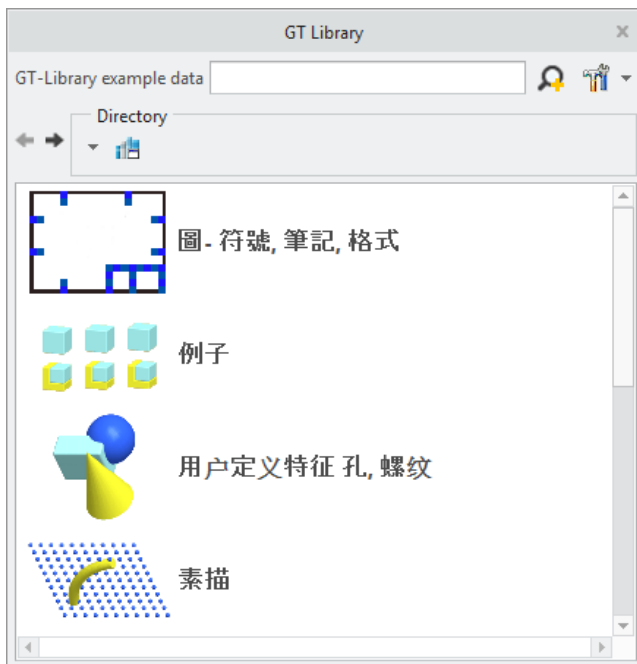
User interface

The GENIUS TOOLS Library user interface depends on the language in which Creo Parametric was started. Currently the languages German and English are available.

When Creo Parametric is started in another language, the GENIUS TOOLS Library user interface is displayed in English.

Multi-lingual databases

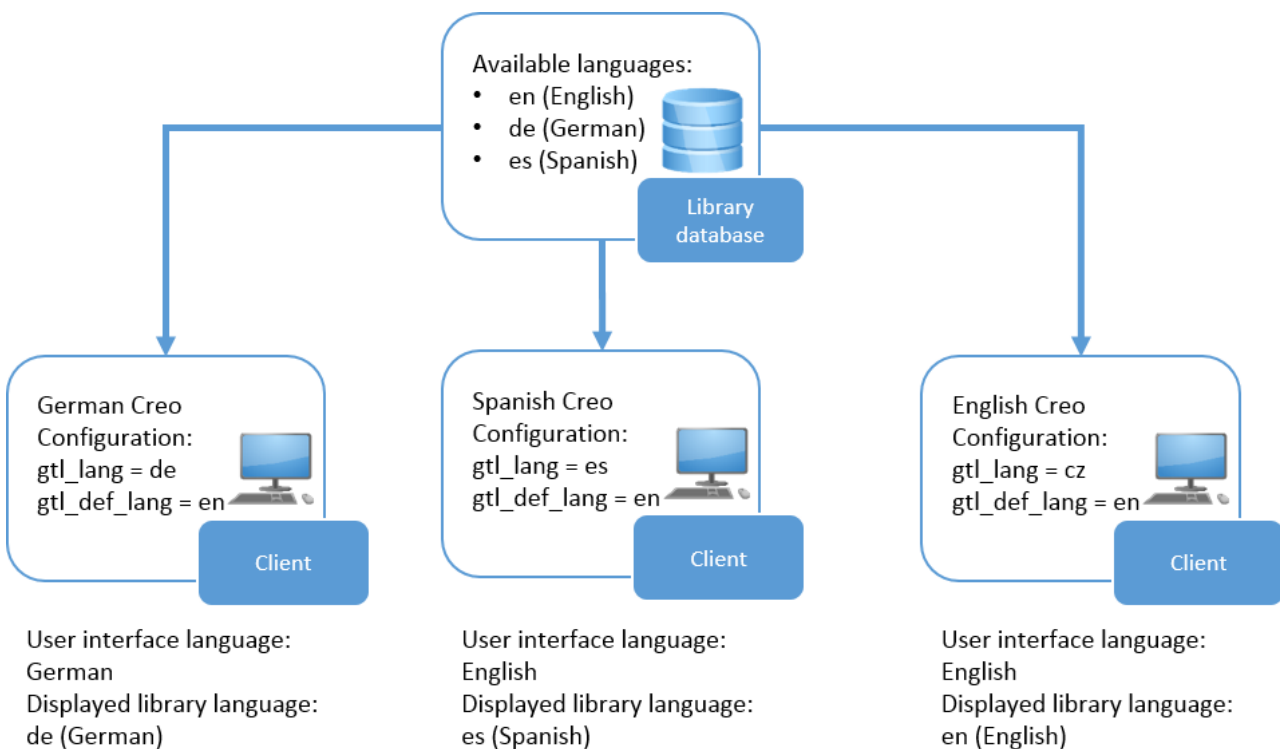
GENIUS TOOLS Library databases can contain information in various languages. New languages can be added to library databases via the Library Editor using language codes.



GENIUS TOOLS Library with Chinese data

The language displayed in GENIUS TOOLS Library depends on the client computer configuration. The Configuration option `gtl_lang` specifies (via language code) which language to display.

The Configuration option `gtl_def_lang` specifies a language via language code to be used as a fallback variant if the language specified in `gtl_lang` is not found in a library database.



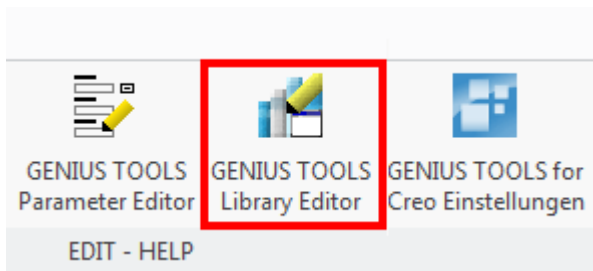
Displayed localization of GENIUS TOOLS Library and the database contents

9.3.4 Library Editor

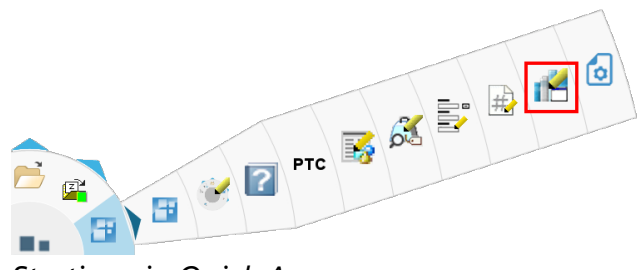
With GENIUS TOOLS Library Editor you manage your libraries and library content for of GENIUS TOOLS Library.

Starting the program

Start GENIUS TOOLS Library Editor via the GENIUS TOOLS ribbon menu or via GENIUS TOOLS Quick Access ([<] key).



Starting via the ribbon menu



Starting via Quick Access

Saving changes

The changes made in the editor are saved in the resource directory.

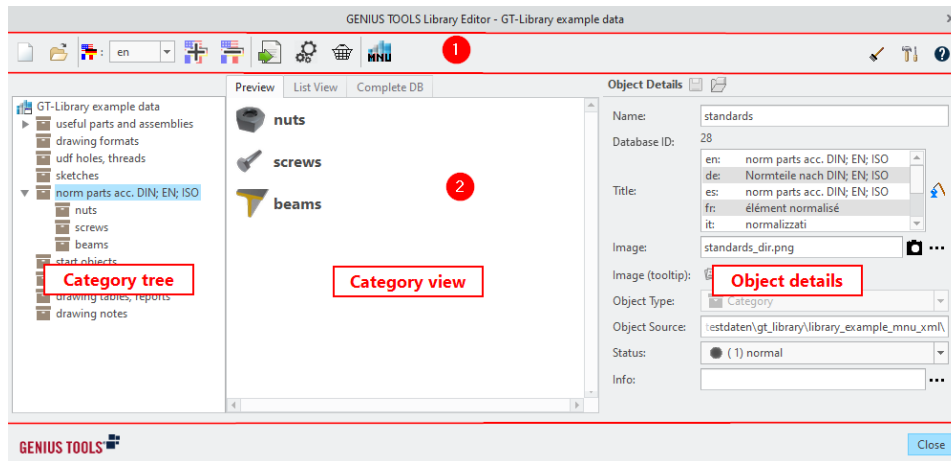
When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer (Caddepot). Path: `<operatingenvironment>/parametric/configuration/gt_resource_folder`.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

9.3.4.1 User interface

The user interface of GENIUS TOOLS Library Editor consists of the following elements:











1. Command bar²¹⁶
2. Library management with
 - Category tree²¹⁷
 - Category view²¹⁹ with the tabs Preview, List View, Complete DB
 - Object details²²²

9.3.4.2 Command bar

The following buttons are contained in the command bar:

Icon	Name	Description
	Create new database	Creates a new database in the Library resources directory. Note: The database is opened immediately. A database already opened is automatically saved and closed.
	Open database	Opens a database. Note: A database already opened is automatically saved and closed.
	Change displayed language	Changes the displayed language of library elements in the editor.

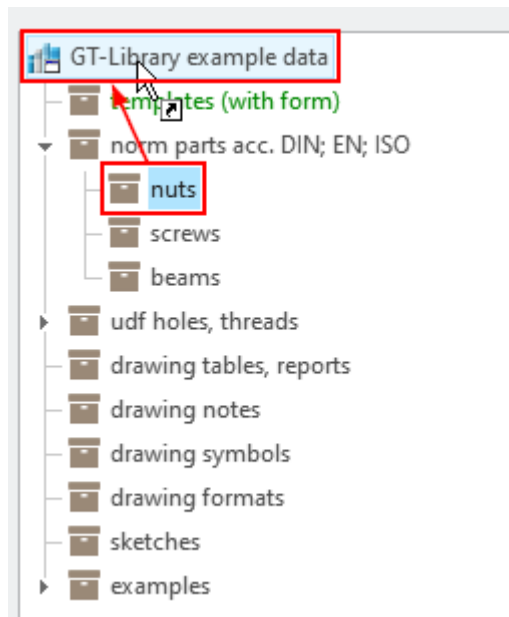
Icon	Name	Description
	Add/remove a language	<p>Adds an additional language to a library or removes an existing one.</p> <p>Note: Languages set via the configuration (gtl_lang and gtl_def_lang) cannot be deleted from a database.</p>
	Import objects into current category ²⁴³	Imports additional objects into the currently selected category of the library. Objects are not copied but represented by the corresponding library objects.
	Configure and execute batch mode ²⁴⁰	<p>Starts a batch run across any library objects.</p> <p>Preview images, parameters or family table information are automatically read or generated.</p>
	Create MNUs ²⁴³	Opens the MNU export dialog.
	Open collector ²⁴⁴	Opens the collector for batch mode of library objects.
	Clean up data ²⁴⁷	Opens the dialog for cleaning up library databases, e.g. deleting images.
	Restrict search function ²⁴⁷	Opens a list that allows you to exclude parameters and dimension from the search index.
	Help	Opens the Help.

9.3.4.3 Category tree

The category tree shows all categories of the current library.

Please note: The editor always displays the previously opened library at first.


Use Drag-and-Drop to rearrange categories. Drag categories back to the root node to display them in the first level again.



Drag a category and drop it on the root node to display it on first level again

Adding objects to the category tree

To add an object to the category tree, find the object using the search function in the *All objects* tab, then use Drag-and-Drop to place it in the required categories.

The *All objects* tab has a dedicated function for finding all objects that cannot be reached via the category tree .

Linking library categories

Categories can be linked to other categories or to other libraries. Library objects of the linked category are displayed in the other library or category. Alternatively, categories can be linked several times in the same library.

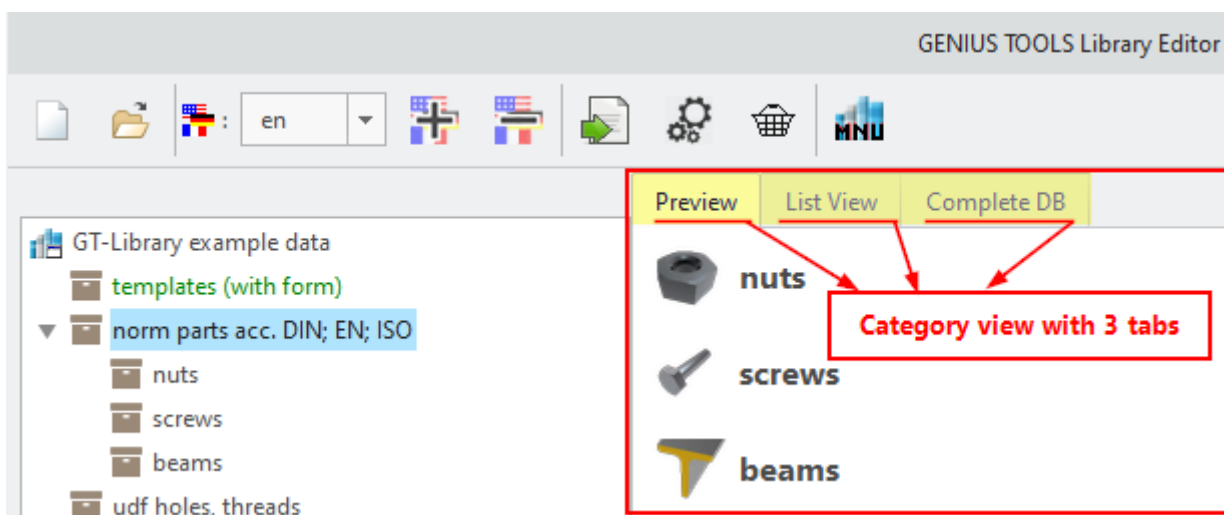
Proceed as follows to link a library category:

1. Open the library, containing the category you want to link.
2. Open the context menu in the category tree and copy the link information.
3. Open the library in which you want to create the link.
4. Create a new *Link* type library object.
5. Add the link information to *Object Source* and create the object.

Always check the object type (1) when linking.
Copy the link information into the field "Object Source" (2)

9.3.4.4 Category view

The Category view displays library objects of the current selection in the category tree. Click a library object to display more information in the Details area.




Each of the three tabs shows different information to an object:

Preview: Library objects and categories are displayed with preview image and description text.

List view: Library objects are sorted by type and internal name. In this view, displayed columns can be changed with the context menu.

All library elements are displayed regardless of their status.

All objects: The tab allows a search over all library objects in the current library.

Advanced search for objects that cannot be found in the category tree. 

Sorting objects

Use drag-and-drop to rearrange library objects in the Preview area. The order defined in the preview area is used for displaying library objects in the library browser.

Use drag-and-drop to assign library objects to categories.

If you drag objects from the tabs *Preview* or *List view* from the current category to another one, the selected objects will be moved.

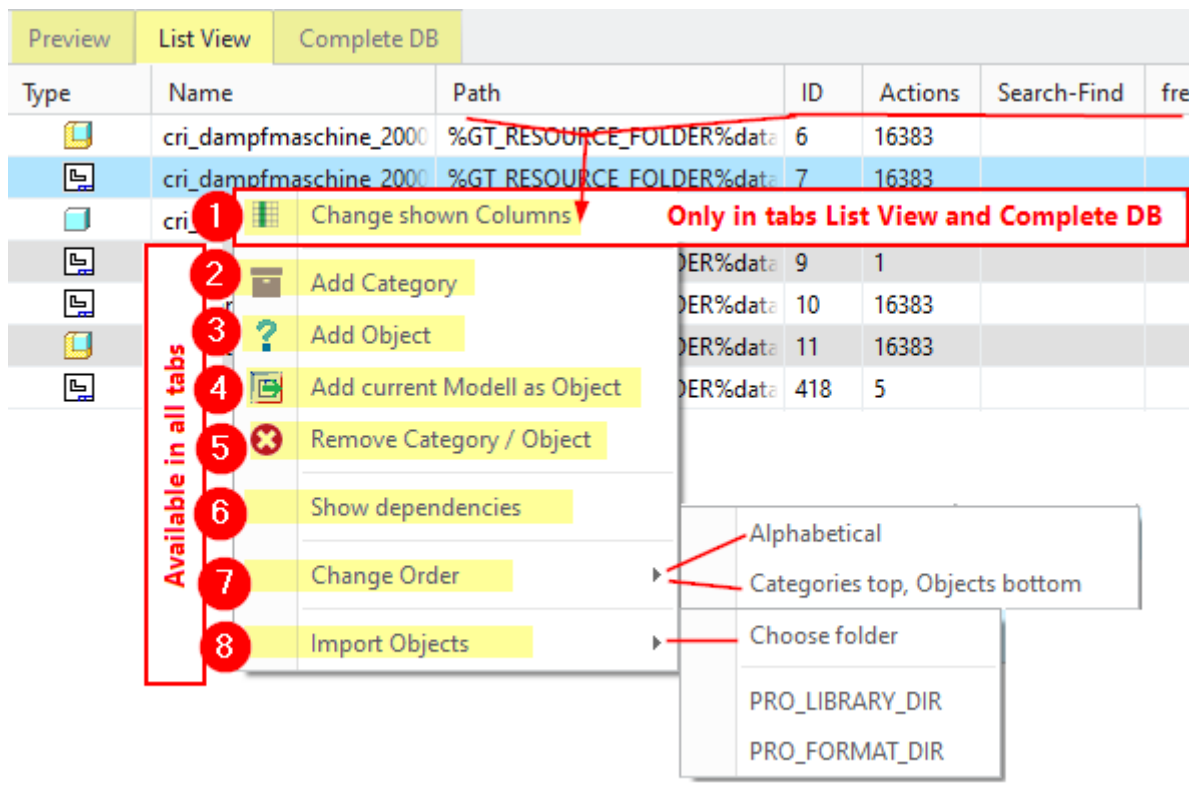
From the tab *All objects*, you can link objects to multiple categories.







You can also use drag-and-drop to put objects into the object collector dialog.

Use the tabs above the Category view to display different views of the library items.

Further actions in the context menu

Use the context menu to perform further actions.



Type	Name	Path	ID	Actions	Search-Find	fre
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	6	16383		
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	7	16383		
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	9	1		
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	10	16383		
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	11	16383		
	cri_dampfmaschine_2000	%GT_RESOURCE_FOLDER%date	418	5		

Available in all tabs

- Change shown Columns
- Add Category
- Add Object
- Add current Modell as Object
- Remove Category / Object
- Show dependencies
- Change Order
 - Alphabetical
 - Categories top, Objects bottom
 - Choose folder
- Import Objects
 - PRO_LIBRARY_DIR
 - PRO_FORMAT_DIR

Only in tabs List View and Complete DB

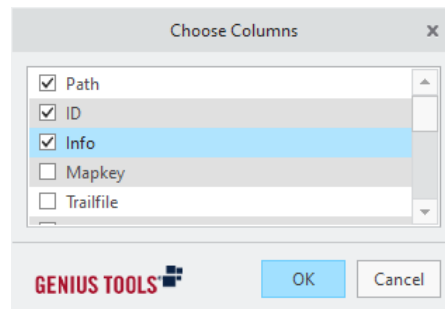
The following options are available in the Category view context menu:

1. Change displayed columns

Functions in tabs List View and Complete DB

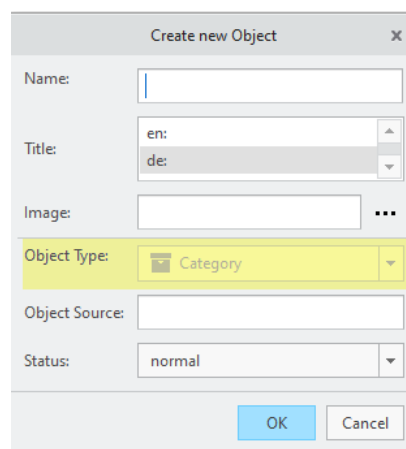
Select the columns that are to be displayed by checking the box. The columns may differ in the list view and the all-objects views.

The selected columns are taken over to the next session.



2. Add category

Adds a new (sub)category into the selected category.



3. Add object

Adds a new library object into the selected category.

4. Add current model as object

Inserts the currently opened model into the selected category as a library object.

5. Remove category/object

Removes the selected object/category from the library (with subcategories and objects). For objects with family tables, the instances will be deleted as well.

Tip: As an alternative to deleting a category or an object, the database reference to an object or a category can be removed instead. the category or object is still in the database, but is not included in the tree structure.

6. Show dependencies

Displays parent and child objects of the selected library object.

7. Change order






Sorts the displayed elements. Combine the two possible views by selecting the options several times.

8. Import objects

Imports library objects into the selected category.

9.3.4.5 Object details

The section *Object Details* displays information on the selected element. The command bar contains the following buttons:






Icon	Name	Description
	Save object details	Saves modified object details into the database.
	Open model in Creo	Opens the current library object in Creo.
	Edit selection list ²²⁵	Opens the editor to customize selection lists for the current library object. (Not available for categories.)
	Edit model list for copy purpose ²²⁷	Opens the editor to customize copy rules for the current library object. (Not available for categories.)
	Run objects through batch mode ²⁴¹	Opens directly the dialog <i>Batch Processing</i> > <i>Settings</i> without the intermediate step of selecting the object. This is useful when adding individual models to the database.

Tip: Do not forget to save the changes made to a library object.

The information displayed differs for

1. Categories and

2. Library objects

Object Details     

Name:

Database ID: 15

Title:

en: Riveted plates
de: Vernietete Platten
es:
fr:
it:





Image:  ...





Image (tooltip):   ... -

Image (detail):   ... -

Object Type:


Assembly

Object Source:

Status:

(1) normal



Info: ...

Mapkey: 






Trailfile: ...




Webcode:

none




 

Actions:

Parameters   

Name	Value
PART_NO	-

Dimensions   

Name	Value
------	-------

Detail area for categories and library objects (yellow)

1. Details for both categories and library objects

The following object details are editable:

Name: Name of the selected object. After an import, the name corresponds to the file name for library objects; for a category, the name corresponds to the folder name.

Database ID: The internal ID of the object in the library database (only for information).



Title: The localized names of a library object. Standard texts can be used via the button  (Description of the standard text selection dialog ).



Image: Chose a preview image for a category or library object or create it. (See [Working with images](#) .)

Image (tooltip): An image for display as a tooltip for a category or library object. (See [Working with images](#) .)

Object type: The current type of a library object. Always check the type of library objects. The type defines how a library object is opened by Creo and what actions can be performed on it. For example, manufacturing assemblies (MFGs) have the same file extension as normal assemblies (.asm), but must be treated differently by Creo.

Please note: The object type "Category" cannot be changed after it has been created.


Object source: Path of the library object. For categories, the folder name must be omitted.

Example for a category in a library that corresponds to a path:

- Full path of the category: D:\Parts\StandardParts
- Object source: D:\Parts
- Name: StandardParts

Use categories to open paths in the Library browser in Windows Explorer.

Warning: Check the validity of the path for categories that are not solely used for logical structuring. The path of a category is composed of the object source and the name.

Status: Select one of the pre-defined status colors. (Meaning and choice of colors can be edited, see [Defining object status](#) .)

Info: Document assigned to a library object or category with additional information. For a stored document, the Info button is displayed in the Detail window of a Library object. For categories, documents can be accessed via the context menu of the library browser.

Please note: Information documents must be available on the client computers so that they can be opened! Viewing applications are controlled by the client computer.

2. Additional details for library objects

Image (detail): An image for the display in the detail window of a library object. (See [Working with images](#))²³²




Mapkey: A mapkey for a library object. The mapkey is executed after clicking the mapkey button in the details window.

Trailfile: A trailfile for a library object. The Trailfile is executed after clicking the Trailfile button in the detail window.

Webcode: Webcode of a Form to be linked with the library object. For information on imports read chapter [XML-interface for Form values](#)²⁴⁸.

Actions: Specifies different actions that can be executed on the object in the GENIUS TOOLS Library. Details window. Different actions can be executed on the individual object types. (See [Actions on Library objects](#))²³⁶

Parameters and Dimensions: These two tables are used to add parameter-value pairs or dimension-value pairs to the library object. The parameters or dimensions given here are displayed in the GENIUS TOOLS Library Details window as additional information and can be searched for.

Use the  and  buttons to add or remove value pairs. Value pairs can also be adopted from the current model using the *Read* button. 

Please note: The parameter tables and dimensions tables can be filled automatically for generic models. Use the function in *Read family table cell information* in the batch processing.

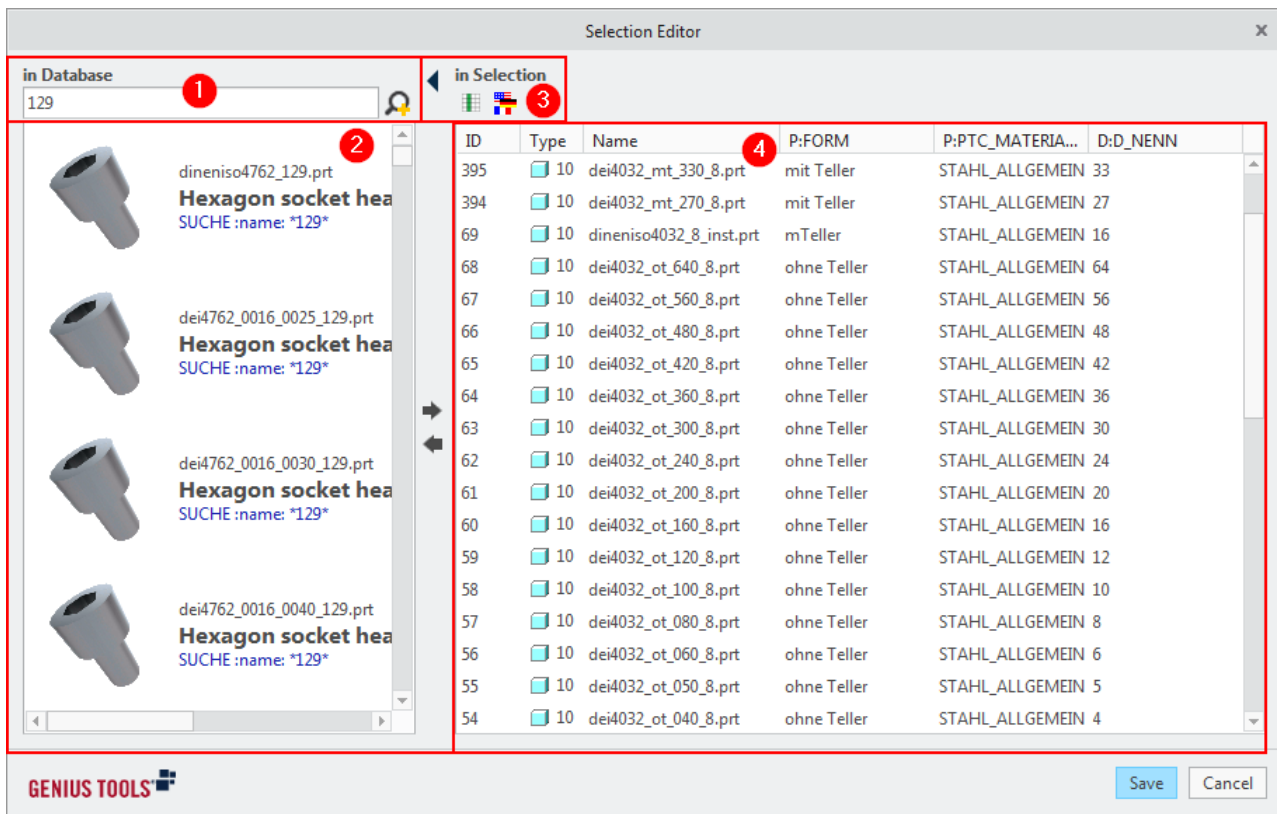
Parameters can contain multiple character-separated values. Library objects that contain multiple character-separated values in one parameter will be found in a search for either of the parameter values. The separator character is defined in the configuration option `gtl_parameter_multiple_value_separator`.

Selection list editor

Selection lists for library objects are created in a special editor. Library objects are compiled in selection lists. They are available in the detail window in the Selection tab as variants of a library object.

In Library Editor, select a library object and click the *Edit selection list* button under Object Details.

The editor is divided into the following areas.



1. Search
2. Library browser
3. Options
4. Selection list





Tip: Selection lists can be filled automatically for generic parts with family tables. Use batch mode.

Use the search (1) to search the current database. The search is identical to the search of GENIUS TOOLS Library.

On the left side of the dialog is the Library browser (2). It displays library objects of the current database. Next to the Library Browser are the options (3). The library objects of the selection list (4) are displayed on the right.

Drag and drop library objects into the selection list or use the arrow buttons to add and remove library objects. Library objects in categories are added *recursively* (all contained library objects) or *explicitly* (only library objects that are directly in a category - without subcategories).

Click into table cells to edit empty or incorrect parameters and dimensions manually.

1781		10	d172_a_015_016_st.prt	STAHL_ALLGEMEIN	15	16
1780		10	d172_a_014_028_st.prt	STAHL_ALLGEMEIN	14	28
1779		10	d172_a_014_016_st.prt	STAHL_IST_GEMEIN	14	16
1778		10	d172_a_013_016_st.prt	STAHL_ALLGEMEIN	13	16

Click into the table cells to edit their content manually



Parameters and dimensions changed in selection lists are automatically changed in the corresponding library objects.

Modified values are not transferred back to models.

Use the tab key to navigate within selection lists and localization dialogs.

Options

Use the table symbol to show or hide columns.

Use the localization button to edit translations for columns. You can use the options for default texts  or for automated localization .

If library objects in the selection list are generics, use the third button: *Merge Selection Lists*. Nested family tables are read out by batch processing and processed for the selection list. In addition, parameters can be inherited from generics and the generics can be removed from the selection. This functionality is useful for consolidating nested family tables.


You can also merge selection lists manually. To do so, use the context menu to search the selection list for

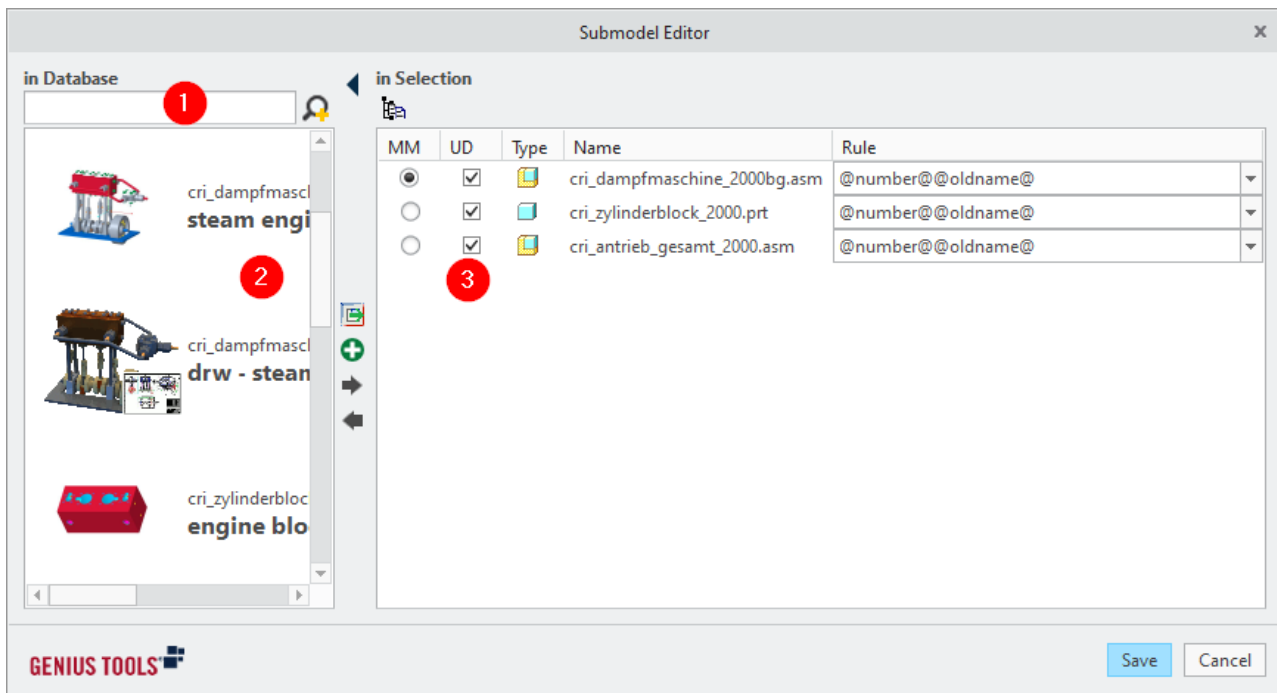
- all instances
(instance information will only be written when importing or when using batch mode)
- all selection objects
(that is, all objects that are part of the selection list for the selected object)

If the selection table meets your requirements, click *Save*.

List of models to be copied

This list contains copying instructions for models to be copied together. Models linked in this way will be copied into the work directory together with the selected object if you execute a *Copy* action.

There is a separate editor dialog for models to be copied together. To open it, select a library object in Library Editor and click *List of models to be copied together* .



The editor contains:

1. Search field
2. Library Browser
3. List of library objects in the selection: objects to be copied



Use the Search field (1) to search the current database. The search function is identical to the search function used in GENIUS TOOLS Library.

The left pane of the dialog shows the Library Browser (2), listing the library objects in the current database. The right pane shows the library objects in the current selection (3).

Drag-and-drop library objects into the current selection, or use the arrow buttons to move library objects to or from the current selection.

The following buttons are available.

Icon	Name	Description
	Copy submodel definition into selection	Use this dialog to define replacement rules that will be inherited by all instances of the selected model.
	Add dependent file that is not in library	<p>Add dependent models for the current Creo model to the copying list</p> <p>Please note: The model has to be the currently active Creo model!</p>

Icon	Name	Description
	Add file that is not in library	Add any model to the copying list
	Add to current selection / remove from selection	Use the arrow buttons to add or remove library objects

Creating copy definitions in the selection

1. In the list of selected objects, define the main model (MM) for the copy definition. The main model determines which form will be used and which model will be opened after copying.
2. Next, define whether instances that you have added to the selection should be unlinked from their generic model on copying (UD = undock).
3. Last, define the copying rule under *Rule*. Add an individual copying rule for each model, or use the dropdown list to select already configured copying rules.

Variables in copying rules

You can use all GENIUS TOOLS for Creo [variables](#)⁶⁸⁵. Additionally, there are two variables specifically for copying rules:

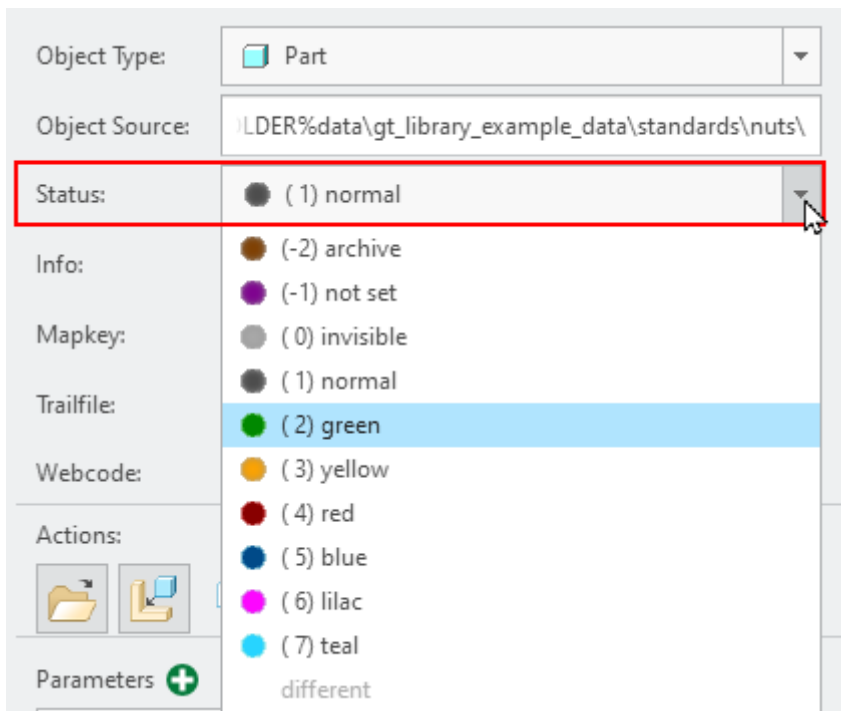
@oldname@: returns the old file name of a library object.

@copyno:<model_name>@: applies the copying rule of the model given in the variable to the current model.

Please note: All changes are saved to the object when closing the editor.

Defining object status

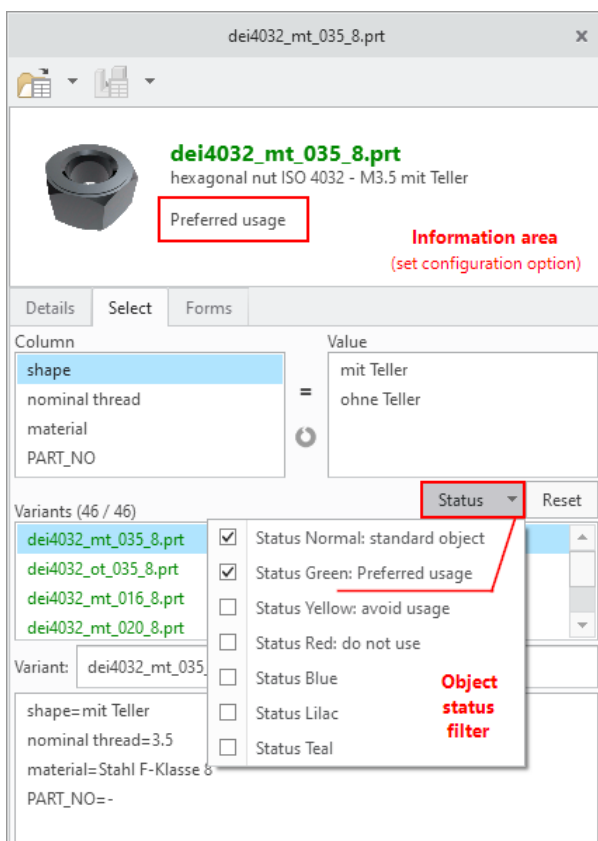
You can assign an object status for library objects and for categories, in the [Object details](#)²²⁴ section in GENIUS TOOLS Library Editor.



Selecting status in the section Object details

Objects can be filtered by this status

- with the object status filter in the library browser¹⁹³ and
- with the object status filter in the selection tab of the details window.²⁰³



Available status

The status values *Normal*, *Green*, *Yellow*, *Red*, *Blue*, *Lilac* and *Teal* can be assigned freely. In the start settings the following descriptions are assigned to these colors.

Status	Description (EN)	Description (DE)
Normal	Standard object	Standardobjekt
Green	Preferred usage	Vorzugsweise verwenden
Yellow	Avoid usage	Verwendung vermeiden
Red	Do not use	Nicht mehr verwenden

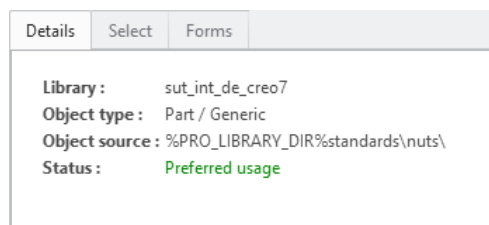
Modify status description

You can change the text for the description of these status using the configuration options `gtl_<StatusColor>_text`, e.g. `gtl_filter_yellow_text = obsolete`. These changes can only be displayed in one language.

Configure display

The status of an object is visible in the [detail window](#)^[20]

- in the upper information area (screenshot above): this display is activated with the configuration option `gtl_detail_window_show_status_in_head_area`.



- in the details tab: always

The respective colors can be changed with the configuration options `gtl_*_color`.

Tip: Document the meaning of the freely definable statuses in the work instructions. Assign descriptions using the configuration options `gtl_<StatusName>_text` of GENIUS TOOLS Library.

The following three status values are administrative and cannot be searched by the object filters:

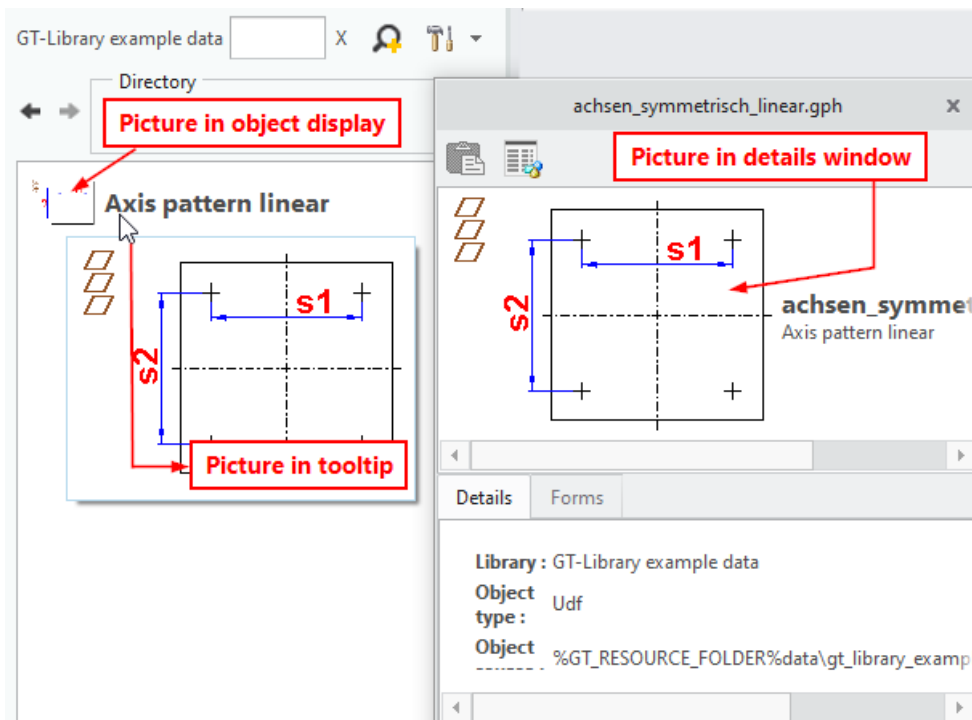
- Archive: archived library objects are not displayed in the Library browser.
- Not set: library objects without status are displayed in the Library browser and can be searched for.
- Invisible: invisible library objects are not displayed in the Library browser and cannot be searched for. Invisible library objects and categories are not displayed in the GENIUS

TOOLS Library Editor preview either, but can be found via the *Lists* view and *All objects* tabs.

Working with images

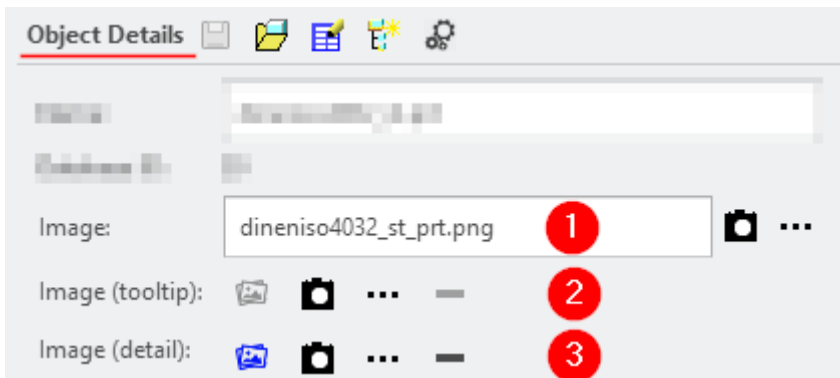
Images can be embedded in GENIUS TOOLS Library

1. in the object display area ("Preview image")
2. in tooltips
3. in the details window²⁰¹






Three possibilities of using images

You can add images to the object details in the GENIUS TOOLS Library Editor or create them directly via snapshot.



Use the following symbols to work with.



Symbol	Name	Beschreibung
	Open existing image	The symbol turns blue when an image has been uploaded.
	Create image	Creates a screenshot of the currently opened model. Replaces an already existing image.
...	Add / replace image	The selected image is automatically copied / converted to the respective folders (<i>img</i> , <i>img_detail</i> , <i>img_tooltip</i>) under the same name and type.
	Delete current image	Removes the assignment of the uploaded image to the object.

1. Images in object display (Preview image)

Name of image file: Displays the name of the image from the directory

GT_RESOURCE_FOLDER>/library/<BIBLIOTHEK>/img_w40. Use a language variable, e. g. *gt_lang*, in the path specification for displaying images language-dependently, e. g. *start_bolt_\${gt_lang\$.png}* (See section below.)

Create image: Preview images are created by default in the size of 40*30 pixels.

Please note: Preview images can be created automatically in **batch mode**  .

Add / replace image: Select an image from a directory.

Size of image: The display size can be changed, see [Changing the sizes of images](#)²³⁵.

2. Images in tooltip

Open existing image: Opens the image from the image directory *img_tooltip*. The image must have the same name as the object, using the pattern

`<OBJECT_NAME>_<OBJECT_EXTENSION>.png` (e.g., *dei4032_mt_020_8.prt* >

dei4032_mt_020_8.prt.png). The default directory is

`<GT_RESOURCE_FOLDER>/library/<BIBLIOTHEK>/img_tooltip` and can be changed with the configuration option `gtl_tooltip_image_folder`.

Create image: Tooltip images are created by default in the size of 200*200 pixels. You can change this size with the configuration option `gtl_img_create_tooltip_size`.

Add / replace image: The selected image will be automatically copied / converted with the same file name and type to the *img_tooltip* folder.

Size of image: Tooltip images are displayed in original size.

3. Images in details window (Detail image)

The *details window*²⁰¹ can either display the image from the object display (Preview) or an image that is especially for this window. If there is an extra image for the details window, you have the possibility to hide the display of the object title with the configuration option `gtl_detail_window_detail_image_show_title`.

Open existing image: Opens the image from the image directory *img_detail*. The image must have the same name as the object, using the pattern

`<OBJECT_NAME>_<OBJECT_EXTENSION>.png` (e.g., *dei4032_mt_020_8.prt* >

dei4032_mt_020_8.prt.png). The default directory

`<GT_RESOURCE_FOLDER>/library/<BIBLIOTHEK>/img_detail` can be changed with the configuration option `gtl_detail_image_folder`.

Create image: Details images are created by default in the size of 200*200 pixels. You can change this size with the configuration option `gtl_cimg_reate_detail_size`.

Add / replace image: The selected image will be automatically copied / converted with the same file name and type to the *img_detail* folder.

Size of image: The default size for displaying detail images is 100*75 pixels. This size can be modified in two ways:

1. The size of all detail images is similar: Use the configuration options `gtl_detail_window_detail_image_height` and `gtl_detail_window_detail_image_width` to define the height and width of all detail images.

2. The size of the individual image file should be used: Set both the configuration options `gtl_detail_window_detail_image_height` and `gtl_detail_window_detail_image_width` to 0.

Images are retrieved in the following order:

- detail image of the instance
- detail image of the generic
- image of the instance
- image of the generic

Please note: Alterations of the detail image (after the initial preview) will be displayed after restarting Creo Parametric.

Changing the sizes of images

Images are by default available in small size (for object display) and in large size (for detail window). You may set the object display from small to large images and you may also modify the standard size of all images.

Images are stored in format 4:3. The pixel values in the configuration options refer to width.

Type of image	Standard for display size	Configuration options
Object display images	small (40*30 Pixel)	<code>gtl_img_switch_size:</code> defines the sizes of images in the objekt display. (Default width: 40 pixels) <code>gtl_img_size:</code> Changes size of object display images from small=40 to large=100
Details window images	large (100*75 Pixel)	<code>gtl_img_detail_size:</code> defines the size of detail images (Default width: 100 pixels) For larger images: Create separate <i>Detail</i> directory

Type of image	Standard for display size	Configuration options
Tooltip images	Original size	Original size is used up to 800*800 pixels.

Warning: The configuration options modify the size of all images of an image type and change the name of the storage folder. Therefore, take care not to change sizes of several image types simultaneously, i. e. do not make changes to both configuration options `gtl_img_detail_size` and `gtl_img_switch_size`.




Display images language dependent

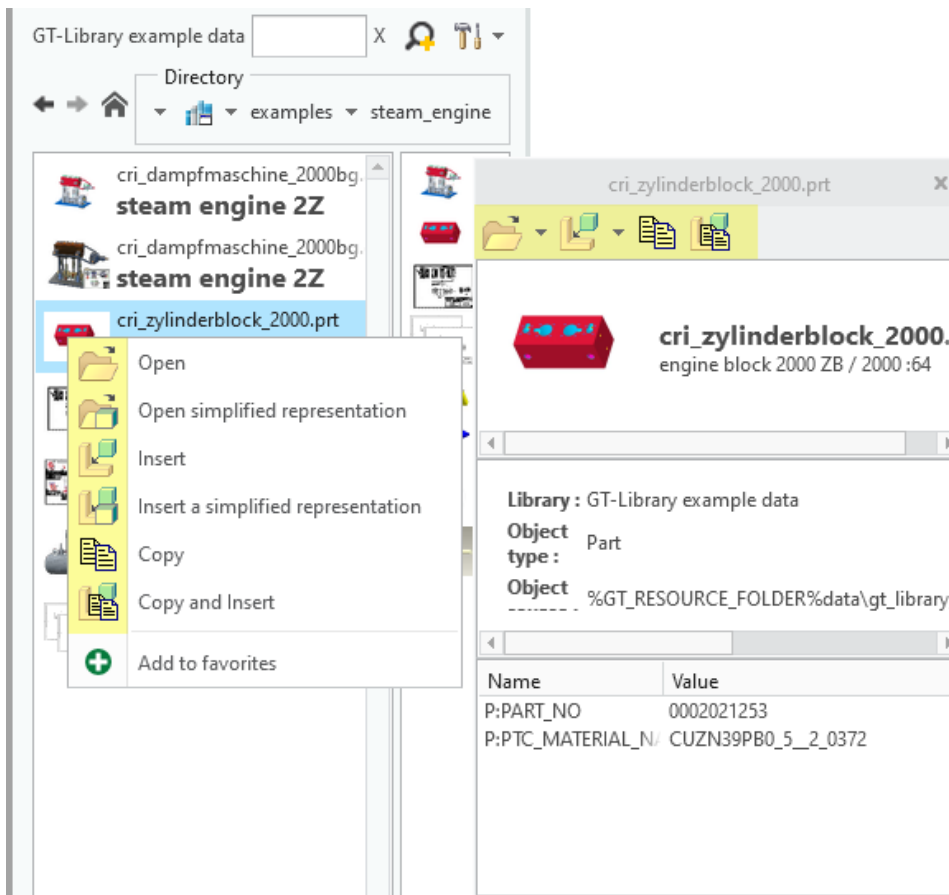
You can display images depending on a language. This can be useful when images contain textual information.

- For detail and tooltip images: Create a folder per language and link to that folder.
- For preview images: Add to the image file name a language abbreviation, which is output by the variable `gt_lang`, and enter the variable in the notation `gt_lang` in the name of the file (1).

9.3.4.6 Actions on library objects

Library object actions are specified in the GENIUS TOOLS Library Editor. They are available in a library objects Details window or in the context menu.

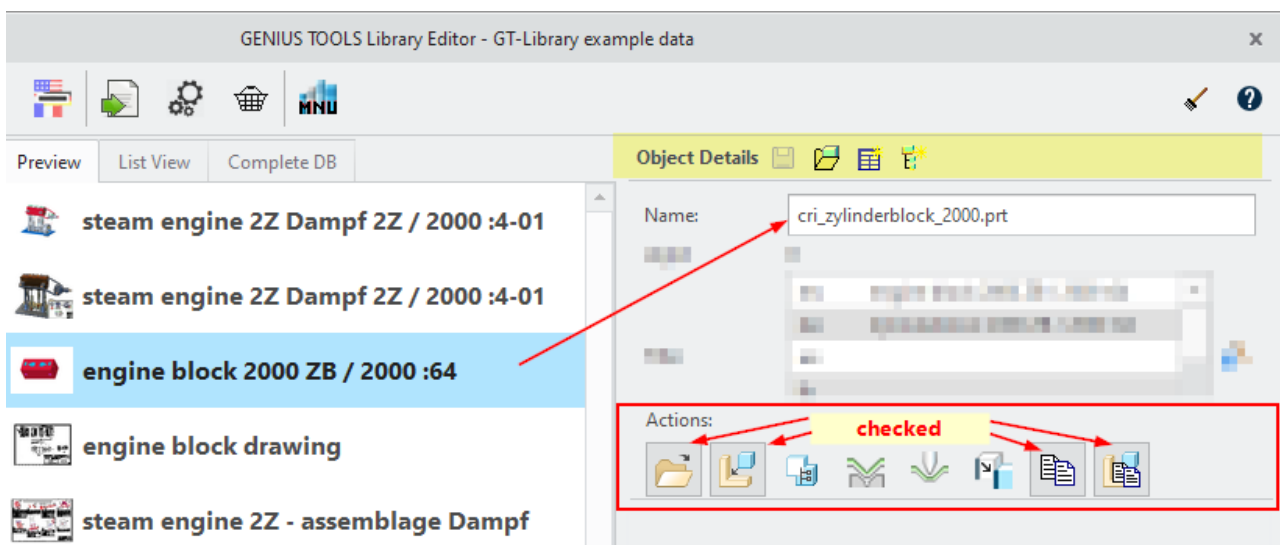
Example: The part *engine block 2000* can execute the actions Open , Insert , Copy  and Insert Copy.



View in the context menu (left) and details window (right)

These actions are activated in the Library Editor by clicking on the corresponding icons:

- either for multiple objects simultaneously by using the **object collector** ²⁴⁴ or
- for individual objects in the object details > Actions, see screenshot below.



Please note: Different types of library objects support different actions.

Batch file for models that cannot be retrieved

You can use the configuration option `gtl_retrieve_run_batch` to specify the path to a batch file. This batch file will be called if a model cannot be retrieved by GENIUS TOOLS Library. After the batch file has run, Library again tries to retrieve the model data. In the batch file, the variables `@name@` for the model name and `@path@` for the model path can be used.

Open



Allows to open models or to open models as simplified representations. Assemblies can also be opened with simplified graphics.

Open with defined Representation open the model with the representation that is defined by the configuration option `gtl_action_open_defined_simp_rep`.

You can use the following terms as option values:

Explicit specified : e.g. LAYOUT or FLYER

DEFAULT ENVELOPE REP

AUTOMATIC REP

MASTER REP

DEFAULT

Notes, symbols and selections cannot be opened.

Insert



Allows to insert library objects into assemblies, parts, sketches, drawings and reports.

Additionally activates the action **Insert as simplified representation**.

Library object type	Insertable in
PRT	ASM
ASM	ASM
GPH	ASM, PRT
SEC	SEC
FRM	DRW

Library object type	Insertable in
TBL	DRW, FRM
SYM	DRW, FRM
TXT	DRW

Copy models



Copies a library object into the current working directory with a new name using GENIUS TOOLS Name Generator. The copy is opened in Creo.

The action is available for parts and assemblies. This action copies models from the lists of models for copy purpose as well.

If a library object contains a Form, the action *Copy the selected file* will additionally be activated for the library object. This copies the model and values in the form are applied to the model. If you have defined copying rules for the model, these will be applied.

The workflow in combination with GENIUS TOOLS Name Generator (`gtl_gtng_*`) and GENIUS TOOLS Forms (`gtl_gtf_*`) can be defined using configuration options. For more information, please refer to [configuration](#)⁶¹⁵.

Copy drawings



Copies a drawing with a new name (using GENIUS TOOLS Name Generator) and without copying the model in the current working directory.

Insert file with note reference



Inserts a TXT file with hint arrow to a drawing (DRW) or a report (Rep).

Merge/Inherit



Inserts a library object (parts) as an inheritance into another part.

Die



Inserts a library object (parts) as a die for a sheet metal part.

Punch



Inserts a library object (parts) as a punch for a sheet metal part.

Copy geometry



Inserts a library object as a copy geometry for a sheet metal part. If the library object has multiple copy geometries defined, the user is offered a selection.

Execute mapkey



Executes a mapkey deposited in the library object.


Trail file



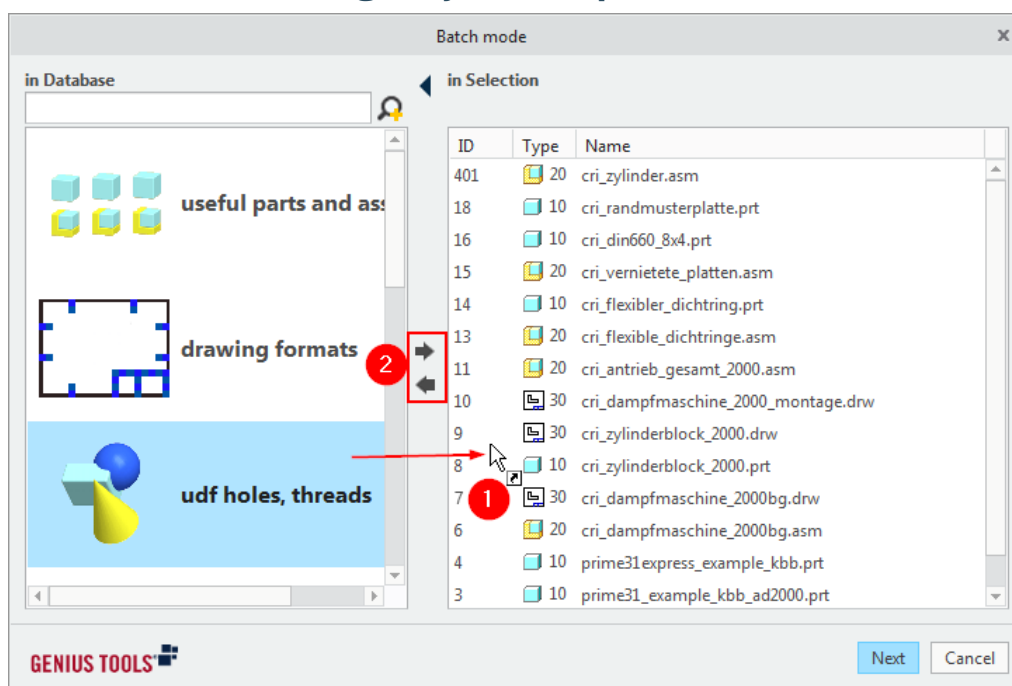
Executes a trail file deposited in the library object.

9.3.4.7 Wizard: Batch mode

There is a batch mode wizard guiding you through the configuration of the batch mode. The wizard consists of two dialogs: Object composer and Batch mode configuration.

In the Library Editor, click on the *Configure and execute batch revision* button  to open the configuration dialog of a batch revision.

1. Batch mode dialog: Object composer



First dialog box of Batch mode: Object selection

In the first dialog, you can specify the library objects for batch revision by drag and drop (1) or by using the two arrow buttons (2)

The dialog is divided into two parts. The library objects already selected (in Selection) are displayed on the left side. The browser displaying the library objects of the current database is located on the right side.

You can either drag and drop library objects into the selection list, or use the arrow buttons to add library objects to the selection list or to remove them.

When adding library categories to the selection list, select whether the library objects included should be added recursively (all library objects underneath the category and underneath included categories) or only from the selected folder (all objects underneath the category).

Use the Search bar to search in the database. The search function is the same as in the GENIUS TOOLS Library Search and filter area.

Click *Next* to open the next wizard dialog.

2. Batch mode dialog: Configuration


The second dialog configures the batch mode actions. Select the actions to be executed on the library objects.


The following options are available:

The options are divided into options for loaded models and options for models in the database. Options for loaded models require loading the entire model in Creo and are more time consuming. Options for models in database are only executed on the database.

You can also save the configuration for a batch mode job, load it later and run it again or make adaptations to the configuration.

Saving and loading the configuration

Save configuration:  Writes the current settings for batch mode into a configuration file of the type: GTLE_BATCH_CFG. The default storage location for Library is preselected.

Load configuration:  Loads a configuration file of the type GTLE_BATCH_CFG and restores the settings for batch mode stored in the selected file.

Options for loaded models

Create instances as objects: Specifies whether instances of a generic part are also included in the library database as independent objects.

- Add new instance objects to batch list: Specifies whether newly created library objects are included in the current batch list.
- Add existing instance objects to batch list: Specifies whether library objects being already known instances of generic parts are included in the batch list.

- Do not revise existing generics: Excludes generics from further batch mode.

Read family table cell information: Specifies whether family tables are read from parts and added to the database.

Read parameters: Specifies the parameters to be read from the library objects. They will be displayed in the GENIUS TOOLS Library Details window.

Map parameters to existing language: Specifies one or more parameters for each available language to be used as the library object title in GENIUS TOOLS Library.

Specify individual parameters by name. Multiple parameters require a percent sign before and after the parameter name, for example:

- *parameter1*
- *%parameter1% - %parameter2%*
- *%parameter1% - Generic part*

Create preview images: Creates a preview image (image for the object display area) for each library object.

By changing the configuration option `gtl_editor_use_black_on_white_for_screenshots` you can decide whether the preview image will be created with the system colors "Black on White" (1) – i. e. the images will have a white background – or with the current color settings (0).

Only if no preview image exists: Excludes library objects that already have a preview image from image creation.


Index – Form: Creates an index for the form (bounding box side lengths) of library objects for [3D search](#)¹⁹³.

Index – Voxel: Creates an index of voxel information (spatial record) of library objects for [3D search](#)¹⁹³.

Options for models in the database

Assign info documents from the info folder: Looks up info documents of the model name in the folder specified in `gtl_info_folder` and links them to library objects of the same name in the database. Supported file formats: PDF, HTML, DOC, DOCX, XLS, XLSX and XLSM.

Please note: Paths are always calculated on the client. Use the configuration option `gtl_info_folder` to specify a path for multiple databases.

Change path: With this option enabled, path segments (left input field) are replaced by an input (right input field). Use this option when Creo files being referenced by library objects have been moved and cannot be found via the search paths. For changing a path to a directory we recommend using the function Replace path substrings by in the [Cleaning up data](#)²⁴⁷ dialog (broom button ).

Change status: Changes the status of all library objects to the status provided here.

Click *Save configuration* if you want to execute a batch list several times. The configuration will be saved for later executions.

Following batch mode configuration, click *OK* to start it.

Please note: A batch mode may take a long time depending on the selected options.

9.3.4.8 Wizard: Import

The Object Import dialog exists for importing library objects.

Warning: Always check the object type of library objects after an import.

Click the button *Import Objects in current category* to open the dialog. Alternatively, use the context menu in the Preview area.

The following options are available.

Folder: Defines the file system folder that is to be imported.

Import selected folder as category: Defines whether the folder is imported as a separate library category below the current category.

Import folders only: Imports a folder structure without including the contained data in the library database.

Import folders and files: Imports both: folders and files.

Configuration for MNU import: Defines the language assignment in MNU files.

Configuration for STTOOLS XML import: Defines the language assignment in the XML files of the Library Viewer of the TOOLBOX.

Check file existence: Checks if all referenced files are contained in the folder structure.

Warning: Checking all referenced files takes a long time.

9.3.4.9 Wizard: MNU export

There is a dialog for exporting MNU files that guides you through the configuration of the export. In Library Editor click on the *Create MNUs* button to open the MNU export dialog.

The following options are available in the dialog:

PRO_LIBRARY_DIR: This is the directory in which the file *index.mnu* is created, which contains the content of the first level of GENIUS TOOLS Library.

Warning: Only use the environment variable %PRO_LIBRARY_DIR%, if this directory corresponds to the entry in the configuration file (*config.pro*). If it is not the same, use an absolute path.

Category-based mode: An MNU file is created for each category in a library. The files are stored in the path of the category (object source).

File-based mode: A folder and the associated MNU file are created for each category of a library. The structure corresponds to the categories of the library. Use this mode, if you have entered an absolute path in the field PRO_LIBRARY_DIR.


Categories: Select the categories to be exported here.

First language/Second language: Define here the language assignment for the MNU files.

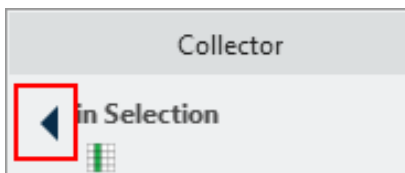
Confirm the dialog with OK to start the MNU export.

9.3.4.10 Wizard: Object collector

For revising library objects, the Object collector dialog exists. The object collector is used to edit properties such as status, actions or preview images of several library objects simultaneously.

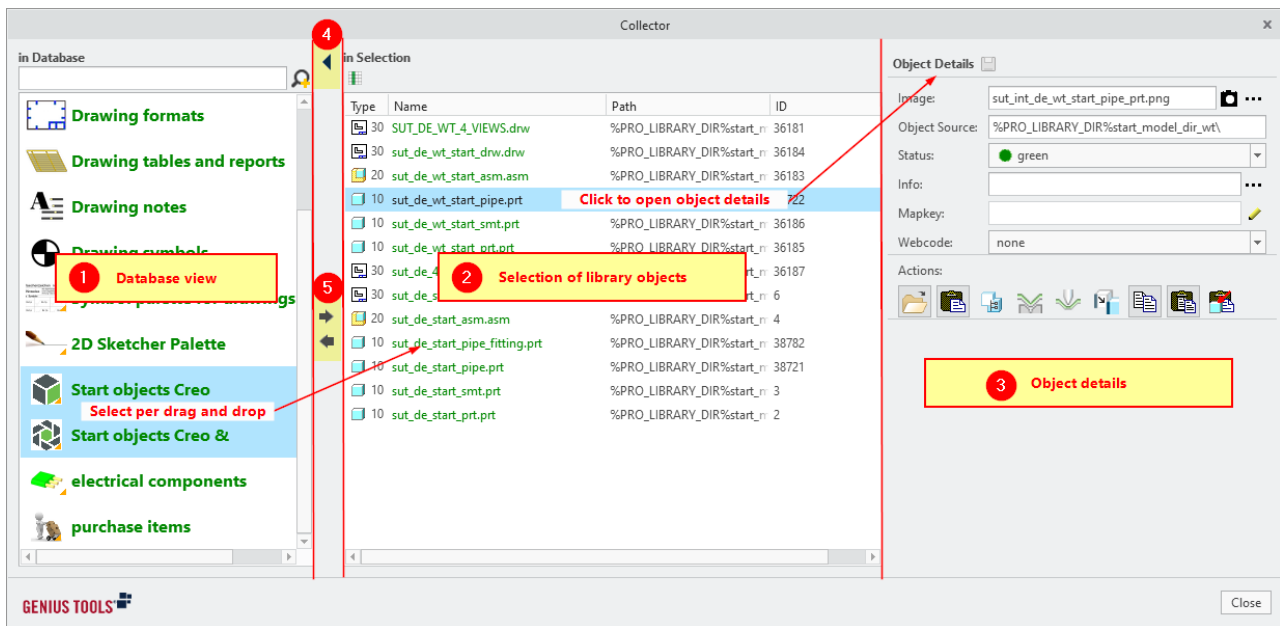
Click the *Open object collector* button  in GENIUS TOOLS Library Editor.

Click on the arrow button to display the database view.



In the database view (1), library objects and categories are displayed as in the library browser.

Add library objects to be revised to the selection list (1) via drag and drop or use the arrow buttons (5). Use the Shift or Control key to select multiple categories or objects.



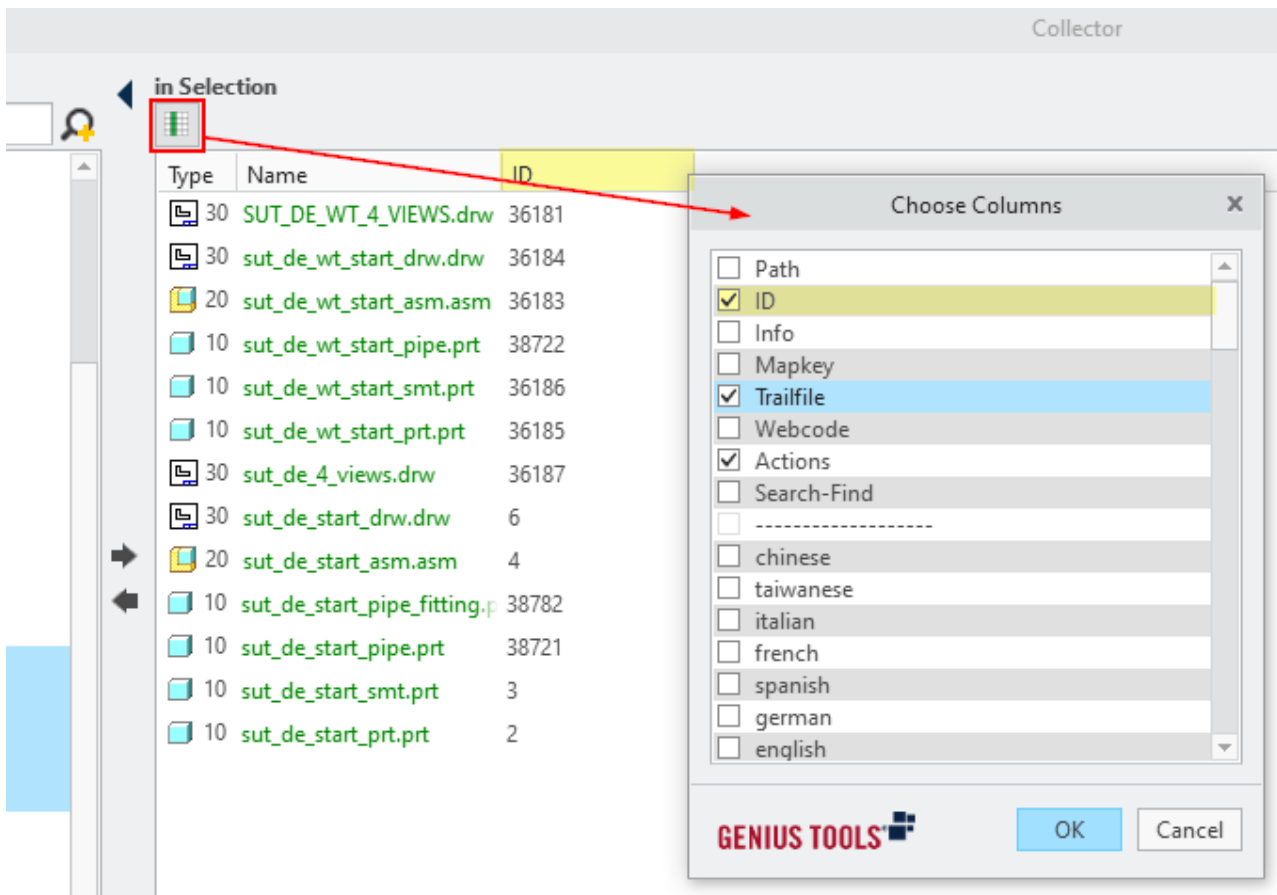
Collector dialog window

The object details (3) will be displayed if at least one library object is selected in the selection.

Library objects in selection and Object details

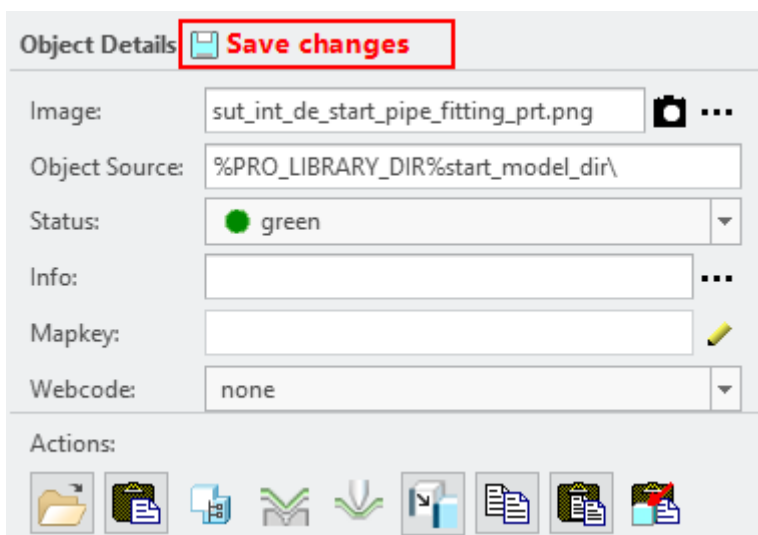
This section displays the library objects that have already been selected.

Use the button *Show Columns* to show additional information and select them by checking the boxes.



Display additional information in the collector


Select the library objects to be edited in the selection. Use the keys Shift and Control for multiple selections. To the right, modify object details to be revised.

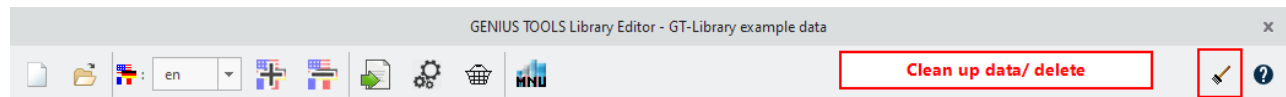


Object details

Click the button *Save Object* to save changes in the library database. Reload the library database in Library to work with the updated data basis.


9.3.4.11 Cleaning up data

The broom icon  opens the dialog box for cleaning the database. Deleted data cannot be recovered. The following actions are available.

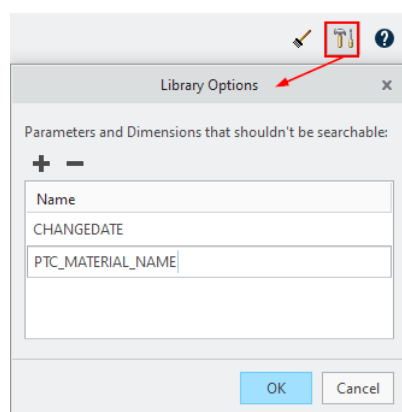


1. Set all instance images to the generic image (does not delete files)
2. Delete unused images from data directories as well as from all image directories (img, img_w40, img_w100)
3. Delete parameter / dimension: **from all models** in the database. This action can be used to delete e. g. incorrect parameters or dimensions.
4. Replace path substrings by
The default settings in GENIUS TOOLS: the path set by the variable *PRO_LIBRARY_DIR* is replaced by *%PRO_LIBRARY_DIR%*.
5. Compress database

9.3.4.12 Customizing the search function

You can search all library objects by entering a term in the **search field** . This can lead to unintentionally many results, e. g. when a search term is found both in the name of the object and in the material parameter.

In the command bar, you can use the tool icon to open a dialog and enter individual parameters and dimensions that should be excluded from the search.



Please note: Entries in this dialog will change the search strings in the entire database for GENIUS TOOLS Library and Library Importer. This may take some time.

9.3.5 XML interface for Form values

Values determined in advance can be read in for *GENIUS TOOLS Forms*⁸⁵ via an XML interface. Whether and for which forms this interface is used must be determined on a company-specific basis.

For users, the function *Form -Load values (.xml)* from an XML file is available in the tool menu of the library browser.

An XML file for reading Form values must be in a specific XML format and UTF-8 encoding. This section describes the required XML format.

Root element

The element `root` has the following attributes:

- `modelname` (optional): name of the model that contains the target Form, without extension
- `webcode`: webcode of the target Form
- `new_name`: name for the copy of the model that contains the target Form

Form values

Each value to be set in the target Form is defined by an element `element` with the following attributes:

- `name`: name of the Form element
- `id` (optional): ID of the Form element. If no ID is given, the name is used for determining the Form element.
- `value`: value to be set in the Form

`element` is an empty element.

To access the names and IDs of the target Form elements, you can export the Form as an XML file via the tools menu of *GENIUS TOOLS Forms*.

Models to be copied

The list of models to be copied with the model that contains the target Form is bracketed by an element `submodels`.

Each model to be copied with the target model is defined by an element `submodel` with the following properties:

- attribute `new_name`: name for the copy of the model, without extension

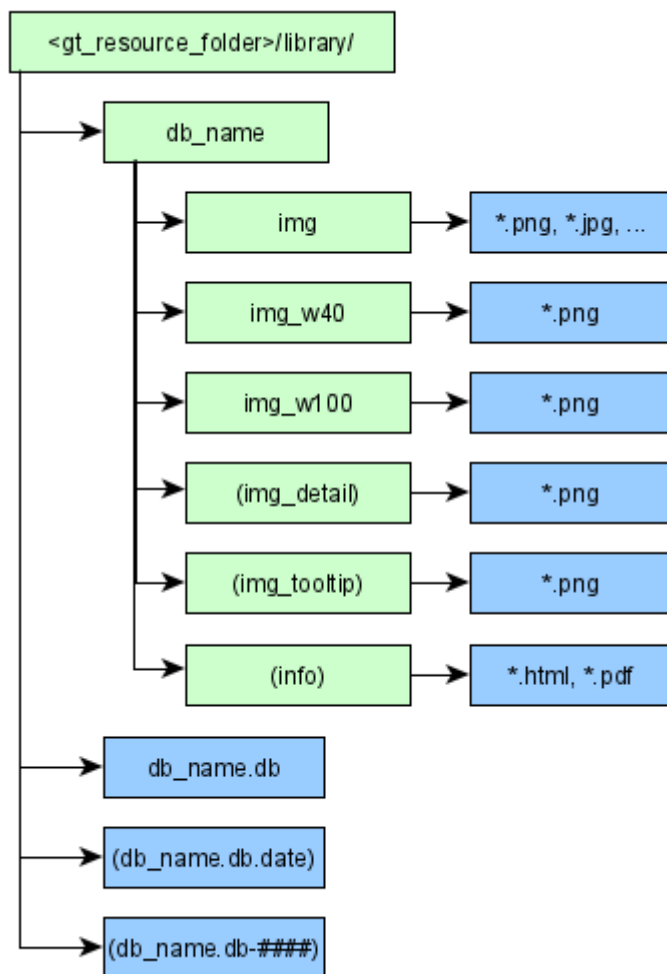
- attribute `copy` (optional): This attribute can contain the value `y` for yes, be empty, or be absent in order for the model to be copied. The model is not copied if you set `copy="n"`.
- element content: original name of the model to be copied, with extension

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<root modelname="SUT_INT_DE_START_CUBE" webcode="sut_int_de_start_cube"
  date="30.10.2018 9:28" new_name="model_copy">
  <element name="DESCRIPTION_1_DE" id="para|DESCRIPTION_1_DE"
    value="Startwürfel" />
  <element name="DESCRIPTION_1_EN" id="para|DESCRIPTION_1_EN"
    value="base cube" />
  <element name="length" id="dim|0" value="20" />
  <element name="width" id="dim|1" value="20" />
  <element name="thickness" id="dim|2" value="20" />
  <submodels>
    <submodel new_name="model_copy_drawing"
      copy="y">sut_int_de_start_cube.drw</submodel>
  </submodels>
</root>
```

9.3.6 Data structure

The data that is needed by Library is saved inside a folder located in a configurable location, with the default at `<gt_resource_folder>/library`. Every database that is usable inside Library has its own storage structure inside this folder.



GT Library - Structure tree

Files are represented in blue and folders in green in the figure. Images that are linked by the database are saved inside the image folders. Folders and files that are shown in brackets can be there, but do not have to be.

Back up copies

Images in the image folder and back up copies of the database (`db_name.db.####`) can be deleted. This has no impact on Library.

By default a back up copy is created once a day. This can be deactivated by setting the configuration option `gtl_editor_create_db_security_copy_once_a_day`.

Synchronization

The file `db_name.db.date` is important for synchronization. It need to be re-written at every change of the database. This is done automatically by the Editor or GENIUS TOOLS Data Importer. If there is no `db_name.db.date` file the database `db_name.db` will be synchronized on every folder change. In this case, you should deactivate the synchronization (see also [Configuration Options](#)⁵⁹³).

Images / Convert images

Different sizes of each image are saved pre-formatted in the folders *img_w40* and *img_w100*. If these do not exist when Library Editor is started, they will be created and the pictures from the respective other directory will be converted into them. This behavior has been implemented for legacy data, but it can also be used to clean up inconsistencies. Its important to remember that images from the *img_w40* folder cannot be resized to a bigger size at the conversion.

Detail images can be saved in the *img_detail* folder. You can find more information under [Library Editor - Object pictures](#)²³².

Info files

Info files can be saved in the info folder and linked automatically with objects by a batch process. You can find more informations about batch modes under [Library Editor - Object details](#)²²² and [Library Editor - Wizard: Batch mode](#)²⁴⁰.

9.3.7 Setting configuration options

The following options allow you to define further settings. These options can be found in the Library module in [GENIUS TOOLS Configuration Utility](#).⁵⁵⁸

gtl_gtp_start_gtp_after_model_creation

Determines whether the [GENIUS TOOLS Parameters](#)³³⁴ dialog box opens automatically after a new file is created. Specify the desired file types separated by commas. Possible file types: prt, asm, drw. Default: None.

The parameter dialog does not open if you create a new file and insert it into an assembly in one step.

9.3.8 Use cases

In this section, you will find short instructions on tasks relating to Library Editor.

9.3.8.1 Creating a new library

Proceed as follows to create a new library:

1. Open GENIUS TOOLS Library Editor.
2. In the command bar, click *New*.
3. Enter a descriptive name for the new library. Be sure not to use special characters. The database and the required sub-directories are created in the resource directory.

4. In the next step, check the languages set in Library Editor. A standard library has German and English language set.
5. Click *Add a language* to add new library languages using language codes. Use the *Remove a language* button to remove the language codes not needed.

Please note: The language code en and the language code specified via the `gtl_lang` configuration option cannot be removed.

Your new library is now ready for use. Proceed with adding categories and library objects to your database. They can either be created manually or be imported.

9.3.8.2 Creating a new category

1. Open the GENIUS TOOLS Library Editor.
2. Click *Open database* and select a database to be edited.
3. In the category tree, navigate to the position where you want to create a new category.
4. Open the context menu in the Preview area (right mouse button) and click *Add category*.
5. Enter the Name for the new category. Enter the *folder name* as the Name if the category corresponds to a folder.
6. Enter a Title for the category. Fill in the language dependent input fields.
7. If the category is to contain an image, choose the appropriate preview image by directly entering its location into the input field or by selecting a preview image via the (...) button.

Tip: When you create the category, the preview image is copied to the database image folder in the resources directory with a new name. When you open the category in the editor, another image name is displayed.

8. Check if the object type is set to category.
9. Enter the object source. Enter the *path*, without the last folder name, as the object source if the category corresponds to a folder.
10. Specify a status for the category.
11. Click *OK* to complete the category creation.

Your new library category is now ready for use. Proceed by adding library objects to your database. They can be created manually or be imported.

9.3.8.3 Creating a new library object

1. Open the GENIUS TOOLS Library Editor.

2. Click *Open database* and select a database to be edited.
3. Navigate to the location in your database structure where you want to create a new library object.
4. Open the context menu in the Preview area (right mouse button) and click *Add object*.
5. In the next dialog, enter the following information:
6. Enter the name of the object to be created or select a Creo object via the *three dotted button*.
7. Enter a Title for the object. Fill in all language dependent input fields.
8. If the library object should contain an image, choose the appropriate preview image by directly entering its location into the input field or by selecting a preview image using the (...) button.
Transferring title information from parameters to library objects and creating preview images can be automated via batch mode.
9. Check the object type and adjust if needed.

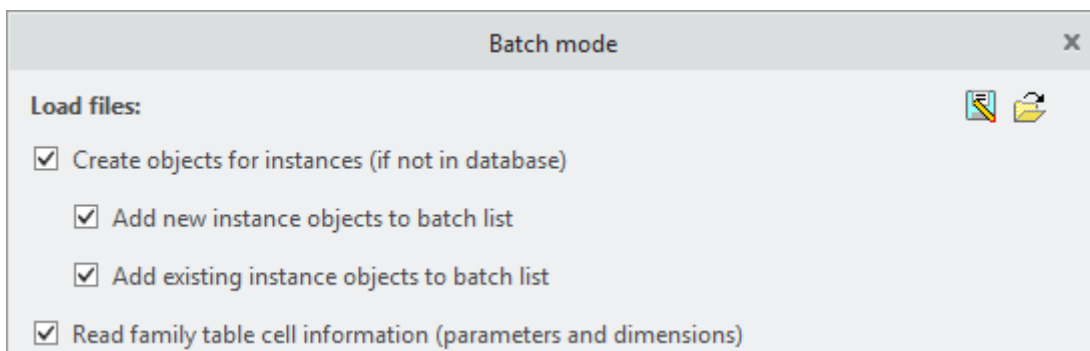
Please note: Pay special attention to the object type with manufacturing assemblies.

10. Specify a status for the library object.
Use the preset status *Invisible* as long as you have not completed the revision of your library objects. Invisible library object are not displayed in the editor preview, switch to list view to see them.
Setting statuses can be automated via batch mode.
11. Confirm the dialog with *OK*. Your new library object is now created.
12. Check the object details in the Library Editor and adjust if necessary.
13. Add an information document to the new object: Click on the *three dotted button* after **Info** and specify a document that can later be opened from the GENIUS TOOLS Library Details window.
14. Specify a mapkey or trail file with a mapkey for your library object: Click the *Pen icon* after **Mapkey** to enter a mapkey directly the next dialog (e.g. via Copy and Paste) or click on the *three dotted button* after **Trail file** to select a trail file from your file system.
15. Specify the executable actions on the library object: Click the actions that should be available for the library object.
16. Automatable revisions for your library object: Modifying title information, preview images and status can be automated via batch mode. Instances and family tables can be captured automatically as well.

9.3.8.4 Importing family tables

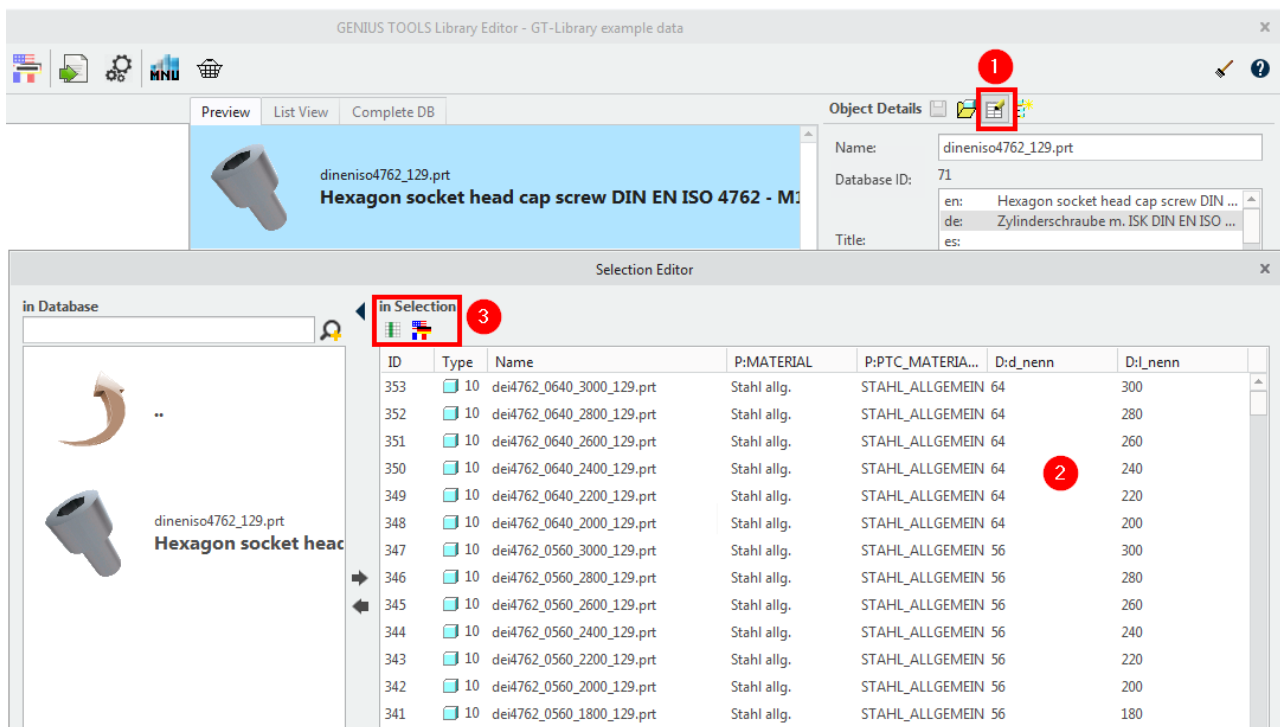
Proceed as follows to import family tables of a generic part:

1. Create a new object in your library structure and select the generic part. The generic part should have previously been saved to the directory using Creo Parametric (*.idx updated). Verifying the family table and setting up a search path guarantee error-free use of the variants in Creo Parametric.
If the instances should have the same preview image as the generic part, specify the preview image prior to the batch mode. Then the image will be inherited by the instances.
2. Start a batch mode. Add the library object you have just created to the selection. Then, click *Next*.
3. Activate the following options for the batch mode:
 - a. Create objects from instances (if not in database)
 - b. Add new instance objects to batch list
 - c. Add existing instance objects to batch list
 - d. Read family table cell information (parameters and dimensions)
 - e. If each instance should have an individual preview image, also activate the *Create preview images* option.



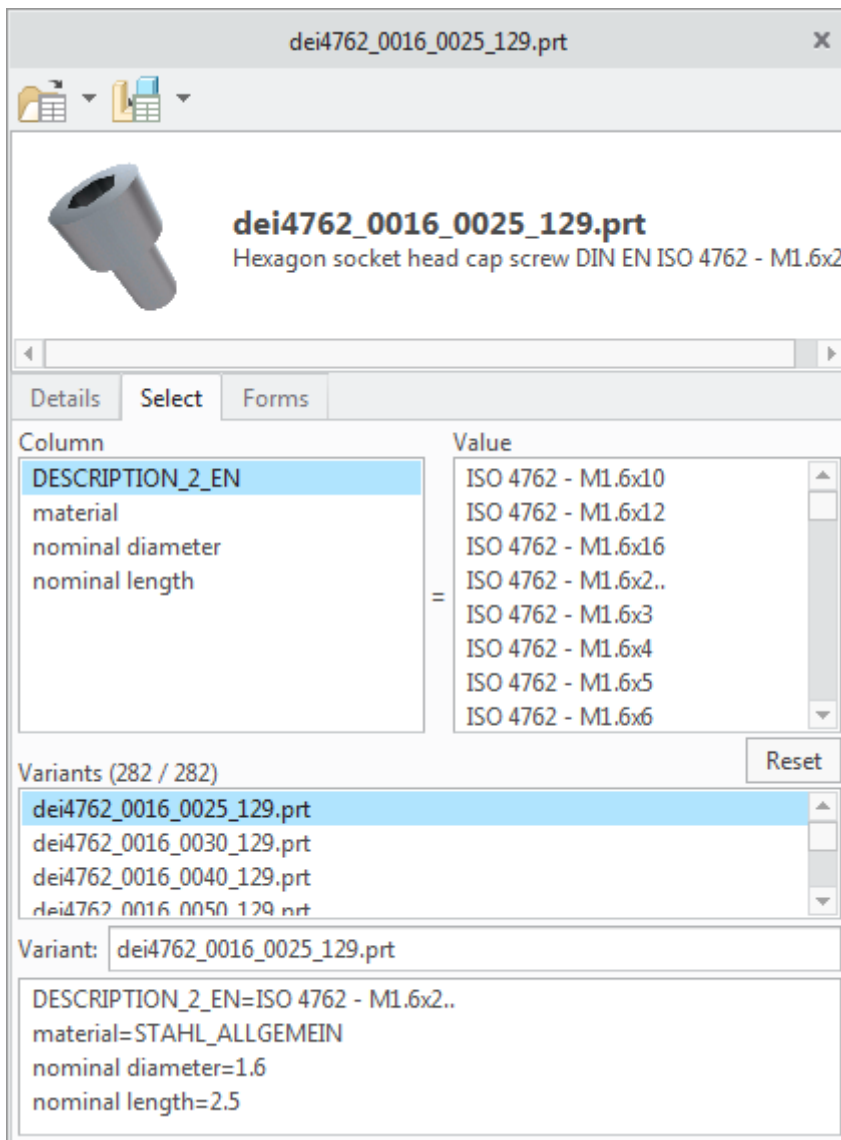
Activate these options to import family tables

4. Click *OK*. The family table is read. This may take some time.
5. Select the generic part in the editor and click *Edit selection list* in the Object Details area.
6. Check the created selection list. If needed, complete missing information. Confirm the dialog after making changes.



In the editor, click "Edit selection list" (1). Then check the selection list (2). You can customize the columns display and assign multilingual column names using the command bar buttons (3).

The family table is now imported completely and is available as a selection table in the library object.



The selection list is ready for use

9.3.8.5 Updating objects

When model information changes, this changed information must also be revised in your library database.

Use the Library Editor batch mode to update your library objects.

Using batch mode you can, for example:


- Update modified parameters
- Map modified parameters to multilingual titles
- Update preview images for your library objects

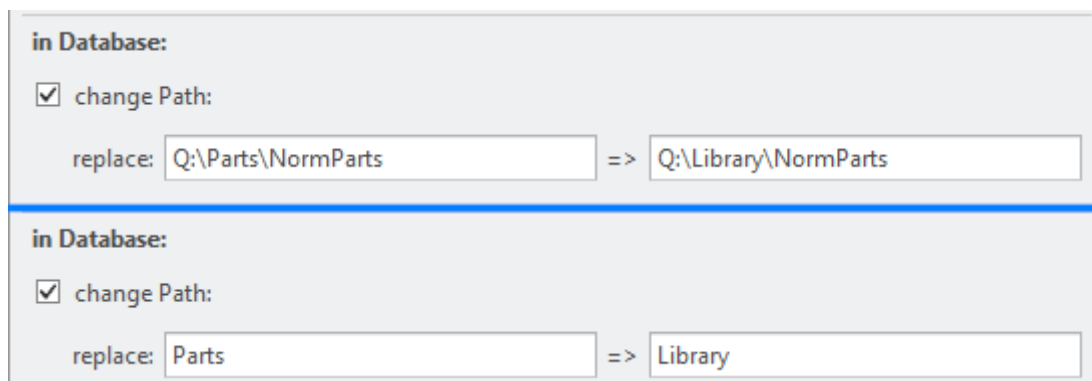
For these three scenarios, proceed as described at [Importing additional object information](#)²⁵⁸. Existing preview images, titles and imported parameters will be overwritten.

Updating file paths

File paths can be updated – first, by using the *batch mode*²⁴⁰ function (for single library objects) and second, by using the function *Replace path substrings by* in the *Cleaning up data*²⁴⁷ dialog.

Updating file path of single library objects: batch mode

1. Open Library Editor.
2. In the command bar click *Execute batch mode* .
3. Select the library elements whose file paths have changed the same way. Confirm your selection by clicking *Next*.
Example: All standard parts have been moved from Q:\Parts\NormParts to Q:\Library\NormParts.
4. In the second dialog box deselect all options except for *change Path*.
5. In the first input field, enter the whole file path (1) or the segment of the old file path (2) that should be changed.
6. In the second input field, enter the replacement for the segment.
7. Click *OK* to start the batch mode.



in Database:

☒ change Path:

replace: Q:\Parts\NormParts => Q:\Library\NormParts

in Database:


☒ change Path:

replace: Parts => Library

Segment in the second dialog box of the Batch Revision function

Please note: Entire file paths or just segments of a path can be replaced. Make sure there are no library objects included in the batch mode for which the file path segment should not be replaced.

Updating file path of all library objects in a directory: Clean up data

1. In the command bar click *Cleanup data* (broom button ).
2. Select the checkbox *Replace path substrings by*
3. In the first input field, enter the segment of the old file path that should be changed.
4. In the second input field, enter the replacement for the segment.
5. Click *OK* to start the cleanup function.

9.3.8.6 Importing additional object information

Use batch mode to import additional information for library objects into the database.

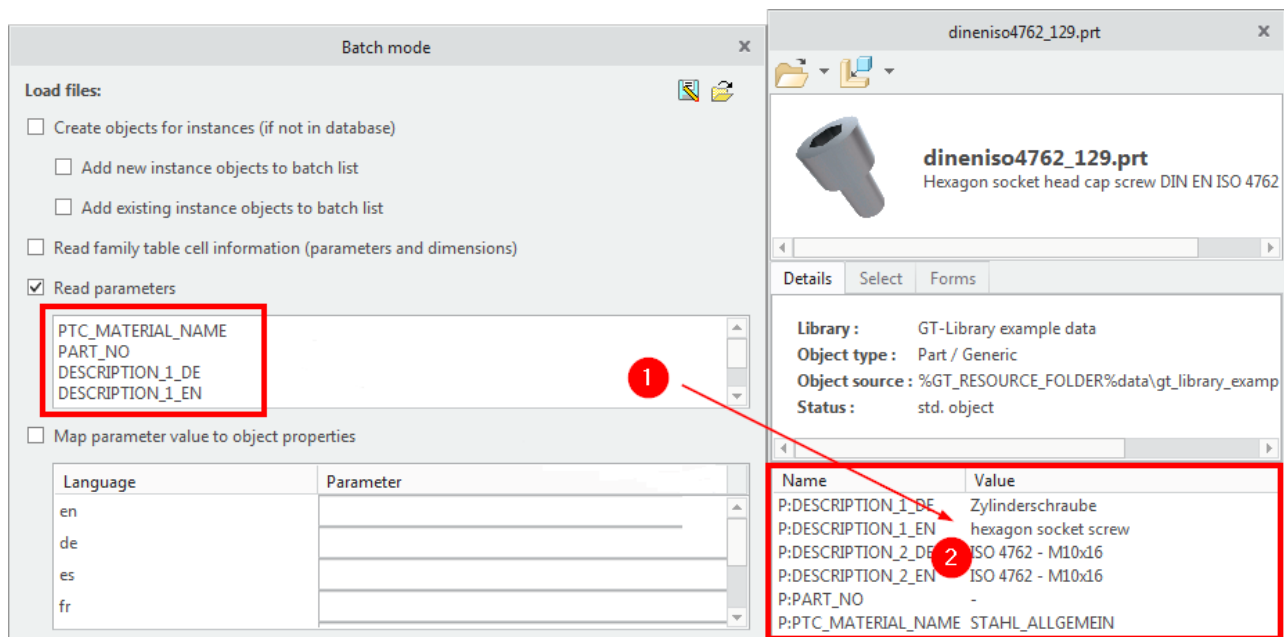
1. Open Library Editor.
2. Click *Execute batch mode*.
3. Select the library objects for which the additional information should be imported into the database. (individually or an entire category)
4. Click *Next*.
5. Select the additional information to be processed. In the dialog, deactivate all revision options that are not needed.
6. Click *OK* to start the revision.

Importing parameters for search optimization

Imported parameters are saved for each library object. They are displayed in the Details window when selecting a library object and can be used in the search, in advanced search and in selection tables.

1. Activate the *Read parameters* option.
2. In the input field below, enter the parameters that should be imported. Enter one parameter per line.

Please note: Make sure you spell all parameters correctly. Parameters not found in a library object will be ignored.



All parameters read into the batch revision (1) are imported into the library objects and are displayed in the Details window, for example (2)

Importing parameter as library object titles

Parameters from Creo data can be used to generate multilingual titles for library objects. A library object's title increases retrievability and can be used in the search and in advanced search. You can also search for titles in languages not displayed.

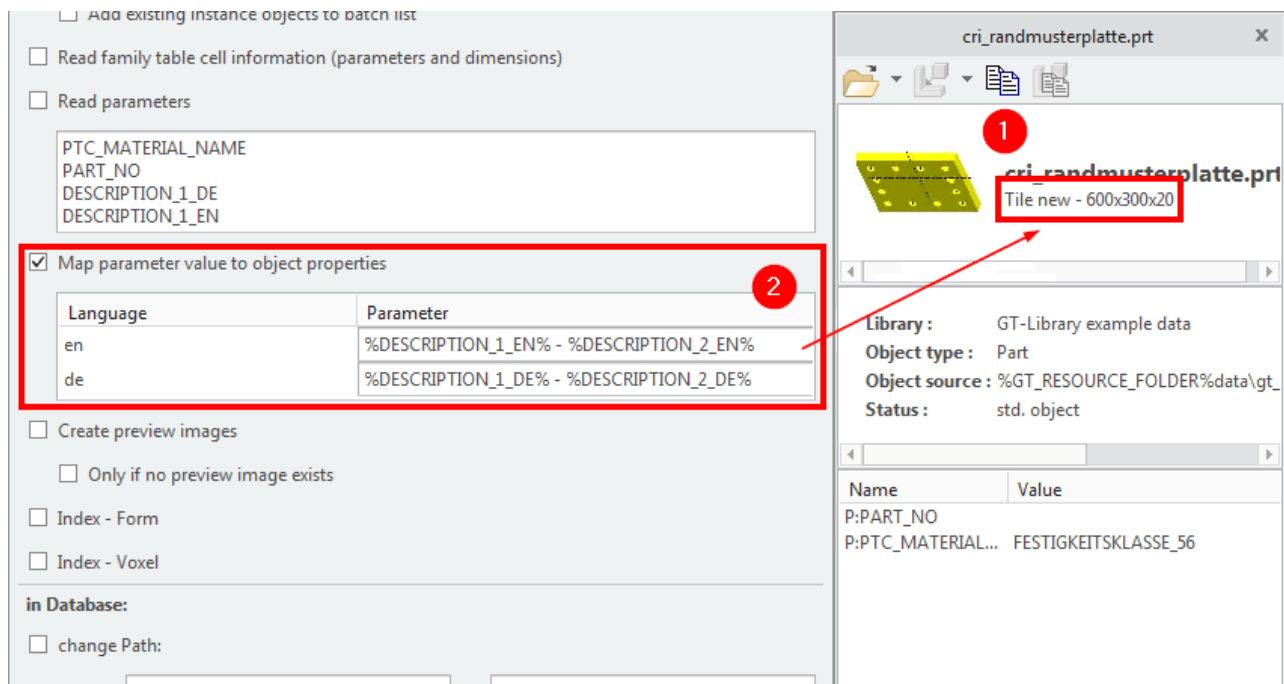
1. Activate the *Map parameter value to object properties* option.
2. In the table, enter the parameter to be used as the title after the individual language codes.

You can also combine parameters and text, and use multiple parameters as the title. Use percent signs (%) to separate parameters from each other and from static text.

Examples

%parameter1% - %parameter2%

%parameter1% - Generic part



Parameters mapped to a language (1) are available as title in library objects (2)

Creating preview images automatically

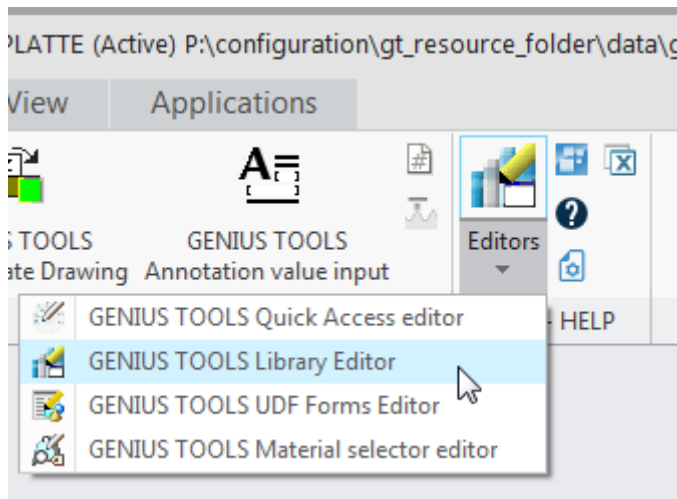
Preview images increase retrievability of library objects in the Library browser.

1. Activate the *Create preview images* option.
2. If for some of the library objects preview images do already exist, activate the additional option *Only if no preview image exists*.

9.3.8.7 Importing Library Viewer libraries

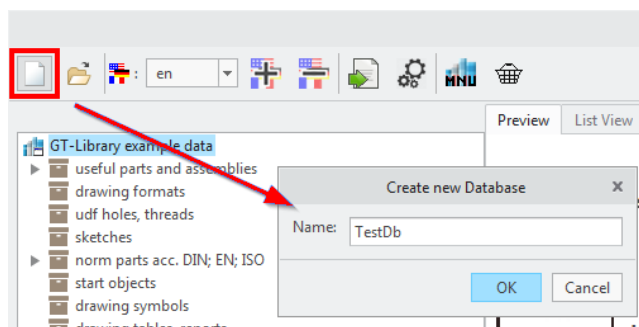
GENIUS TOOLS Library can import the file-based libraries of the Startup TOOLS TOOLBOX Library Viewer. These step-by-step instructions explain how to import a Library Viewer library using an example.

Start the Library Editor.



Start GENIUS TOOLS Library Editor from the ribbon

Create a new database and enter a descriptive name. The name entered is also the database filename.

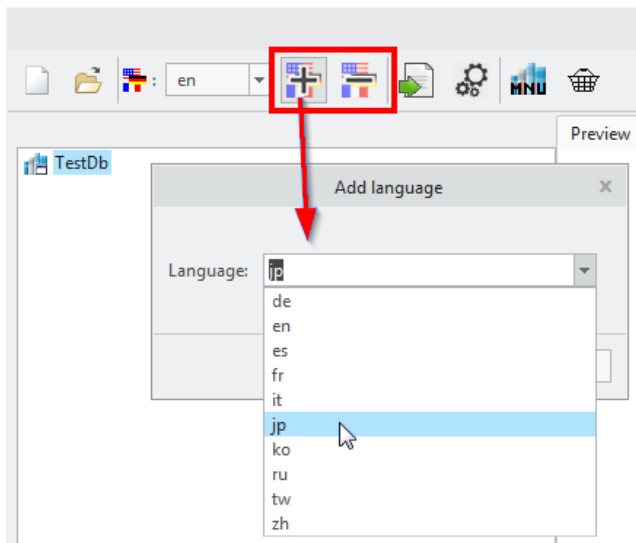


Create a new database

Warning: Make sure nobody else is working on the database during import when using an existing database.

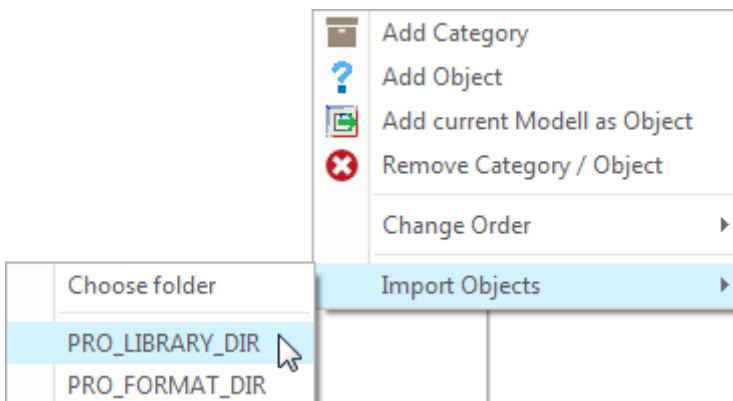
Create the additional languages needed in the database. Select a language code from the drop-down list or enter any custom language codes. This step ensures that all additional information maintained can be reused in the new database.

Please note: The input does not accept special characters and numbers.



Manage the included languages from the flag buttons

Open the context menu in the library preview (center area of Library Editor). Select the menu entry *Import Objects* and select *PRO_LIBRARY_DIR*.



The context menu in the preview area opens via right mouse button

Deactivate the option *Import chosen folder as category*. Otherwise, the *PRO_LIBRARY_DIR* will be displayed as a category above the existing library structure in the Library browser.

Make sure to select the proper library directory when working with Startup TOOLS and multiple projects. To find out which folder is currently referred to, check the config.pro option *PRO_LIBRARY_DIR* in the current Creo session.

Import Objects

Import to Node: TestDb - 0

Folder: D:\Lib ...

☒ Import chosen folder as category

☐ Import Folders only (folders: 77)

☒ Import Files and Folders (folders: 77 | files: 396 | variants: 11336)

Configure MNU import:

1. Row: en

2. Row: de

Configure STOOLS XML import:

DB-Language	XML-Language
en:	english
de:	german

Options:

☐ Check file existence

Cancel Next

*Check the directories and options in the import dialog.
Do not forget the language assignments!*

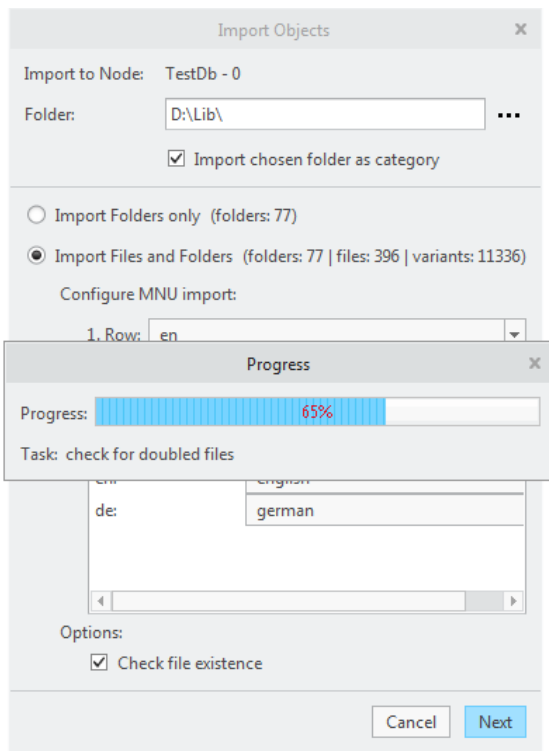
Please note: Check the assignment of the language settings.

Activate the *Check file existence* option if multiple manual changes have been made in your library. With this option, you can ensure all files do exist. This action is executed immediately as you activate it.

Do not use this option with Windchill as the library files will not be found.

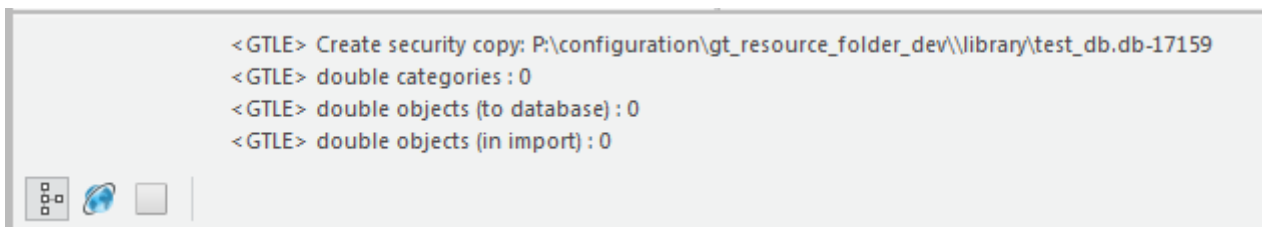
Click *Next*.

The library directory is checked for duplicate files.



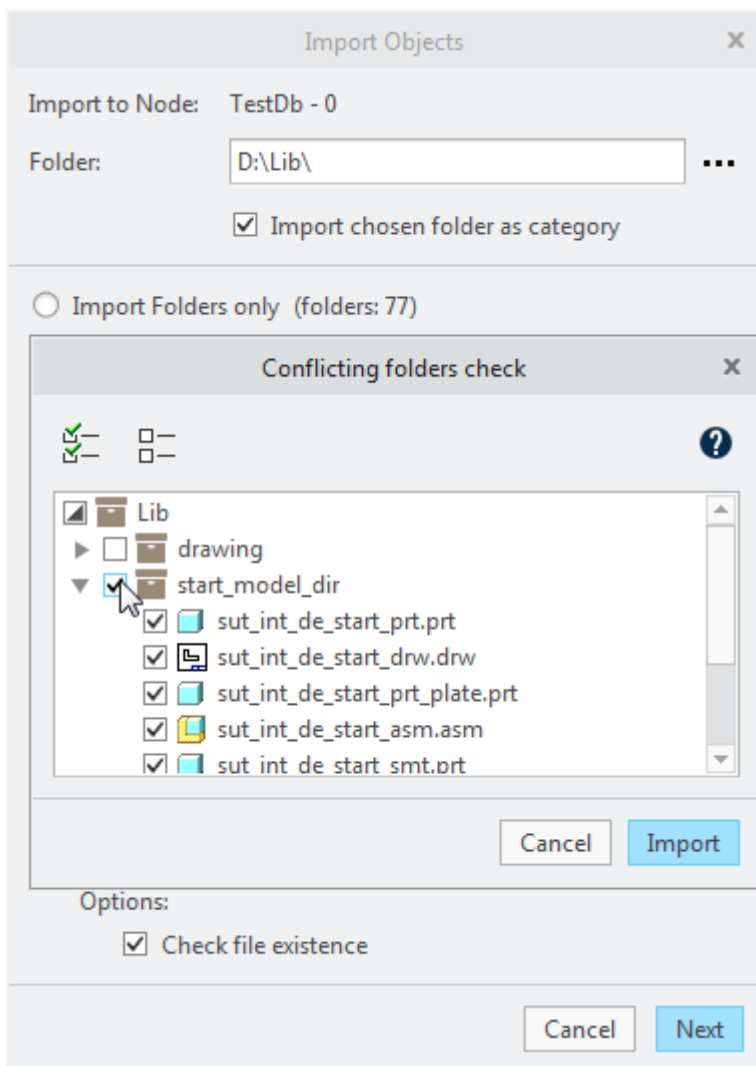
With bigger libraries, checking for duplicate files takes some time

Additional information on the importing directory can be found in the Creo Parametric info area after the check



Refer to the Creo info area for additional information

Select the elements to be imported in the tree structure.

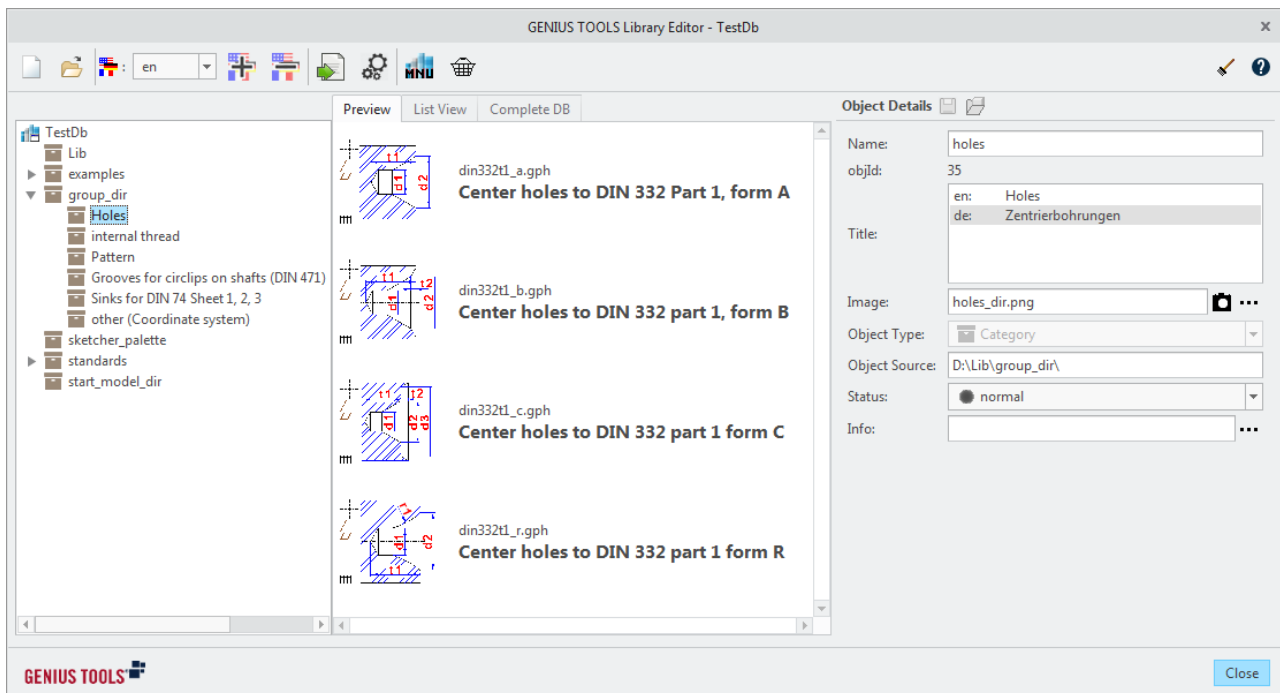


Select the files and directories to be imported and start the import

Click *Import*.

Please note: Checking and importing may take some time. This depends on the storage medium containing the library data as well as the access speed to the library.

The import of the Library Viewer library is now complete. The library functionality (except for ModelUIs) is now established.



After importing, the library database is ready for use

To include additional information for searching and selection (parameters, family table values and variants) in the library, execute a batch mode.

9.3.8.8 Importing MUI files

Proceed as follows to import MUI files into a library.

The MUI template models have to be present in a GENIUS TOOLS Library library, e.g., because you imported them. Also, the model for which you want to import an MUI has to be loaded in Creo.

1. Open the library containing the imported template models in the GT Library Editor.
2. Search for the required template model (ASM or PRT) for the MUI you want to import. Open the model in Creo.
3. Click *Convert STTOOLS MUI* in the object details. A dialog is displayed. Select the MUI and confirm.

Verifying the conversion

1. Close the Library Editor and select the model in Library. In the details window, select the tab *Form*.
2. Create a new model using the form.
3. Verify the properties of the newly created model. Were the correct models copied with it (asm, prt, drw) and the correct names assigned?
4. Verify the copying rules in the Library Editor und the [Library configuration options](#)⁶¹⁵:

- a. gtl_action_copy_set_file_as_common_name
 - b. gtl_copy_drawings_with_same_name
 - c. gtl_copy_check_existence
 - d. gtl_gtf_save_forms_in_model
 - e. gtl_gtng_overwrite_std_number_definition
 - f. gtl_gtng_standard_db_filter_for_file_copy
5. Verify the form as displayed in the Library details window
- a. Are all form elements present?
 - b. Do all form elements work?

Please note: Forms that have been changed have to be re-assigned to their library object.

10 Material

GENIUS TOOLS Material is available in part mode with the following features:

- selection of materials based on material properties
- adjustment and localization of the material data display in the editor

10.1 Introduction

In Creo materials are assigned to a model by selecting an MTL file from the material directory. Mostly, these files have cryptic names, such as *10143_s275j0.mtl* for an unalloyed steel.




By using the module GENIUS TOOLS Material users cannot only view materials by alternative, more descriptive names, but also filter them by properties, such as material group, delivery standard or revision parameter. These functions can be accessed in GENIUS TOOLS Material Selector in part mode.

To begin with, administrators should maintain and update all material properties in the material files (MTL files). With GENIUS TOOLS Material Selector Editor it can be determined which materials and properties will be displayed to users. In addition, administrators can translate properties into any language and add information to a material (as a PDF or web link), e. g. standardization requirements.

Please note: GENIUS TOOLS Material does not support MAT files. You can use the free software GENIUS TOOLS Material Browser by INNEO Solutions to transform MAT files into MTL files.

Assigning materials in Creo 7.0 and later versions

As of Creo version 7.0 a part can consist of several bodies. Materials can be assigned to a part as well as to a body. Hence, a part may consist of different materials. If no material is assigned to a body, the body inherits the material of the part.

The icons used in Creo are:  (for material in a model),  (for currently assigned material/ active material/ Master in Creo 7.0) and  (for inherited material). These icons are also used in the interface of GENIUS TOOLS Material Selector.

The assigned/active material is called "master" in Creo 7.0. If a part has not been assigned any material, Creo defines the master material as being the [system material](#)²⁷ *PTC_SYSTEM_MTRL_PROPS*.

Components of GENIUS TOOLS Material

The module GENIUS TOOLS Material is made up of a user dialog and an administrator dialog.



GENIUS TOOLS Material
Selector

- assigns material to a part
- can filter material list by one or more properties
- reads material properties from database *material.db*



GENIUS TOOLS Material
Selector Editor

- determines, which materials are available to users and which properties are displayed
- writes selected material properties from the MTL-files of the material directory into the database *material.db*

Read the chapter [Functioning](#)²⁶⁸ for further information.

Checking materials

GENIUS TOOLS Material provides for the automatic checking of materials in Creo-models, e.g. their versions. The chapter [Checking materials in models](#)²⁷⁹ explains this function.

10.2 Functioning

In a nutshell

- ✓ GENIUS TOOLS Material Selector edits a Creo model, using data from the database *material.db*.
- ✓ GENIUS TOOLS Material Selector Editor edits and updates the database, using data from the MTL files.
- ✓ GENIUS TOOLS Material Browser edits MTL files.

In Creo material properties are stored in a model, i. e. in the file of a part (PRT file). The content of these properties is read from material files (MTL files) in the material directory.

With GENIUS TOOLS Material material properties are also sourced from MTL files, but are additionally copied into an SQLite database (*material.db*) in order to make them available the users in the user component GENIUS TOOLS Material Selector.

When loading material files from the material directory into the database subdirectories are also copied, i. e the structure of the directory is preserved.

The database *material.db* is located in `%GT_RESOURCE_FOLDER%\material\`

This additional step of a database is necessary to restrict the number of materials to Creo users, to display defined material properties in a favored language and to add extra

information such as documents. These functions are available in the administrative component GENIUS TOOLS Material Selector Editor.

Also, this structure allows for checking material information in a model against material information in an updated material directory.

Please note: To provide users with material information you have to configure GENIUS TOOLS Material Selector Editor.

Updating the material database material.db


Please note that the material database is not updated automatically. This prevents possible problems with PDM systems such as Windchill. You have to update the database manually in the Material Selector Editor by clicking on the button *Update database*

(*Integrity check*) . (See also [Integrity check](#).)²⁸⁵

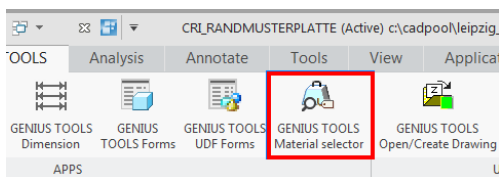
10.3 Usage

The user element GENIUS TOOLS Material is the material selection dialog *Material Selector*. It lists all materials and their properties, which are stored in the material database and which are approved of by the administrator in the editing tool *Material Selector Editor*²⁸¹. For an easier selection process, materials can be filtered by one or more of their properties and corresponding values.

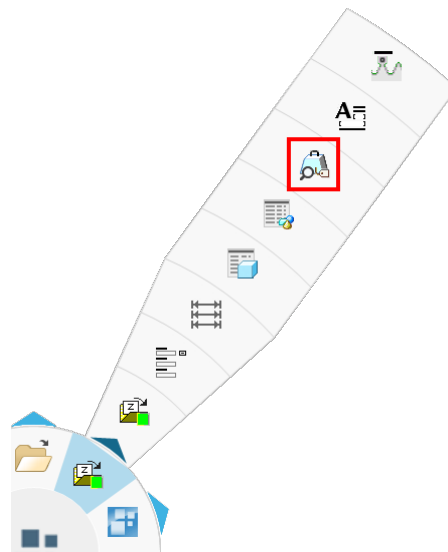
Starting the program: in part mode

Click the button *GENIUS TOOLS Material Selector*  in the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).

The button is available in part mode only since materials cannot be assigned in any other mode (such as assembly or drawing mode).



Starting from the GENIUS TOOLS ribbon menu



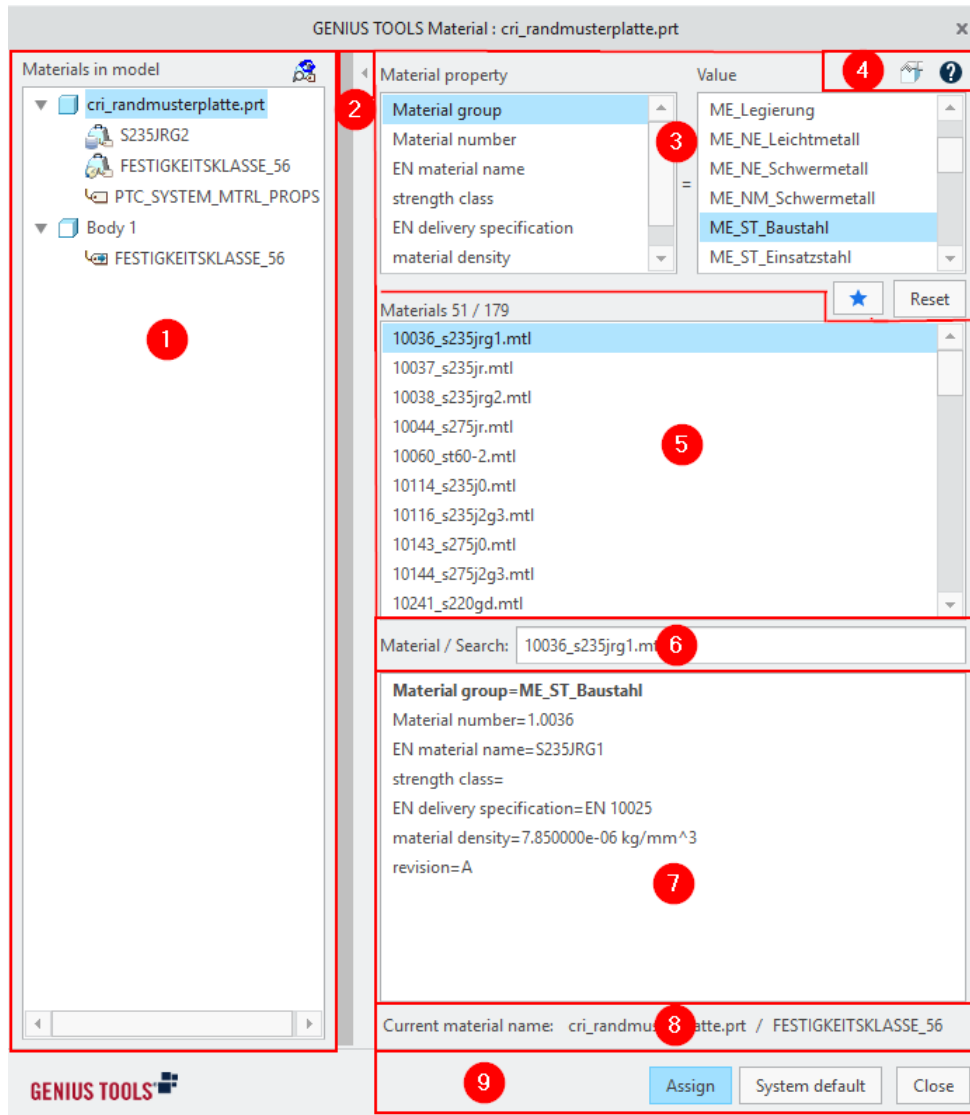
Starting from GENIUS TOOLS Quick Access

Please note: If you have not opened a part file yet, you are in Creo standby mode where the GENIUS TOOLS Material ribbon menu displays the button GENIUS TOOLS

Material Selector Editor  .

10.3.1 User interface

The user interface of GENIUS TOOLS Material Selector consists of the following elements:



User interface of GENIUS TOOLS Material Selector

1. Display of all materials in a model
2. Arrow for collapsing the left segment
3. Material properties²⁷² and corresponding values
4. Creo material dialog box and help
5. Material list²⁷³ from database or filtered/ favored material
6. Display of selected material and search function²⁷³
7. Display of properties and values of the selected material (filtered properties/values are bold)

8. Display of currently assigned (active) material to the part or body
9. Execute buttons: Assign²⁷⁴, System default²⁷⁷, Close

System default

The system default is

- for parts: assigning the system material²⁷⁷,
- for bodies: inheriting the assigned/ active material (master) from the part.

Display of all materials in the model

In the left segment *Materials in model* (1) all parts and – from Creo version 7.0 – bodies with their assigned materials are listed. (To assign a material to a part, choose it from the material list on the right.)

The material that is assigned to a part has the icon ➔ (active or standard material/ "master" in Creo 7.0). All other materials in a model receive the icon 📄. Material in a body that is inherited from the part has the icon 📄.

Additionally, the following icons can be displayed as a result of material checks: 🚧 - 🚧 - 🚧. You have to activate the material check and a deviation of a material has to be found for one of those symbols to appear. Find a list of the meaning of the symbols in chapter *Checking materials in a model*.²⁸⁸

Option: Stop closing the dialog box

After clicking the *Assign* button, the dialog box closes. If you want to keep the dialog box opened, e.g. because you assign materials to several bodies, set the configuration option `gtm_close_at_set_material` to 0. (Go to module GENIUS TOOLS Configuration Utility and then, in standby mode, to GENIUS TOOLS Reread Configuration.)

Option: Do not display materials in model

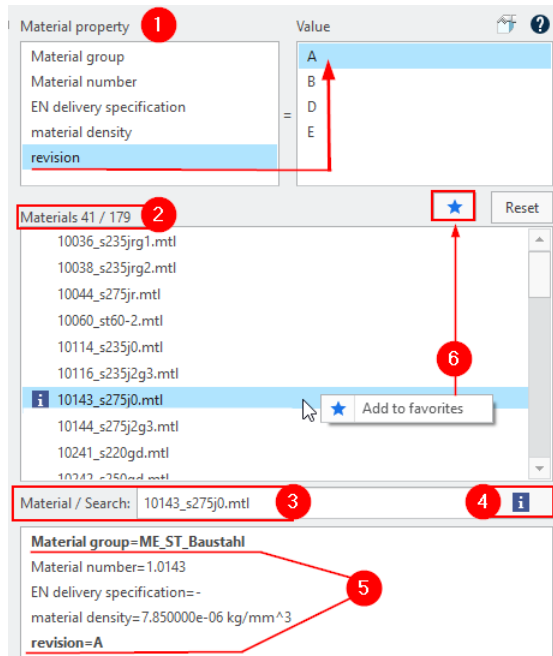
The left segment *Material in model* can be collapsed with the ☐ icon (2). You can define that it is already hidden when opening the dialog box by setting the configuration option `gtm_show_body_selection` to 0.

10.3.2 Selecting materials

1. Filter by material property and value

The two fields *Material attribute* and *Value* restrict the materials displayed in the material list. First, click a material attribute and then select the desired value, e.g. material group =

steel ("Baustahl" in German). All materials matching the selection are displayed in the material list. Repeat this step with additional material attribute-value pairs to further restrict the selection, e.g. delivery specification = DIN EN 10123.



Filtering by material group = steel ("Baustahl") and revision = A

The selection of your filters is displayed in the bottom section (1) in bold.

Click *Reset* to restart the selection.

Material attributes are displayed with localized names. If no localization exists, the name of material attribute in the database is used. Displayed attributes and values are defined in GENIUS TOOLS Material Selector Editor²⁸¹ which manages the material database.

2. Material list (as filtered)

The material list displays the results of the material attribute/value filtering.

The number of results (2) and the number of all available materials is displayed above the material list.

Click on the required material to select it and *Assign* to assign it to a model.

The selected material (3) is displayed below the material list.

3. Search

The search field is located below the material list. Click into the input field and enter any keywords. The search comprises text search in the file names and in all values. Confirm your input by pressing the Enter key.

Please note: Only materials meeting the current filter are searched. If you want to search all materials, click first *Reset*.

The search results are displayed in the material list.

Without active search, the currently selected material file is displayed in the search field.

If a material is selected in the "Materials in model" area on the left, it is immediately inserted in the search field and displayed as a result in the material list. This makes it possible to quickly assign a material that is already contained in the model to a body or the entire component. (Bodies are available as of Creo Parametric 7.0.)

4. Informational documents

If a material with an informational document is selected, an info icon appears on the right of the material/search field. Click to open it. Info documents can be assigned to the materials via the editor.

5. Display of material properties

The properties of the currently selected material are displayed below the search field. Properties in bold (1) are those that were used as filters.

Displayed properties depend on the displayed material attributes. If you want to add properties to the display, go to the editor.

6. Create favorite materials

A material can be defined as a frequently used material ("favorite") with a right-click and queried with the star symbol. (6)

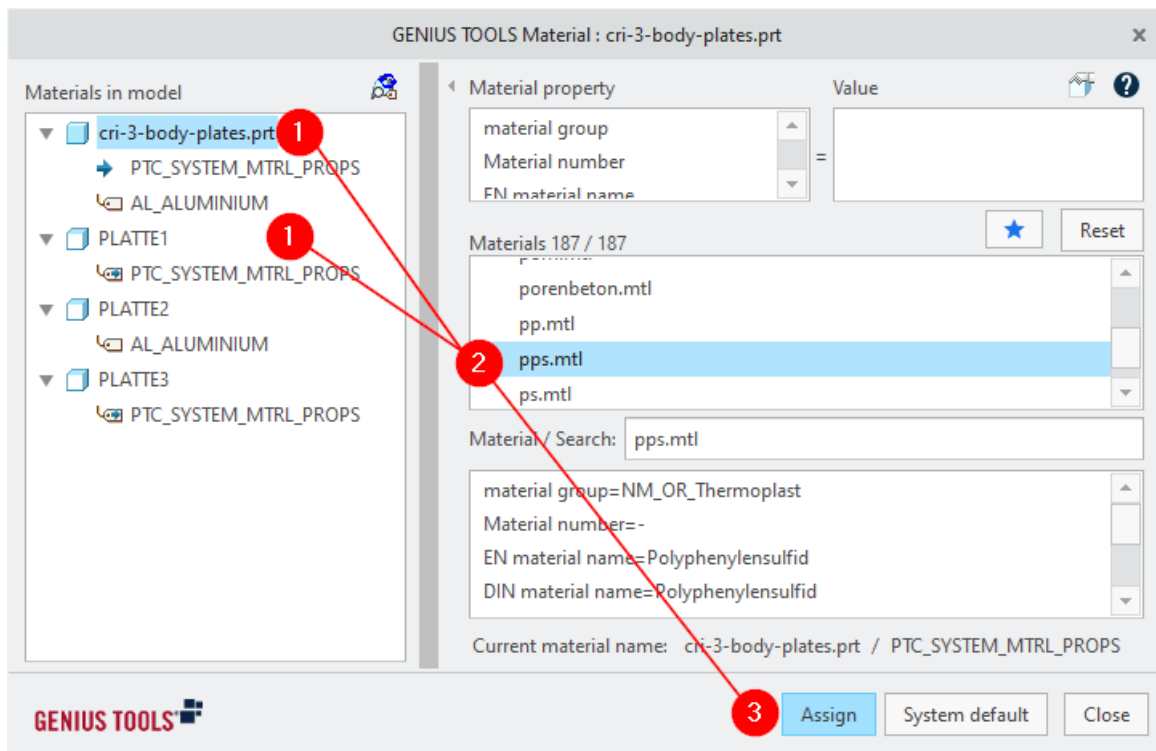
This does not apply any filter criteria. Instead, the favorite materials displayed are read from a specially created file. By default, the file *filter.txt* is created under %appdata%\<OperatingEnvironment>\GENIUSTOOLS\for_Creo. In the configuration option *gtm_favorite_file*, you can specify a different file and thus distribute, for example, a predefined file to specific sites or business units.

The configuration option *gtm_allow_edit_favorites* is used to prevent users from editing this list of favorites. (Default is 1 = Allow)

10.3.3 Assigning materials

Assigning a material to a part

1. In GENIUS TOOLS Material Selector select a part or a body in the left segment *Materials in model* (bodies are available as of Creo Parametric 7.0).



Selecting the part "cri-3-body-plates" and the material "pps"

2. Select the required material in the material list²⁷³ on the right.
3. Click Assign.

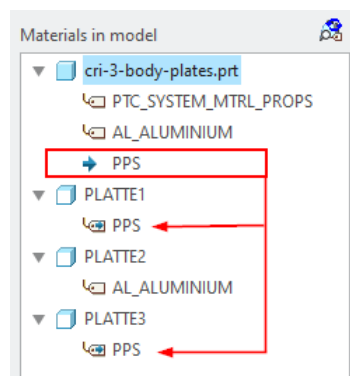
Results: Part versus body

– Assign material to the entire part:


The newly assigned material PPS is set as the standard / master material for the part (cri-3-body-plates) and gets the ➔ icon.

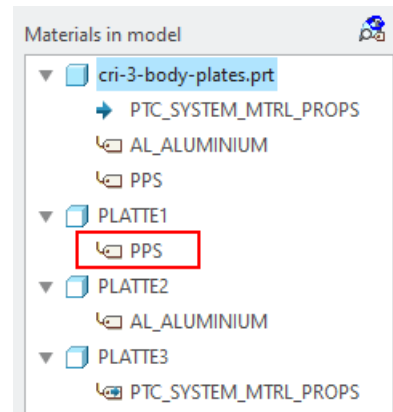
The new standard material is inherited to all bodies where an inheritance has previously been defined, as shown by the 📄 icon. (Here: bodies "PLATTE1" and "PLATTE2")

Materials that are no longer active remain in the model and are marked with the 📄 icon. (here: PTC_SYSTEM_MTRL_PROPS)



– Assign material to a body:

The newly assigned material PPS is assigned to the selected body ("PLATTE1") and marked with the  icon.

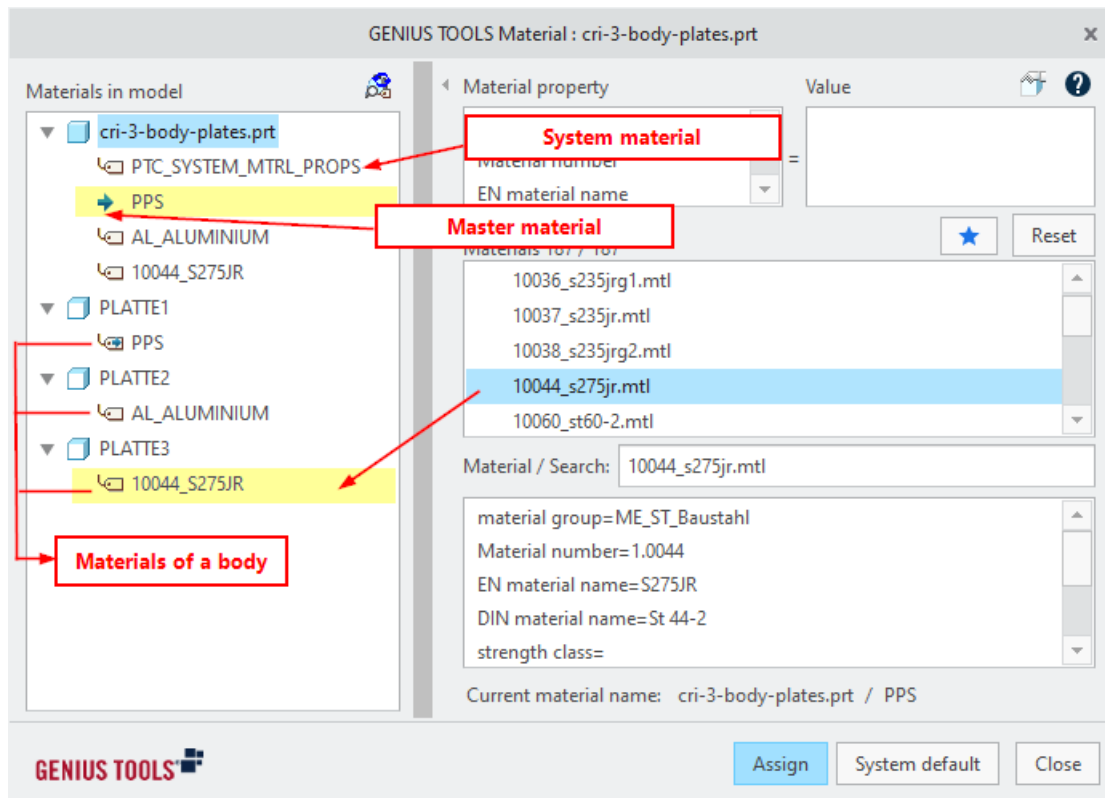


4. To display the newly assigned material in GENIUS TOOL Material Selector, it must be explicitly activated in the Material selection editor, see [Display settings of Material Selector dialog](#) ²⁸⁶.

Assigning material to a body

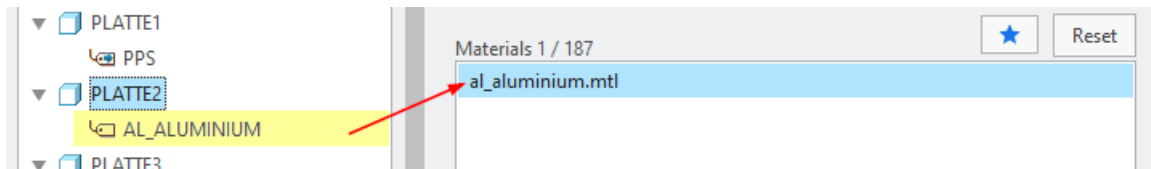
As of Creo version 7 a part can consist of several bodies and, hence, of different materials. If no material is assigned to a body, the body inherits the material (master) assigned to the part.

To assign material to a body, click on the body in the *Materials in model* segment before selecting a material.




Assigning an existing material

If you want to assign a material, which already exists in the part, you can click the desired material in the area *Materials in model* on the left. This will display the material in the material list on the right, so that you do not have to search for the material in the material list.



Assigning material in Creo Material dialog

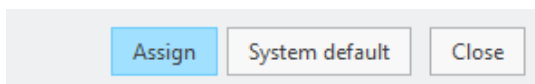
You may also assign material in Creo's Material dialog, which opens when clicking the icon  on the top right corner or in Creo model tree (*Materials* > *right clicking on Edit Materials*).

System material


In Creo 7.0. and later versions there are no longer parts without assigned material. All parts receive the system material *PTC_SYSTEM_MTRL_PROPS*. The properties of the system material can be edited, but the material itself cannot be deleted in a model.

A part can receive the system material by clicking the *System default* button.


System default button




The system default is



- for parts: assigning the system material ²⁷⁷, i.e. system material *PTC_SYTEM_MTRL_PROPS* becomes master material
- for bodies: inheriting the assigned/ active material (master) from the part.

10.3.4 Deleting material

To delete a material that is in the model, open the Creo Material dialog via the icon  at the top right. Select the material to be deleted and delete it by *right-clicking* > *Delete*. This is a standard Creo function that checks if there are dependencies in a model. If dependencies are present, the material cannot be deleted.


Delete the material assigned to the model (active material, master), marked with the icon , Creo from version 7 automatically sets the system material as master. In previous version, the default settings apply if the model has no material.


Please note: *GENIUS TOOLS Material Selector* will be closed when opening the Creo material dialog.

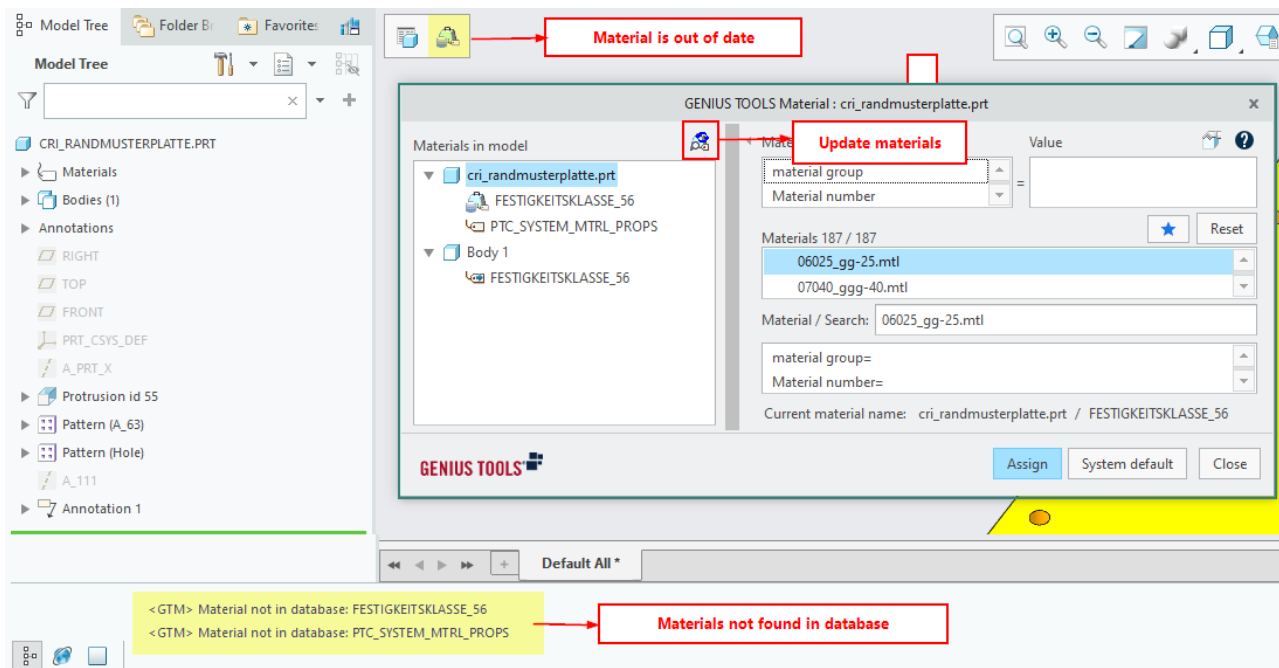
In all versions, the *No material found* notification appears with the icon , if no material has been explicitly assigned to the model and the **material check**  is activated.

Warning: The `gtm_delete_not_current_materials=1` configuration option (default: 0) deletes all unused materials from a model - regardless of valid relations, family tables, bodies or other dependencies. A configuration option set to this value does not check whether such dependencies exist. Deleted materials cannot be restored!
However, it is not recommended to use this configuration option. Make sure to use this configuration option only if you are sure that there are no dependencies in your model.

10.3.5 Updating materials

If a model contains obsolete materials, the *Update-materials* button  will appear in the Materials in model area. This function searches the material database for MTL files of the same name with a newer revision.

1. Click the button to automatically assign these material files.
2. Check if the update was successful: if the assigned material was not found in the database, this will be visible in the main Creo window by the  icon as well as in the Creo message bar. (See yellow fields in the picture.)



Materials cannot be updated if they are not in the database.

Disable "Update materials" function

If you want to exclude a material from the update function, list the material name without file extension in the configuration option `gtm_exclude_material_from_update_all`. This can be important for materials with variable density or for free materials.

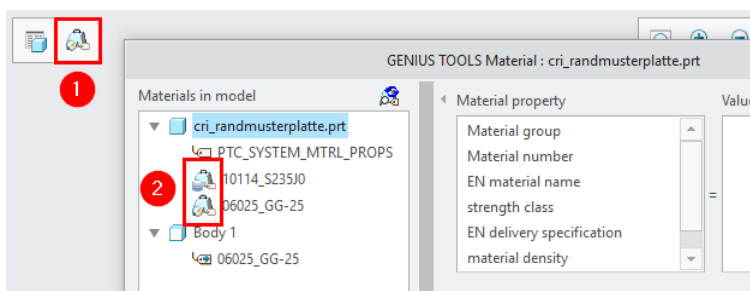
10.3.6 Checking materials in a model

GENIUS TOOLS provides for the automatic checking of materials in a model. It can test whether the material is up-to-date, whether it corresponds with the information in the database and whether the model has a material at all.

Read the chapter [Activating material check](#)²⁸⁸ on how to configure this function.

Results of a material check

If material check is activated and the checked material versions diverge, an icon will be displayed in the Creo Parametric main window on the left (1). Also, the icon will be displayed next to the material in the left segment in GENIUS TOOLS Material Selector dialog (2).



All materials in a model are checked and the result generates an according warn icon. These can be switched off with separate configuration options, see [Activating material check](#).²⁹⁰

If a model contains several materials, the highest of the following results of the material check is displayed. (This behavior can be turned off with the configuration option below.)

	Icon	Message	Meaning
1	No icon	–	No error
2		Material is dated	The revision parameter in the model is not the same as in the database. It is not detected that the revision in the model is older, only that it differs.
3		Several materials found with the same name	Several materials found in the database, e.g. in subdirectories.
4		Material cannot be found in database	
5		Material has no revision parameter	
6		Part has no material	This notification appears up to Creo version 6.
7		Material is Creo system material	As of Creo version 7 all parts need a material. This message comes up if no material has actively been assigned by any user and the system material ²⁷⁷ is set by Creo by default.

Option: Check only current material

If you want to receive a material check message only for the current/active material in a model, set the configuration option

`gtu_ui_change_check_material_check_only_current_material` to 1. (Go to the module **GENIUS TOOLS Configuration Utility**⁵⁵⁸ and then, in start mode, click on **GENIUS TOOLS Reread Configuration**.)

Please note: This configuration does not change the display of icons in the Material Selector dialog. When opening Material Selector all materials in a model are checked.

10.4 Configuration

In this section you find information on configuring GENIUS TOOLS Material with the administrative component GENIUS TOOLS Material Selector Editor, on configuring the automatic material check and on updating material files.

10.4.1 Material Selector Editor

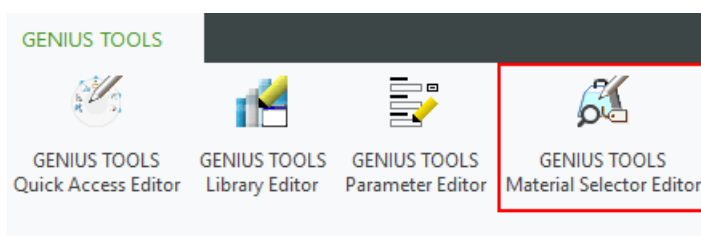
With GENIUS TOOLS Material Selector Editor, you can manage the number and the display of materials and material attributes in the dialog box of GENIUS TOOLS Material Selector.

Please note: GENIUS TOOLS Material Selector Editor does not edit material files.

10.4.1.1 Starting the program

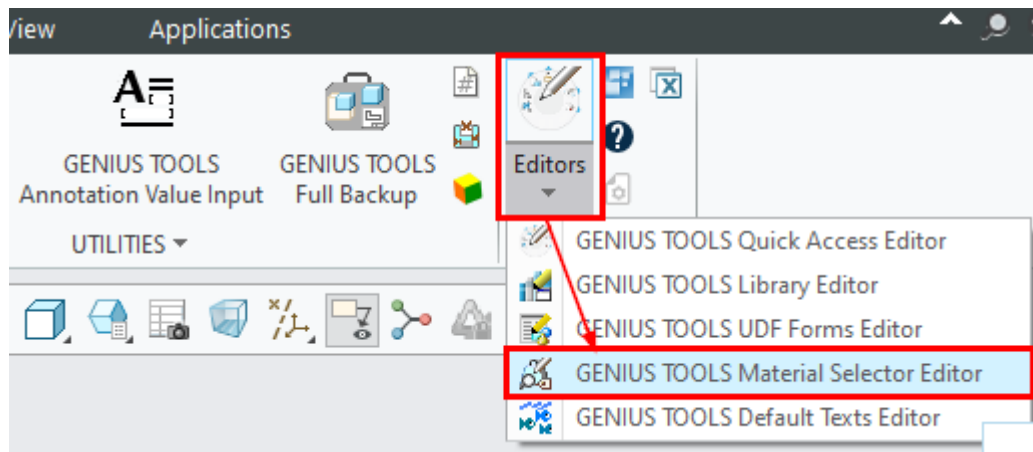
GENIUS TOOLS Material Selector Editor can be started in the following ways:

- in Creo standby mode (i.e. without having opened any file): in the GENIUS TOOLS tab in the ribbon menu

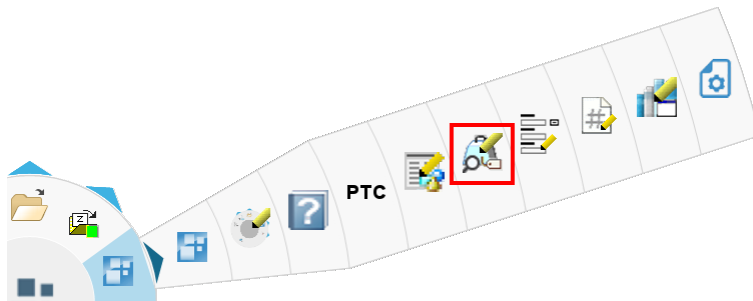


GENIUS TOOLS ribbon in start mode

- in part mode (i. e. in an opened PRT-file): from the GENIUS TOOLS tab in the ribbon menu in the segment *Editors*



– in GENIUS TOOLS Quick Access (key [<])



Saving changes

The changes made in the editor are saved in the resource directory.

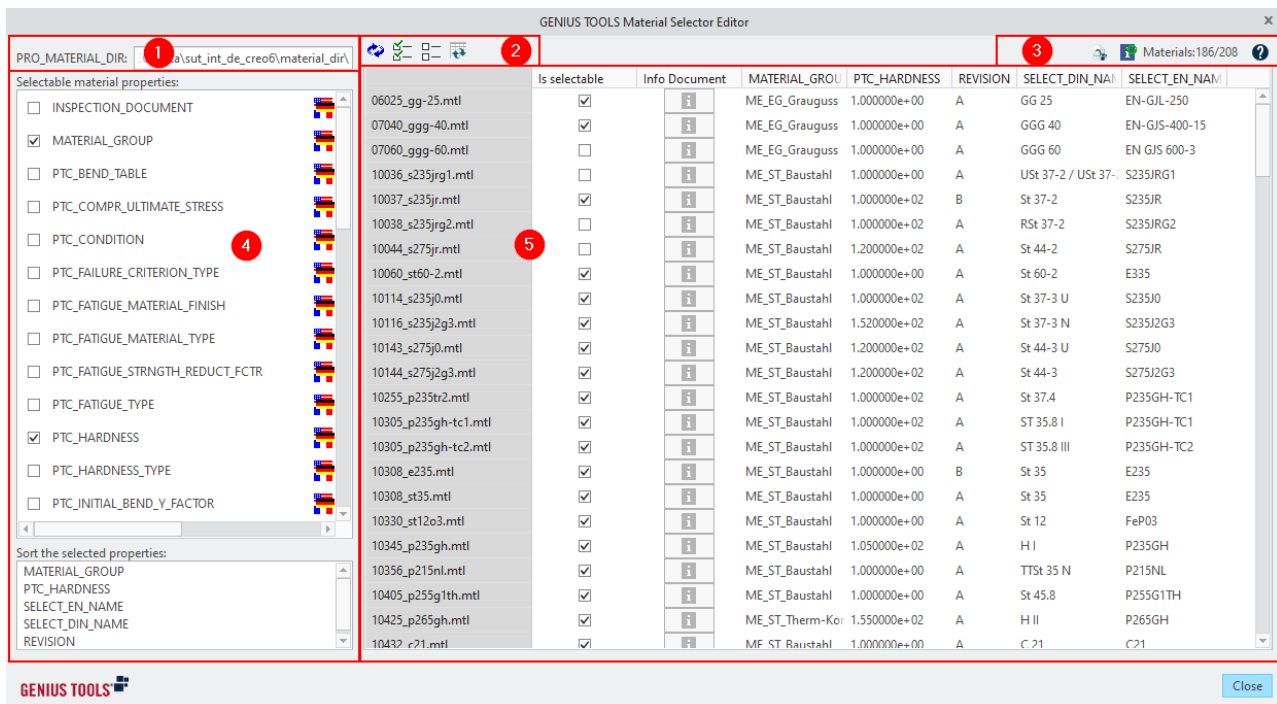
When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer (Caddepot). Path: `<operatingenvironment>/parametric/configuration/gt_resource_folder`.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

10.4.1.2 User interface

The user interface of GENIUS TOOLS Material Selector Editor consists of the following elements:



User interface of GENIUS TOOL MATERIAL EDITOR

1. Material directory (pro_material_dir)
2. Buttons for reloading material files, de-/selecting materials and updating database
3. Additional functions: Material Browser²⁹¹, Info documents²⁸⁶, Number of material files²⁸⁴, Program Help
4. Selectable properties and display order
5. List of all materials (MTL-files) from the material directory, including sub-directories

Warning message when opening the editor

If the number of material files in the material directory is less than in the material database, an error message is prompted. Update the material database in GENIUS TOOLS

Material Selector Editor by clicking on *Check database attributes/ values* . (See also Updating complete database: Integritycheck²⁸⁵)


10.4.1.3 Material directory


GENIUS TOOLS Material Selector Editor displays all material files available from the material directory, including sub-directories, after starting. The current directory is displayed on the left of the dialog box (1).

You can edit the path of the material directory by setting the `pro_material_dir` configuration option.

All files in the material directory are displayed in GENIUS TOOLS Material Selector Editor. (See next chapter.)



10.4.1.4 Defining selectable materials

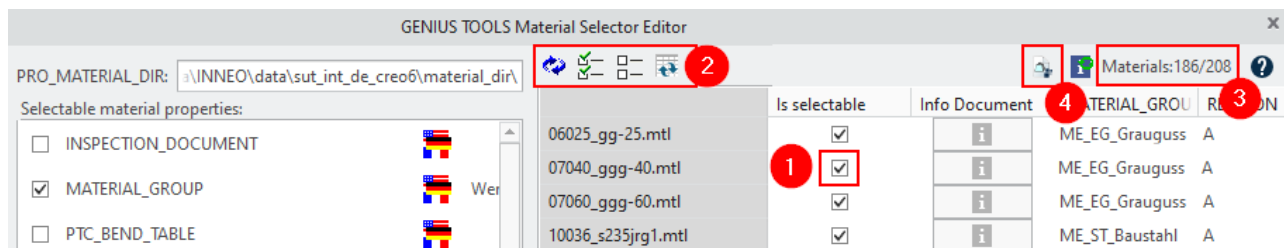
In GENIUS TOOLS Material Selector Editor you define the list of materials, which is available to users as well as the properties displayed. These specifications are written into the material database and update it correspondingly. To update the complete database – which checks all content of the database against the content of the selected materials in the editor – an integrity check is needed, which is generated by the command  *Update database*.

Please note: With the exception of the command  *Reload material files into Material Selector Editor* all actions in the editor change the material database.

Selecting material from material list

The segment on the right displays all material files of the material directory (`pro_material_dir`) including sub-directories. Activating the check box (1) next to a material file (column *Is selectable*) copies all selected material properties of the file into the material database. The material is then available to the user in GENIUS TOOLS Material Selector Editor.

By clicking the buttons (2)  *Select all materials* or  *Select no material* you can quickly check and uncheck all boxes.




Define material selection in GENIUS TOOLS Material Selector Editor

Materials are displayed with the properties checked on the left segment *Selectable material properties*.


Number of material files displayed (3)

The top-right corner (3) displays the total number of materials in the database and the total number of materials in the material directory. If the first number is higher than the second, it will mean that the database is not up-to-date.


Updating database with integrity check

By clicking on the button  (segment 2) the database is updated which includes an integrity check. This process

- deletes non-existing MTL files (files that are in the database but not in the material directory),
- copies changes in MTL files into the database (if these are changes in the selected material properties).

If you add material files, reload the editor with  (see next).

Updating material files vs updating database

If you edit one or more MTL files and want to make these changes available to the user, click the icon  (segment 2). This does not update the database, but rereads all material files from the material directory into the Editor.



Reload material files in
Material Selector Editor


- uploads changes in MTL files and new MTL files into Material Selector Editor
- corresponds to a restart of Material Selector Editor (Opening afresh the dialog box)
- does not load content from MTL files into the material database



Update database (integrity –
check)

- Copies afresh all MTL files and properties, as selected in the editor, into the material database and checks all entries with an integrity check

Edit material files

With the  button (4) you can edit the material files directly in the material directory if you have installed the separate freeware GENIUS TOOS Material Browser in version 1.0.4.0 or newer. For older versions, Material Browser will open, but the material directory must be entered manually.

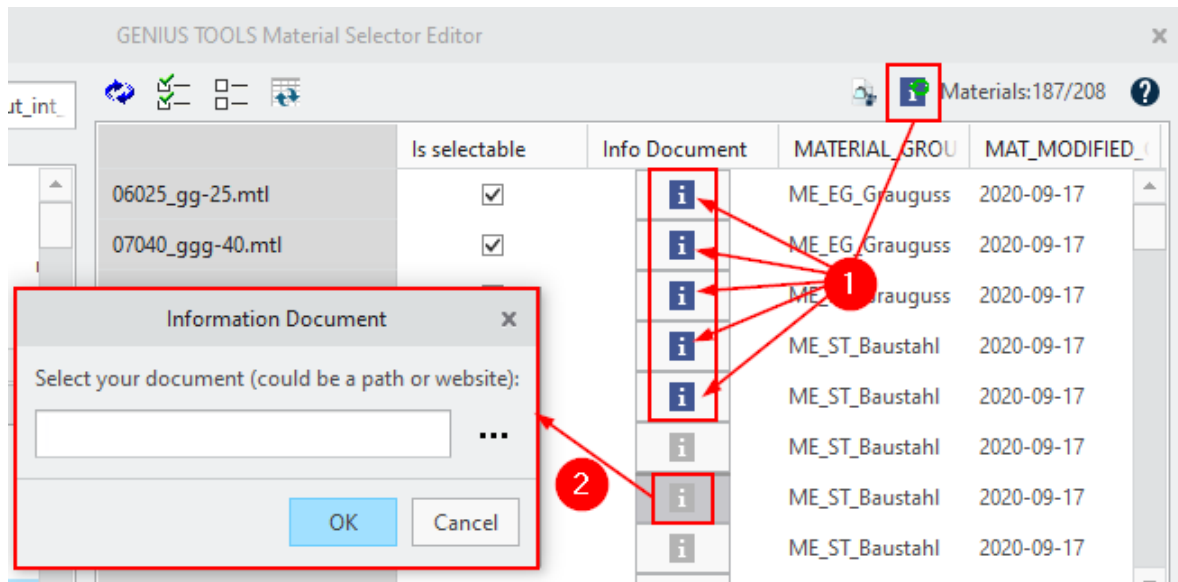
The button is displayed if GENIUS TOOLS Material Browser is installed and the path to the executable file *GT_Material_Browser.exe* is set correctly in the configuration option `gtm_editor_material_browser_path`.

If you work with the PDM system Windchill, consult additionally the chapter [Edit material files in Windchill](#) ²⁹².


Adding info documents

You can add documents to individual materials. These info documents can be retrieved in the GENIUS TOOLS Material Selector dialog by the user.

You can upload all available documents automatically (1) or upload a document for a single material (2).



Providing documents for materials

(1) Use the  button to activate the automatic upload of all documents whose name corresponds to the name of any material file. This function searches for documents of all file types in a specific folder. The path to this folder is set by the configuration option `gtm_infoDoc_folder`. (Default: `%gt_resource%\material\info`).

(2) To upload info documents individually, click the info icon next to the material and enter a file or web address in the following dialog. To be able to open websites and network documents, you have to specify the appropriate protocol (`http://`, `https://`, `ftp://` etc.).

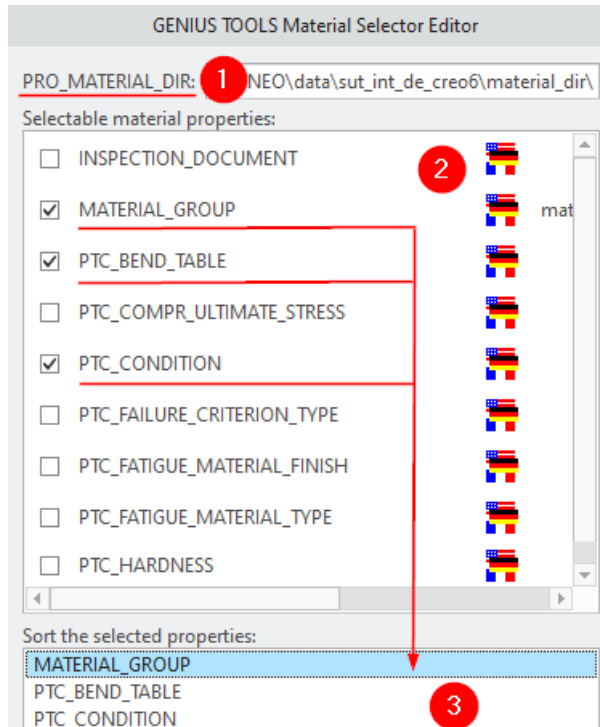
10.4.1.5 Display settings of Material Selector dialog

In GENIUS TOOLS Material Selector Editor you can define which material properties are displayed to the user in GENIUS TOOLS Material Selector and you can translate material properties.

Specify material attributes and arrange their display order

The material attributes which should be displayed to the user are listed in the segment *Selectable material properties* (2). Activate the check boxes next to the required material properties.

Material properties are read from the parameters in an MTL-file and can be Creo parameters (e.g. PTC_BEND_TABLE) as well as company-specific parameters (e.g. SELECT_EN_NAME).




Selection of properties MATERIAL_GROUP, PTC_BEND_TABLE und PTC_CONDITION

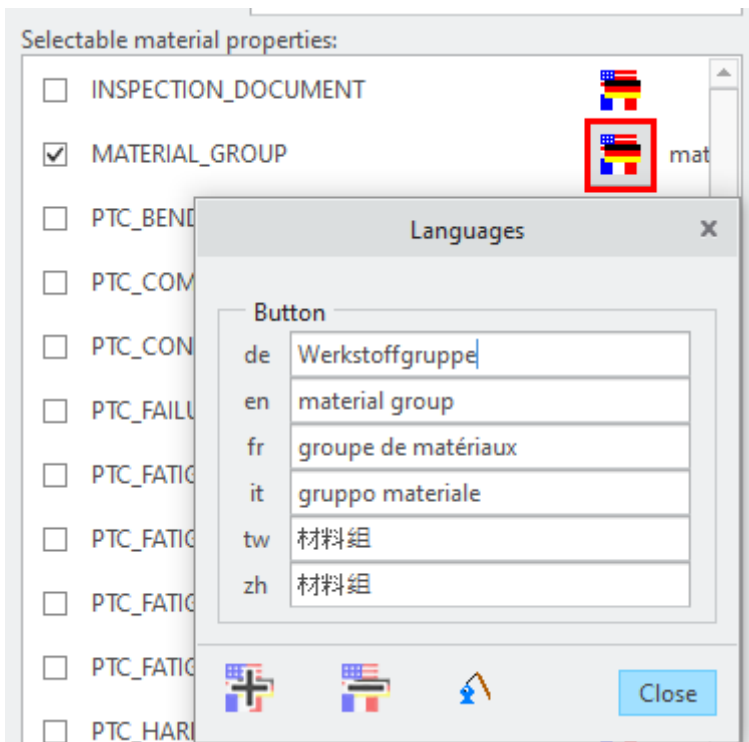
Please note: The material database is rewritten, as soon as you activate/deactivate a material property. With numerous selected materials, this action may take some time.

The display order of the selected attributes is displayed below (3), i.e. the order in which the properties appear in the filter function of the GENIUS TOOLS Material Selector dialog. Drag and drop the properties to the required position.

Translating material properties

Material properties can be localized, i. e. displayed in a specific language. Click the flag icon behind the property terms to open the localization dialog.

You can add or delete languages by clicking on the + or - icon in the localization dialog. With the button  you can choose standard text. (See also [Set standard text](#)⁴⁹⁷.)



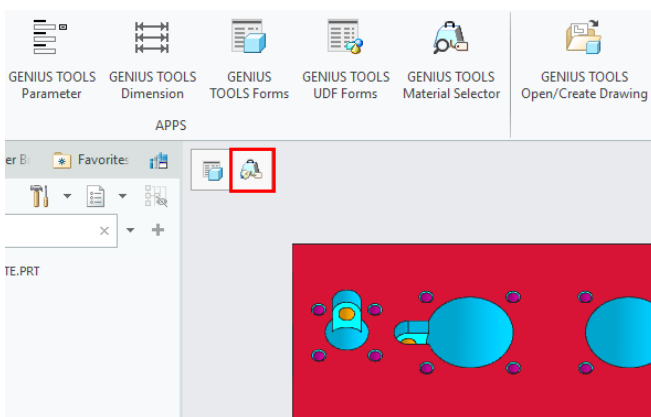
Localization dialog

Properties that are not translated receive the term saved in the database.

The set language (via configuration option `gtm_db_lang[261]`) determines which localization is used in GENIUS TOOLS Material Selector.

10.4.2 Activating material check

GENIUS TOOLS Material allows for automatically checking materials in models. The result of this check is displayed as an icon on the top left corner in the Creo Parametric main window as well as next to assigned materials in the Material Selector dialog. You find a list of all possible results in the chapter [Checking materials in a model^{\[279\]}](#).



Notification icon in Creo Parametric

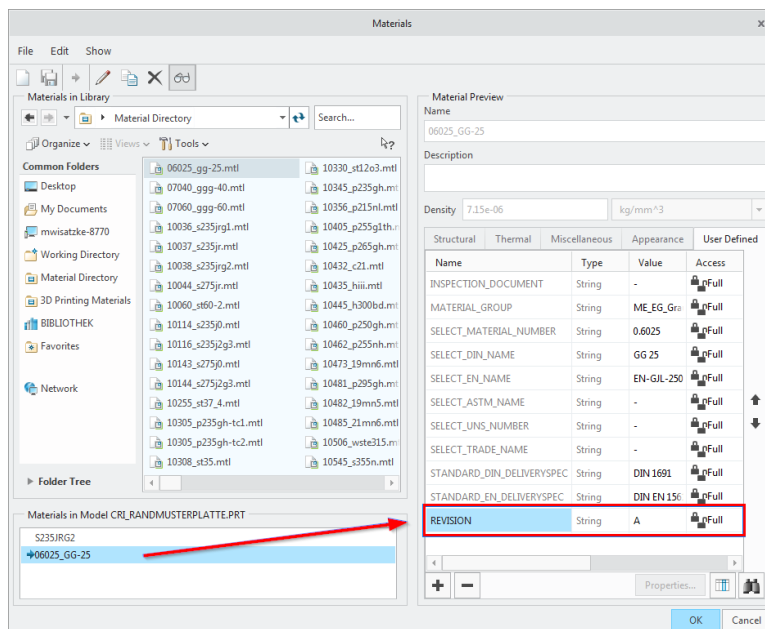
Please note: Materials in a model are not replaced by this function.

The function material check requires a material parameter for storing the revision information of the material. When opening a part, this parameter is read from the material in the model (i.e. from the PRT file) and compared with the parameter in the material saved in the material database.

Activating material check

Proceed as follows to use material checking:

1. Determine a material parameter in which to store the material revision.
2. Maintain the parameter in your materials, either in the MTL files in the material directory or with [GENIUS TOOLS Material Browser](#)²⁹¹ or in the Creo Materials dialog. Make sure the parameter is of type String.



Maintaining the material parameter using Creo's material editor

3. Set the configuration option `gtu_ui_change_check_material_version` to 1. (You can use [GENIUS TOOLS Configuration Utility](#)⁵⁵⁸ .)
4. Specify the material parameter in the configuration option `gtu_ui_change_check_material_version_parameter`.

Option: Disable display of notification “Material is Creo system material”

You can turn off the display of the icon when it notifies you that a part only has system material by setting the configuration option `gtu_ui_change_check_material_system_material_is_wrong` to 1.

Option: Check only current material

If you want to receive a material check message in the main Creo window only for the current/active material in a part (in Creo 7.0: "master material"), set the configuration option `gtu_ui_change_check_material_check_only_current_material` to 1.






Please note: Do not activate this configuration if a message for dated material in a body should be displayed in the main Creo window.

This option does not affect the behavior of the material check function in the Material Selector dialog, i. e. when opening GENIUS TOOLS Material Selector all materials in a model are checked and message icons displayed next to a material.

Switching off options for the material check

The following configuration options check materials and generate the according warn icons.

The default settings are 1 (on).

Icon	Message	Meaning, configuration option
	Material is dated	The revision parameter in the model is not the same as in the database. It is not detected that the revision in the model is older, only that it differs. <code>gtu_ui_change_check_material_check_old_material</code>
	Several materials found with the same name	Several materials found in the database, e. g. in subdirectories. <code>gtu_ui_change_check_material_check_multiple_materials</code>
	Material cannot be found in database	<code>gtu_ui_change_check_material_check_material_not_in_db</code>
	Material has no revision parameter	<code>gtu_ui_change_check_material_check_material_has_no_param</code>
	Part has no material	This notification appears up to Creo version 6. <code>gtu_ui_change_check_material_check_no_material</code>

10.4.3 Edit material files


Material files (MTL files) can be located in the material directory locally or on a server or in Windchill.

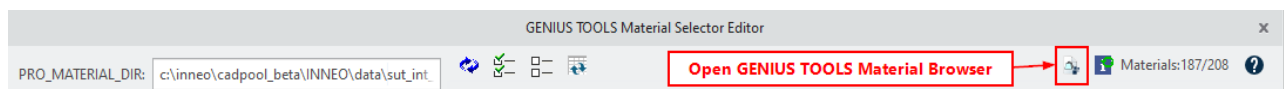
Without Windchill you can edit material files in two ways:

- Write new or make changes to existing material properties (parameters) manually into an MTL file.
- Edit larger quantities of MTL files with the separate freeware GENIUS TOOLS Material Browser.

Information on editing material files in Windchill are found in the next chapter.

10.4.3.1 Edit material files with GENIUS TOOLS Material Browser

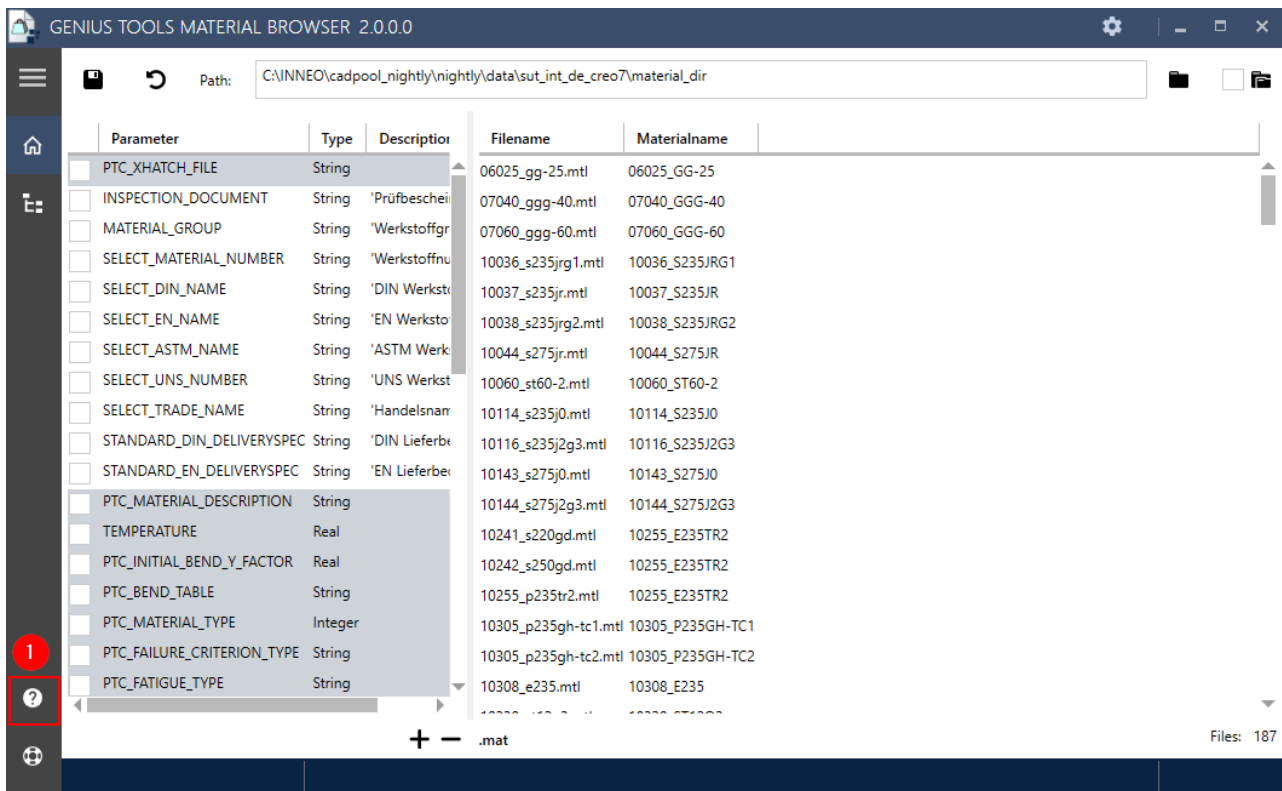
You can edit material files (MTL files) directly in the Material Selector Editor. The  button opens the separate freeware GENIUS TOOLS Material Browser and displays all files in the material directory. This function is available with Creo version 7.0..



The button is displayed if GENIUS TOOLS Material Browser is installed and the path to the executable file *GT_Material_Browser.exe* is set correctly in the configuration option `gtm_editor_material_browser_path`.

GENIUS TOOLS Material Browser version 1.0.4.0 and newer opens the material directory with which you work in the current session. For older versions, the material directory must be entered manually.

Information about the Material Browser can be found in the program in the help icon (1). The application can be downloaded free of charge from the download area of the Inneo website. (<https://www.inneo.de/de/services/technischer-support/genius-tools-downloads.html>)

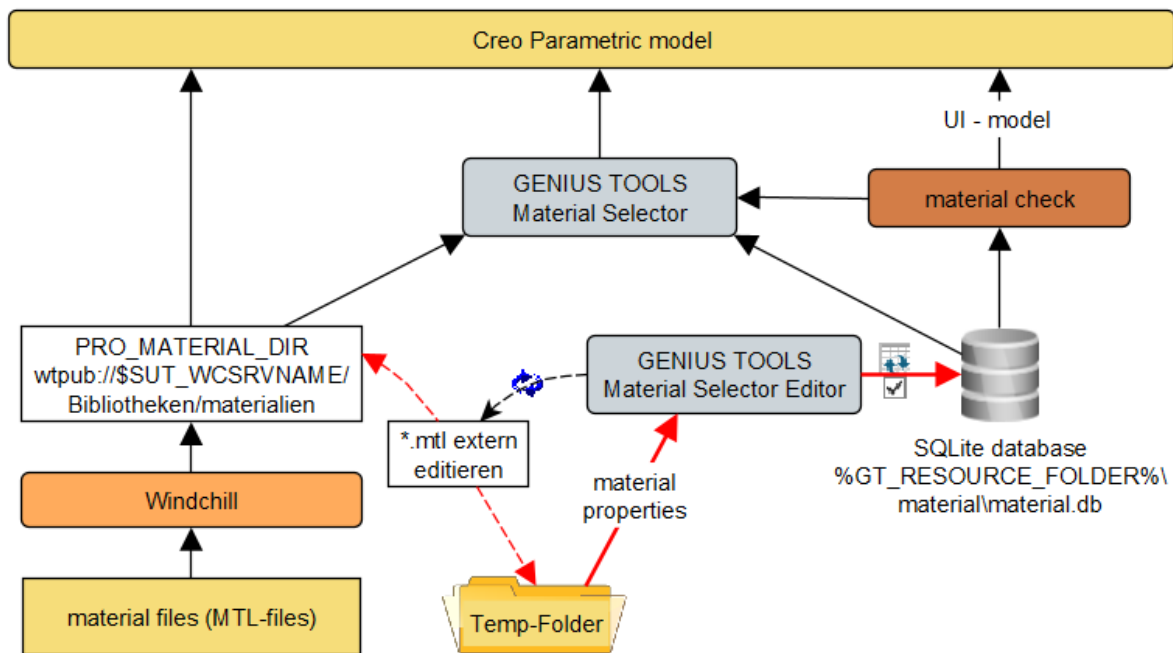


10.4.3.2 Edit material files in Windchill

If you use a material directory which is located in Windchill you may edit material files the following way. (This function is available with Creo version 7.0..)

Prerequisites:

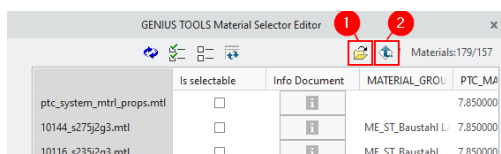
- an active Windchill server
- Windchill is connected
- the path to the material directory specified by the configuration option
`pro_material_dir` is set to a windchill directory, e. g.
`wtpub://$SUT_WCSRVNAME/Bibliotheken/materialien`



Procedure:

Warning: Keep GENIUS TOOLS Material Selector Editor open during the whole process.

1. Open GENIUS TOOLS Material Editor. This will load all MTL-Files of the material directory in a temporary directory Temp folder. The material directory is transferred in its existing structure, i. e. including all subdirectories.
2. Open Temp Folder by clicking *Open temporary folder* (1)



3. In Temp folder edit the MTL files (manually or with GENIUS TOOLS Material Browser).
4. Return to the open Material Editor and upload the edited MTL files into the workspace by clicking on *Upload edited files* (2)
5. Check in the edited MTL files into the workspace.
6. Upload the edited MTL files in GENIUS TOOLS Material Selector Editor by clicking *Reload material files*.
7. Update material database by clicking on *Update database (Integrity check)* ²⁸⁵.

11 Multibody to Assembly

GENIUS TOOLS Multibody To Assembly allows you to transfer parts that have been created with bodies to an assembly structure. Every thus created part includes an external copy geometry feature that references exactly one body.

Please note: The function GENIUS TOOLS Multibody To Assembly is only available with subscription licenses for GENIUS TOOLS for Creo. The Creo Advanced Assembly Extension (AAX) is also required.


There are two modes existing:

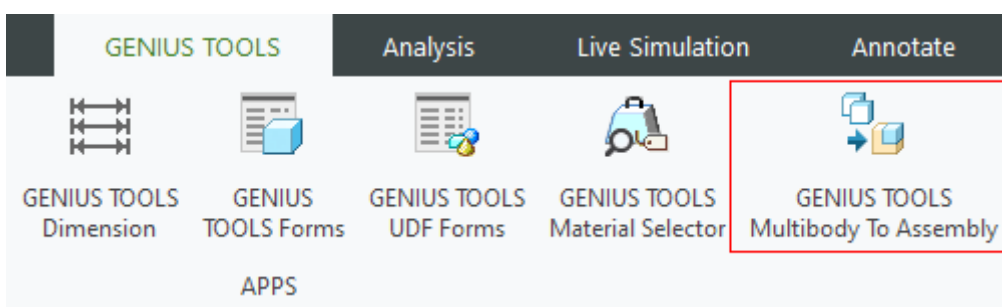
- initial creation: New export of bodies to an assembly that has not yet been created (empty user interface)
- update mode: Exported assemblies are expanded to include new parts

Please note: Components that have already been exported cannot be removed or edited with GENIUS TOOLS Multibody To Assembly.

11.1 Usage

Starting the program: in Creo part mode or assembly mode

GENIUS TOOLS Multibody To Assembly  can be started in the GENIUS TOOLS ribbon menu.




Configuration of the start state

With the configuration option `gtmba_select_last_export_at_start` the user interface can be configured.

0 = GENIUS TOOLS Multibody To Assembly starts with an empty user interface (initial creation)

1(default) = GENIUS TOOLS Multibody To Assembly starts with the last exported assembly (update mode)

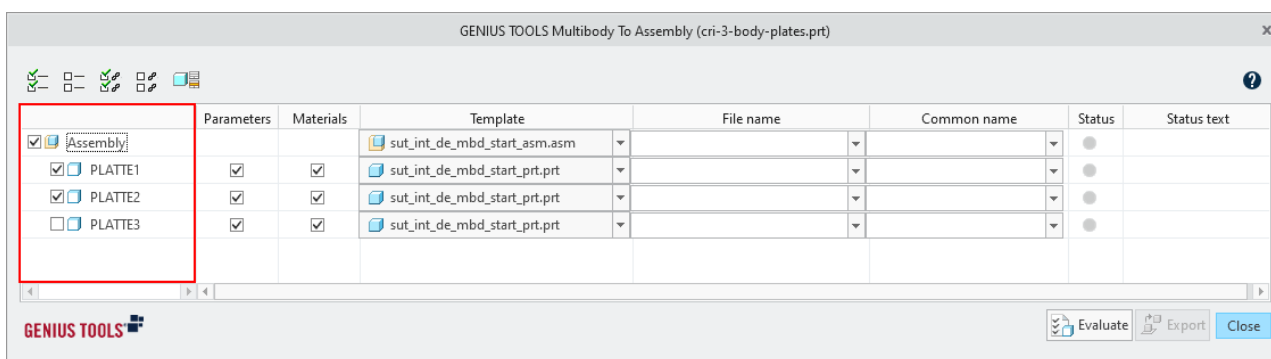
Display configuration


Use the configuration option `gt_start_multibody_to_assembly` to hide the button . Default: 1 = On

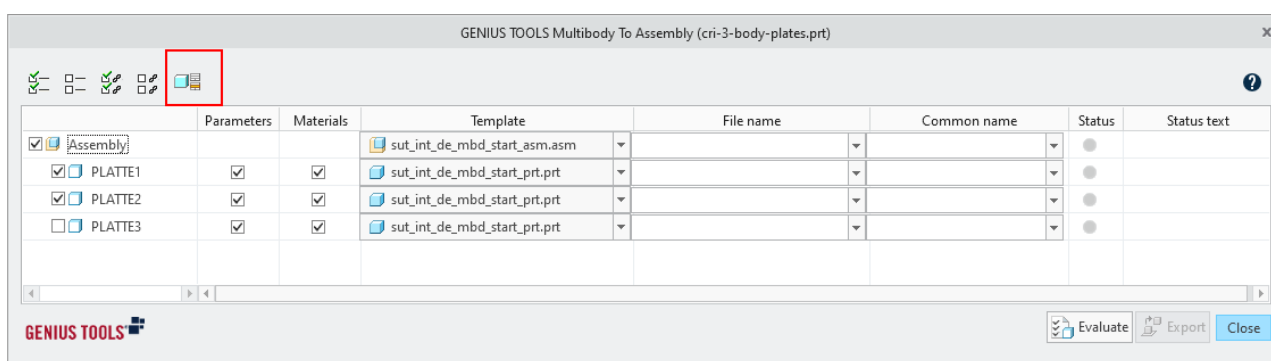
Typical process for initial creation / export

To create a new assembly, open a part with several bodies and start GENIUS TOOLS Multibody To Assembly via the ribbon menu in the GENIUS TOOLS tab.

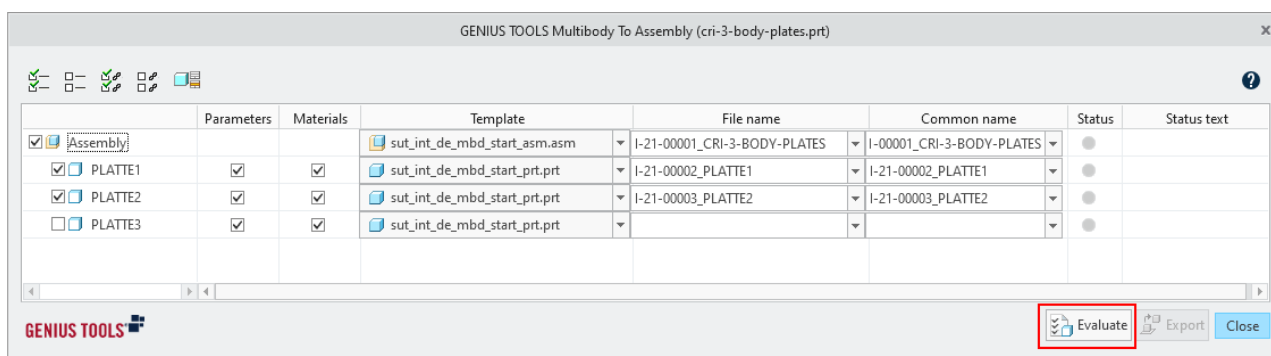
Select the bodies to be exported.



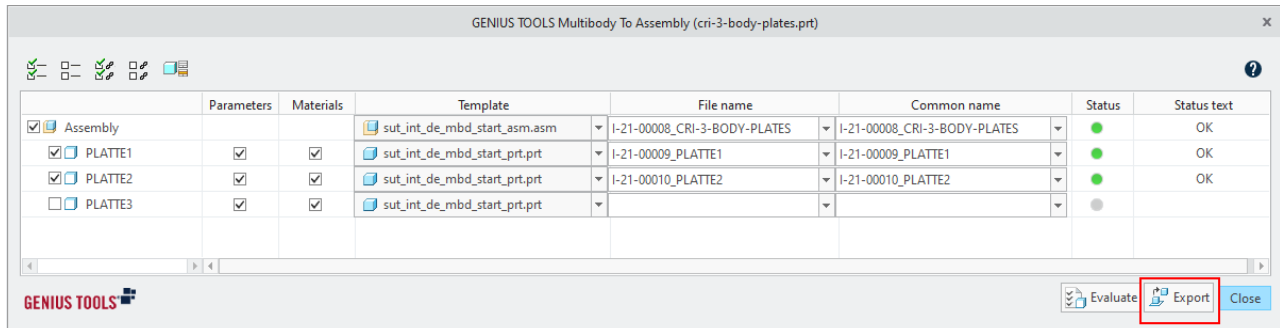
If you press the *Generate export name* button  the names are generated in the *File name* and *Common name* column.



Then click the *Evaluate* button below.



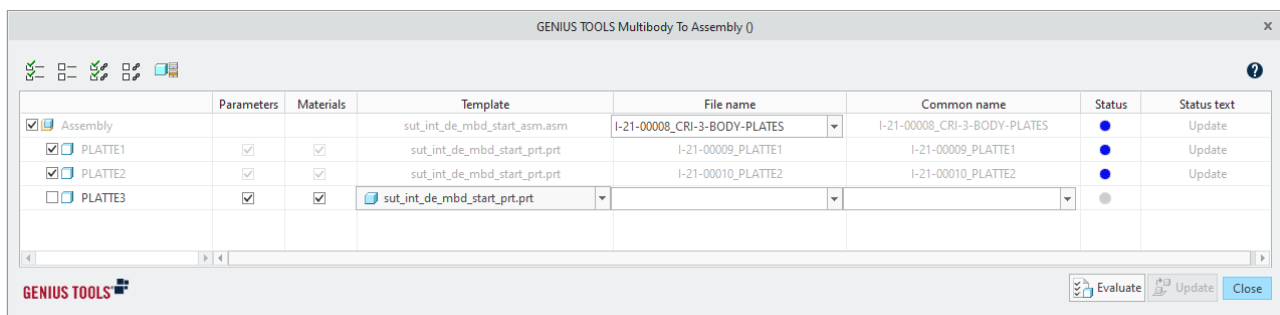
The colored status message and the associated status text appear. If the status is *green* = OK, the new assembly can be created using the *Export* button. The dialog closes automatically.



Typical sequence of the update process

In this mode, assemblies exported with GENIUS TOOLS Multibody To Assembly can be expanded with new parts.

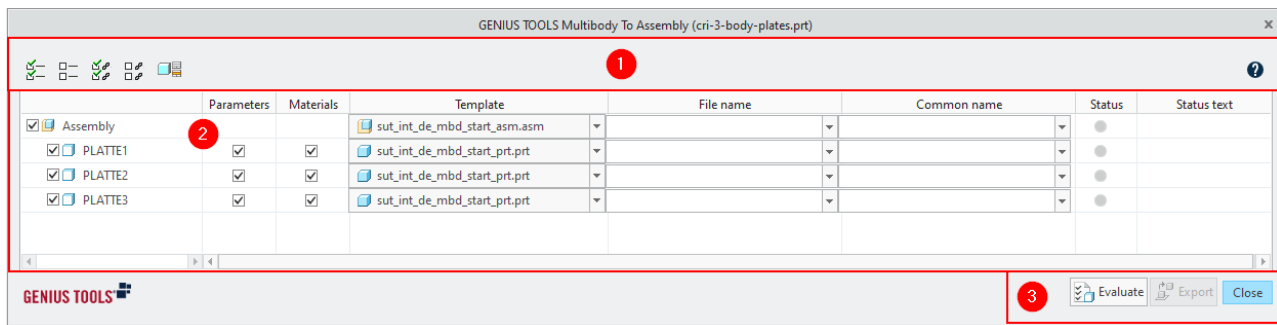
Call GENIUS TOOLS Multibody To Assembly again from the ribbon menu after you have created a new assembly. GENIUS TOOLS Multibody To Assembly now starts in update mode. You can now make further changes / adjustments by selecting the desired parts. The update mode can be called up from the part created or from the assembly.



If you want to exit the update mode, delete the entry in the *File name* field, confirm with enter or click in the user interface. The dialog is closed.

11.1.1 User interface

The GENIUS TOOLS Multibody To Assembly user interface consists of the following elements:



1. Command bar²⁹⁷ with Help button
2. Body export settings²⁹⁸
3. Buttons for Evaluate, Export³⁰⁰ and Close



11.1.2 Command bar

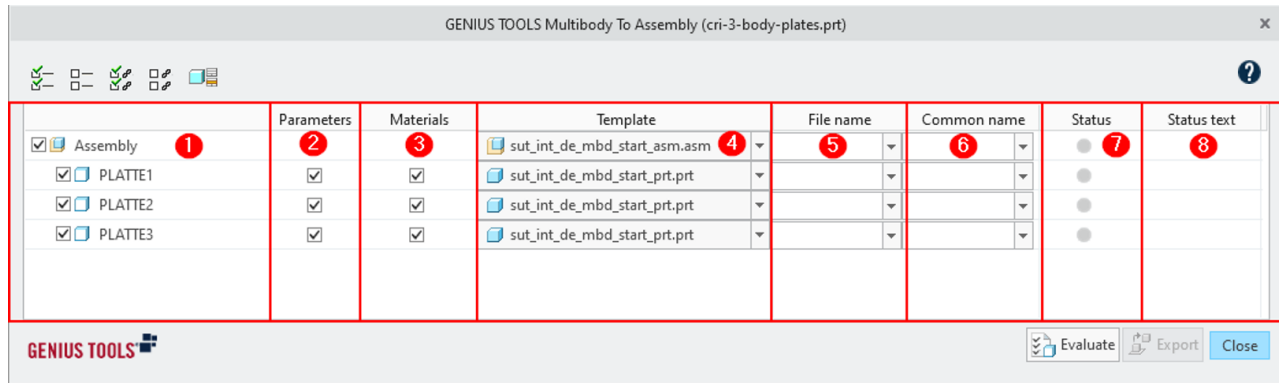
The GENIUS TOOLS Multibody To Assembly command bar consists of the following elements:

Symbol	Name	Description
	Check all elements	Selects all bodies of the assembly group.
	Uncheck all elements	Deselects all bodies of the assembly group.
	Check all referencing bodies	Selects the referencing bodies (patterned, copied etc.)
	Uncheck all referencing bodies	Deselects the referencing bodies (patterned, copied etc.)
	Generate names for elements without a name	Generates the export name for fields, where no export name was set.

11.1.3 Body export settings

1. Selecting the bodies

Select with the checkboxes which bodies shall be exported. The assembly group cannot be deselected. With the help of the button  you can select all bodies. With the button  you deselect all bodies.



2. Parameters

With these checkboxes the parameter export of the body is activated / deactivated. The default can be set with *gtmba_parameter_checked*.

The transfer of parameters to the exported part works in two steps. First, the parameter values of the source model are transferred and then the body parameters. Non-existing parameters are created. Parameters that already exist are given a new value. If the parameter types or units differ, there is an output in the message area. Parameters beginning with PTC_ are not transferred. The parameter sets of model and body can be influenced by the configuration options *gtmba_part_parameter_export_rule* and *gtmba_body_parameter_export_rule*.

gtmba_part_parameter_export_rule

Defines the rule for determining which part parameters are to be exported.

gtmba_body_parameter_export_rule

Defines the rule for determining which body parameters are to be exported.

Examples of values for the configuration options:

Configura- tion value	Description
^.*\$	All parameters are transferred
^\$	No parameters are transferred

Configura- tion value	Description
<code>^((Parameter3) (Parameter4)).*\$</code>	Parameter1 and Parameter2 are transferred
<code>^(?! (Parameter1) (Parameter2)).* \$</code>	Parameter1 and Parameter2 are not transferred

Exemplary task:

No model parameters are to be transferred. The parameters DESCRIPTION_1_DE and DESCRIPTION_1_EN are to be transferred from the body.

```
gtmba_part_parameter_export_rule=^$
```

```
gtmba_body_parameter_export_rule=^((DESCRIPTION_1_DE)|(DESCRIPTION_1_EN)).*$
```

Tip: Body parameters can be maintained with GENIUS TOOLS parameters.

3. Materials

These checkboxes activate / deactivate the material export of the bodies. The default can be set with *gtmba_material_checked*.


The body material becomes the model material if the checkbox is activated.

4. Template

Select the templates that will be used to create the assembly and parts. By default, the settings of Config.pro options *template_solidpart* and *template_designasm* are used.

Specific start templates can be defined with the options *gtmba_start_model_dir*, *gtmba_template_solidpart* and *gtmba_template_designasm*.

5. File name

The file name can be assigned manually. If using the button  *Generate names*, the names are created by GENIUS TOOLS Name Generator. By default, @number@@oldname@ is used.


@number@: replaced by GENIUS TOOLS Name Generator

@oldname@: name of the body

A specific name generator can be defined with *gtmba_gtng_filter*.

Example: *gtmba_gtng_filter=body_file_name*

6. Common Name

The file name can be assigned manually. If using the button  *Generate names*, the names are created by GENIUS TOOLS Name Generator. By default, @filename@ is used.

@filename@: file name created under point 5

But also @number@ and @oldname@ are possible.

Example: gtmba_gtng_common_name_rule=@number@@oldname@

7. Status

Colored display of the bodies status:


-  = Undefined
-  = Update
-  = OK
-  = Warning
-  = Error

8. Status text

The status text shows the colored status in text form:

-  = Undefined
-  = Update
-  = OK
-  = Warning

- File (*.prt) already exists

-  = Error

- Circular reference
- Empty name
- Duplicated name
- Existing file suffix (*.asm)

11.1.4 Evaluate, Export and Update

1. Evaluate

Before exporting, it is necessary to evaluate. Click on the button *Evaluate*

The evaluation checks:

- whether the File name and / or Common name is filled and already used
 - assembly: cannot be created
 - part: will be inserted instead of creating a new part
- whether the template can be opened

2. Export / Update

The assembly will be opened after the export and the Multibody To Assembly dialog is closed.

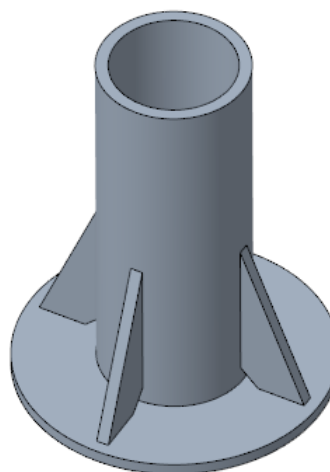
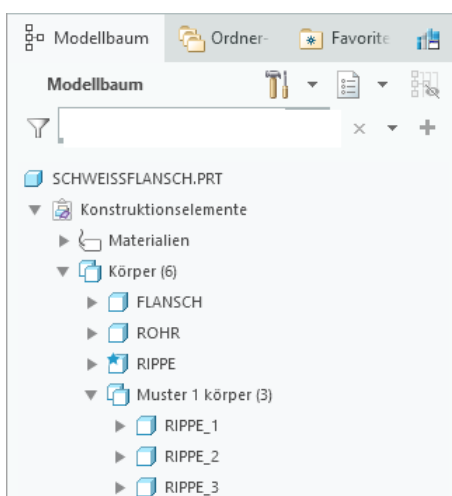
Please note: After executing Export / Update, the export settings are written into the multibody part and the created assembly.

11.1.5 Use Case

This chapter describes the application of GENIUS TOOLS Multibody To Assembly as an example.

11.1.5.1 Export of referenced bodies

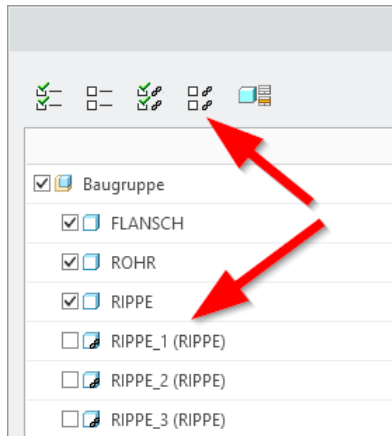
In this example, a multibody model with referenced (identical) bodies is to be exported. Identical bodies can arise when a body is patterned, copied or mirrored and no further changes have been made to the variants. Identical bodies are to be represented by the same part in the target assembly. GENIUS TOOLS Multibody To Assembly can only recognize referenced bodies. Only the user can decide whether further changes have been made to the referenced bodies.



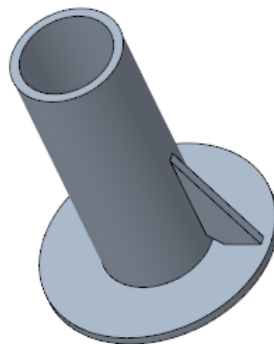
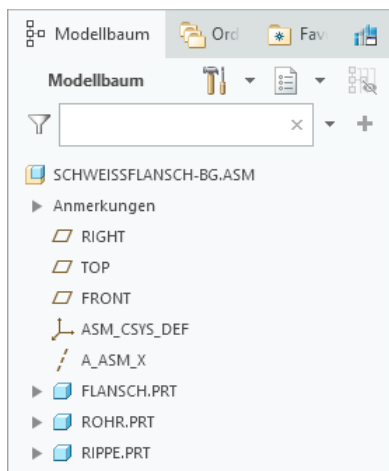
In a "real" assembly, the rib exists once and is then patterned.

Initial assembly generation with identical ribs

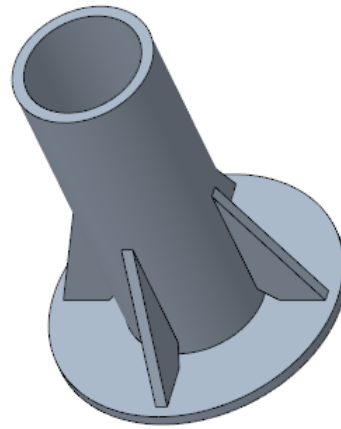
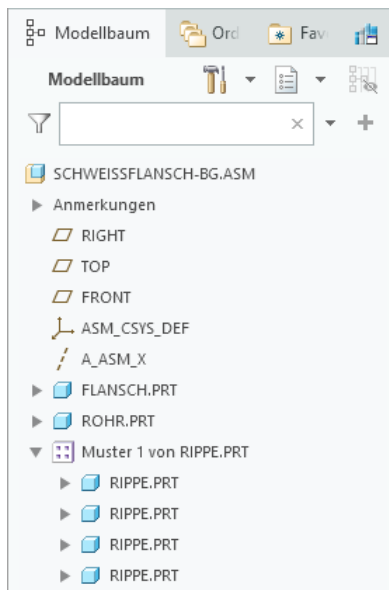
1. Open GENIUS TOOLS Multibody to Assembly.
2. Select only the original bodies of the identical bodies and export them. Referenced bodies can be recognized in GENIUS TOOLS Multibody To Assembly and selected or deselected together.



3. After the export you will see the original part as well as the corresponding model tree.

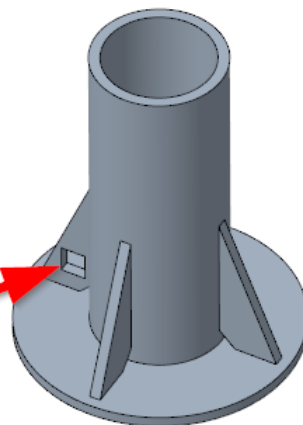
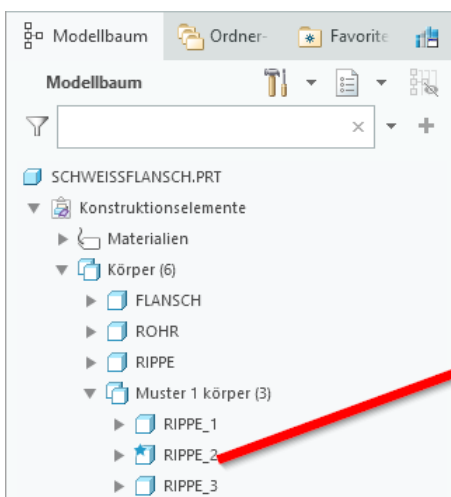


4. In the assembly, duplicate the source part according to the body definitions.

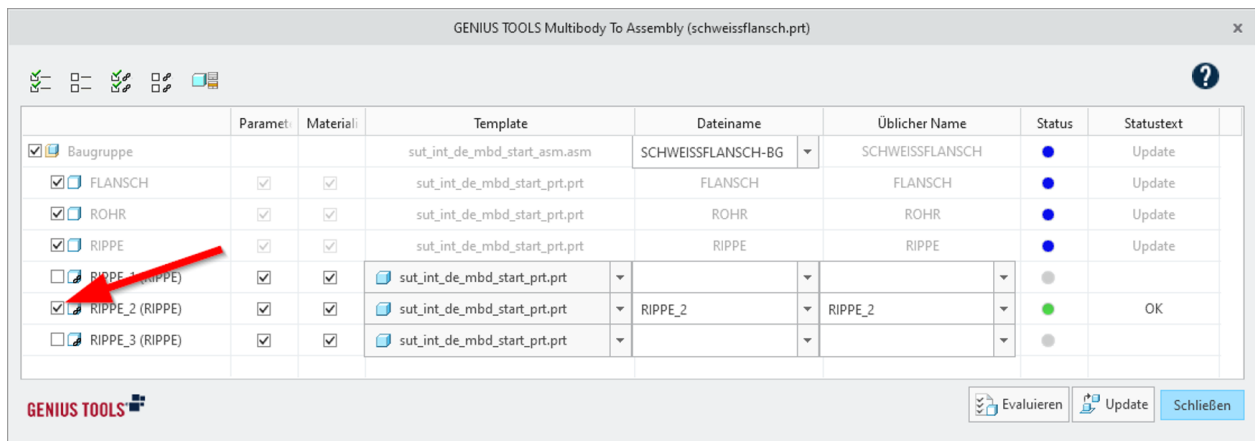


Update of an assembly

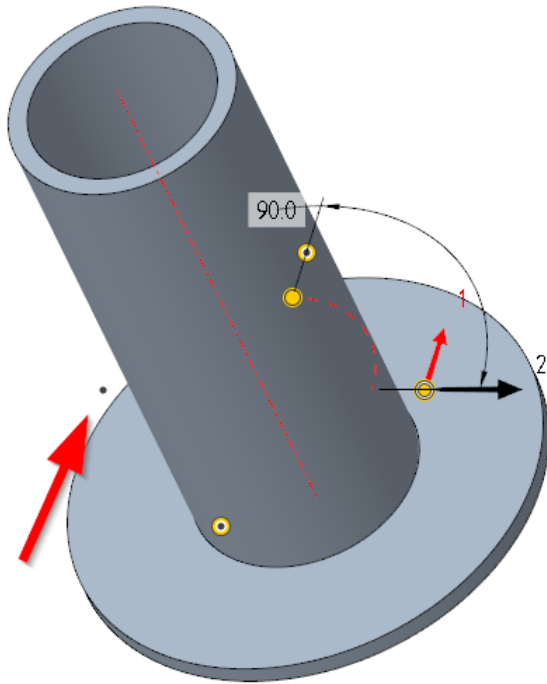
A breakthrough is now to be added to the assembly (see red arrow).



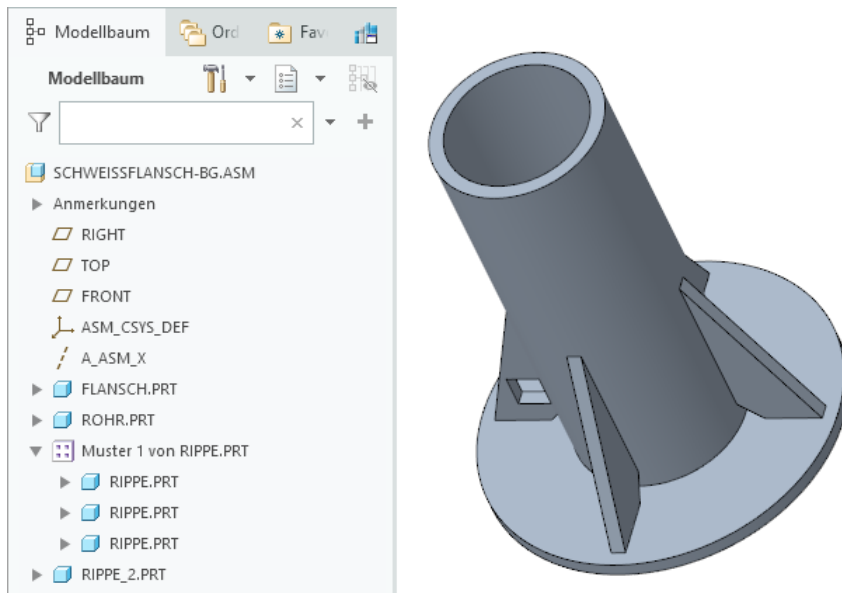
1. Open GENIUS TOOLS Multibody To Assembly. If an assembly has already been exported, the module starts by default in the update mode.
2. Select the relevant rib and update it.



3. In the assembly, there is now a collision of the pattern variant with the new part. You must now redefine the pattern. By redefining, the colliding variant can simply be hidden (see red arrow).

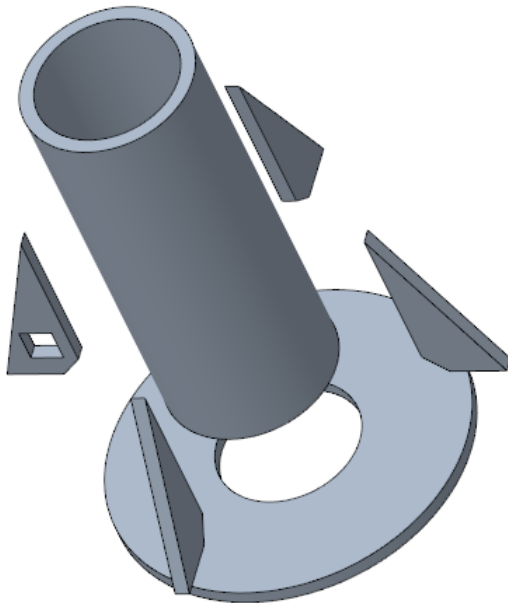


4. The result looks like this in the model and model tree:



Advantages of the assembly

The assembly can now be used like any "classically generated" assembly. This means, for example, parts lists can be derived or exploded views can be generated.



12 Name Generator

Creo Parametric is only to a certain extent able to ensure the continuous naming of individual objects. Especially if no product data management solution is in use in a company, it is difficult to achieve consistent and logical numbering.

GENIUS TOOLS Name Generator is an intuitive tool for sequentially assigning names with a numbering for file names of individual parts, sheet metal parts and assemblies. However, the program can also be used for any other files and objects for the construction. In this way, you ensure unique assignment and enable sequential numbering in accordance with working guidelines.

Name Generator can be used individually on stand-alone workstations (locally) as well as in a network (globally) for all Creo workstations.

GENIUS TOOLS Name Generator is available with following features:

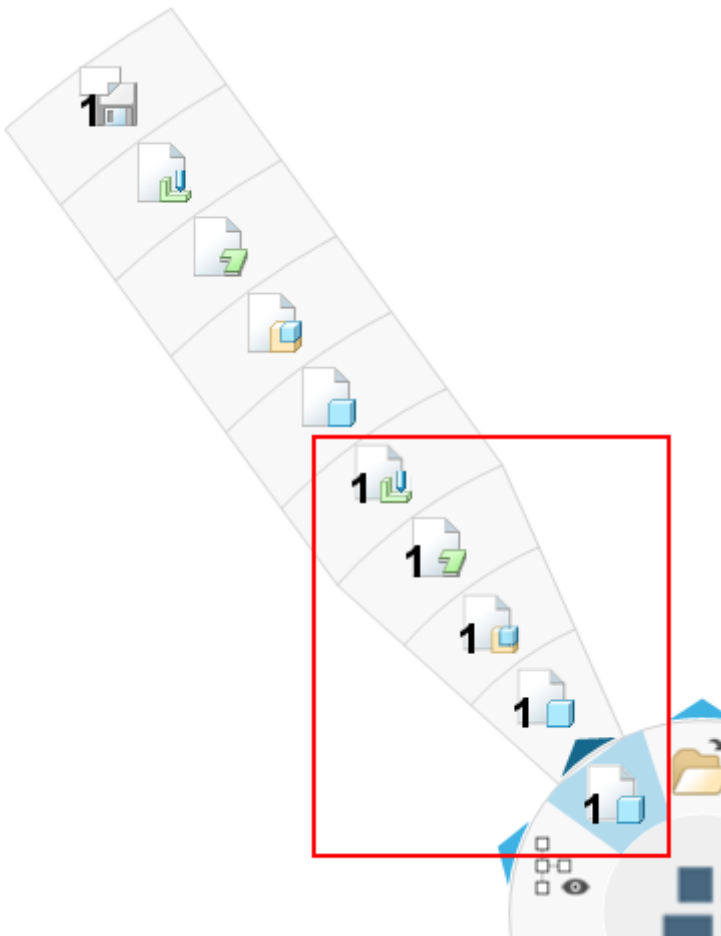
1. Usable in intelligent Mapkeys in the GENIUS TOOLS Modules [Quick Access](#)¹⁸¹, [Library](#)¹⁸¹, [Forms](#)⁸⁵, [Parameter](#)³²⁷ and [Multibody To Assembly](#)²⁹⁴.
2. Manual application in various design steps when
 - assigning names for family table variants,
 - creating parts in assemblies.

12.1 Usage

Starting the program to create new models

The main application of the Name Generator is the generation of filenames for new assemblies and parts.

In Quick Access ([<] key), the buttons for creating models with Name Generator are already active in standard configuration. You can identify those buttons by the small number One.



One click creates models of the specified type with the preset name configuration. The Creo default dialog for creating new models is opened. Instead of the typical Creo proposal a generated name proposal is pre-entered.

Please note: Make sure not to change the model type afterwards. (Additions within the scope of maximum name length may be reasonable.) Different name configurations for generating the model names can be deposited.

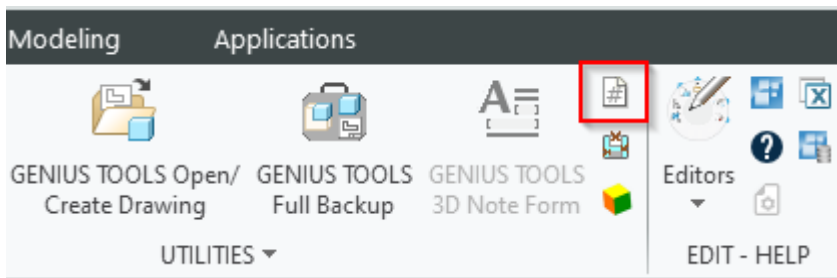
Automated application in other GENIUS TOOLS modules

GENIUS TOOLS Name Generator is embedded in the GENIUS TOOLS for Creo standard configuration for the following modules and can be used automatically for the creation of new parts, assemblies or drawings.

- In GENIUS TOOLS Quick Access and GENIUS TOOLS Forms the Name Generator is used with customized smart mapkeys.
- In sample data of GENIUS TOOLS parameter the name Generator is included.
- Via the GENIUS TOOLS Library sample libraries, new parts are created with Name Generator.
- GENIUS TOOLS Multibody To Assembly²⁹⁴: Creation of parts from an assembly group.

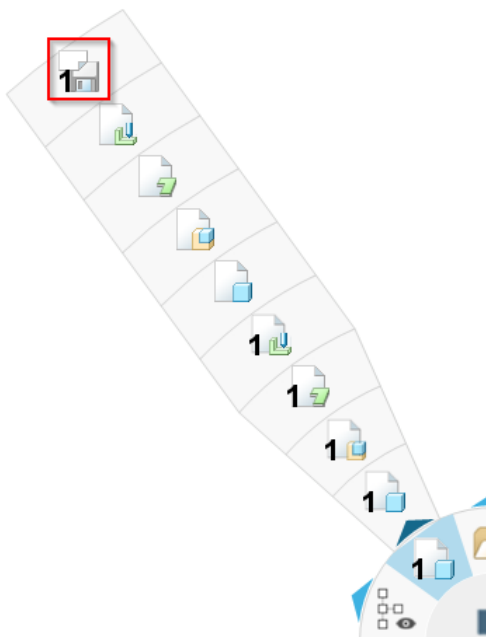
Special case: Starting the program from the ribbon menu

GENIUS TOOLS Name Generator can be started from the Ribbon menu in the tab GENIUS TOOLS. This is necessary if you generate a number independent from the module and want to paste it via the clipboard. For example to paste the generated name in family tables.

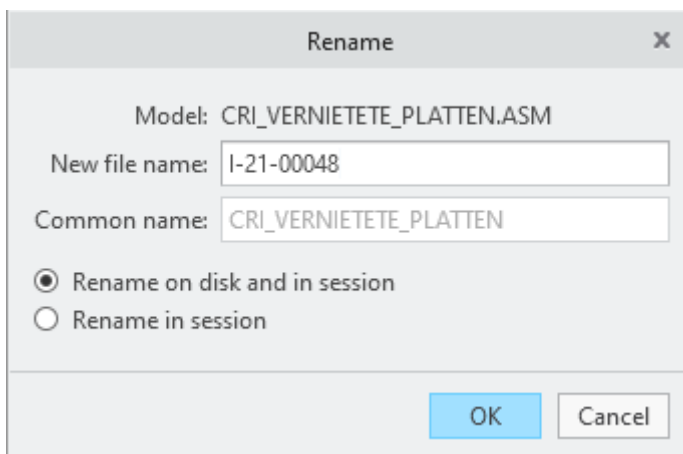


Renaming models using Quick Access

To rename the current model use the button *Rename current model* in the Quick Access menu.



Clicking this button opens the Name Generator.

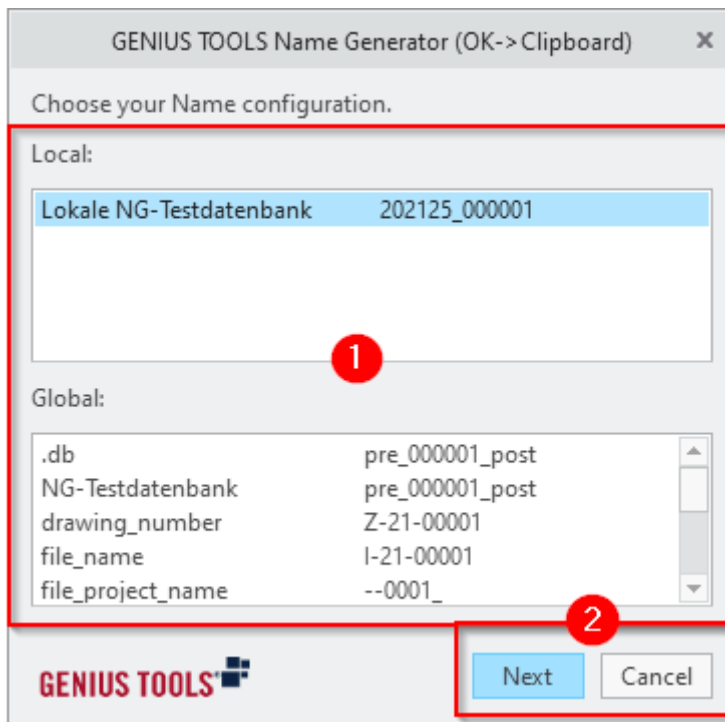


Select the required name configuration and confirm the dialog. The Creo default dialog for renaming is opened with the newly generated file name pre-entered.

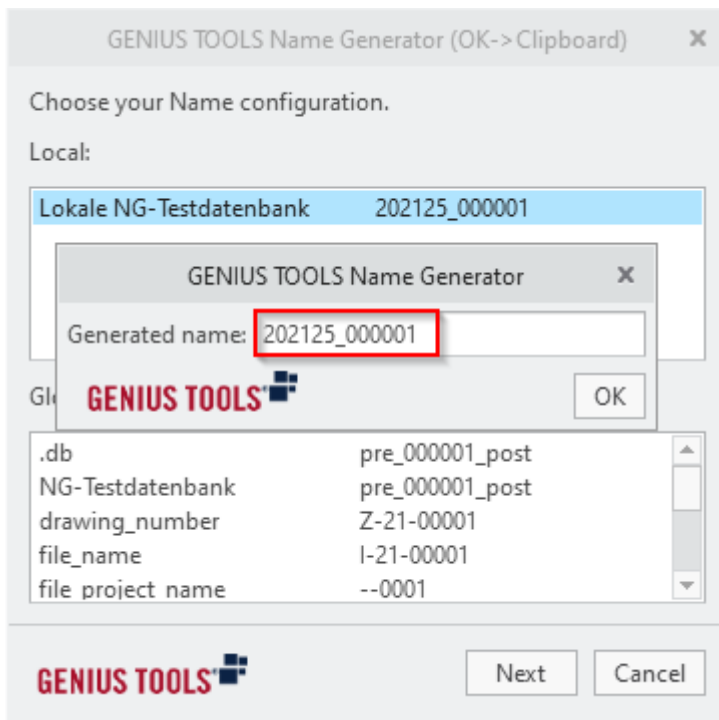
12.1.1 User interface

The user interface of GENIUS TOOLS Name Generator consists of the following elements:

1. Local and global name configurations
2. Buttons for *Next* and *Cancel*.



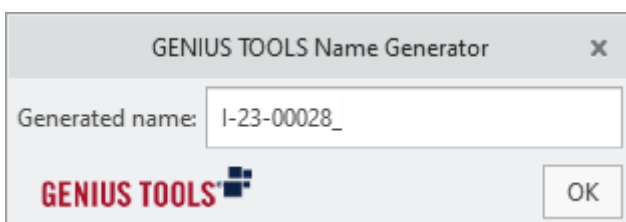
A generic preview is displayed behind each name configuration. Select one of the available name configurations (1) and confirm your selection (2). Customize the generated name with your own entries in the next dialog and confirm the dialog.



Customize the generated name if necessary. After clicking **OK**, the name is in the buffer memory. Use the generated name via **CTRL+ V** or the context menu (*right-click -> Paste*).

Controlling the user interface display

The appearance of the user interface depends on the `gtng_db_name_filter` configuration option. This configuration option allows you to specify values by which name configurations are searched. Then only the name configurations that match the stored value range are displayed. If you have stored a value in the `gtng_db_name_filter`, only the name configurations that match or correspond to this value are displayed. If exactly one name configuration corresponds to the value range, this name is already suggested. The user interface then opens in this form:



12.2 Configuration

In this section you will find information about the configuration of the GENIUS TOOLS Name Generator module, which is done in GENIUS TOOLS Name Generator Editor³¹².

12.2.1 Name Generator Editor

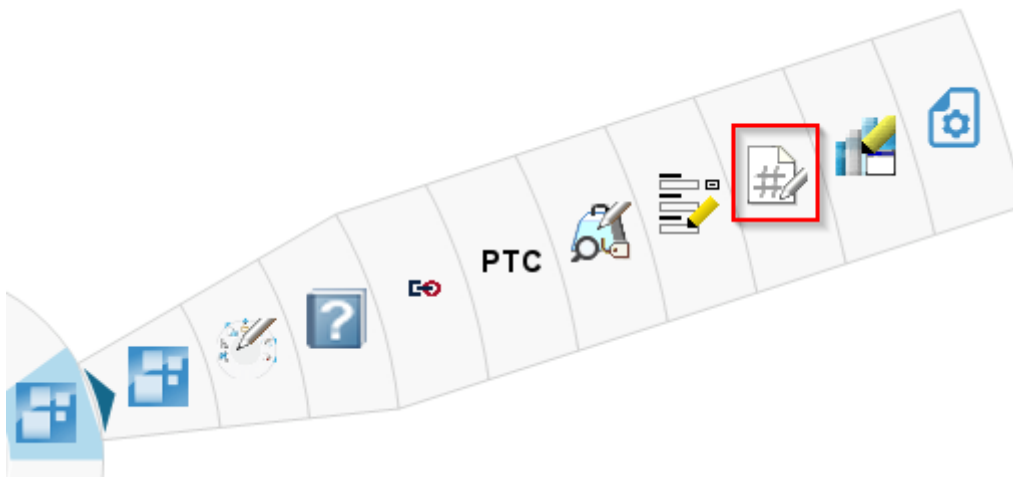
GENIUS TOOLS Name Generator Editor is used to create name configurations and to save them either locally or globally.

- Local name configurations are saved locally in the appdata directory of the current user.
- For users of Startup TOOLS global configurations are saved in the Main server *caddepot\serveronly*. (If you use GENIUS TOOLS for Creo without Startup TOOLS, you can find the global configuration in the resource folder.)

Warning: Global name configurations require read and write access to the files for all users.

Starting the program

Start GENIUS TOOLS Name Generator Editor via GENIUS TOOLS Quick Access ([<] key).



Saving changes

The changes made in the editor are saved in the resource directory.

When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer (Caddepot). Path: `<operatingenvironment>/parametric/configuration/gt_resource_folder`.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

12.2.1.1 User interface

The user interface of GENIUS TOOLS Name Generator Editor consists of the following elements:

The screenshot shows the 'GENIUS TOOLS Name Generator Editor' dialog box. It has a title bar with a close button (X). The main area is divided into two columns. The left column has a 'Local:' section with a text box containing 'Local Test' and a 'Global:' section with a list box containing 'test_windchill', 'drawing_number' (selected), 'file_name', 'file_project_name', and 'ident_number'. The right column has a 'Generate number from:' section with radio buttons for 'Database', 'Server' (selected), and 'Creo / Windchill'. Below this are input fields for 'Server URL:', 'Current counter:' (1), 'Minimum value:' (1), 'Maximum value:' (99900), 'Number of digits:' (5), and a 'Template:' field with the value 'Z-\$\$sub\$\$2\$\$4\$@date@\$\$-@number@'. At the bottom, there are '+', '-' buttons, a 'Save and preview' button (with a glasses icon), and a 'Close' button. Red circles with numbers 1, 2, and 3 highlight the 'Global:' list, the 'Generate number from:' section, and the 'Save and preview' button respectively.

1. Available name configurations ³¹³
2. Configuration details ³¹⁴
3. Save and preview. Click on the glasses-symbol to see a preview of the generated file name.

12.2.1.2 Available name configurations

Existing name configurations are displayed on the left side of the dialog. Click an entry in the lists to display the associated options in the configuration details. Use the plus button to create new name configurations. Select a name configuration and click the minus button to delete it.

Local and global name configurations

In the configurations chose either the option for local name configurations (`gtng_local_folder`) or the option for global name configurations (`gtng_folder`).

For different usage scenarios, there are local and global name configurations.

Local name configurations are suitable for constructions in which no other persons are involved or which are edited exclusively with one user account on one workstation.

Local name configurations, by default, are saved under %appdata%\INNEO\GENIUS_TOOLS\for_Creo\name_generator on a local computer.

Tip: Customize the storage location of local and global name configurations with the GENIUS TOOLS for Creo configuration options.

Global configurations are stored by default in the main server under caddepot\serveronly\gt_name_generator and are applicable for all users.

In order to create new names with the current counter, there always must be an active network connection.

Warning: Global name configurations require read and write access to the files for all users.

12.2.1.3 Configuration details

The current configuration of the selected element is displayed in the configuration details.

1. Generate number from

Specifies whether names are generated from the name generator databases or a Webserver.

If you are working with Windchill, use the option *Creo / Windchill*. This option lets you request numbers from Windchill without installing a customization on the Windchill server. The functionality uses the currently connected server and the current Creo user.

2. Second line

- **If Database is selected: no specifications possible**

- **If Server is selected: Server URL (2.)**

The URL of the Webserver. The URL must return a number. This number is adopted into a name configuration instead of @counter@. The URL only has to be specified when you are using the Server option.

- **Selection Creo / Windchill: filter**

Specify whether you want Windchill to number parts (PRT), assemblies (ASM) or drawings (DRW).

GENIUS TOOLS Name Generator Editor

Generate number from: ☐ Database ☐ Server ☒ Creo / Windchill

Filter (PRT,ASM,DRW):

Current counter:

Minimum value:

Maximum value:

Number of digits:

Template:

The filter only has to be specified when you are using the *Creo / Windchill* option to address a specific Windchill-number generator. If nothing is selected, PRT (part) is used automatically. Ignore this setting if numbers are generated from databases.

3. Current counter

Displays the current counter of the name configuration. If necessary, the counter can be customized in this field.

Warning: The current counter must at least equal the minimum value.

4. Minimum value

The minimum value is a name configuration's start value. Enter the smallest number for name generation.

5. Maximum value

The maximum value is the last number that can be assigned in the name generation.

6. Number of digits

Specifies the number of digits for number output.

7. Template

Specifies the name configuration. Use the variable @counter@ for upwards counting name configurations.

Warning: In older GENIUS TOOLS for Creo versions @number@ was used for the counter. To ensure compatibility, the variable can still be used.

You can additionally use all variables and string operations of GENIUS TOOLS for Creo. See the [variables list](#). All GENIUS TOOLS for Creo variables can be used in name configurations as desired. However, filenames are limited to 31 characters. Where variables return longer values, the generated name is truncated.

Sample for name generation

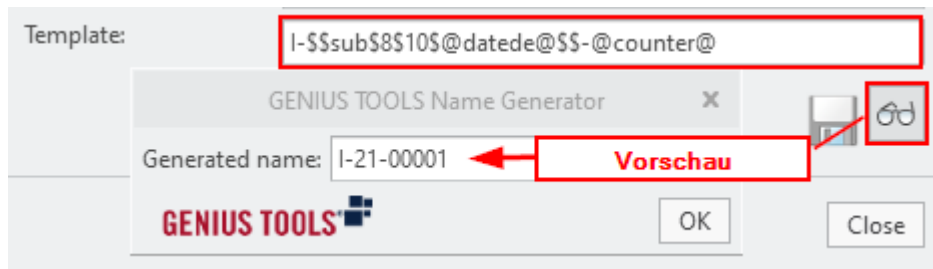
Enter the following variable: I-\$\$sub\$\$8\$10\$@datede@\$\$-@counter@ under Template.

\$\$sub\$\$8\$10\$: shows the 8th to 10th digit from the date format, whereas the 0 from the 8th digit is not displayed.

@datede@: date format dd-mm-yy

@counter@: generates a consecutive number

Result: The generated file name is: I-21-00001.



12.2.2 Use cases

In this section you will find use cases for the GENIUS TOOLS Name Generator module.

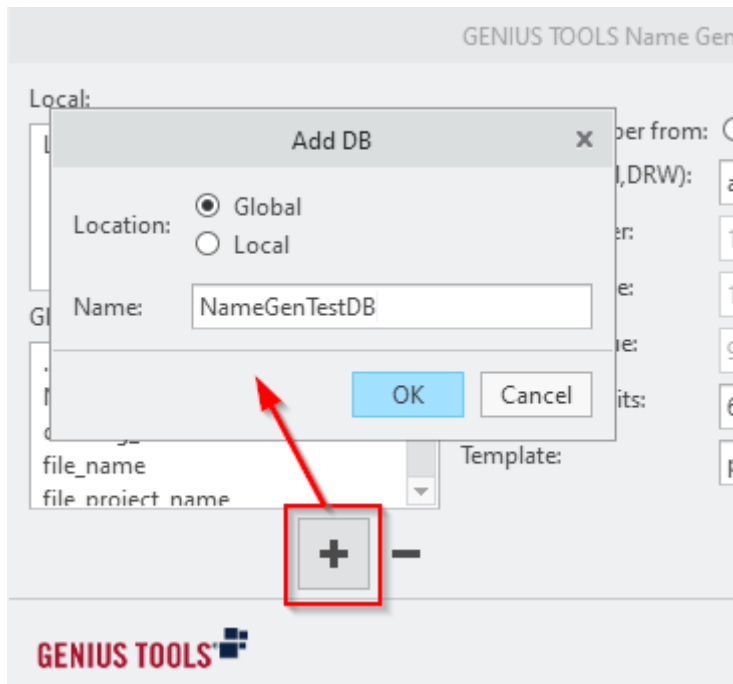
12.2.2.1 Global name configurations with fallback

A name configuration is created in this example. With this name configuration, a parameter is read from a model that is truncated and returned together with the counter. If the parameter does not exist in a model, a static fallback text is returned instead.

Proceed as follows to create a new name configuration with fallback:

1. Open GENIUS TOOLS Name Generator Editor.
2. Click the plus button in the command bar.

- Set the storage location to *Global*. Enter a name for the new name configuration. Make sure to use only special characters that can be used for filenames in windows.



- Confirm the dialog with *OK*.
- Select the newly created name configuration at *Global*.
- Configure the details of the name configuration.

GENIUS TOOLS Name Generator Editor

Generate number from: ☒ Database ☐ Server ☐ Creo / Windchill

Server URL:

Current counter:

Minimum value:

Maximum value:

Number of digits:

Template:

(0)

Close

7. Save the name configuration. The global name configuration with fallback is ready to use for all users.

Explanation of the created name rule

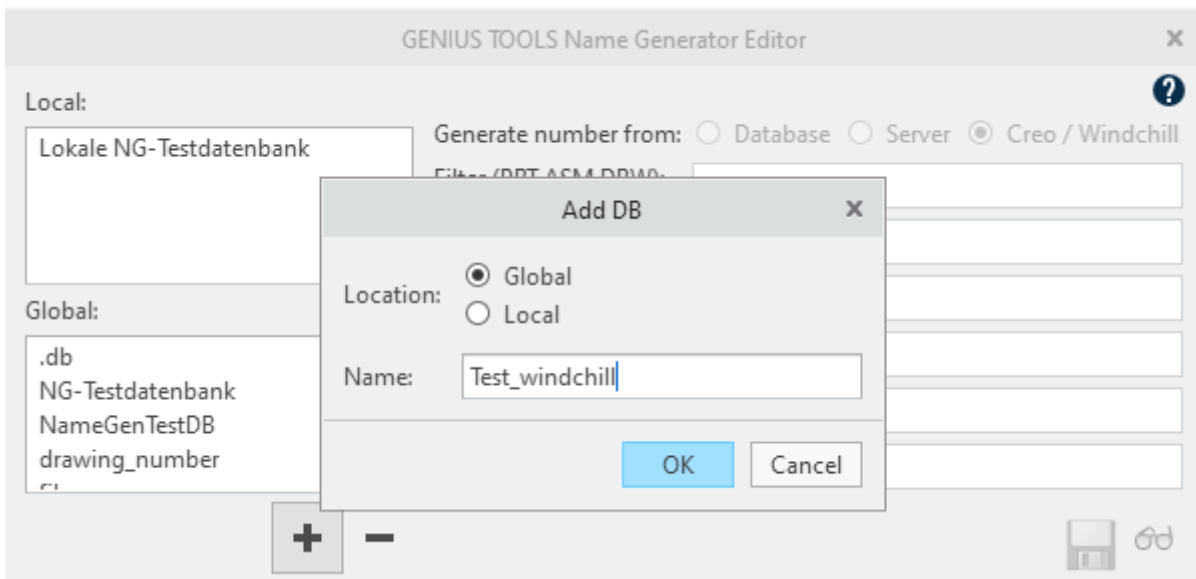
%PART_NO#NotAssigned%_@counter@

A parameter is read from the models and after the hyphen _ the counter is incremented. If no parameter is found, the static text NotAssigned is written in front of the counter instead.

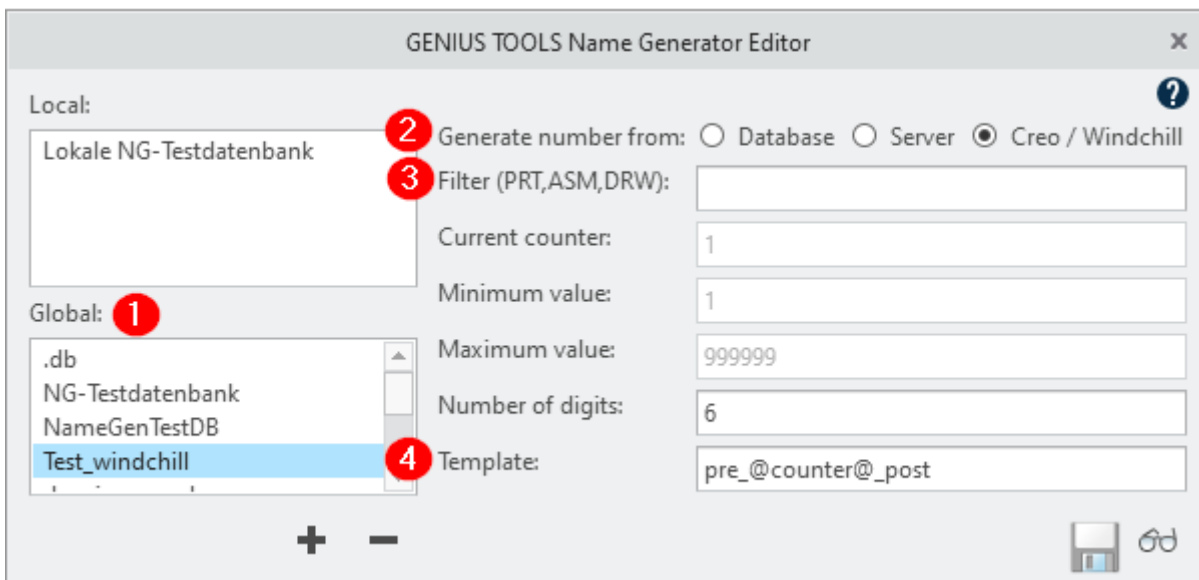
12.2.2.2 Global name configurations with Windchill

A global name configuration is created in this example. In the name configuration a number from a Windchill server is used.

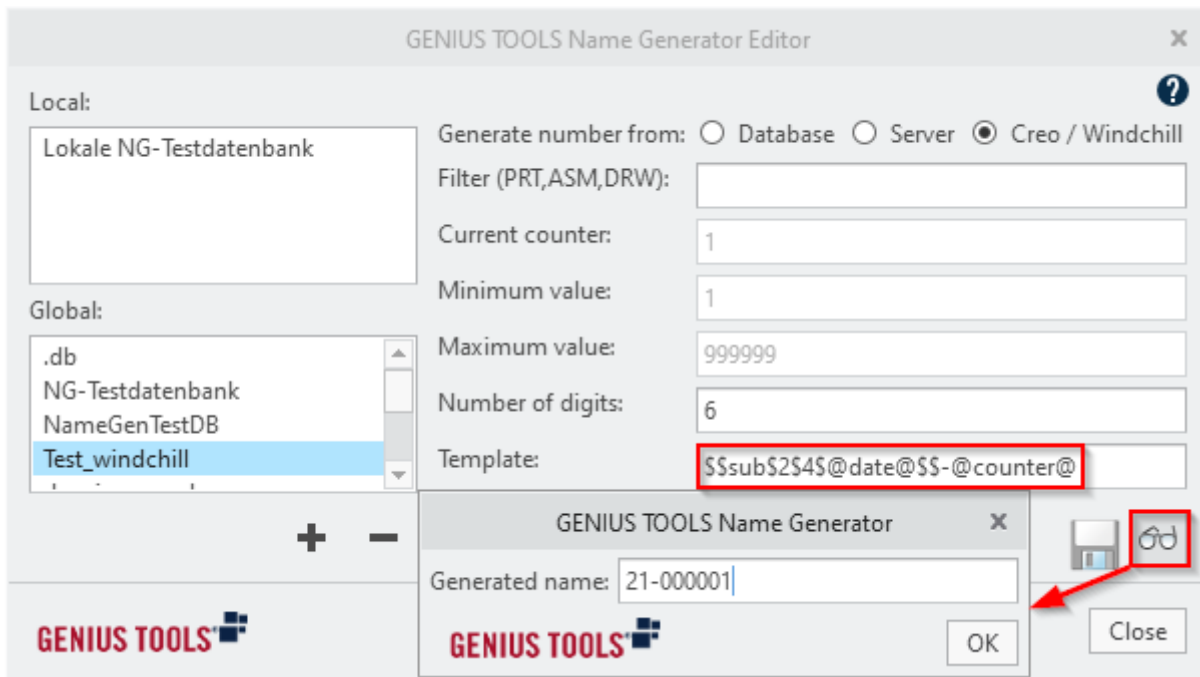
1. Open GENIUS TOOLS Name Generator Editor.
2. Select the Plus-button in the command line. A dialog opens.
3. Set the storage location to Global. Enter a name for the new name configuration and confirm the dialog with OK. Make sure to use only special characters that can be used for filenames in windows



4. Select the newly created name configuration under *Global* (1). Under *Generate Number from* (2) select the option *Creo / Windchill*.
5. Chose the *Filter* (3), select the specific Windchill-number generator. If nothing is selected, the Windchill-number generator for PRT (parts) is used automatically.
6. Under *Template* (4) an example is already entered (here: *pre_@counter@_post*), which can be customized.



7. Configure under *Template* (4) the name configuration to your requirements. Enter for example: `$$sub$2$4$@date@$-$@counter@`. In this operation (`$$sub$2$4$`) the 2nd to 4th digit is cut out from the date format `yyyy-mm-dd` (`@date@`). The 0 from digit 2 is not displayed. With `@counter@` a consecutive number is generated and attached.



8. With the preview-symbol (glasses) a window opens and shows the generated filename. In this example the filename is: 21-000001.

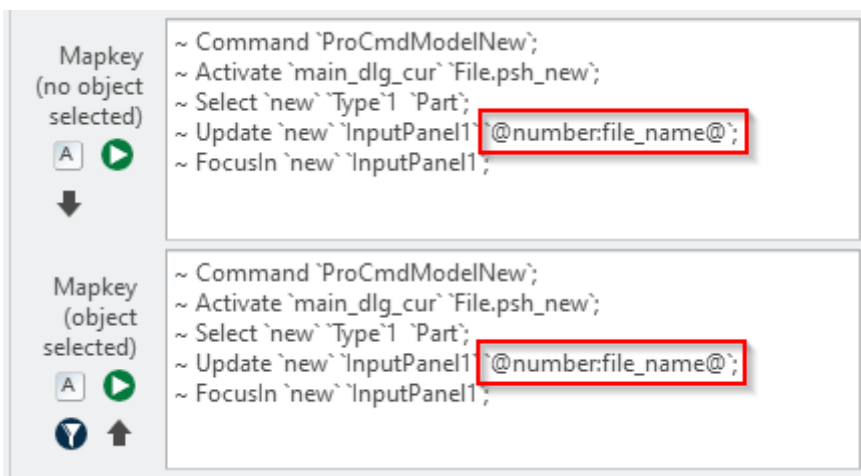
9. Save the name configuration (disc-symbol).

The counter @counter@ in the name configuration now uses Windchill's number generator.

Tip: Windchill name generators do not have to be on the global Mainserver.

12.2.2.3 Integrating name configurations in Quick Access

GENIUS TOOLS Quick Access uses intelligent Mapkeys extended with variables and operations that can only be used with GENIUS TOOLS for Creo.



Customize the Mapkeys in Quick Access (Quick Access Editor).

For being able to use any name configurations, the Mapkeys have to be customized in Quick Access.

The variable `@number:...@` needs to be customized in the predefined Mapkeys.

The Mapkey for *Create new part with number*:

```
~ Command `ProCmdModelNew`;
~ Activate `main_dlg_cur` `File.psh_new`;
~ Select `new` `Type`1 `Part`;
~ Update `new` `InputPanel1` `@number:file_name`;
~ FocusIn `new` `InputPanel1`;
```

Now customize the variable `@number:...@` in the Mapkey.

Variant 1: With name configuration selection

`@number@` opens Name Generator without pre-selected configuration. After selecting the configuration, the Creo dialog for creating parts is opened with a pre-entered name.

Variant 2: With defined name configuration

`@number:nameconfiguration@` directly opens the dialog for part creation. A filename – created with the entered name configuration – is predefined and can be customized.

Please note: If the name configuration is not unique, a name configuration will still have to be selected via the selection dialog.

12.2.2.4 Using project numbers

In this example, a project number with a consecutive number is to generate new names using the GENIUS TOOLS Name Generator. This new and unique name is used, for example, for new parts and assemblies.

Variant A: There is only one consecutive number for all project numbers.

Variant B: Each project number has its own consecutive number (number range).

Query: Where can the project number be obtained from automatically?

Method 1: Project number comes from a parameter of the current model

Set project number - initial assignment

The initial assignment is done manually, in which a defined model parameter receives the project number as a value. By changing the active model (there can only ever exist one in the Creo session), the value of the defined model parameter will also change, and thus with it the project number.

Advantage

If models from different projects are edited in one session, the "correct" new name is automatically created if the "correct" model is active.

Disadvantage

If there is no model in the Creo graphics window (e.g. in the start window), no project number can be determined.

Method 2: The project number is defined in the session environment (operating system variable)

Set project number - initial assignment

If a defined operating system variable is assigned a value before Creo is started, Creo Parametric must be restarted every time the project changes.

If the defined operating system variable is to be changed in the current session, you can define a Javascript-function, that changes the value (see below) or use GENIUS TOOLS Model Processor. By default, it is not possible to change an environment variable in a Creo Parametric session.

Advantage

The project number can be determined in any Creo Parametric situation.

Disadvantage

Additional software functionality is required to change the project number. When changing models from different projects, it is necessary to manually set the project number each time you change.

Solution for variant A: One counter for all project numbers

Method 1: Model parameter value as project number

The model parameter is specified in the GENIUS TOOLS Name Generator Editor.

GENIUS TOOLS Name Generator Editor

Local:

- project_envvar_name
- project_mdparam_name

Global:

- drawing_number
- file_name
- file_project_name
- ident_number
- prj0815
- nri4444

Generate number from: ☒ Database ☐ Server ☐ Creo / Windchill

Server URL:

Current counter:

Minimum value:

Maximum value:

Number of digits:

Template:

(0)

Close

Please note: The GENIUS TOOLS Name Generator can only generate a valid name if a part or assembly is active that also has the corresponding model parameters.

Create or change model parameter value

The first assignment or change of the model parameter value takes place via the parameter dialog of Creo Parametric or GENIUS TOOLS parameters.

Method 2: Operating system variable value as project number

The operating system variable is specified in the GENIUS TOOLS Name Generator Editor.

GENIUS TOOLS Name Generator Editor

Local:

- project_envvar_name
- project_mdparam_name

Global:

- drawing_number
- file_name
- file_project_name
- ident_number
- prj0815
- nri4444

Generate number from: ☒ Database ☐ Server ☐ Creo / Windchill

Server URL:

Current counter:

Minimum value:

Maximum value:

Number of digits:

Template:

(0)

Close

Please note: The GENIUS TOOLS Name Generator can only generate a valid name if an operating system variable with corresponding model parameters exists.

Generate, change or view operating system variable value

An operating system variable in a Creo Parametric session cannot be created, modified, or viewed using standard Creo Parametric functionality. You can, however, use the JavaScript function *SetVar()*, which creates or changes the value of an operating system variable, e. g. by using [GENIUS TOOLS JavaScript Editor](#)⁵⁷⁷.

```
function SetVar () {
    setEnvVar ("PROJECT_NO", "0000");
}
```

If you combine this with an „intelligent“ mapkey in GENIUS TOOLS Quick Access, users can enter a project number in a separate dialog window (1).

The display of an operating system variable value can also be implemented with an intelligent mapkey (2).

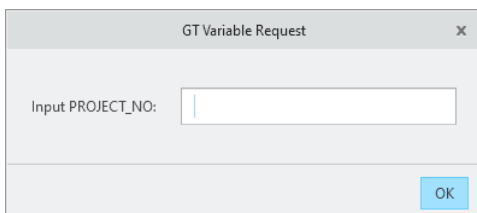
(1) Change operating system variable value by user input

You can create a mapkey or button in GENIUS TOOLS Quick Access that allows users to enter a project number which will be transferred to the operating system variable. (See also [Creating mapkeys](#)⁴¹⁷.)

A GENIUS TOOLS Quick Access command might look like this: (See also [Use of variables for user input](#)⁶⁸⁵ as well as [Javascript function](#)⁵⁸² *replaceVars ()*.)

```
js:var pn = replaceVars("$PROJECT_NO$");
var regexp = new RegExp("^[\\d]{4}$");
do {
    var pn = getStringUI(Input PROJECT_NO:);
    if (!regexp.test(pn)) {
        alert("The project number must consist of 4 numbers.");
    }
} while (pn != "" && !regexp.test(pn));
if (pn != "") setEnvVar("PROJECT_NO", pn);
```

After clicking on the defined mapkey or button in GENIUS TOOLS Quick Access, this dialog appears.

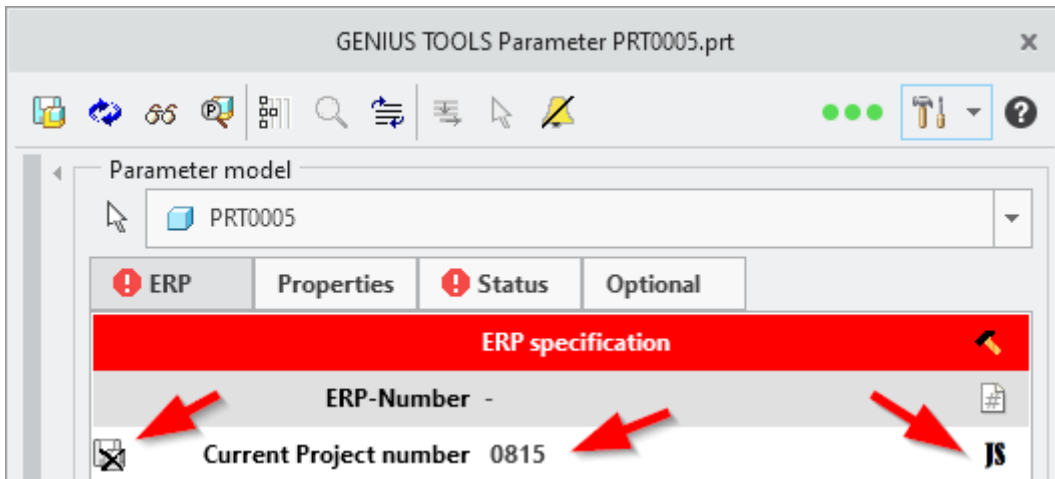


Alternative

A Javascript function can also be triggered in GENIUS TOOLS Parameter, e. g. when opening GENIUS TOOLS Parameters or after changing a model parameter value or as a click function on a parameter.

```
function PostLoad () {
    setInputValue ("PROJECT_NO_ENV", getEnvVar ("PROJECT_NO"));
    return true;
}

function SetVar() {
    setEnvVar("PROJECT_NO", getInputValue("PROJECT_NO_ENV"));
    alert("Project number "+getEnvVar("PROJECT_NO")+" set!");
}
```



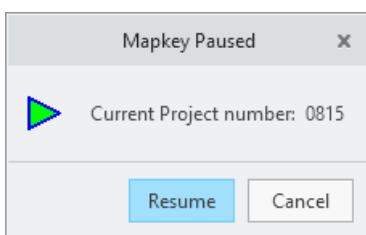
However, a model must always be available for this route. We recommend offering of offering both, GENIUS TOOLS Quick Access as well as GENIUS TOOLS parameters (click function) so that querying and setting project numbers can be carried out conveniently in different ways.

(2) Display operating system variable value

The following intelligent mapkey can be stored in GENIUS TOOLS Quick Access to display the current project number. Mapkeys can include variables by default.

```
@MANUAL_PAUSECurrent project number: $PROJECT_NO$;
```

The output then looks like this:



Solution for variant B: One counter for each project number

If each project number shall receives a counter, a GENIUS TOOLS Name Generator Definition (database) must be created for each project number (e. g. prj0815.db, prj4711.db, prj1234.db). It is irrelevant whether model parameter values or operating system variable values are used.

The manual creation and maintenance of these databases is very time-consuming. GENIUS TOOLS Name Generator offers the possibility to create these databases automatically. The principle works like this:

- Search for a project number counter in data base.
- If it does not exist, a new data base is created using a configured template file.

A template database can be stored with the configuration option

`gtng_copy_template_if_filter_db_not_found` (e. g. project number). The definition can look like the one shown in variant A.

When using GENIUS TOOLS Name Generator within GENIUS TOOLS Quick Access or in the copying rules of GENIUS TOOLS Library, you only need to specify the required generator:

```
@number:prj$PROJECT_NO$@
```

If this is not found, it is generated automatically.

Please note: If this technology is used, every user needs write rights in the directory of the configuration option `gtng_folder`.

Example: Count number for Javascript

If the count number of a project-specific number generator is required in Javascript, the function (e. g. for GENIUS TOOLS parameters) could look like this:

```
function GetProjectSubNo () {  
  var numgenname = getInputValue("PROJECT_NO");  
  setEnvVar("PROJECT_NO", numgenname);  
  var new_num = replaceVars("number:" + numgenname + "");  
  setInputValue("PROJECT_SUB_NO", new_num);  
}
```

13 Parameter

GENIUS TOOLS Parameter is a program to easily edit meta data (parameters) of parts, bodies, assemblies and drawings in Creo Parametric.

In assembly mode, the parameters of any sub-model can be edited. In drawing mode, you can switch between the drawing parameters and the parameters of the drawing models. Parameters to be edited (name, type, displayed text, etc.) are defined in the parameter definitions. They can be created and edited with the included GENIUS TOOLS Parameter Editor.

It is also possible to work with multiple parameter definitions. The parameter definitions that will be used, can be specified manually or automatically, depending on the parameters (selection parameters) of the active model.

GENIUS TOOLS Parameter is available in assembly mode, part mode and drawing mode providing the following features:

1. A classifiable meta data definition including
 - an auto-selection mechanism for parameter definitions
 - freely configurable groupings
2. Easy parameter creation
 - with default properties for all types
 - specification for effective ranges (parts, bodies, assemblies and drawings)
3. Easily editing of parameters by means of
 - free input
 - lists and tables (from files and databases)
 - auto-suggestion function
 - format checks
 - input dependencies
 - adoption of parameters from other Creo models
4. Individual model tree configuration

13.1 Fundamentals

This introductory section gives you an insight into the operating principles of GENIUS TOOLS Parameter. In addition, you will find a glossary of important terms for a better understanding.

13.1.1 Glossary

Selection parameter

The selection parameter is a Creo model parameter. This parameter and the GENIUS TOOLS Parameter parameter definition list are used to determine which parameter definition will be applied to a model. The parameter used as the selection parameter by default is `MC_CHECKTYPE`.

Field function

Field functions are predefined functions to determine parameter values in GENIUS TOOLS Parameter. They are displayed as buttons after the entry fields in the parameter form.

Field functions can set parameter values in different ways, e. g., set the current date or a model parameter value from Creo.

Parameters

Parameters in terms of GENIUS TOOLS Parameter define the display and behavior when assigning values in GENIUS TOOLS Parameter. Creo model parameters are driven depending on the properties of a parameter.

Parameter definition and sub parameter definition

A parameter definition contains parameters, separators and references to sub parameter definitions. A sub parameter definition contains only parameters and separators. The information is stored in an XML data structure.

Parameter model

The parameter model is the Creo model, whose parameter values are currently displayed in GENIUS TOOLS Parameter.

PTC Common Name

The parameter `PTC_COMMON_NAME` is treated like any other parameter in GENIUS TOOLS Parameter. Please note that this parameter is used by Windchill and must not be changed after a model has first been checked in to Windchill. It is recommended to make this parameter read-only.

The possibility to rename Windchill objects in Creo is also controlled by the configuration option `let_proe_rename_pdm_objects`.

Rolling

Rolling allows to pass incremental values stepwise in a Creo parameter chain. The value in the last Creo parameter will be discarded. The purpose of rolling is to obtain history of parameter values.

Separator

Separators are elements used to subdivide the parameters in the element list of GENIUS TOOLS Parameter Editor and the parameter sets in the parameter form in GENIUS TOOLS

Parameter. Separators are also needed for the two functions: *reset* and *execute field function*.

Table connection

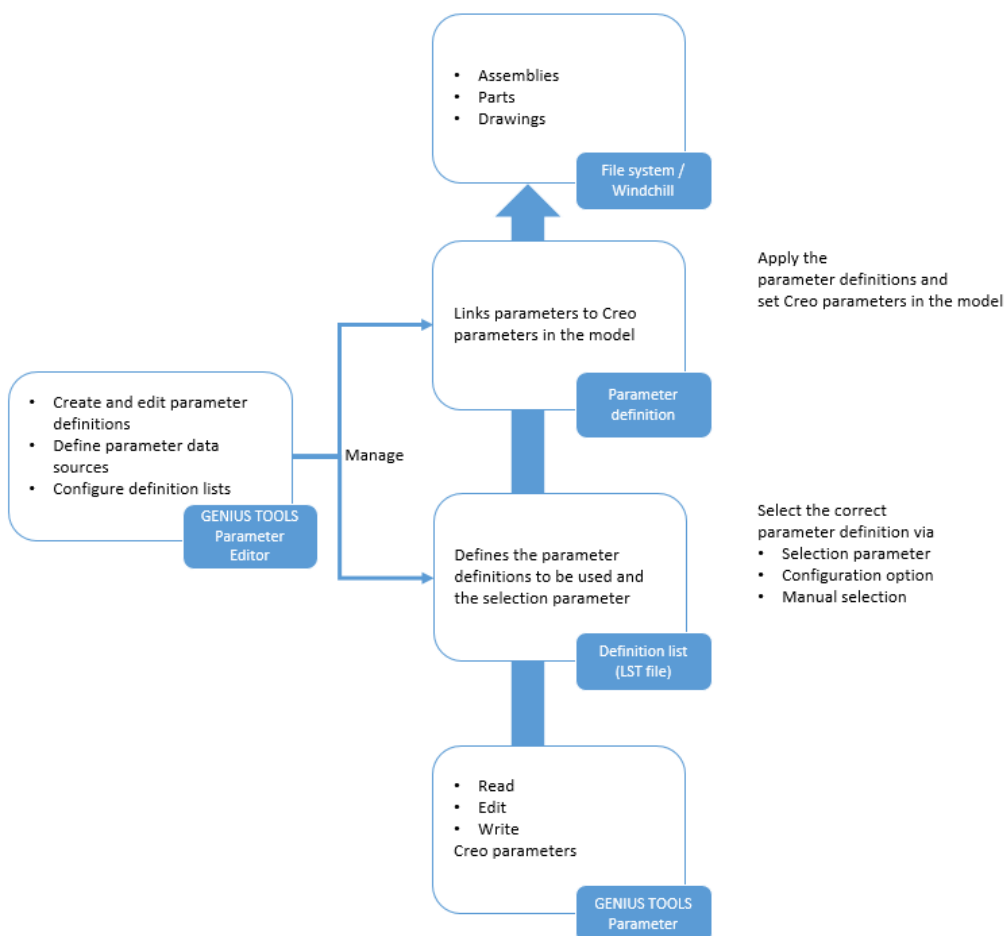
A table connection is the connection between the values of two parameters. Parameter values can either be controlled by another parameter or filtered.

13.1.2 Parameter management concept

One of the major advantages of working with a uniform database in Creo Parametric is that both geometric information and administrative information is pre-defined. The benefit of this is that the information for drawing header, bills of material, and position flags in assembly drawings only needs to be created and maintained once. Administrative information in terms of Creo Parametric are parameters that are saved along with a part, a body or an assembly. Examples of such parameters could be *Name*, *Description*, *Material*, *Weight*, and *Drawing number*.

When creating a drawing of a part or an assembly, the drawing header can be filled in automatically with these parameter values. When building assemblies, a current bill of materials can be generated at any time. The content of the BOMs (number and content of the individual columns) is freely configurable. The position flags on the assembly components can be automatically filled in with any information from these bills of material.

GENIUS TOOLS Parameter was developed to guarantee this uniform database. The program uses parameter definitions, which contain all usable parameters, to display them in a configurable view. For this purpose, a definition exists for each parameter. In addition to the parameter name, these definitions also contain localized descriptions and determine from which data sources (free input, text files, databases, etc.) the parameters can be filled. The parameters are displayed in a form and can be easily filled in by user input.



Steps for preliminary considerations on the parameter concept

- Analyze the existing title boxes
- Analyze the BOMs
- Analyze design details in the ERP

13.1.3 Mechanisms on starting and saving

The parameters that are shown and changed in GENIUS TOOLS Parameter are specified in parameter definitions. It depends on the configuration which parameter definition will be applied to a Creo model and which parameters will be saved to the models.

For a better understanding, this section outlines the general mechanisms on starting GENIUS TOOLS Parameter and when saving parameters.

Start mechanism of GENIUS TOOLS Parameter

1. Determines the parameter definition to be used (see below *Selecting the parameter definition*)

2. Reads parameters from the current model or assembly component and checks PDM status
3. Assigns parameters automatically as specified by the configuration of GENIUS TOOLS Parameter
4. Displays the parameters in the form section of GENIUS TOOLS Parameter
5. Output of status information (status indicator)

Selecting the parameter definition

Automatic selection based on a model parameter

Automatic selection based on a model parameter (selection parameter) is the default behavior of GENIUS TOOLS Parameter. A parameter in the Creo model is compared with a list of parameter definitions.

Important criteria for the selection based on a model parameter are the configuration options `gtp_lst` and `gtp_file_param`.

The configuration option `gtp_lst` specifies the location of the file containing all parameter definitions in list form. By default, this file (*gtp.lst*) is located at `<GTfCInstallation>\gt_resource_folder\parameter`.

The list contains all available parameter definitions in following format:

Location|Description|ParameterValue

Example: %GT_RESOURCE_FOLDER%\parameter\gtp_sut_int_de\pmm_int_de.xml|Production|PRODUCTION

The configuration option `gtp_file_param` specifies the selection parameter. If the value in the model parameter matches with a parameter value in the list, the corresponding parameter definition will be applied to the model.

Please note: If a list file contains duplicate parameter values, only the last entry will be applied.

The chapter [Auto-select parameter definitions](#)³⁹⁴ describes the procedure step by step.

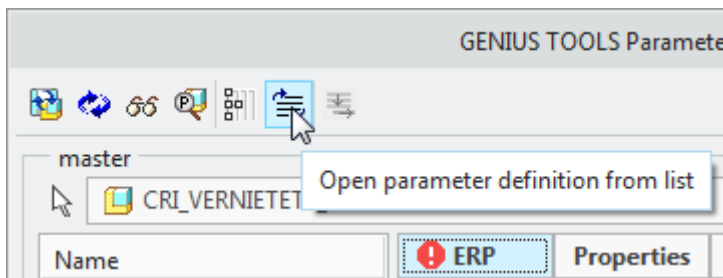
Automatic selection based on a configuration option

If the selection based on a model parameter returns no result, it will be checked whether a parameter definition has been specified via the configuration option `gtp_file`. If a valid parameter definition is returned, it will be applied to the model.

Manual selection

If the automatic selection does not return a valid parameter definition, or if another parameter definition should be used, parameter definitions from the parameter definitions list can also be applied manually to a model.

Make sure to not select a sub parameter definition when selecting manually!



You can also assign parameter definitions to a model manually with this button

Storage mechanism of GENIUS TOOLS Parameter

After changes in the parameter form in GENIUS TOOLS Parameter, the following mechanism will be applied when saving the parameter values:

1. Verification of the parameter values according to the rules defined in the parameter definition
2. Output of warning messages (depending on the configuration)
3. The values are adopted by the model parameters:
 - a. designations are removed (depending on the configuration)
 - b. revised parameter values are adopted into the model
 - c. designations are set
4. The model is being regenerated (depending on the configuration)
5. GENIUS TOOLS Parameter is completely reloaded with the new model parameters
6. If the option is selected, the model can also be saved.

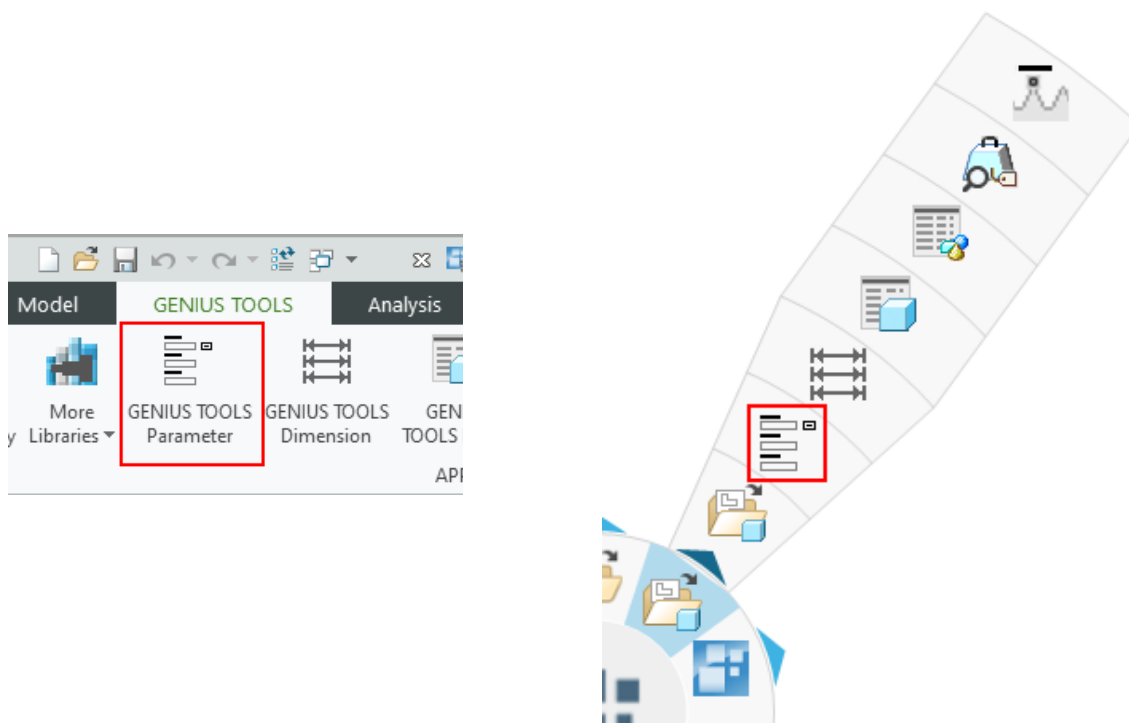
Warning: Changes to the selection parameter or manual selection of a parameter definition may lead to loading a different Parameter definition!

13.2 Usage

This section contains information on using GENIUS TOOLS Parameter. It describes the general structure of the program.

Starting the program

Start GENIUS TOOLS Parameter in the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).



The program opens the parameter form of a specific parameter definition. The selection can be carried out manually or can be automatized, see the section [Selection of parameter definition](#)³³¹.

Automatic start when creating new models

The GENIUS TOOLS parameter dialog box can be set to open automatically after creating a new part, assembly or drawing, by listing the desired file types in the `gtl_gtp_start_gtp_after_model_creation` configuration option, separated by commas. The parameter dialog does not open if you create a new file and insert it into an assembly in one step.

Modality of the dialogs

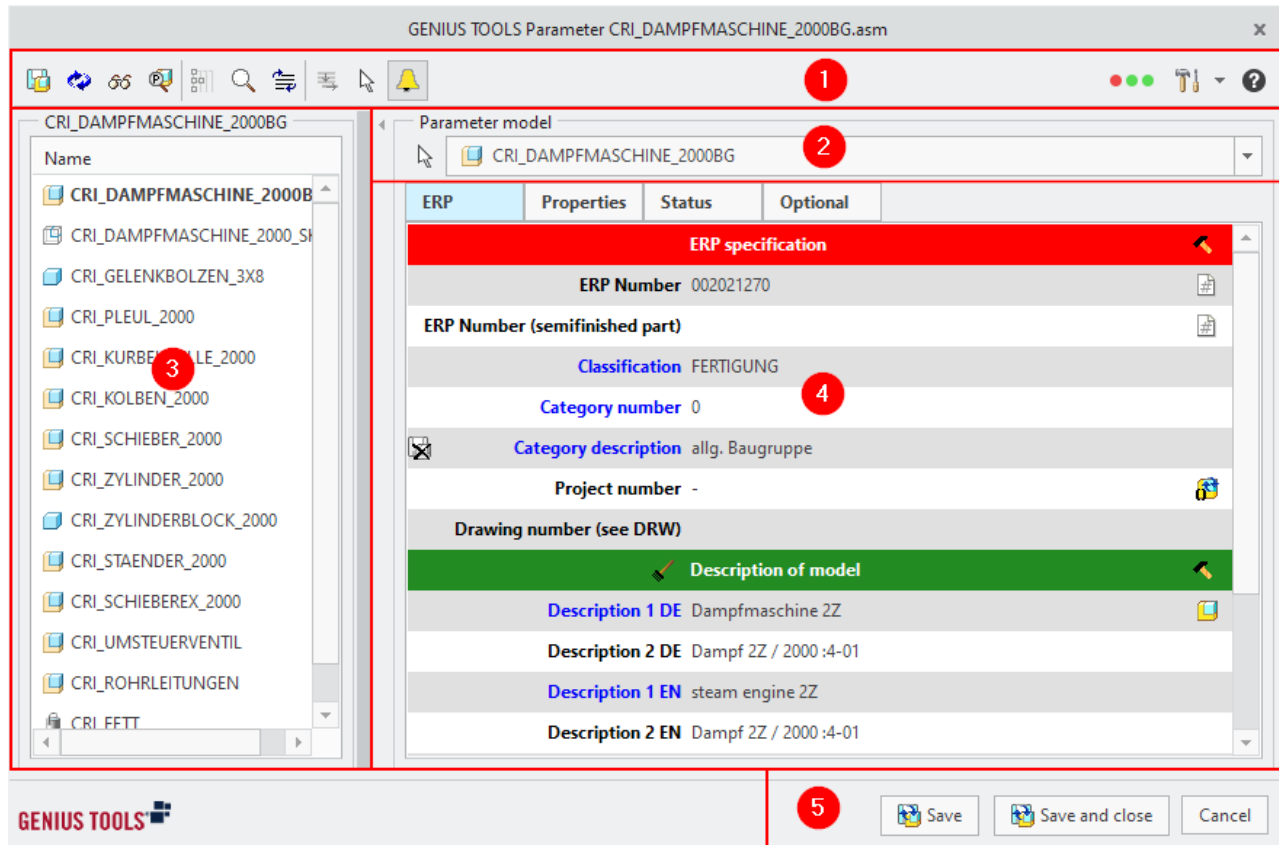
The modality of the Parameter dialogs (locking the rest of the application or individual dialogs) is determined by the configuration option `gtp_dock_dialog_to_md1`. You can edit multiple models using the same Parameter dialog and also open and operate the editor at the same time. If you change modality to set `gtp_dock_dialog_to_md1=0`, GENIUS TOOLS Parameter will remain open and available when you change the model.

Click *Discard changes and reload* to make the current Creo model the parameter model and load the corresponding parameter definition.

Warning: You can only change parameter values if the active model is the parameter model! Save any changes to the parameter form before you change the active model.

13.2.1 User interface

The user interface of GENIUS TOOLS Parameter consists of the following elements:



1. Command bar³³⁵ with status indicator³³⁸ and tools menu³³⁷
2. Model selection³³⁹
3. Model list³⁴⁰
4. Parameter form³⁴²
5. Save / Cancel buttons³³⁴.

Save and cancel

Provides the option to save the model if the option is set.

The *Save* button turns green to indicate changes to the parameter form. Click on the button to apply changes in the parameter form to a Creo model. The status display is updated. *Save and close* skips updating the status display. If errors occur, the parameters are not saved.

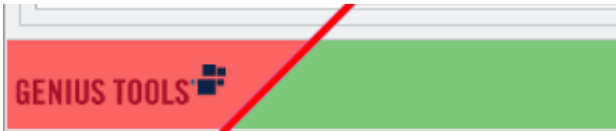
Depending on the save status, the footer of the parameter window is colored:

Green: Parameters were transmitted, no warnings or errors occurred during transmission. The PreSave method returned `true`.

Yellow: Parameters were transferred, but warnings or errors occurred during transfer. The PreSave method returned `true`.

Red: Parameters were not transferred because conflicts exist and the configuration options `gtp_do_not_save*` are set to 1.

Alternatively, the PreSave method returned a `false` and therefore no parameters were transferred.













The footer of the Parameter window shows the save status.




Warning: Note the modality of the Parameter window! By default, GENIUS TOOLS Parameter is closed when other windows in Creo Parametric (Creo standard window and GENIUS TOOLS) are opened.

13.2.2 Command bar


The command bar contains the following buttons:

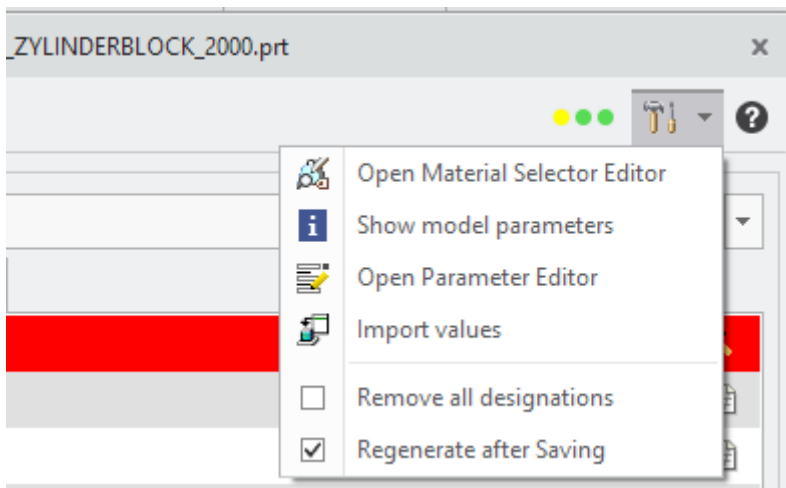
Icon	Name	Description
	Save	Saves all parameters and corresponding parameter values in the model. Note: Depending on the configuration, saving may be possible only if there are no conflicts.
	Discard changes and reload	The current parameter definition and the parameters of the model are reloaded. Note: Unsaved changes will be discarded!
	Adopt model parameters	Model parameters are read again, the parameter form is refilled. Note: Unsaved changes will be discarded!

Icon	Name	Description
	Select model and adopt parameter values	<p>After selecting a model in the Creo window its Model parameters are read and displayed in the current parameter form.</p> <p>Note: Unsaved changes will be discarded!</p>
	Update model tree	<p>Reloads the model tree and displays it in the Creo Parametric sidebar.</p> <p>Parameters, flagged for the model tree in the editor will be additionally shown in the model tree.</p> <p>The model tree can only be adjusted if the GENIUS TOOLS Parameter is not bound to the Creo Parametric window (<code>gtp_dock_dialog_to_md1=0</code>).</p>
	Highlight current model	Highlights the model selected in the model list in the Creo window.
	Open parameter definition ³⁵⁵ from list	Opens an existing parameter definition and applies it to the model parameters of the current model.
	Roll values	<p>Executes rolling when configured in the current parameter definition.</p> <p>Note: Button is only activated in drawings.</p>
	Select start model	Displays a model selected in the Creo window or in the model tree in the parameter form.
	Activate or deactivate Save message	Activates/ deactivates in assemblies a warning message for unsaved changes in parts.

Icon	Name	Description
	Status indicator ³³⁸	Shows the current status for <i>loading</i> , <i>working</i> and <i>saving</i> with traffic light colors and opens the status dialog.
	Tools	The Tools menu contains various supporting functions.
	GENIUS TOOLS Help	Opens the Help.

Tool menu

Supporting functions are available in the tool menu :



Open Material Selector Editor

Opens the GENIUS TOOLS Material Selector Editor to modify the material properties.

Show model parameters

Shows model parameters of each opened part (prt, asm, drw) in an extra window. The function only opens a view. No changes can be made, but parameter values needed can be copied.

Open Parameter Editor

Opens the GENIUS TOOLS Parameter Editor³⁵¹ to edit the parameter definitions.

Import values

Imports a text file or a CSV file whose content is written into the corresponding parameters.

Remove all designations

Specifies whether all designations are removed in Windchill mode. The display of this checkbox can be hidden by setting the configuration option `gtp_designate=0`.

Regenerate after saving

Specifies whether models will be automatically regenerated after clicking *Save*. The display and standard setting of this checkbox can be changed in the configuration option `gtp_regen` as follow:

- 0: display unchecked box
- 1: display checked box
- 2: automatic regeneration and no display of checkbox
- 1: no automatic regeneration and no display of checkbox

13.2.3 Reading parameter values from text files

You can import parameter values from text files into the parameter form. To do so, use the function *Import values* in the tool menu.

The file *GTP_values.txt* is read from the work directory and the values are written to the corresponding parameters. After import, table connections and JavaScript OnChange functions are executed.

Warning: The function *Import values* also writes read-only fields!

Values can be read from files with the following character encodings:

- UTF16LE/BE
- UTF8 BOM
- ANSI

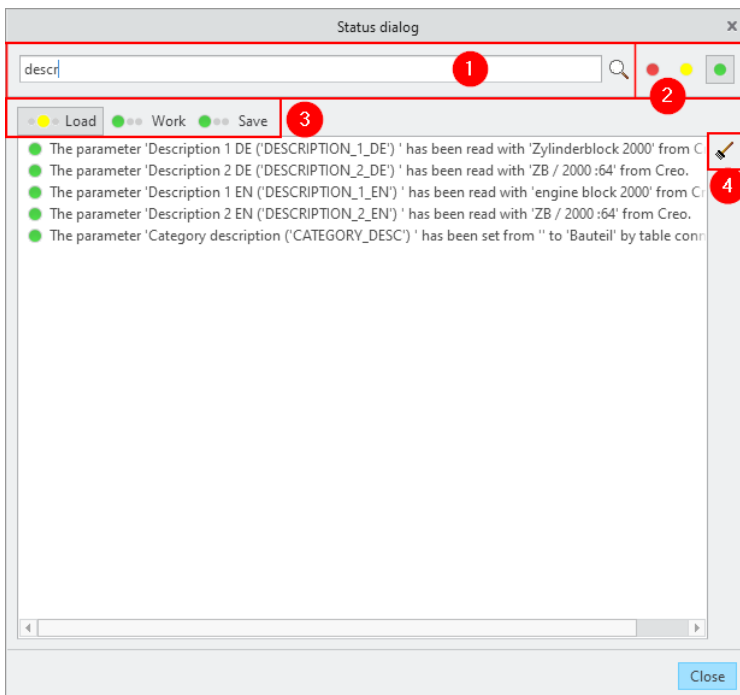
The text file has to follow the format given below with a parameter name and a value separated by a colon in each line.

Example

```
DESCRIPTION_1_DE:Abdeckkappe  
DESCRIPTION_1_EN:Cover cap  
PART_NO:004526-12  
ARCHIVE:YES
```

13.2.4 Status dialog

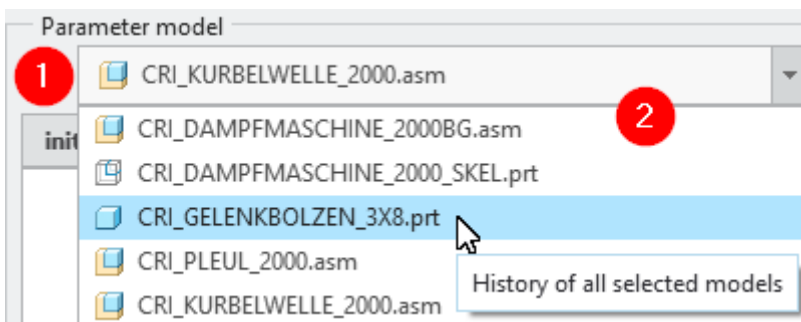
The command bar contains the status dialog for tracking successful and faulty work steps. The functionality of the status dialog is the same for all modules.



- 1. Text search:** searches for message content in all tabs. Press ENTER or click the magnifying glass icon to start the search.
- 2. Traffic light buttons:** Use the three traffic light buttons (2) to turn the displayed messages (errors, warnings and/or successes) on or off.
- 3. Load, Work and Save:** The messages are divided into these three tabs. For each tab, a separate status traffic light provides an overview of the status of that work step. Whether there is a warning or an error is indicated by the traffic light color in the tab.
- 4. Reset:** deletes the displayed status messages. These messages are only displayed again after a restart of the module.

13.2.5 Model selection

The model selection consists of two elements:



The model selection with object selection (1) and drop-down list (2)

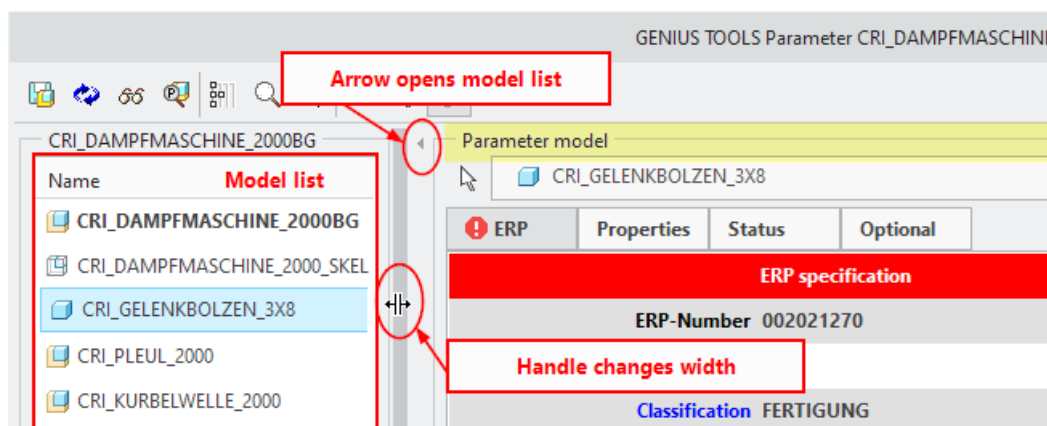
Select models using the Object selection (1). Their parameters will be read into the parameter form.

The drop-down list (2) shows the current selection. Open the list and select from the recently selected models.

Above the model selection, the name of the current parameter definition is displayed. Place the mouse on the definition name to see its storage location as a tooltip.

13.2.6 Model list

The model list opens on the left below the command bar when clicking the arrow next to *Parameter model*.



Model list and parameter model CRI_GELENKBOLZEN_3X8

Click on a sub-model to display its parameters in the parameter form. The model, for which the parameter values are displayed, is referred to as *parameter model*.

Mode dependency of the model list

The model list depends on the configuration option `gtp_show_md1_list`.

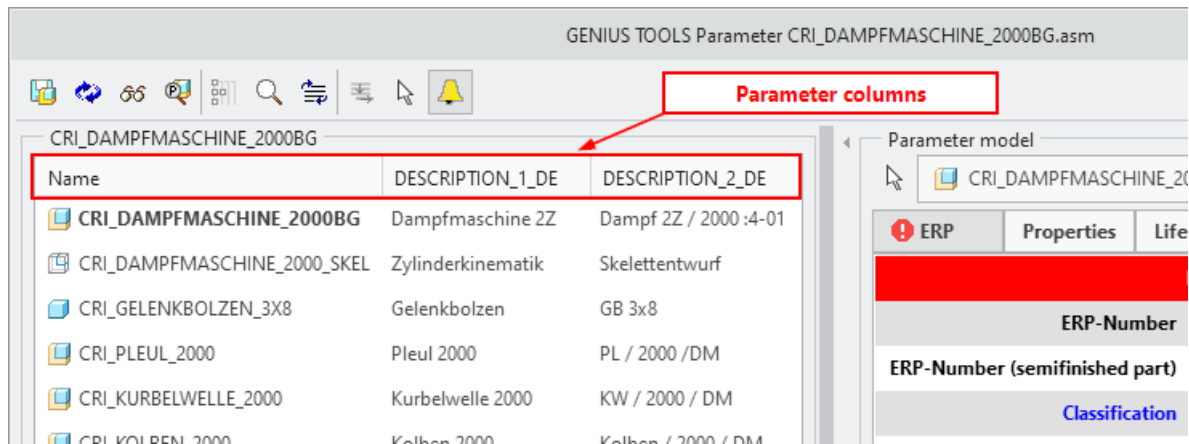
If the configuration option is set to 0, the model list is not displayed. If the configuration option is set to 1, the model list is displayed. If the configuration option is set to 2, the model list is displayed or not, depending on the model type (for example, show list for assemblies and for generic parts with instances).

If the configuration option `gt_window_size_position_save` is set to 1, the size of the model list is saved.

Adding extra columns for parameter

You can add additional columns for parameters in the model list by entering the parameters needed, separated by comma, into the configuration option `gtp_model_tree_columns`. The option `gtp_model_tree_column_width` allows you to set the

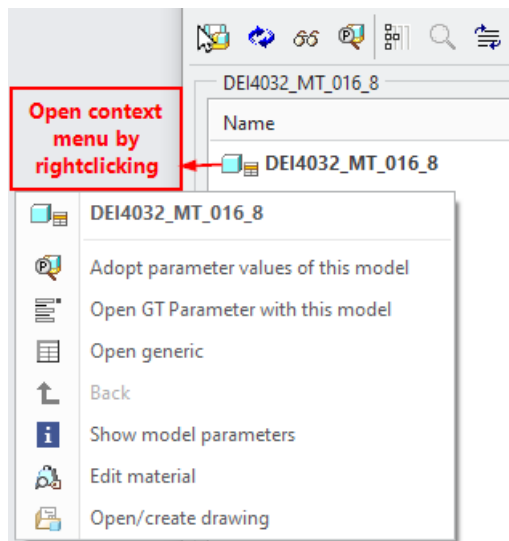
width of these additional columns, measured in the Windows standard width of a character. The default value is 10.



Additional columns for the parameters DESCRIPTION_1_DE and DESCRIPTION_2_DE

Context menu

The model list has its own context menu which appears when right-clicking on an element.



- **Adopt parameter values of this model:** The parameters of the selected model are transferred to the fields of the parameter model in the parameter form. Save after executing this action to ensure that the values are applied into the parameter model.
- **Open GT Parameter with this model:** Opens the selected model in GENIUS TOOLS Parameter.
- **Open generic:** Opens the instance as a Generic.
- **Back:** Switches the view to the previous model.
- **Show model parameters:** Opens a new window and displays an overview of all parameters of the selected model. It is not possible to make changes via this window. If you need parameter values from this overview, copy them.

- **Edit material:** Opens GENIUS TOOLS Material for the selected model.
- **Open/Create Drawing:** Opens the models drawing or generates a new drawing.

13.2.7 Parameter form

The parameter form displays parameter sets of the currently selected model – depending on the [parameter definition](#).³⁶⁵ Only those parameters specified in the editor are displayed in the parameter form.

A **parameter set** consists of the localized title of a parameter (1) and the parameter value of the model in the input field (2).



Click into input field to activate it.


Use the tab key (TAB) for switching to the next input field and SHIFT+TAB for switching to the previous field.


Click the localized parameter name to display additional information about this parameter.

The background color of the heading bar (3) depends on the configuration in the parameter definition.

Input fields

Enter the parameter values either manually or via predefined automatic field functions (5). These two functions *Auto-fill*  and *Reset*  affect input fields up to the next separator.

With the *broom symbol*  you reset the input fields to their initial value. Whether an input field can be reset depends on its configuration. The setting is made in the parameter definition under *Resettable*. See [Adding parameter > Field: Value](#) ³⁶².

Execute field functions  automatically executes the field functions that have been defined for fields. If more than one function has been configured for a field, the first one will be executed. Only functions that do not require user interaction will be executed.

To create the hammer symbol, consult the chapter [Adding separators](#) ³⁷⁵.



The different field types of the input fields are listed in the section [Field types](#). ³⁴⁶

Tabs

Tabs will show up above the parameter form if a parameter definition contains [parameter subdefinitions](#) ³⁵⁵. The tabs subdivide parameters for better accessibility.

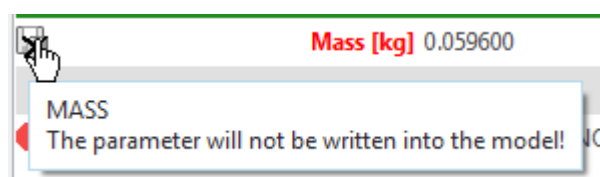


If there are pending tasks (e.g. a mandatory field not filled in) for a sub parameter definition, a warning symbol (exclamation mark) is displayed before the name in the tab.

If you have previously opened GENIUS TOOLS Parameter in a Creo model, the program remembers in which tab you stopped working.







Info symbols in parameter sets

Info symbols are displayed before the localized name to indicate important information about a parameter. Move the mouse pointer over the icon for further information.



Info symbols before the localized name

The following info symbols are displayed:

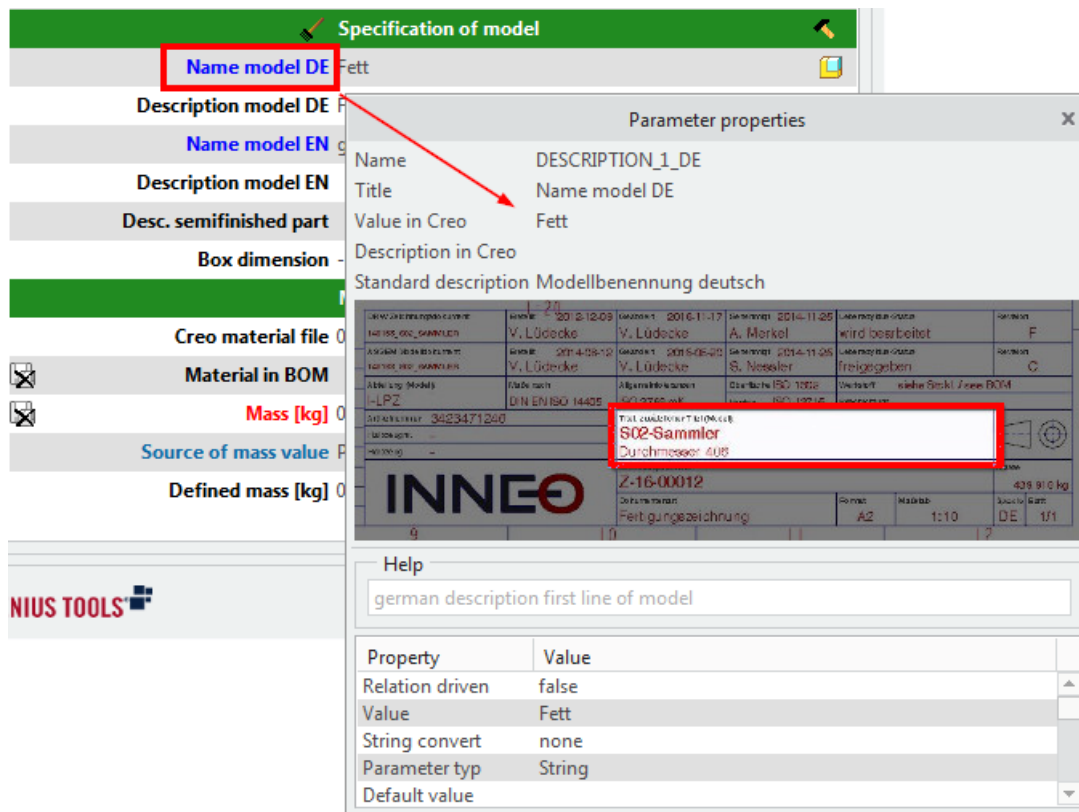
Icon	Description
	The parameter will not be adopted into the model when saving.
	The parameter is not contained in the model. Saving creates the parameter and transfers the parameter value into the model.
	The input field is a mandatory field and must be filled in.
	The input does not correspond with the specifications for the input field. (e. g.: letters in a number field).
	The parameter is locked by a relation.
	The model is an instance, the parameter is not in the family table of the generic.

Localized parameter titles are displayed in different colors as soon as one of the following conditions occurs:

- Dark blue: The parameter is filled by database connection.
- Light blue: The parameter is filled by a CSV file.
- Orange: The parameter is controlled or locked by a relationship.

Parameter properties

In addition, you can add further information on parameter sets. Click on the localized parameter title to open the dialog *Parameter properties*. The dialog shows properties and information about the selected parameter.



Properties dialog of the parameter

13.2.7.1 Calculating / assigning masses

There are several ways to calculate mass. The parameter `mp_source` is displayed as the Source of mass value.

- By default, the mass of the part is calculated based on the density (Source of mass value: GEOMETRY). In this case, if a Creo-Material file is selected, the density is automatically filled in and the mass is calculated.

Material, density & mass	
Creo-Material file beton.mtl	
d=	Material (BOM) Beton
	Source of mass value GEOMETRY
<input checked="" type="checkbox"/>	Density [g/cm³] 2.300000
d=	Mass (BOM) 7.938500

- Alternatively, you can specify as Source of mass value: PARAMETERS. In this case, the density is removed from the parameter form and the mass is taken from the bill of materials. If not specified, the value 0 is entered.

Material, density & mass	
Creo-Material file	beton.mtl
d=	Material (BOM) Beton
Source of mass value	PARAMETERS
Defined mass [kg]	5.450000
d=	Mass (BOM) 5.450000

13.2.8 Field types

A parameter value is displayed in an input field, which can be predefined and also contain automated [field functions](#)³⁷⁰.

The values for parameters can be filled in and reset automatically, see [input fields](#)³⁴³.

The input fields can have a different function depending on the configuration in the editor:

Input field

Regular input fields accept any string for input. Restrictions (e. g. numbers only) are possible through configuration in the editor.

Description model EN	Steam 2Z/2000:4-01
----------------------	--------------------

Drop-down list

Drop-down lists contain a selection of possible inputs. They are specified through text/CSV files or databases.

General tolerance	ISO 2768-m
Chem. element	ISO 2768-mH
Thickness [µm]	ISO 2768-mK
Brightness / color	ISO 2768-mL

Combined input field

Combined input fields allow free input and to select from a drop-down list. If a combined drop-down is assigned to a database, the drop-down list entries will be filtered as you type.

Supplier	IN
Project no.	INA
	INNEO

List selection

A list selection automatically displays a list with appropriate entries for the current input. This requires a table connection. The displayed columns are specified in the configuration

and are retrieved from a CSV file or a database. When entering text to a list selection with database connection, the list will be filtered for matching entries. You can also enter regular expressions (RegEx) for filtering. Clicking on the entry applies the matching value to the field.

Please note: A selection list provided by a webserver as a JSON file cannot be filtered further in the parameter form.

EN*	DE
straight pin	Zylinderstift
strainer	Sieb
strainer basket	Siebtopf
-	-

Material input field

The behavior of material input fields is similar to combined drop-down lists. All materials and subfolders of the material folder are displayed in the drop-down list. Subfolders can be selected to navigate the tree. In addition, you can open the material selection via the magnifier icon.

Read-only input field

Read-only input fields cannot be edited.

Mass [kg] 0.059600

Warning: The function *Import values* or JavaScript functions can edit read-only fields.

Checkbox

Checkboxes allow to choose from two options. This can be Yes/No decisions, for example.

PDM information

Online available? ☒

Field functions

Field functions are predefined functions for determining a parameter value in GENIUS TOOLS Parameter. Field functions can set parameter values in different ways, e. g. for setting the current date or transfer a model parameter value from Creo .

Click on a field function icon to execute the function. Find a list of all functions in chapter Field functions.

Connected fields

Input fields can be interlinked. Then, they are dependent on one another. If the value of one of the fields is modified, appropriate values will be automatically entered into the other fields. Information from databases or CSV files will be evaluated in the background.

ERP	Properties	status
ERP specification		
ERP no. -		
ERP no. (semifinished part) -		
Classification		
Category no. 0		
Category desc allg. Baugruppe		
order code -		
Project no. -		
Drawing no. (set in DRW) -		
Description of model		
Name model DE Ausrichtflansch (3)		
Description model DE -		
Name model EN *flan (1)		
Description model EN -		

Valid values	
EN*	DE
alignment flange	Ausrichtflansch
centering flange	Zentrierflansch
chuck flange	Futterflansch
clamp flange (2)	Spannerflansch
clamping flange	Aufspannflansch
flange	Flansch
flange seat	Flanschaufnahme
flange socket	Flanschdose
flanged wheel	Bordscheibe
hub flange	Aufname-Flansch
motor flange	Motorflansch

Connected fields	
Name	Column
DESCRIPTION_1_DE ->	DE (4)

Following the input (1) appropriate values (2) are displayed. After selecting a value, the corresponding value is entered into each connected field (3). The names of the connected fields (4) are displayed below the valid values.

Click on a connected field and enter a search term. You can also use the * placeholder to search for subterms. Then, select the value to be entered from the list. Appropriate values will be adopted by each connected field.

13.3 Configuration

This section contains further information on

- available regular expressions for the Parameter Editor, and
- the structure of the Parameter Editor

In addition, you will find information on configuring GENIUS TOOLS Parameter and the parameter definitions in the section [Use cases](#)³⁷⁷.

13.3.1 Regular Expressions

Use Regular Expressions in GENIUS TOOLS for Creo to check value inputs or allow only rule-compliant inputs to be saved.

Character	Description
\	Indicates the following character as a special or verbatim character. For example "n" is corresponds to the character "n". "\n" corresponds to a line-break character. The sequence "\\" corresponds to "\", "\(" corresponds to "(".
^	Corresponds to the beginning of the input.
\$	Corresponds to the end of the input.
*	Corresponds to the proceeding character zero or multiple times. For example "zo*" matches either "z" or "zoo".
+	Corresponds to the proceeding character one or multiple times. "zo+" for example matches "zoo", but does not match "z".
?	Corresponds to the proceeding character zero or one time. For example "a?ve?" matches the "ve" in "never".
.	Corresponds to all single characters except for a line-break character.

Character	Description
(Pattern)	Matches Pattern and saves the equivalent. The compared substring can be retrieved from the resulting matches listing using the elements [0]...[n]. For comparing of characters put in parentheses () use "\" or "\".
x y	Corresponds to either x or y. For example matches "l red" either "l" or "red". "(l r)ed" matches "led" or "red".
{n}	n is a positive integer. Corresponds to exactly n times. "o{2}" for example does not match the "o" in "Robert" but the first two "o"s in "Boooooat".
{n,}	n is a positive integer. Corresponds to at least n times. "o{2}" for example does not match the "o" in "Robert" but all "o"s in "Boooooat". "o{1,}" is equivalent to "o+". "o{0,}" is equivalent to "o*".
{n,m}	m and n are positive integers. Corresponds to at least n and maximum m times. For example "o{1,3}" matches the first three "o"s in "Boooooat". "o{0,1,}" is equivalent to "o?".
[xyz]	A group of characters. Corresponds to any of the included characters. "[abc]" for example matches the "a" in "falling".
[^xyz]	A group of excluded characters. Corresponds to any character not included. "[^abc]" for example matches the "f" in "falling".
[a-z]	A character range. Corresponds to any character in the specified range. For example, "[a-z]" matches any lowercase alphabetic character in the range from "a" to "z".
[^m-z]	An excluded range of characters. Corresponds to any character not included in the specified range. "[m-z]" for example matches all characters not included in the range from "m" to "z".

Examples

Regular expression	Description	Example
[a-zA-Z]*_[a-zA-Z]*	Any alphabetic string with an underscore	user_tbx
[0-9]{5}	Five random numbers	12345
^. {7}\$	7 random characters	t_p.prt
^[A-Z]{1}[a-z]{2,10}	A capital letter at the beginning followed by two to ten lower case letters	Tuser
[0-9]{2}\.[0-9]{2}\.[0-9]{4}	Date format	01.08.1975

13.3.2 Parameter Editor

The GENIUS TOOLS Parameter Editor is used to specify parameter definitions for use in GENIUS TOOLS Parameter. Parameter forms consist of one or more parameter definitions.

Starting the program

Start GENIUS TOOLS Parameter Editor from the tools menu in the GENIUS TOOLS Parameter window or via GENIUS TOOLS Quick Access ([<] key).



Saving changes

The changes made in the editor are saved in the resource directory.

When working with Startup TOOLS, the data of the local resource folder (in the Cadpool directory) is synchronized with the resource folder of the administrator computer

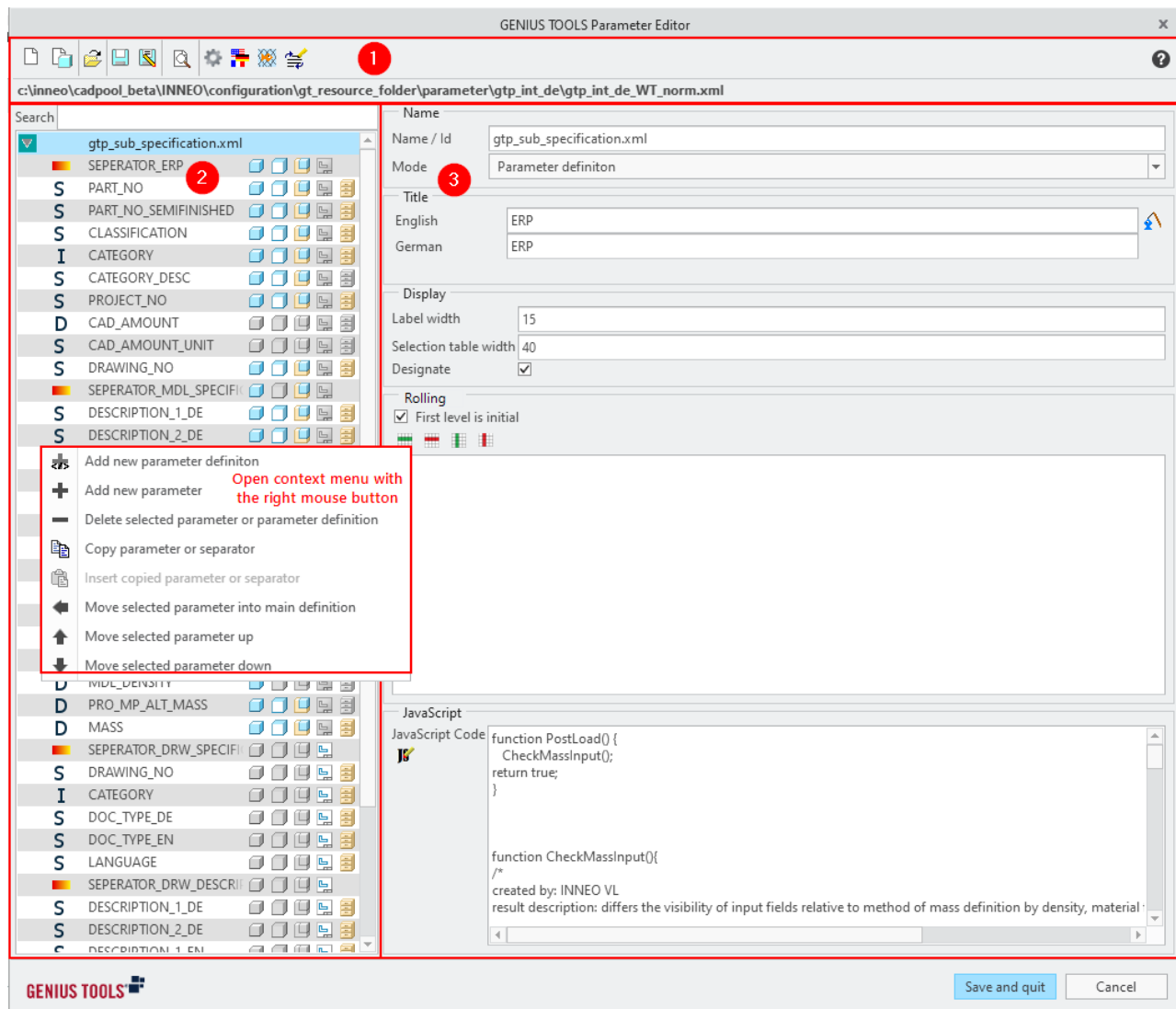
(Caddepot). Path: <operatingenvironment>/parametric/configuration/gt_resource_folder.

To ensure that the changes are applied, you should work directly in Caddepot.

Warning: If you are using satellites, note that the Caddepot of a satellite is overwritten by the Caddepot of the main server.

13.3.2.1 User interface











The user interface of the GENIUS TOOLS Parameter Editor consists of the following elements:




1. Command bar ³⁵³
2. Element list ³⁵³ with context menu ³⁵³
3. Detail view for parameter definitions ³⁵⁵, parameter ³⁵⁹ and separators ³⁷⁵

13.3.2.2 Command bar

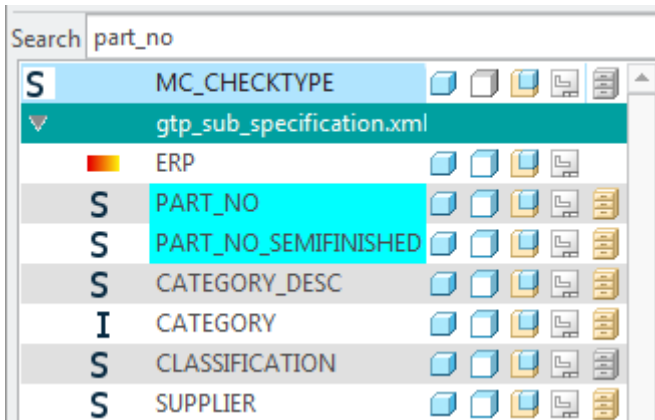
The following buttons are contained in the command bar:

Icon	Name	Description
	New	Creates a new parameter definition in XML format. Note: The file is only created after saving.
	New parameter definition from model	Creates a new parameter definition based on all parameters of the currently opened model.
	Open parameter definition	Opens an existing parameter definition for editing.
	Save	Saves the current parameter definition.
	Save as	Saves the current parameter definition with a new name.
	Preview	Displays the result of the current configuration in GENIUS TOOLS Parameter without the need to save.
	Properties	Opens the dialog to edit the parameter definition properties.
	Manage languages	Opens the dialog to manage the languages of a parameter definition.
	Automatic standard texts	Opens the dialog for automatically setting default texts for all parameters and separators. The names of the parameter definitions are used as separators in the table. The parameter names are used as the translation key. See Set standard texts for multiple elements ⁴⁹⁷
	Edit list of parameter definitions	Opens the dialog to edit the list of existing parameter definitions ³⁷⁶ .

13.3.2.3 Element list

The element list displays all parameters and sub-parameter definitions  contained in the current parameter definition. Click on an element in the list to display it in the detail view. Click on the arrow symbol before a sub-parameter definition to expand it and display the contained parameters.

A search field is located above the element list. Use the search field to search elements in the list by parts of their name. Search results will be highlighted in color.



Color highlight when searching "part_no"

You can move parameters within the same parameter definition using Drag-and-Drop. Press CTRL to select multiple parameters.

The availability settings are displayed next to the element on the right. They indicate if an element (parameter or separator) will be displayed in the parameter form in parts, bodies, assemblies or drawings. Click on the symbols to show or hide an element in the corresponding mode.

Parameters have a fourth icon. It specifies whether designation is allowed for a single parameter. This function is only relevant if a PDM system such as Windchill is used.

Context menu

The element list has its own context menu. Right-click on an element to open the context menu.

You can select multiple elements using CTRL.

The following options are available:

Add new parameter definition: Adds a new sub-parameter definition to the element list.

Add new parameter: Creates either a new parameter definition or a separator.

Delete selected parameter or parameter definition: Deletes the currently selected element.

Copy parameter or separator: Copies parameters or separators to the buffer memory.

Insert copied parameter or separator: Pastes the previously copied element into the element list at the current position.

Move selected parameter into main definition: Moves a parameter from a sub parameter definition into the main parameter definition file.

Moving the selected parameter up/down: Moves a parameter or separator one position in the list.

13.3.2.4 Parameter definitions



Various parameters can be defined in a parameter definition. Since such a list can become confusing, parameters are organized in subparameter definitions. The subparameter definitions are displayed as a *tab* in the parameter form ³⁴³.

Master parameter definition

A parameter definition that contains subparameter definitions is called a master parameter definition.

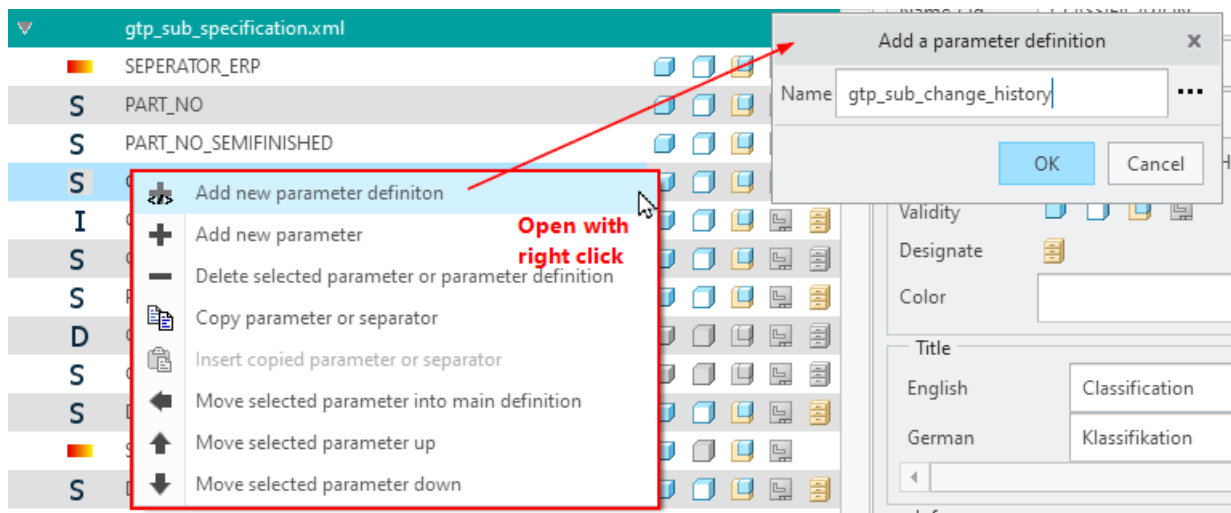
All definitions are stored in an XML file.

Create a parameter definition

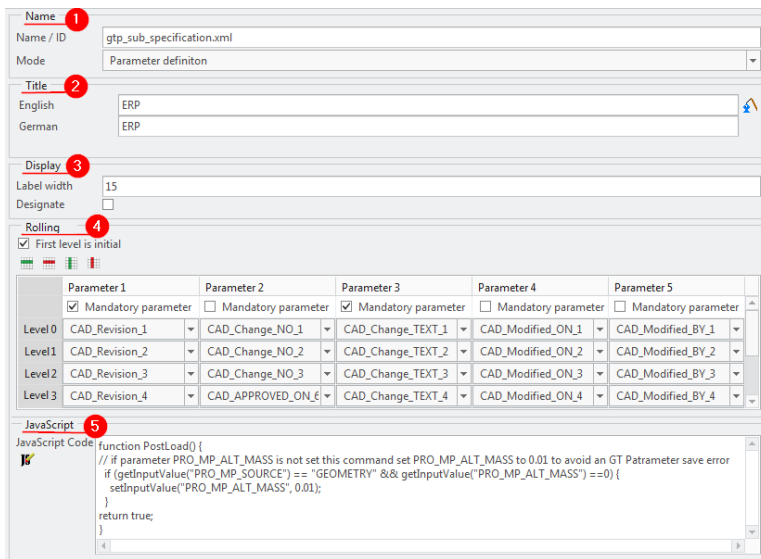
Open the list of all parameter definitions ³⁷⁶ via the  button. Create a new entry with the + Symbol and select this new definition in the parameter dialog with the function *Open parameter definition from list* . Open the editor for configuration.

Create a parameter subdefinition

In the element list open the context menu and select *Add new parameter function*.



In the Detail view select the mode *Parameter definition*. The dialog is divided into the following sections.



The screenshot shows a software interface for defining sub-parameters. It includes fields for Name, Title (English/German), Display (Label width/Designate), and a table for parameter levels. A JavaScript code editor is at the bottom.

	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
	<input checked="" type="checkbox"/> Mandatory parameter	<input type="checkbox"/> Mandatory parameter	<input checked="" type="checkbox"/> Mandatory parameter	<input type="checkbox"/> Mandatory parameter	<input type="checkbox"/> Mandatory parameter
Level 0	CAD_Revision_1	CAD_Change_NO_1	CAD_Change_TEXT_1	CAD_Modified_ON_1	CAD_Modified_BY_1
Level 1	CAD_Revision_2	CAD_Change_NO_2	CAD_Change_TEXT_2	CAD_Modified_ON_2	CAD_Modified_BY_2
Level 2	CAD_Revision_3	CAD_Change_NO_3	CAD_Change_TEXT_3	CAD_Modified_ON_3	CAD_Modified_BY_3
Level 3	CAD_Revision_4	CAD_APPROVED_ON_4	CAD_Change_TEXT_4	CAD_Modified_ON_4	CAD_Modified_BY_4

```

JavaScript Code
function PostLoad() {
    // if parameter PRO_MP_ALT_MASS is not set this command set PRO_MP_ALT_MASS to 0.01 to avoid an GT Parameter save error
    if (getInputValue("PRO_MP_SOURCE") == "GEOMETRY" && getInputValue("PRO_MP_ALT_MASS") == 0) {
        setInputValue("PRO_MP_ALT_MASS", 0.01);
    }
    return true;
}
  
```

Detail view of a sub-parameter definition

1. Name


Name/ID: The name represents the filename of the parameter definition file. If the filename value is changed and then saved, this will be similar to a *Save as* command. The parameter definition will be saved with a new name and the file with the old name remains unchanged.

Warning: The mode of sub parameter definitions cannot be changed at a later date.

2. Title

Specifies the language-dependent (localized) names that will be displayed as tabs in the parameter form.

The number of available input fields depends on the language configuration.

Standard texts can be added via the button  (Description of the standard text selection dialog ⁴⁹⁷).

3. Display

Label width: Specifies the width (in characters) of the localized parameter title in the parameter form.

Selection table width: Specifies the width of the tables (for example, for database connections) that are displayed next to the parameter form. The width is given as number of characters.

Designate: Specifies whether designations in PDM mode should be priorly removed for all parameters contained in the definition.

4. Rolling

In the *Rolling* section, parameters are specified whose values are transferred to other parameters. Rolling allows to pass incremental values stepwise (per level) in a parameter chain.

Please note: The function for rolling parameter values is only available in drawing mode.

The parameter values are transferred FIFO (first in, first out). This allows to create a change history, for instance. For each editing process, the parameter value is moved one step. When the end of the configured chain is reached, the parameter values are deleted.



Rolling



☒ First level is initial

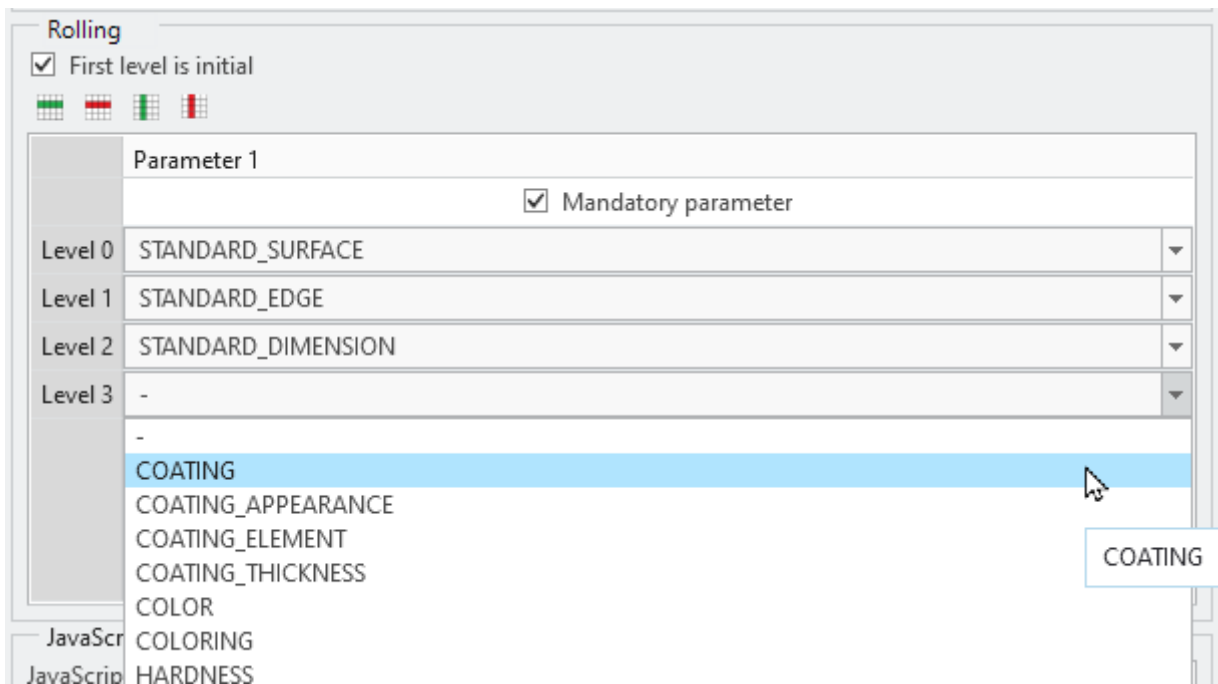
	Parameter 1	Parameter 2	Parameter 3
	<input checked="" type="checkbox"/> Mandatory parameter	<input type="checkbox"/> Mandatory parameter	<input type="checkbox"/> Mandatory parameter
Level 0	CAD_Revision_1 ▾	CAD_Change_NO_1 ▾	CAD_Change_TEXT_1 ▾
Level 1	CAD_Revision_2 ▾	CAD_Change_NO_2 ▾	CAD_Change_TEXT_2 ▾
Level 2	CAD_Revision_3 ▾	CAD_Change_NO_3 ▾	CAD_Change_TEXT_3 ▾
Level 3	CAD_Revision_4 ▾	CAD_Change_NO_4 ▾	CAD_Change_TEXT_4 ▾
Level 4	CAD_Revision_5 ▾	CAD_Change_NO_5 ▾	CAD_Change_TEXT_5 ▾
Level 5	CAD_Revision_6 ▾	CAD_Change_NO_6 ▾	CAD_Change_TEXT_6 ▾
Level 6	CAD_Revision_7 ▾	CAD_Change_NO_7 ▾	CAD_Change_TEXT_7 ▾
Level 7	CAD_Revision_8 ▾	CAD_Change_NO_8 ▾	CAD_Change_TEXT_8 ▾

The parameter values are transferred step by step

First level is initial: If this checkbox is set, the parameter values of the first step will not be overwritten after the parameter has been filled once. This means that the configured chain starts at level 1.

Add or remove rows:  /  Adds rows to the table. You can add rows to determine after how many editing steps a parameter value is deleted.

Add or remove columns:  /  Adds columns to the table, defining more parameters for which values should be transferred.




Determine parameters for Rolling

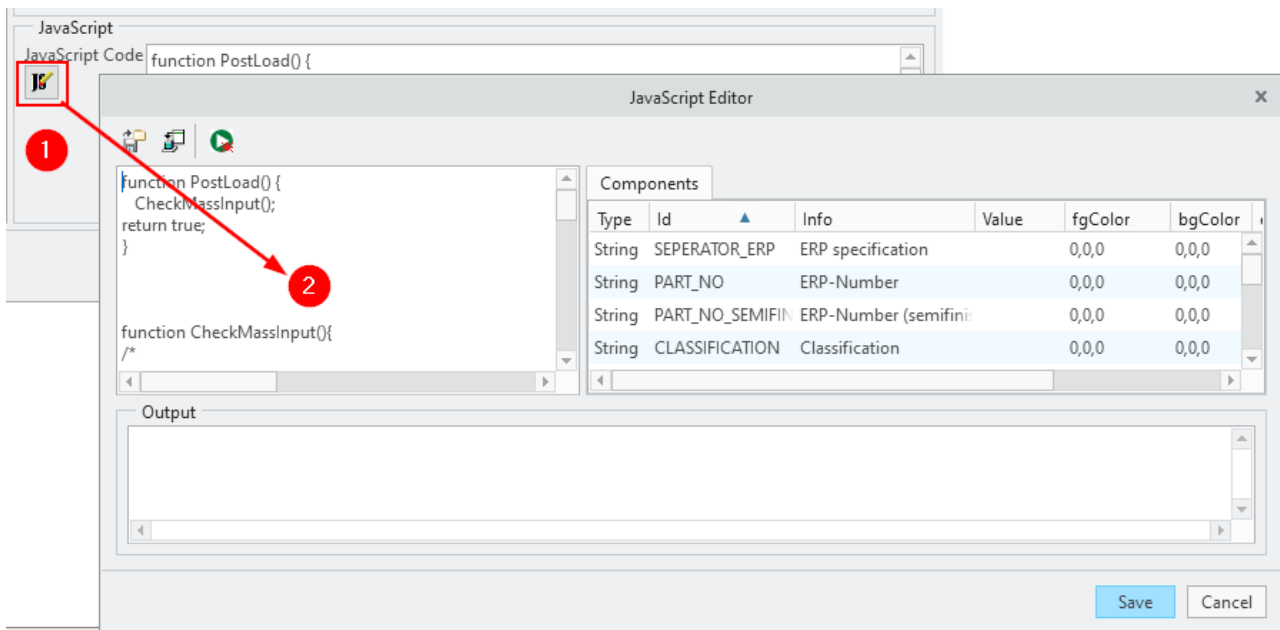
5. JavaScript in parameter definitions

You can add executable JavaScript code to each parameter definition. It is only available in the respective parameter definition.

Parameter definition: Open the dialog *Properties* and navigate to the section *JavaScript*.

Sub parameter definition: Select the sub parameter definition in the element list and navigate to the JavaScript section in the detail area.

Enter the JavaScript code directly into the input field or click on the JavaScript icon  to open the [JavaScript Editor](#)⁵⁷⁷. Go [here](#)⁵⁸² for an explanation of JavaScript functions and short examples. Under [Use cases](#)³⁹⁶ you will find a complex example of using JavaScript with *GENIUS TOOLS Parameter*.



Execution time of functions: JavaScript can be executed at different times.

Time of execution	Function
After loading a form	PostLoad
Before saving values in a form	PreSave
After saving values in a form	PostSave
After editing a value or after clicking enter	OnChange

Several of these functions can occur in the JavaScript code. The names of the PostLoad and PreSave functions are fixed. Functions of type OnChange can be named freely.

Table connections are also covered by OnChange functions. If a table connection changes parameter values when GENIUS TOOLS is started, no OnChange function is executed. To correct these parameter values, the PostLoad function must be used.

Please note: Use the JavaScript function *creoMapkeyAddToStack* only as a PostSave function, because it executes mapkeys and mapkeys usually close windows.

13.3.2.5 Adding Parameters

Parameters in GENIUS TOOLS Parameter contain all settings to specify whether and how a parameter is saved in a Creo model and which input field type is to be used for input.

Field functions such as *Name Generator* or *Material search* are also stored in parameters.

Name (1)

Name / Id: DESCRIPTION_1_DE

Mode: Parameter

Type: String

View: ☐ Read-only ☐ Hidden ☒ Save

Validity: ☐ ☐ ☐ ☐

Designate: ☐

Color:

Title (2)

English: Description 1 DE

German: Benennung DE

Info (3)

Image: images\title_description.jpg

English: Model description 1 german

German: Modellbenennung deutsch

Value (4)

Value:

Parameter unit:

Resettable: ☒

Only from list: ☐

Description: MDL Benennung

String convert: None

Mandatory field: ☒

Format: ^[^\|@]*\$

Maximum length: 30

Show in model tree: ☒

Column width in model tree: 15

List (5)

List: SQLITE database

Database name: gtp_int_de.sqlite

Table: description_custom

Column: DE

Displayed columns: DE,EN

Table connections (6)

Connected parameters

Parameter	Column
DESCRIPTION_1_EN	EN

Filters

Parameter	Column
-----------	--------

Functions (7)

OnChange js-function:

Parameter functions

Function
Model name

Save and quit Cancel

Example for a detail view of a parameter

The detailed view for parameters is divided into different sections:

1. Name

Name: Specifies the Creo model parameter that is addressed by the parameter.

Mode: Specifies whether it is a separator or a parameter.

Type: Specifies the parameter type to be saved in Creo models. Possible types include: String, Integer, Real number, Yes/No and Material.

Warning: Parameters can contain values in exponential notation. Check the Creo configuration option `param_dec_places`.



View Impact: Specifies general visibility settings of the parameter:

- Read-only: The input field cannot be modified.
- Hidden: Specifies whether the parameter is displayed. Hidden parameters are displayed in the *Hidden* tab if the configuration has been adjusted accordingly (gtp_show_hidden_params). Also saving these parameters depends on the configuration (gtp_save_hidden).
- Save: Specifies whether a parameter is adopted into the model when saving.

Tip: Use the *Save* option to specify temporary parameters to allow a conditional selection for another parameter without saving the temporary parameter in the Creo model.



Validity: Specifies the use of a parameter in parts, bodies, assemblies and drawings. Parameters that are invalid in an object type can neither be saved nor used.

Designate: Specifies whether to allow or prevent designations in PDM mode. When saving in GENIUS TOOLS Parameter, the parameters are flagged *Designate* in the Creo model.

Color: Specifies an individual color for the input field. Type the color name directly into the entry box or choose an appropriate color via the color circle  to select a color (Description of the color dialog  ⁴⁹⁰).

2. Title

Specifies the language-dependent (localized) names that will be displayed in the GENIUS TOOLS Parameter Name column.

Standard texts can be added via the button  (Description of the standard text selection dialog  ⁴⁹⁷).

3. Info

Language-dependent (localized) is stored under *Info*. In addition, an image can be stored for each parameter. The image and the localized information are displayed when the parameter name is clicked. URLs are automatically converted into clickable links in GENIUS TOOLS Parameter.

Use images and localized information to add support information about parameters.


Always specify URLs including the network protocol (http://, https://).

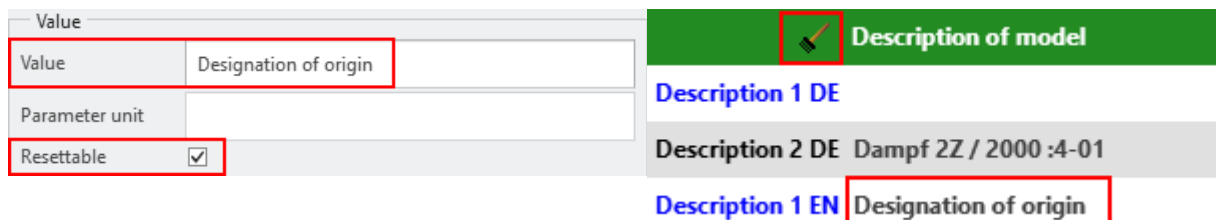
If images are changed later on, the change will only be displayed after restarting Creo.

4. Value

Value: Defines a default value. It is used when the parameter is not in the model and the field in the parameter form is not filled. This can be for example: "-", "not defined" or "n. d.".

Parameter unit: When the parameter type is *real number*, a unit can be specified here. The unit will be adopted by the model parameter when saving.

Resettable: Allows resetting the input value in the parameter form to the default value with the *broom symbol*  in the parent separator. The broom symbol is displayed if the Resettable checkbox is activated in at least one parameter. The function takes effect until the next separator.

Value		 Description of model Description 1 DE Description 2 DE Dampf ZZ / 2000 :4-01 Description 1 EN Designation of origin
Value	Designation of origin	
Parameter unit		
Resettable	<input checked="" type="checkbox"/>	

Activated function in the
GENIUS TOOLS Parameter Editor

Applied function in the
Parameterformular

Warning: The Reset function can only be used if there is a parent separator. The reset icon can only be displayed there.


Only from list: Specifies whether users can only select from predefined list items. It is not possible to directly type into the input field. This option requires a database or a list connection.

With database connections typing into the input field can be used for searching, but only list values will be accepted. When this option is disabled, a field will also accept free input.

Description: The description entered here will be adopted by the model as parameter description on saving.

String convert: Allows automatic conversion of the input to upper or lower case. This option is only available for string type parameters.

Required field: Specifies whether an input field must be filled in by the user. Mandatory field are marked with an asterisk in the parameter form *.

Format: Specifies a regular expression to verify the input. If a format has been specified for an input field, the users will be prompted when the input does not comply with the format .

The format can only be checked if the parameter is configured for writing into the model (Save is set) and is not hidden.

Tip: Activate the checkbox behind the *Format* input field to automatically match values that do not match the format (if possible).

Example

Format: `^\w-\d{2}-\d{5}$`

Meaning: "1 Character", "Minus", "2 Digits", "Minus", "5 Digits"

The following entry is incorrect: ABZ-23-00005 (there are three characters at the beginning and not one).

If the check box is not set, the correction must be made by the user.

If the check box is set, the entry is automatically corrected to Z-23-00005 (i. e. characters are removed at the beginning).

Maximum length: Specifies the maximum length for the input in characters.

Show in model tree: Specifies whether the parameter will be shown in the model tree when the model tree is displayed.

Column width in model tree: Specifies the column width for display of the parameter values in the model tree.

5. List

The list configuration is located on the right side of the detailed view.

The list configuration allows to deposit a list of values for a parameter from the following sources:

- text and CSV files: see [File-based lists for parameters](#)³⁶⁴,
- SQLite or Access databases (until version 2003): see [Database file lists for parameters](#)³⁶⁵,
- webserver database systems: see [Webserver lists for parameters](#)³⁶⁶.
- REST API: see [REST API lists for parameters with Windchill](#)³⁶⁸

Please pay attention that file-based lists have to be under *gt_resource_folder*. Also UNC-paths (`\\servername\sharename\path`) are possible, thereby the accessibility has to be ensured.

The configuration depends on the list type.

Electroplated coating ISO 4042

Parameter	Value
Chem. element	
Thickness [µm]	
Brightness / color	10
Coating code	15
Color	25
Coating or RAL-code	30
	5
	8
	beliebig

A list from a text file was deposited in an input field

ERP Properties Status

Project no. -

Specification of model

Parameter	Value
Name model DE	Dampfmaschine ZZ
Description model DE	Dampf ZZ / 2000 :4-01
Name model EN	steam engine ZZ
Description model EN	Dampf ZZ / 2000 :4-01
Desc. semifinished part	
Box dimension -	

Valid values

EN*	DE
2-jaw chuck	Zwei Backen Futter
3 x signal lamp	Dreifach Meldele...
3-jaw chuck	Drei Backen Futter
3-phase servomotor	Drehstrom-Servo-...
3-phase spindle motor	Drehstrom-Spind...
4-jaw chuck	Vier Backen Futter
6-jaw chuck	Sechs Backen Futt...
abrasive body	Schleifkörper
abrasive cloth	Schleifleinen

Connected fields

A database was deposited in an input field

6. Table connections

In a table connection, two or more input fields are connected. Specify the connected parameters³⁶⁸ and, if necessary, limit the values with a filter list.³⁶⁹

7. Functions

For each input field you can assign an OnChange functions (Javascript)³⁷⁰ as well as various parameter functions.³⁷¹

File-based lists for parameters

File-based lists can be created using plain text files (TXT) or CSV files (Comma-Separated Values).

Text files allow the easy selection of entries from the file (one entry corresponds to one line). CSV files allow a more complex configuration with table connections similar to databases.

To configure a list from a text file you have to specify the file and the file encoding it uses. The list can be used after saving the parameter definition and reloading it in GENIUS TOOLS Parameter.

List	
List	Text/CSV file
File	None
File-coding	SQLite database
Column	Webserver
	Text/CSV file

List configuration with CSV file

Warning: Pay attention to the file encoding when creating list files! Only Unicode and ANSI are supported.

File: Specifies the list file.

Select the list file using the three dots button or create a new file using the Edit button (pen icon). If a file has already been specified, Edit opens the file using the editor specified in the configuration.

Only the file name is displayed, not the path.

File encoding: Specifies the file encoding of the file.

If the selected file is a text file, no further configuration is required.

Column: Specifies the CSV file column to retrieve the values for the input field.

Displayed columns: Specifies the CSV file columns displayed in the parameter form to allow the user to make a selection.

List	
List	Text/CSV file
File	gtp_list_defined_mass.csv
File-coding	Unicode/UTF-16
Column	MASS_SOURCE
Displayed columns	SOURCE_DE, MASS_SOURCE

gtp_list_defined_mass.csv - Editor		
DATEI	Bearbeiten	Format
SOURCE_DE	SOURCE_EN	MASS_SOURCE
berechnet	calculated	GEOMETRY
vorgegeben	defined	PARAMETERS

Column (blue) and displayed columns (red)

Tip: Activate the *Read-only* option for a list field. This way the list entries can be selected but cannot be modified.

Database file lists for parameters

Database files can be used for lists: SQLite or Access databases (until version 2003).

The following information needs to be entered.

Database name: Specifies the name of the database file. Via the button behind the field the file can be selected via the file browser. Only the file name is always displayed, not the entire path to the database.

Table: Specifies the name of the table in the database.

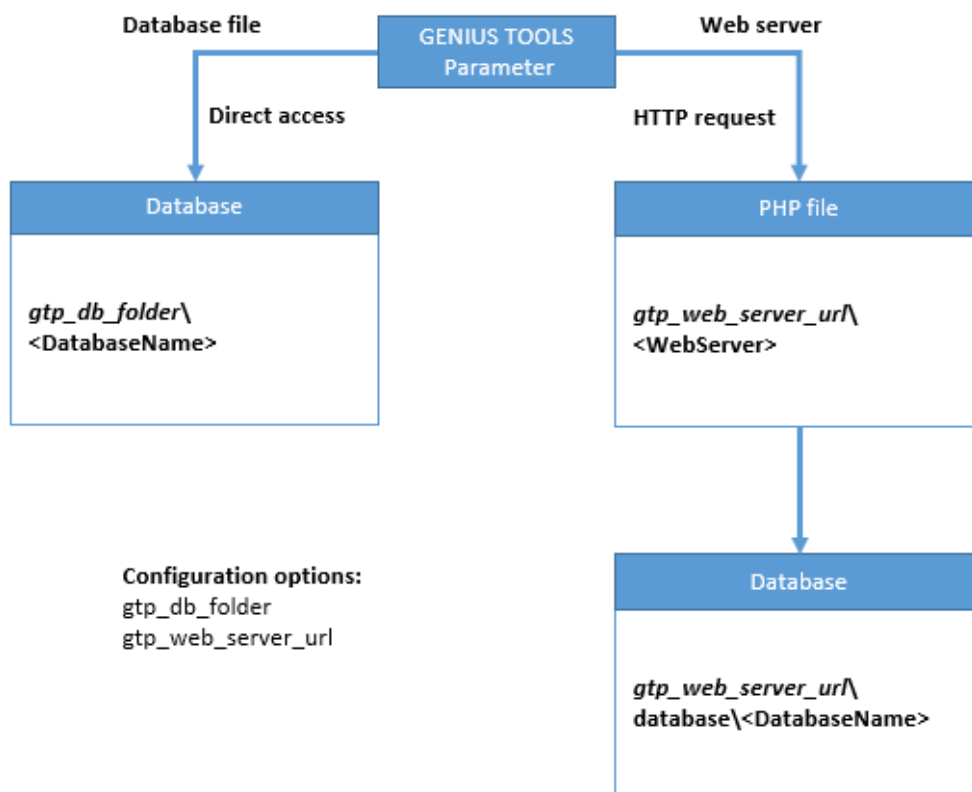
Column: Specifies the table column to retrieve the values for the input field.

Displayed columns: Specifies the table column displayed in the parameter form to allow the user to make a selection. If multiple columns should be displayed, specify them comma-separated.

The configuration of connected parameters is done in the same way as the configuration of CSV files with connected parameters.

Please note: Do not use mathematical symbols +-~/* for describing field names. This can otherwise lead to the problem that field names cannot be recognized unambiguously.

Overview of database access paths for lists



Access to database files or web server

Webserver lists for parameters

In addition to database files, GENIUS TOOLS Parameter can also use databases on a webserver.

An HTTP-POST request is sent to a webserver awaiting a JSON (JavaScript Object Notation) response.

Please note: Any filtering of the data has to be implemented on the webserver. A selection list provided by a webserver as a JSON file cannot be filtered further in the parameter form.

Webserver: Specifies a URL that returns valid JSON data. You can enter a part path. This will be appended to the value of the configuration option `gtp_web_server_url`. If your input contains a colon (as in `http://localhost/testdata.json`), it is assumed to be a complete URL, and `gtp_web_server_url` will not be prefixed. You can find an example PHP file in the installation directory under `Tools/gtp_server`.

Database name: Specifies the database to be used. Is passed as POST parameter `DB`.

Table: Specifies the name of the table in the database. Is passed as POST parameter `TABLE`.

Database name and table may be left blank if they are referred to automatically by the webserver PHP page

Column: Specifies the table column to retrieve the values for the input field. Is passed as POST parameter `SFIELD`. The column names under **Column** and **Displayed columns** are also used to label the columns in the user interface.

Displayed columns: Specifies the table column displayed in the parameter form to allow the user to make a selection. Is passed as POST parameter `FIELDS`. If the column with the values for the input field (**Column**) is not listed under **Displayed columns**, it will be added to the POST parameter automatically.

Example of an HTTP-POST request

`DB=gtp_parameter.db&TABLE=namen&SFIELD=german&FIELDS=english,german&SEARCH=**%`

Argument	Description
DB	Database name
TABLE	Name of the table in the database
FIELDS	Name of the columns to be returned
SFIELD	Name of the search field
SEARCH	Search string

Example of the expected JSON response

```
{ "timedb":10, "debug":"DB=gt_parameter.db TABLE=namen FIELDS=english,german
SFIELDS=german SEARCH=**%", "varr":["Abdeckkappe|cover cap",
"Abdeckscheibe|cover disc","Abdeckung|cover"],"time":Verbrauchte Zeit}
```

The output *varr* contains the data required for the parameter selection. The other parts of the JSON response could also be passed empty.

Argument	Description
timedb	Duration of the database query
debug	Arbitrary text output
varr	Output of the data from the defined fields. Each record is delimited by straight double quotation marks and a comma, the fields are separated by pipe symbols ().
time	Duration of the entire query

REST API lists for parameters with Windchill

The REST API list function is available for PTC Windchill with 2.5 REST Services installed. BasicAuth authentication protocol is supported.

Webserver: Defines the web server. *entityName* and *propertyName* must be customized according to the Windchill configuration. After entering the web server, you will be prompted to authenticate to Windchill.

Languages: Defines the displayed languages. The languages entered must match the browser language codes: e. g. en, de or fr-CH, en-US.

Column: Defines the column from which the values for the input field are taken.

Displayed columns: Defines the columns of the table to be displayed for selection in the parameter form.

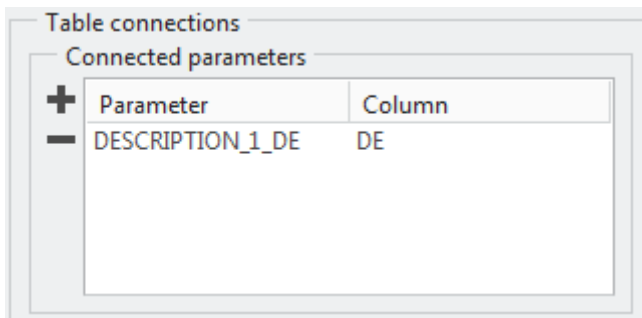
Connected parameters

Lists with table connections can be specified for parameters. A table connection interlinks two or more input fields.

For instance, if a field lets the user make a language-dependent selection, a connected field can be automatically filled with the appropriate entry for another language.

In order to be able to use connected parameters, there has to be a list of values that contains a column for each affected parameter and a row for each combination of values.

All parameters that you want to be connected need to be added to the connected parameters list. You have to specify the corresponding column name for each parameter.



Connected parameter and column in the database

After saving and reloading it in GENIUS TOOLS Parameter the updated list is available with the connected input fields.

Please note: If values for connected parameters are set automatically, the *first* row of the value list is used that has the defined value for the leading parameter. Parameter values can be set automatically, e. g., when a JavaScript function is executed, when values are read from the model, or when the configuration option `gtp_check_connections` is set to add missing values after loading the model.

Filters for parameters

Filters are used for 1:n-relations in tables. Specifying one parameter value controls the selection options of other parameters.

In order to be able to use filters, there has to be a list of values that contains a column for each affected parameter and a row for each combination of values.

Example


The code for a galvanic coating per ISO 4042 is filtered by the coating material, coating thickness and coating gloss and color parameters. The users can now enter an ISO code to fill in the coating material, coating thickness and coating gloss and color automatically. Alternatively, they can enter values for coating material and coating thickness to filter the selection options for coating gloss and color and for the ISO code.

The parameters that are required to filter another parameter need to be added to the list under *Filter*. You have to specify the corresponding column name for each parameter.

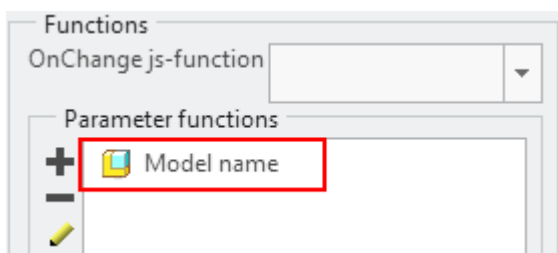
Filters	
+	Parameter
-	Column
	COATING_ELEMENT
	chem_element
	COATING_THICKNESS
	thickness
	COATING_APPEARANCE
	gloss_color

Field functions

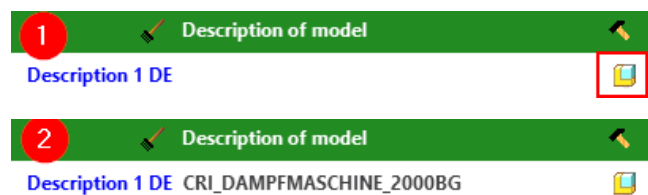
Field functions fill an input field with the value of a parameter form after one click on the symbol at the end of the input field.

With the *fill hammer*  in the header you can fill ["hammer"] all values of a section that have field functions with one click.

Example: Field function *Model name* at parameter *Description_DE*:

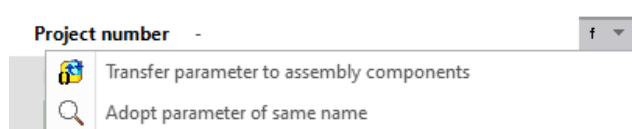
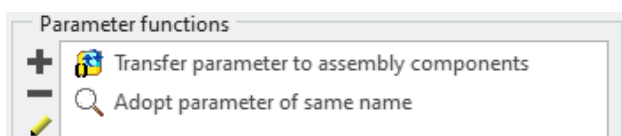


Parameter Editor



Filling the input field in the parameter formular

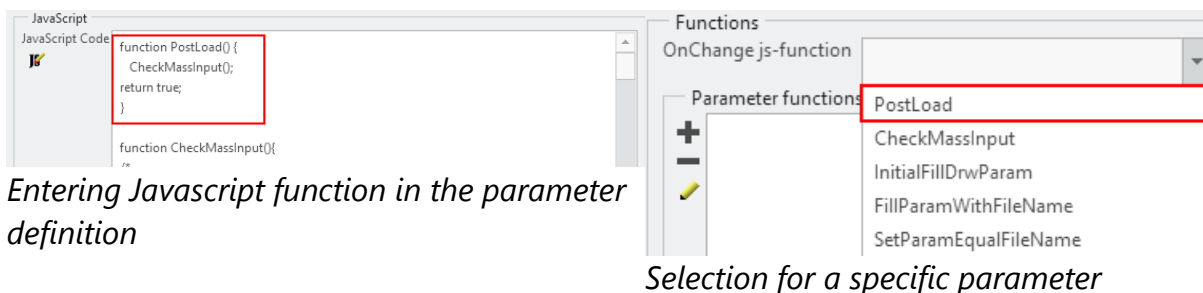
If you configure multiple functions for an input field, these functions will be displayed in a drop-down list for the input field in the parameter form. When field functions are executed using the hammer button, only the first field function will be used.



OnChange functions (Javascript)






Each input field can have OnChange functions attached to it. OnChange is executed when the value of a form element is changed automatically (e. g. by value tables) or manually by leaving the input field or pressing Enter.







OnChange functions have to be entered in the [Javascript](#) section of a parameter definition ³⁸⁸ before they are available for selection as a field function.








Adding field functions

Use the **+** button to add one or more parameter function for an input field. These are available:

Icon	Parameter function	Description
	Date	Allows to apply the current date to the input field. Several prebuilt date formats are available. Alternatively, the required format can be manually entered.
	Login name	Adopts the current Windows user name or STOOLS User (Log/SHORT) into the input field.
	Adopt parameter of same name	Adopts the parameter value of a Creo parameter of the same name from another part or assembly into the input field.
	Free parameter search	Adopts the value of a parameter from the Creo <i>Parameter selection</i> dialog into the input field. Note: The function supports the transfer of only one parameter. If multiple parameters are selected, only the first parameter value will be adopted.
	Model name	Adopts the model name from the currently active model in GENIUS TOOLS Parameter the input field. In a drawing the active model is adopted as parameter value.

Icon	Parameter function	Description
	Drawing name	Adopts the drawing name from the currently active model in GENIUS TOOLS Parameter into the input field. Note: Can only be used in Drawing mode.
	PTC Common Name	Adopts the PTC Common Name from the currently active model in GENIUS TOOLS Parameter into the input field.
	Material selection	Opens GENIUS TOOLS Material and adopts the selected material into the input field. Make sure you have a properly maintained material database. Missing or double materials in the database can cause models that cannot be regenerated. Make sure to only use MTL-files when material selection is used as a field function. GENIUS TOOLS for Creo does not support MAT-files.
	Bounding box	Adopts the bounding box dimensions into the input field. Requires further input, see extra dialog ³⁷³ below.
	Name Generator	Adopts a generated name from GENIUS TOOLS Name Generator into the input field. Parameter looks for the name generator database in the directory specified by the configuration option <code>gtng_folder</code> .
	Use predefined parameter value	Adopts the parameter value of a Creo parameter that must be specified during configuration from a model or part that has to be selected at runtime.

Icon	Parameter function	Description
	Copy parameter value into active drawing model	Copies the parameter value into the parameter of the active model. Note: Can only be used in Drawing mode.
	Copy parameter value from active drawing model	Copies the parameter value from the parameter of the active model. Note: Can only be used in Drawing mode.
	Transfer parameter to assembly components	Opens the GENIUS TOOLS Value Transfer ⁴³¹ dialog window or predefined values from a configured XML file, see below ³⁷⁴ . Note: Can only be used in assembly mode. Note: Parameters of the active sub-assembly are passed on, not of the main assembly!
	Replace variable	Replaces a variable specified in the editor with its value and adopts it into the input field.
	Run JavaScript	Executes a JavaScript function.

Bounding box


The function asks for information on how to specify the dimensions of the bounding box. The table illustrates this for a bounding box with the dimensions x=151, y=133.5, z=90.0.

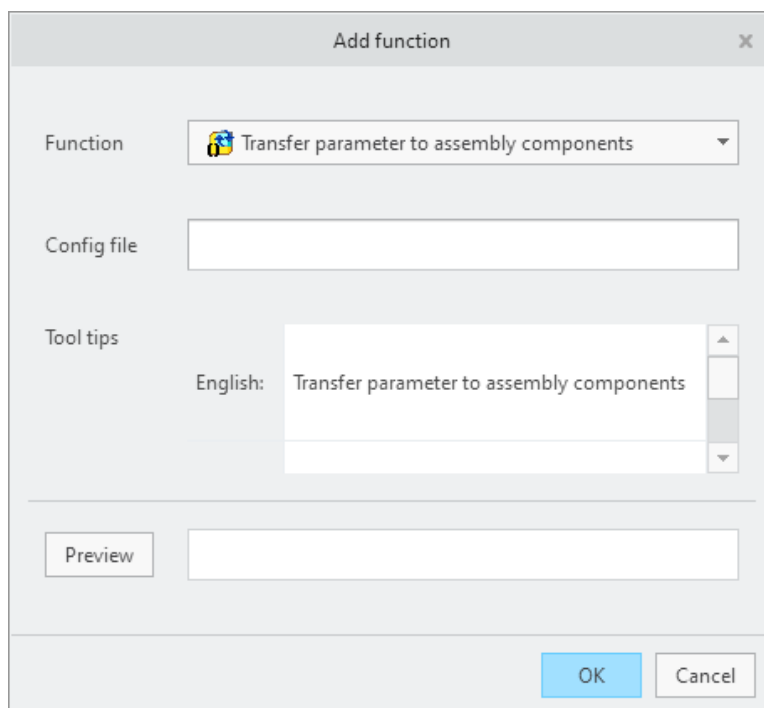
	Input example 1	Input example 2
Complete prefix	Dimensions:	Size:
Prefix X	x=	
Prefix Y	,y=	x
Prefix Z	,z=	x
Complete postfix		
Position after decimal point		1

	Input example 1	Input example 2
Shown values		xyz
Delete followed zeros	ja	nein
Result	Dimensions:x=151,y=133.5,z Size: 151.0x133.5x90.0 =90	

Transfer parameter to assembly component: in assembly mode


This function allows you to transfer parameter values from an assembly to its individual components. The field function can be used

- without predefined values: the GENIUS TOOLS Value Transfer dialog box opens, or
- with predefined values: by clicking on the  icon the values from a configuration file (XML file) are entered. The entire section can thus be filled in with the [fill hammer](#).³⁴²



Procedure for using the function with filtered values

Creating a configuration file

1. Open the GENIUS TOOLS Value Transfer component.
2. In the [Filter section](#)⁴³⁴, add a parameter and define its values, e.g. @mdl@ contains CRI_DE.
3. Using the  button in the command bar, save the specifications as an XML file in a folder, e. g. as in the screenshot above: cri_d.

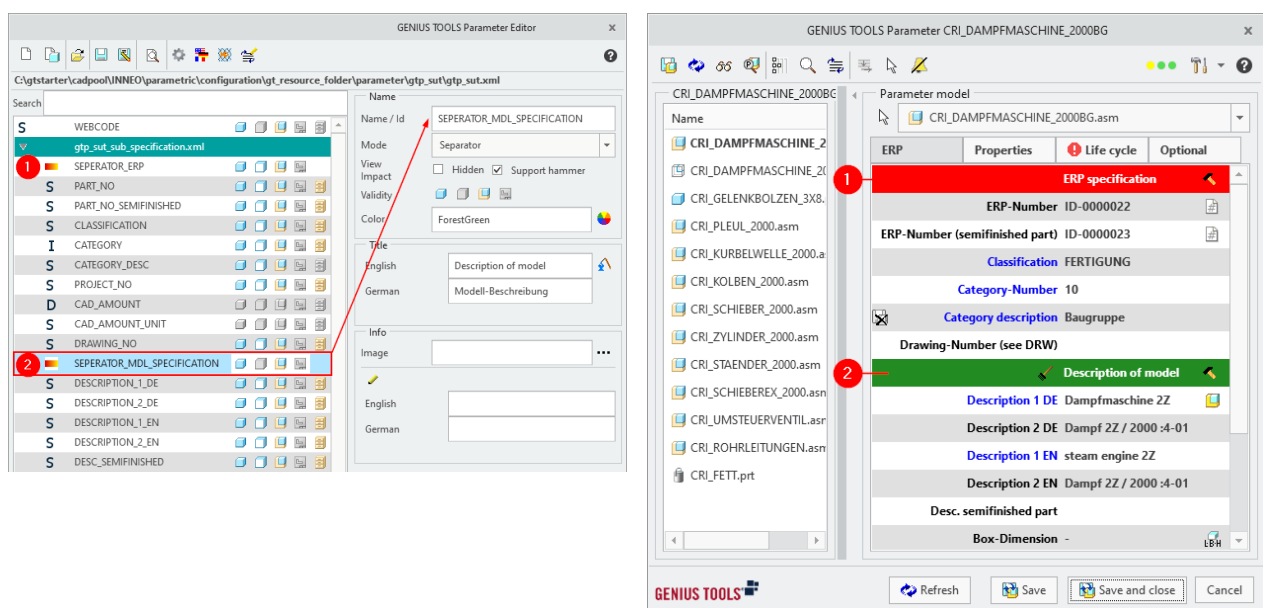
Specifying the configuration file

- In the Parameter Editor in the dialog *Add function* specify the complete path to the XML file or only the file name with file extension (in the screenshot above *cri_d.xml*). Variables can be used in the name specification.

If only the file name is specified, the folder specified in the configuration option `gtp_gtr_rules` will be read. (Default is `%gt_resource_folder%parameter\`).

13.3.2.6 Adding separators

Use separators to logically structure the parameters, e. g. as section headers. Separators are also needed to reset input fields and to have them filled in by field functions automatically.



Setting of a separator in the
GENIUS TOOLS Parameter Editor

Result in the
GENIUS TOOLS Parameter dialog

The detail view for separators is divided into different sections:

Name



Name: Specifies the name of the separator.

Mode: Specifies whether it is a separator or a parameter.

View/Impact: If a separator is marked *Hidden*, it is only used to structure the view in the Editor. It is not displayed in GENIUS TOOLS Parameter.



If the setting *Support hammer* is active, the button for automatically filling in values is displayed in the separator as soon as field functions are defined for parameters.

Validity: Specifies the visibility of the separator in Part, Assembly and Drawing mode.

Color: Specifies a custom color for a separator. Type the color name directly into the entry box or choose an appropriate color via the color selection  (Description of the color dialog  ⁵⁹⁰).

Title


Specifies language-dependent (localized) names that are displayed in the GENIUS TOOLS Parameter forms section. Leave the title empty to create an empty separator.

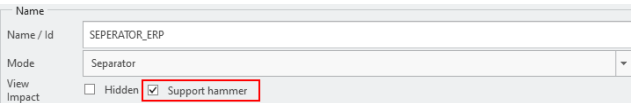

Standard texts can be added via the button  (Description of the standard text selection dialog  ⁴⁹⁷).

Info

Language-dependent (localized information) and a preview image can be deposited at Info. They are currently not displayed in the forms section.


Automatic filling of parameter values: Create hammer symbol

In the parameter form, users can use the *hammer symbol*  to have parameter values filled in automatically. The *hammer symbol* has an effect up to the next Separator. The symbol is displayed if a field function is stored in at least one parameter and if the *Support hammer* setting is set in the Separator. The functions are defined in each parameter. To display the symbol, it is mandatory that a Separator is located above the parameters in the form.


Setting	Result
 <p>The screenshot shows the 'Parameter Editor' interface. The 'Name / Id' field contains 'SEPARATOR_ERP'. The 'Mode' dropdown is set to 'Separator'. Under the 'View Impact' section, the 'Support hammer' checkbox is checked and highlighted with a red box.</p>	 <p>The screenshot shows the 'Parameter form' result. It features a green header bar with a hammer icon and the text 'ERP specification'. Below this, a table lists parameters: 'ERP-Number ID-0000008', 'ERP-Number (semifinished part) ID-0000009', 'Classification FERTIGUNG', 'Category-Number 10', and 'Category description Baugruppe'. A red box highlights the first three rows, indicating the automatic filling of values.</p>

Activated function in the Parameter Editor

Applied function in the Parameter form

The broom symbol for resetting the values is also displayed in the separator, but set individually for each value. As soon as a value can be reset, the broom symbol appears. See Adding parameters > Range: Values  ³⁶².

13.3.2.7 Editing parameter definition list

Open the list of parameter definitions in the command bar with the button . It shows the content of the file *gtp.lst*. Entries can be edited directly in the dialog.

The fields above the list show the defined location of the file (configuration option: gtp_lst) and the current selection parameter (configuration option: gtp_file_param).

Parameter definition list: P:\configuration\gt_resource_folder_dev\parameter\gtp.lst

Selection parameter: MC_CHECKTYPE

	Valid	Parameter value	Description	File
0	<input checked="" type="checkbox"/>	FERTIGUNG	Fertigung	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de.xml ...
1	<input checked="" type="checkbox"/>	NORM	Normteil	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de_norm.xml ...
2	<input checked="" type="checkbox"/>	KAUF	Kaufteil	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de_norm.xml ...
3	<input checked="" type="checkbox"/>	BEISTELLUNG	Beistellung	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de.xml ...
4	<input checked="" type="checkbox"/>	SONSTIGES	sonstiges	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de.xml ...
5	<input checked="" type="checkbox"/>	NORM-HALBZEUG	Norm-Halbzeug	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de_norm.xml ...
6	<input checked="" type="checkbox"/>	BLECH	Bechteil	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de_smt_to_partno.xml ...
7	<input checked="" type="checkbox"/>	SUT_INT_DE	SUT INTERNATIONAL DE	%GT_RESOURCE_FOLDER%\parameter\gtp_int_de\gtp_int_de.xml ...
8	<input checked="" type="checkbox"/>	SUT_DE	SUT DE PMM	%GT_RESOURCE_FOLDER%\parameter\gtp_sut_de\sut_de.xml ...

Save and close Cancel

Add new entries using the (+) button. Click on a line number and use the minus button (-) to remove an entry.

Click in a cell of the table to make changes.

The individual line columns have the following meaning:

Valid: Activates or deactivates the parameter definition. Deactivated parameter definitions are ignored on selection.

Parameter value: Defines the value of the selection parameter for a parameter definition (default: MC_CHECKTYPE).

Warning: Make sure to assign parameter values only once.

Description: A meaningful description of the parameter definition. Descriptions can also be used without a parameter value and a definition as comments or for separation.

File: Specifies the location of the parameter definition.

Click the button  and select a parameter definition to add via the file browser.

Save the parameter definition list to apply changes.

13.3.3 Use cases

In this section you will find use cases of the GENIUS TOOLS Parameter Editor. The tasks in the following sections mainly build upon one another.

GENIUS TOOLS Parameter can be customized for use in companies in different ways. You can customize individual parameters of the parameter definition included with GENIUS TOOLS for Creo and Startup TOOLS or develop individual parameter definitions.

The following sections describe the second approach and also explain how to import parameter definitions from the Web.Link TOOLBOX Parameter manager.

Preliminary considerations on creating new parameter definition files

Before creating new parameter definitions think about your parameter concept:

- Which parameters should be added in the model, and in which mode (prt, asm, drw)?
- How should the parameters be structured (e. g. in sub-parameter definitions)?
- Which separators should be used for structuring the parameters?

Parameter definition backup

To avoid loss of data and restrictions in productive operation, we recommend to save a copy of your existing parameter definitions. In case of misconfiguration you can quickly restore a working configuration.

Customizing the configuration settings

Customize the GENIUS TOOLS Parameter configuration settings to your requirements. Closely inspect the following configuration settings:

gtp_db_folder

Inspect this configuration option if you want to use information from databases in your parameters.

gtp_designate

Adjust this configuration setting if you are using a PDM system such as Windchill.

gtp_file and gtp_file_param

Adjust the configuration options `gtp_file` (path to a parameter definition when no model parameter is found for assignment) and `gtp_file_param` (selection parameter specifying which parameter definition is used) to have your new settings being used.

Creating the parameter definition files

The following section describes the creation of new parameter definitions in various examples.

Warning: Note that sub-parameter definitions can also refer to other parameter definitions, but these references are not evaluated.

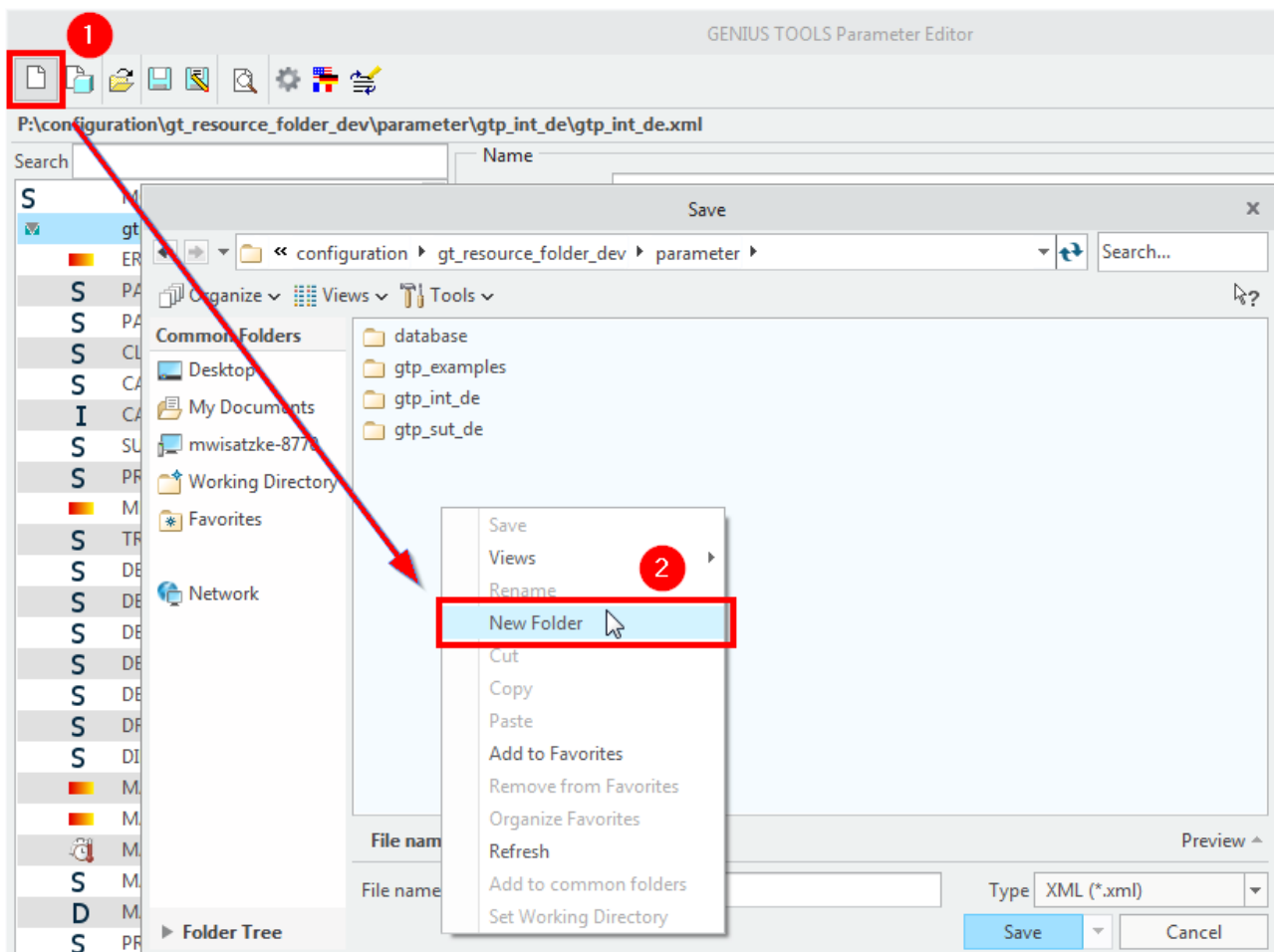
Another example shows the import of an existing parameter definition from the TOOLBOX Parameter manager.

The examples partly build on one another; so depending on the operational scenario the procedure of creating a parameter definition may be combined using the examples.

13.3.3.1 Creating a simple parameter definition

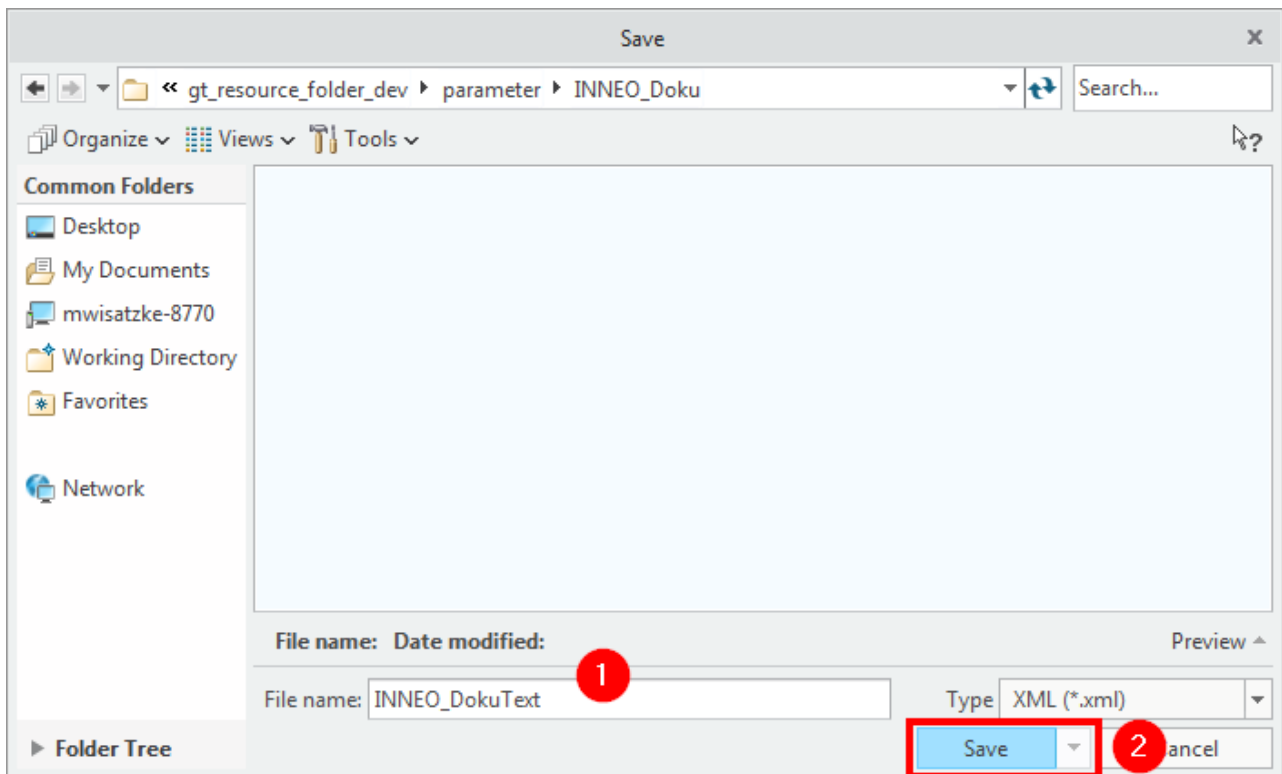
A parameter definition without references to other definitions consists of a single file. All parameters and separators are included in it and are displayed as a list in the parameter form.

1. Open GENIUS TOOLS Parameter Editor.
2. Create a subfolder in the `<GTfInstallationDirectory>\configuration\gt_resourcefolder\parameter\` following the name scheme `companyname_parameterdefinitionname`.



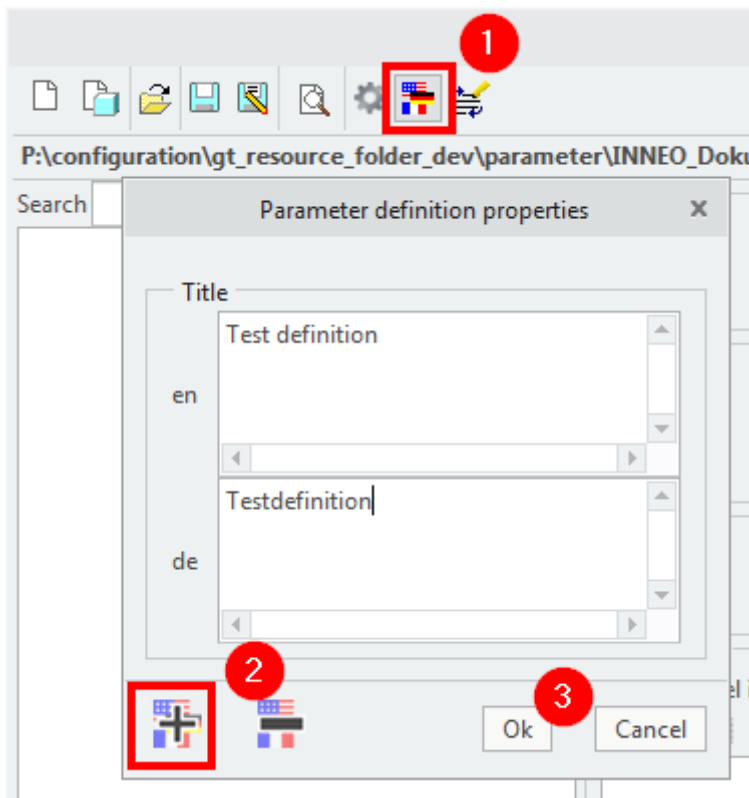
In the editor, click on "Create new parameter definition" (1) and create a new folder (2) via the context menu

3. Create a new parameter definition inside the folder. Click Save.



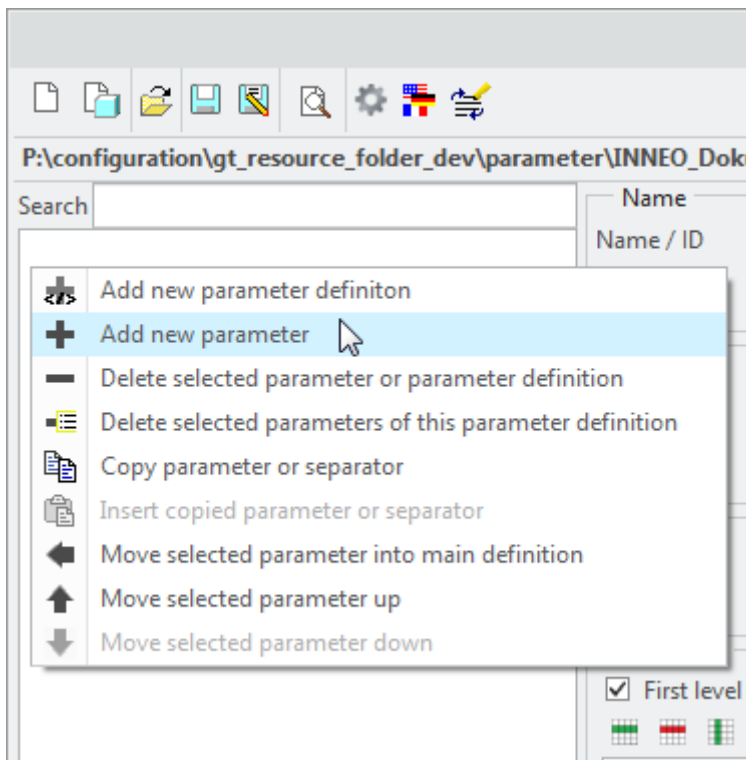
Enter a title (1) and confirm by clicking "Save" (2)

4. Click *Edit definition title*. Add additional languages to the parameter definition and enter the parameter definition titles for these languages. As soon as a new language is added, a name can be assigned to every parameter in this language.



Open the definition titles dialog (1). Add another language (2). Confirm the dialog after entering the definition titles (3)

5. Click Save in Parameter Editor. The file is created.
6. Create your parameters via the element list in the editor.



New parameters are added via the context menu in the element list of the editor

In the following sections, parameters with following properties are created as an example:

- Parameter with simple input with list selection
- Parameter with database connection
- Parameter as a mandatory field

In addition, a colored separator is created that allows to reset input fields.

Parameter with input field - Description

For non-standardized descriptions free input may be suitable. Note that *String* type parameters are restricted to a maximum of 80 characters.

Open the context menu in the element list and create a *String* type parameter. If you need descriptions in multiple languages, create an individual parameter for each language.

Create a new parameter

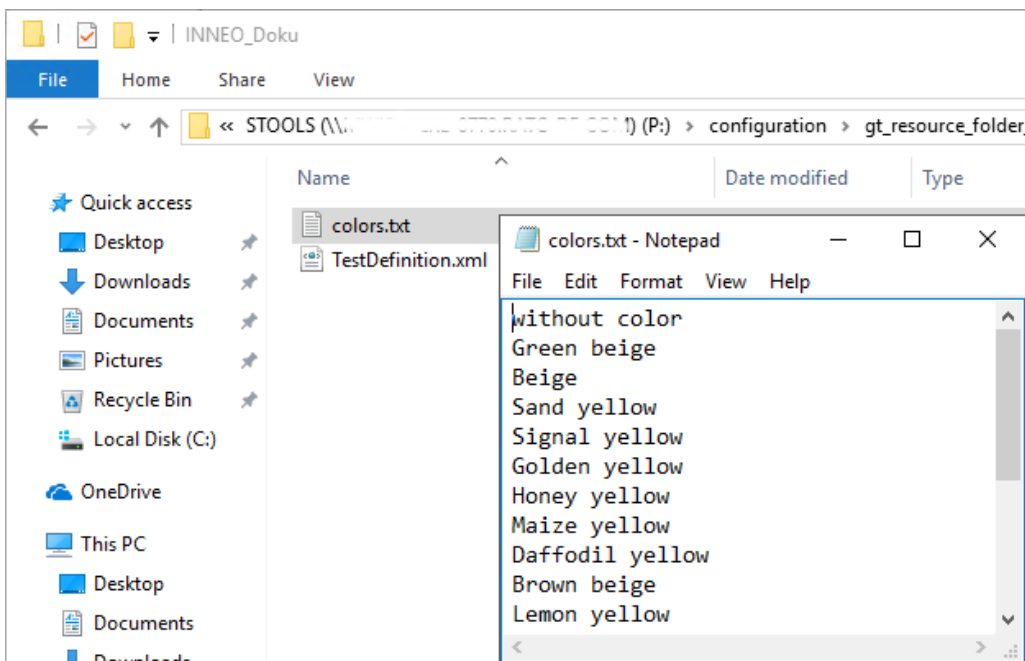
1. Enter a title for the field in *German* and *English* language.
2. Save the parameter definition.

For free input to into the field no further input is required.


Parameter with list selection - Color

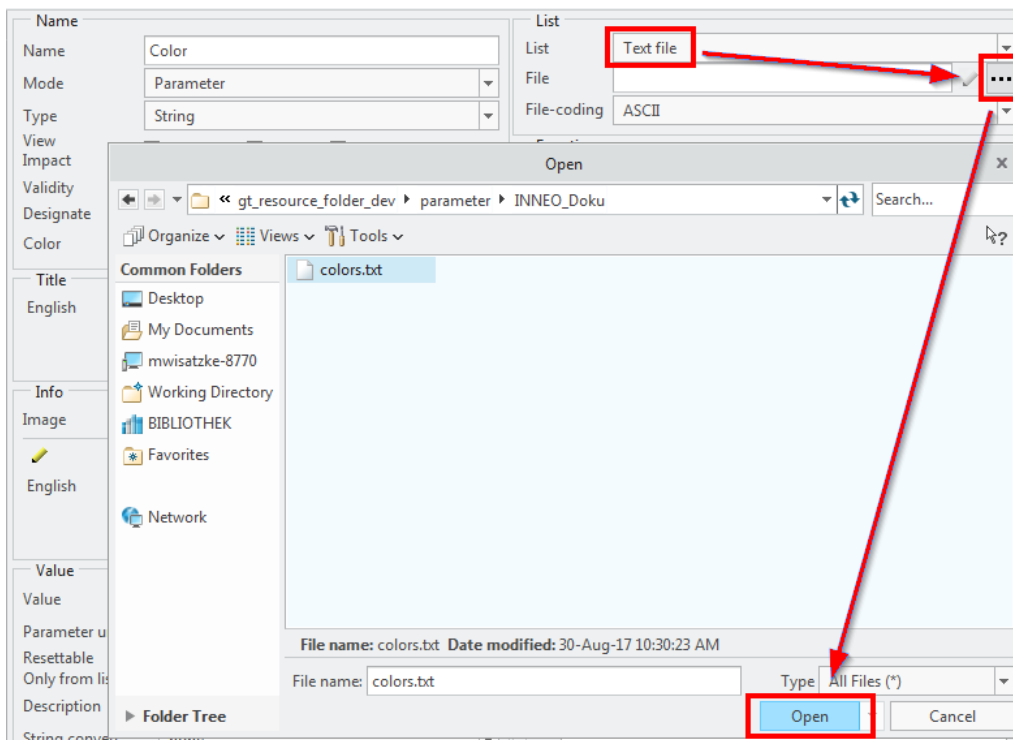
In order to use the same color names in all parts, a list can be deposited for colors.

1. Create a *String* type parameter.
2. Create a new file *colours.txt* in the folder containing the Parameter definitions using the Windows Text Editor.
3. Enter the desired color names and save the file.



Make sure to use the proper file encoding when saving!

4. In the Parameter Editor list section, select *Text file* and select the file you just have saved via the button .



For List select "File", then select the TXT file and confirm your input

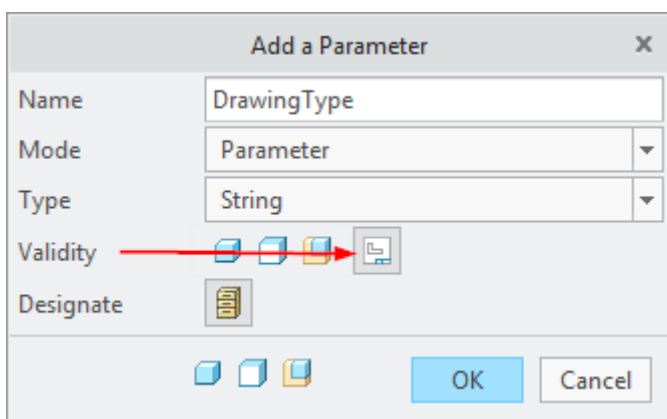
5. In the Value section, select the option *Only from list*.
6. Enter a title for the field in German and English language.
7. Save the parameter definition. The parameter with list selection is fully configured.

Parameter with list from database file - Drawing type


In the `gt_resource_folder\parameter\database` folder there is a file named `gtp_int_de.sqlite`. This file contains a selection of several standardized records related to different topics.

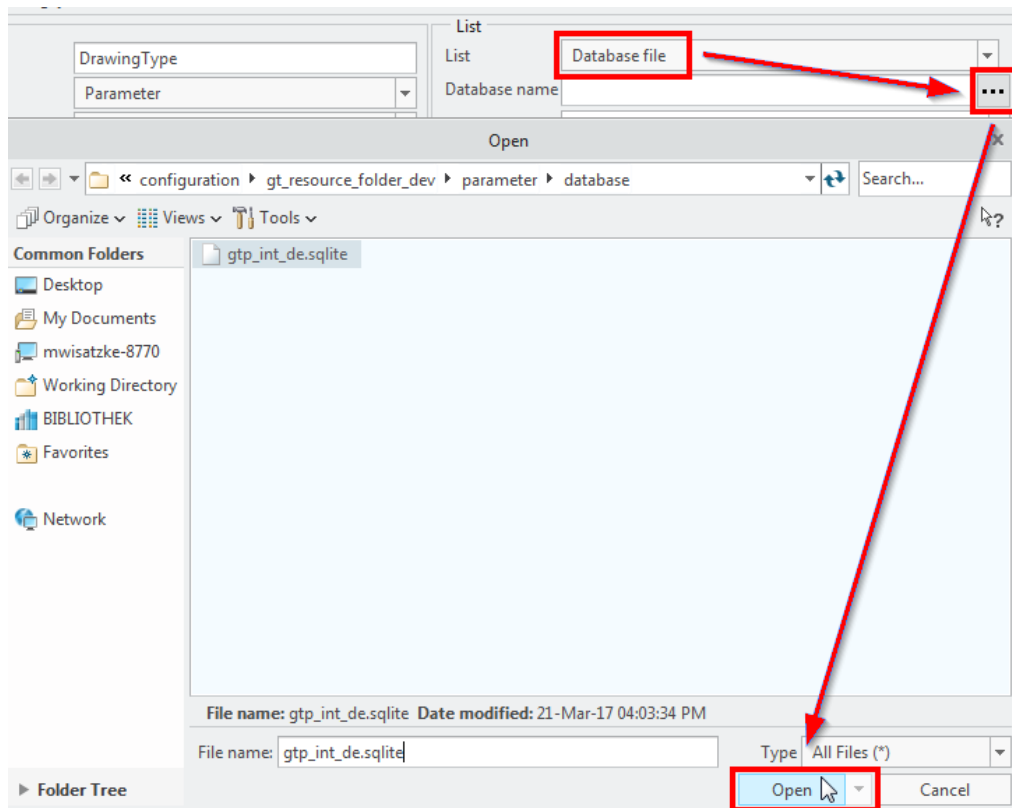
For drawing types there is a prebuilt table in the database that will be used in this example.

1. Create a *String* type parameter. For Validity select *Drawing*.



Make sure to use the proper validity setting

2. In the list section, select *Database file* and select the SQLite file in the ...
 \parameter\database folder via the button .



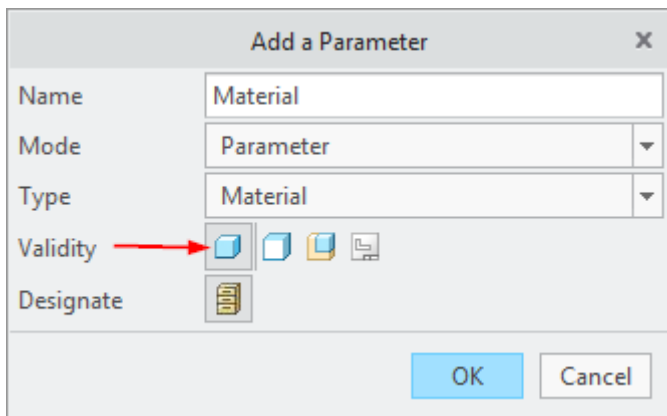
For List, select "Database", then select the SQLite file and confirm your input

3. For Table, select *document_type*.
4. In the *Column* field below, select *DE* for German.
5. For *Displayed columns*, select *DE* and *EN* with the Ctrl key pressed.
6. In the *Value* section, select the option *Only from list*.
7. Enter a title for the field in *German* and *English* language.
8. Save the parameter definition. The parameter with a list from a database is fully configured.

Parameter with mandatory field - Material

Materials are an important property of a model and are usually only added in parts (prt).

1. Create a new *Material* type parameter. Restrict the *Validity* to parts.

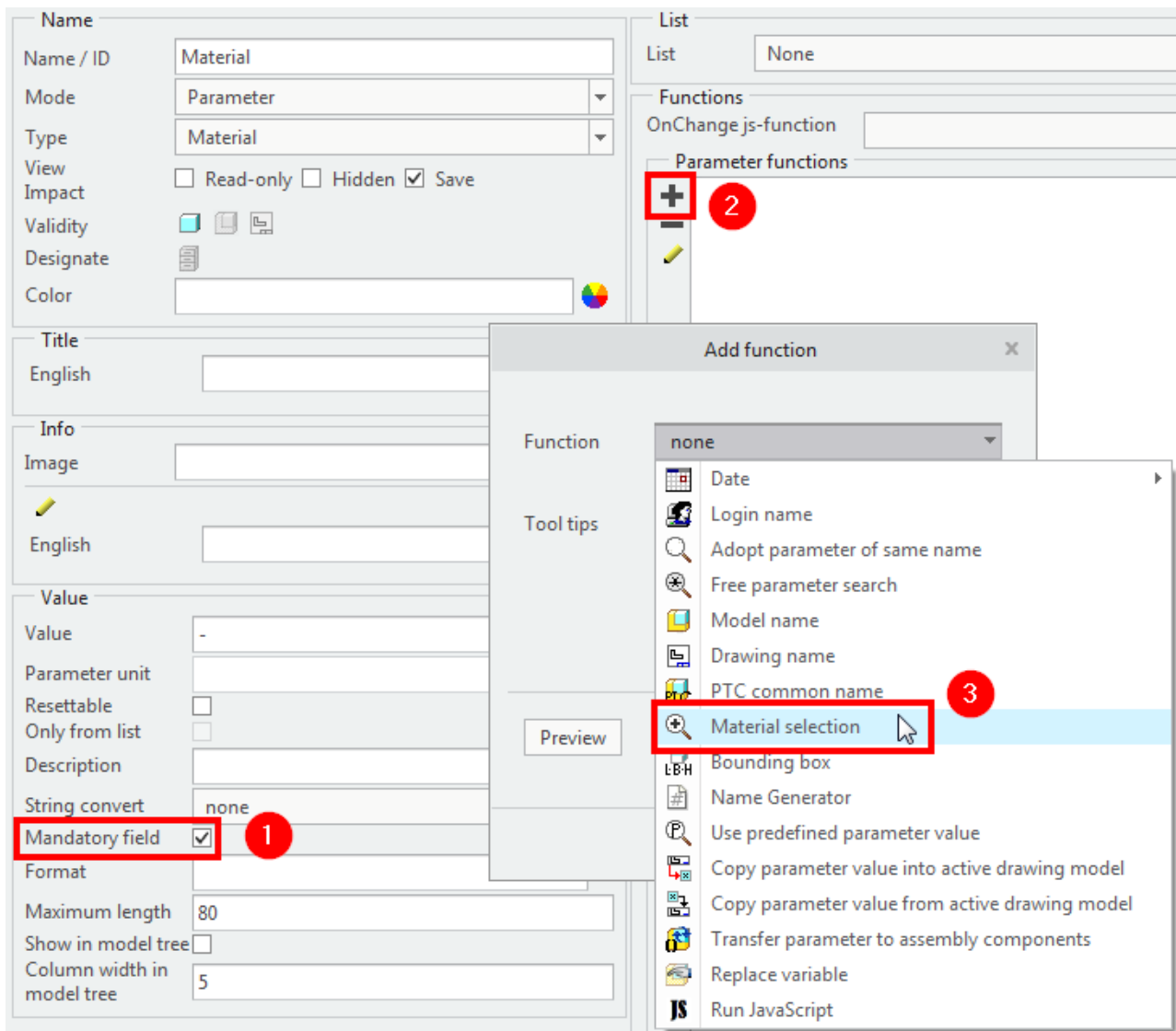


The screenshot shows a dialog box titled "Add a Parameter". It contains the following fields and options:

- Name:** A text input field containing "Material".
- Mode:** A dropdown menu set to "Parameter".
- Type:** A dropdown menu set to "Material".
- Validity:** A section with four icons. The first icon (a blue cube) is selected, indicated by a red arrow pointing to it.
- Designate:** A section with one icon (a yellow cube).
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

Make sure to use the proper parameter type at creation

2. Activate the *Mandatory field* option in the value section.
3. Go to the *Parameter functions* area and click *Add function*
4. and select *Material selection*.



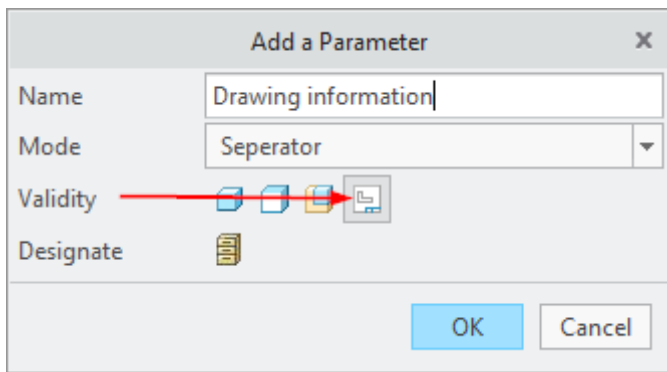
Activate the Mandatory field option (1), click on Add function (2) and select Material selection (3)

4. Enter a *Title* for the field in German and English language.
5. Save the parameter definition. The material parameter as a mandatory field is fully configured.

Colored separator with reset-function

Separators are used for logically structuring the input fields for parameters. If *Reset* and *Fill automatically* are required in a parameter form, at least one separator for each parameter definition has to be created.

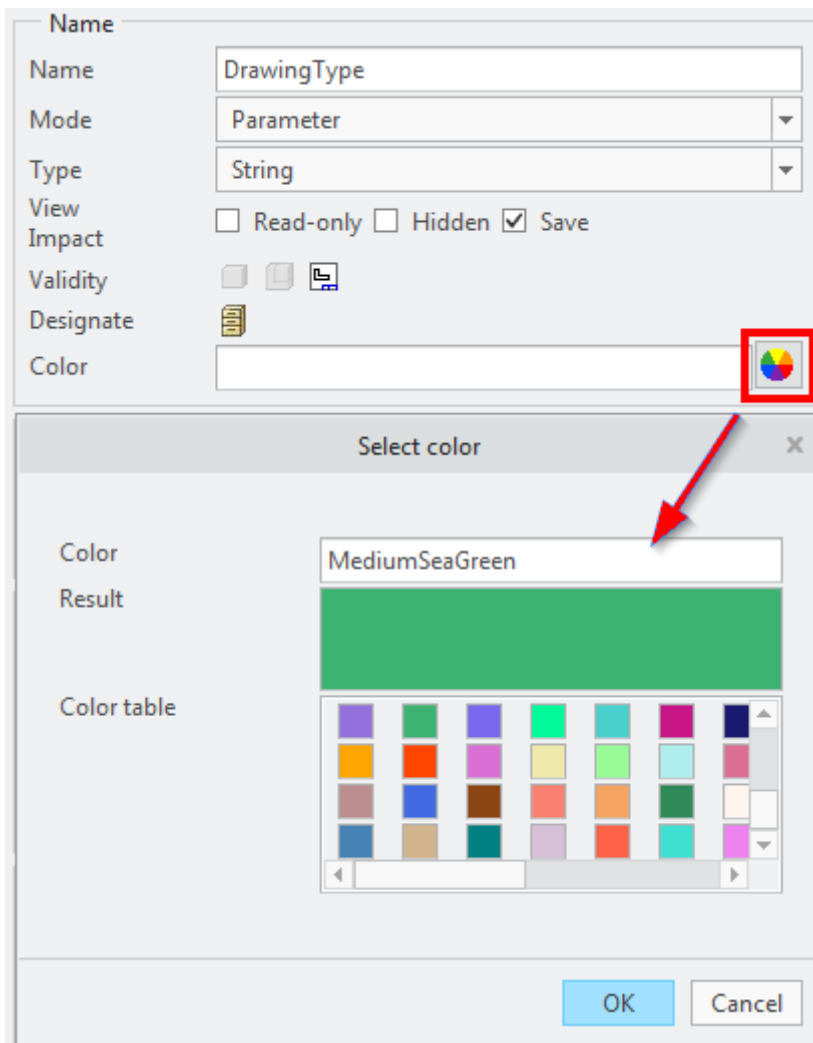
1. Open the element list context menu and click on *Add parameter*.
2. Select the *Separator* mode and enter a name.



3. Drag the separator in the list and drop it before the parameter that should be reset. Move separators by Drag-and-Drop in the parameter definition list.

S	PRO_MP_SOURCE					
D	PRO_MP_ALT_MASS					
D	MP_DENSITY					
	DrawingInformation					
S	DRAWING_NO					

4. Click on the color selection icon to the right of the *Color* field and select an appropriate color. Confirm the selection.



A color can be directly selected via the pen icon

5. Enter a title for the field in *German* and *English* language. The *Reset* function is not specified in the separator itself. The separator is only needed to display the reset button.
6. Click the parameter that should be reset. In this example the *DrawingType* parameter.
7. In the *Value* section, activate the *Reset* checkbox.
8. Enter a default value at *Value*. This will automatically be entered into the field when resetting.

Value	Assembly drawing
Parameter unit	
Resettable	<input checked="" type="checkbox"/>
Only from list	<input type="checkbox"/>
Description	
String convert	none
Mandatory field	<input type="checkbox"/>
Format	
Maximum length	80
Show in model tree	<input type="checkbox"/>
Column width in model tree	5

Enter a default value and activate the "Reset" function

9. Save the parameter definition.

13.3.3.2 Creating a Parameter definition with tabs

GENIUS TOOLS Parameter supports to split a parameter definition into several sub-parameter definitions. These are displayed as additional tabs in the parameter form and increase clarity.

In the following example an existing parameter definition ([Creating a simple parameter definition](#)³⁷⁹) is extended with a sub-parameter definition.

1. Open GENIUS TOOLS Parameter Editor and load the parameter definition from the example under [Creating a simple parameter definition](#)³⁷⁹.
2. Open the context menu in the element list. Click on *Add parameter definition*.
3. Enter a name for the parameter definition.

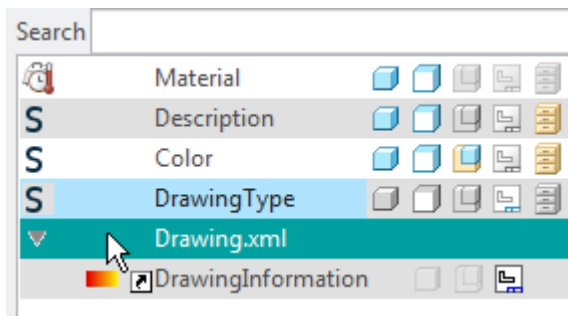
Add a parameter definition

Name: Drawing

OK Cancel

Create a new Parameter definition via the context menu

4. Drag-and-Drop the parameter into the new parameter definition.



Drag and Drop the desired parameters into the new sub-parameter definition

5. Save the parameter definition.

13.3.3.3 Importing old parameter definitions

GENIUS TOOLS Parameter can import parameter definitions that were created with the Web.Link TOOLBOX Parameter manager.

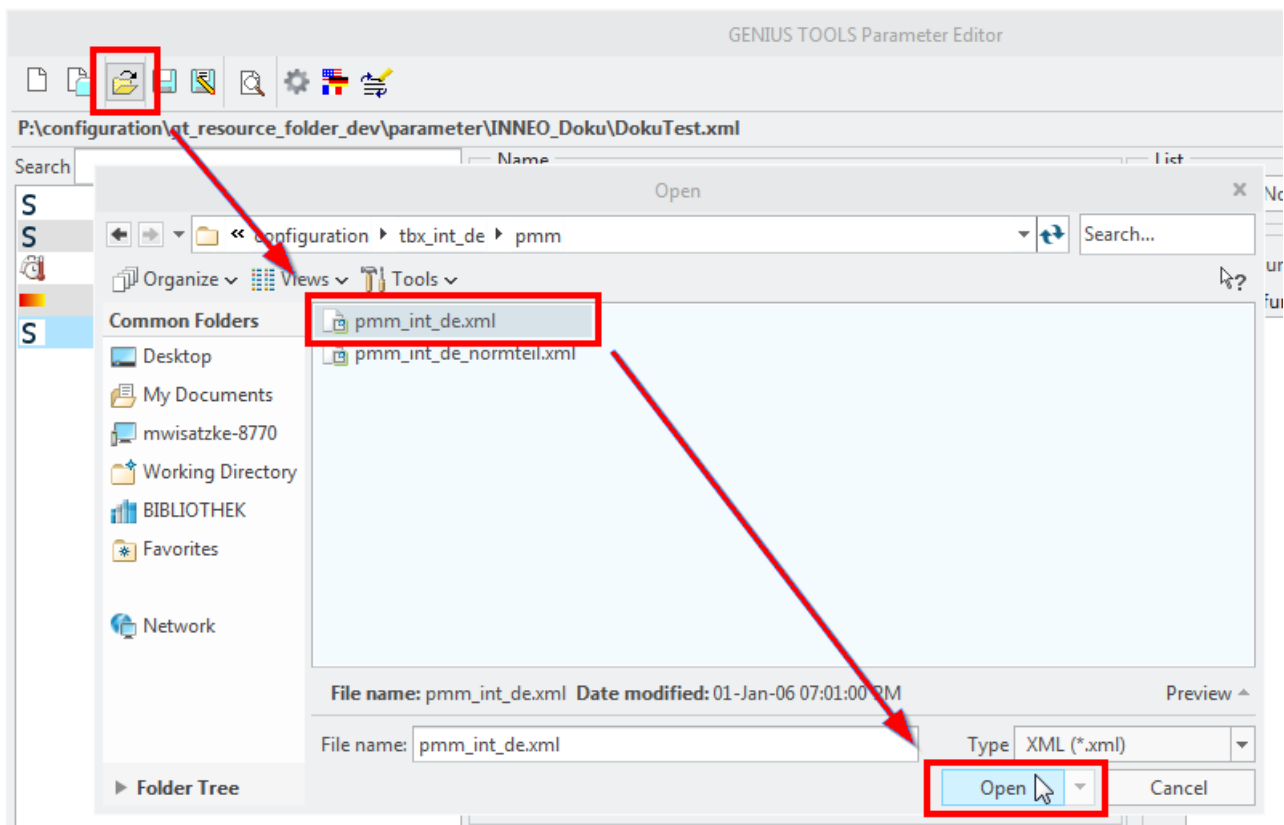
Before importing, the following two preliminary considerations should take place:

- Does the visibility/validity of parameters in the individual modes (prt, asm, drw) need to be customized?
- Should the existing parameters be split into sub-parameter definitions for logical structuring?

Proceed as follows to import a parameter definition:

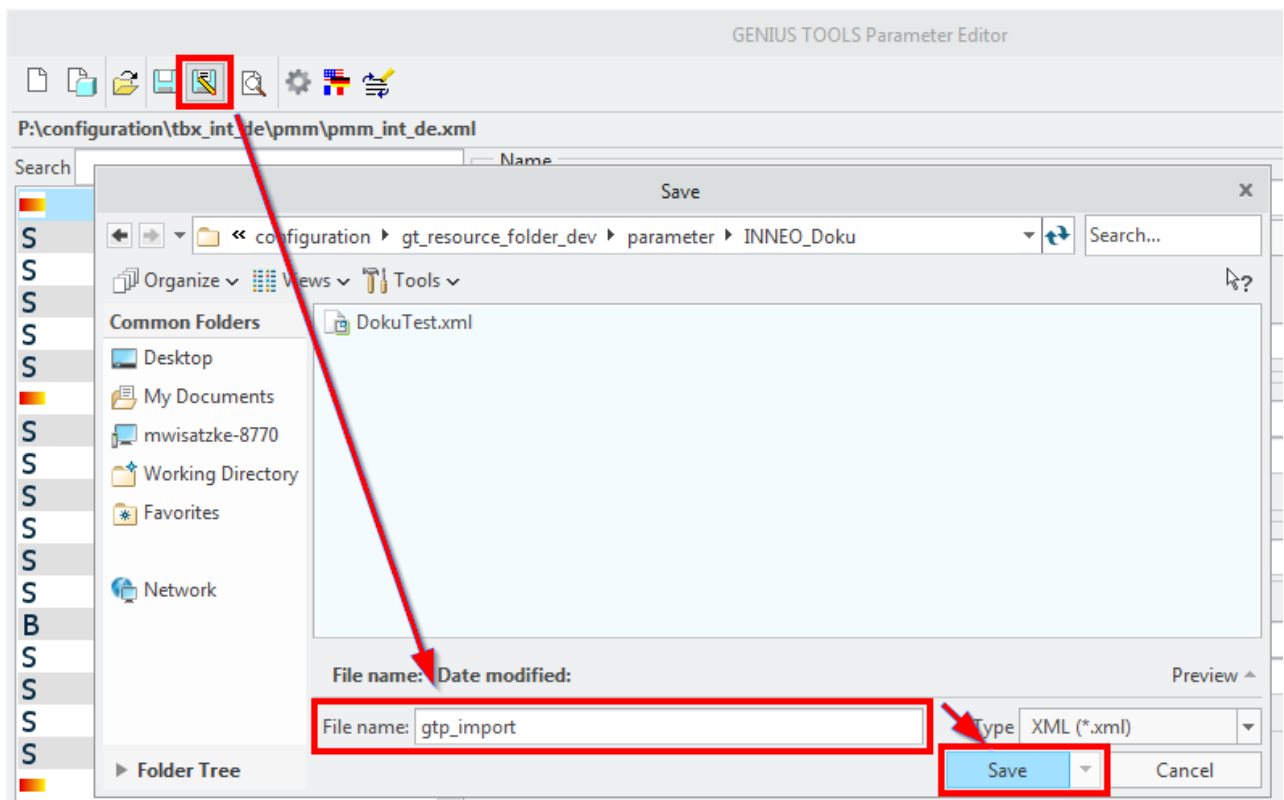
Adopting the parameters

1. In Windows Explorer, create a subfolder under
`<GTfCInstallationDirectory>\configuration\gt_resourcefolder\parameter\` following the name scheme `companyname_parameterdefinitionname`.
2. Open the old parameter definition in GENIUS TOOLS Parameter Editor.



Open the old Parameter definition in the editor

3. Click on **Save as** and save the parameter definition in the newly created folder. The old parameter definition is adopted.

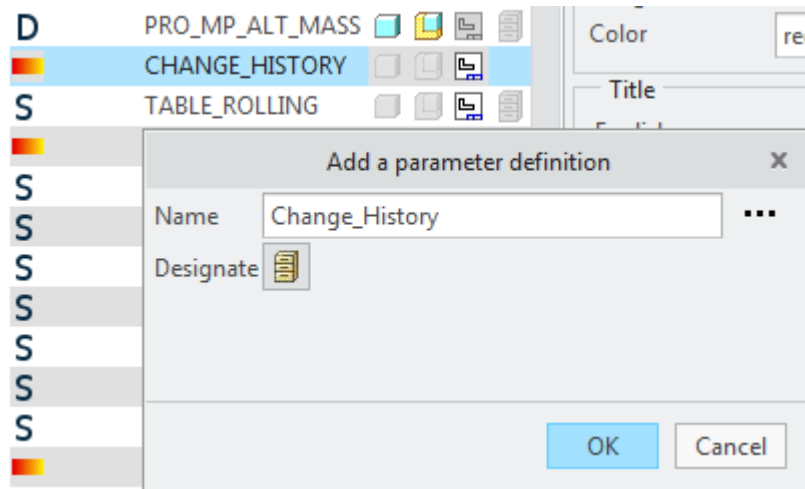


Save the parameter definition with a new name

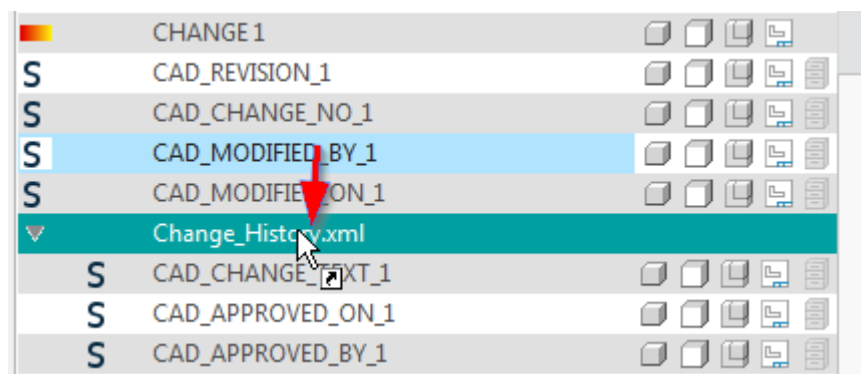
Splitting into sub-definitions

If several tabs should be displayed in the parameter form rather than a long list of parameters, the parameter definition has to be split up into sub-definitions. If separators were used in the TOOLBOX Parameter manager to logically separate different parameters, they can now be grouped into topics via sub-definitions.

Open the context menu in the element list and add the required sub-definitions.



Select parameters or separators that should be moved and drag-and-drop them into a parameter definition.



When the creation of the sub-definitions and moving of the parameters and separators are done, the splitting process is complete.

Customizing the file and database connections

Verify whether the file and database connections in the parameters are up to date and functioning.

Open each parameter by clicking it in the element list and check the settings in the details view. Then, save the parameter definition.

Your new parameter definition for GENIUS TOOLS Parameter is now ready for use.

13.3.3.4 Auto-select parameter definitions

Which parameter definition is applied to a model is specified via the list of parameter definitions (lst file) and a parameter in the model. The list of parameter definitions is specified via the `gtp_lst` configuration option and the selection parameter via the `gtp_file_param` configuration option.

Default for the list of parameter definitions is the `gtp.lst` file; the selection parameter to determine the parameter definition is `MC_CHECKTYPE`.

Additional information can be found in the [Configuration](#)⁵⁰⁰ section.

List of parameter definitions structure

Each line in the list must contain three values separated by a pipe (|) character:

- File path of the parameter definition
- Description of the parameter definition
- Parameter value of the selection parameter in models

Example

```
Parameter definition|Description|Parameter value
%GT_RESOURCE_FOLDER%\parameter\gtp_sut_int_de\pmm_int_de_fe.xml|manufacturing|MANUFACTURING
%GT_RESOURCE_FOLDER%\parameter\gtp_sut_int_de\pmm_int_de_be.xml|provision|PROVISION
%GT_RESOURCE_FOLDER%\parameter\gtp_sut_int_de\pmm_int_de_so.xml|other|OTHER
```

When a model is opened with GENIUS TOOLS Parameter and the parameter `MC_CHECKTYPE` has the value `PROVISION`, the second parameter definition for the model will be used automatically.

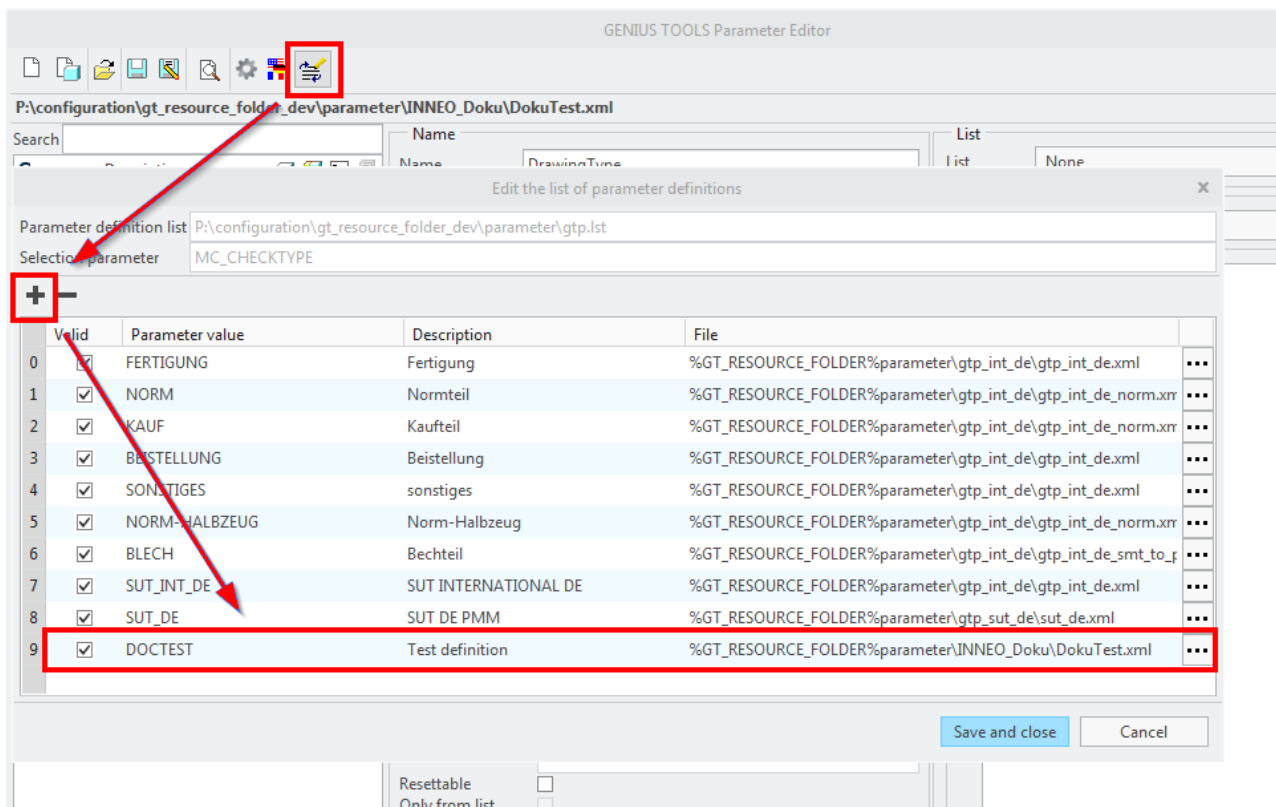
Please note: If the list contains duplicate parameter values, always the last entry will be used.

Expanding the list of parameter definitions and start models

The list of parameter definitions can be edited manually as well as via the Parameter Editor user interface.

Proceed as follows to expand the parameter list in the editor with an additional entry:

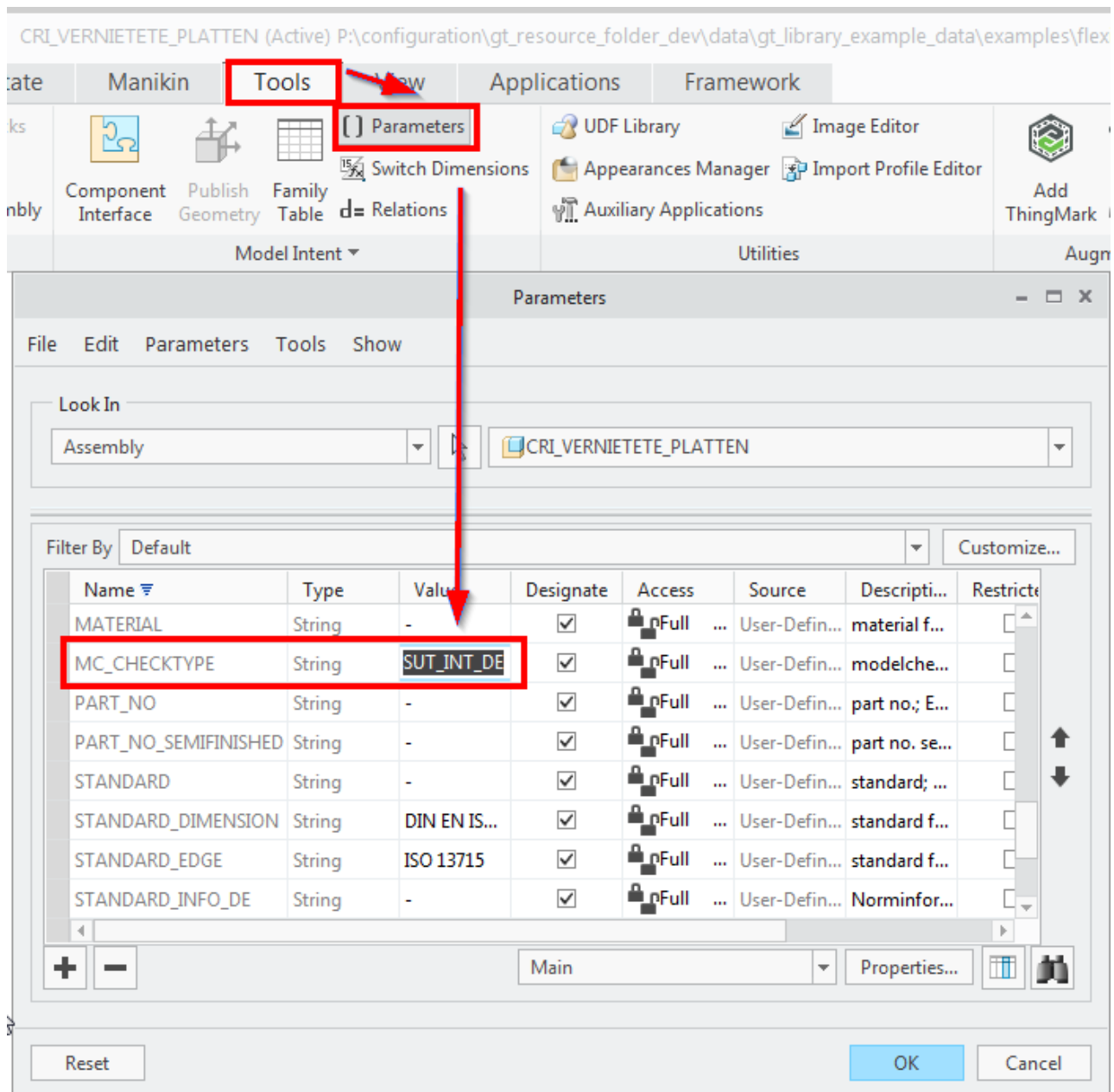
1. Open the Parameter Editor.
2. Click on the *Edit list of parameter definitions* button.
3. Add a new entry to the list.
4. Enter the value for the selection parameter and a short description of the parameter definition.



Open the parameter definition list and add a new entry

Please note: If you have modified the `gtp_file_param` configuration setting, the parameter value must be deposited in the parameter specified there.

5. Navigate to your start models folder in Creo Parametric. You will find the start models (when using Startup TOOLS) at `Library/start_model_dir` (`<ProjectName>\library_dir\start_model_dir`).
6. Open a start model. In the *Tools* ribbon click *Parameter* and search for the `MC_CHECKTYPE` parameter. If you are using an individual selection parameter, create it in the start model.
7. As the parameter value enter the value you have specified in the list file.



Switch to the "Tools" ribbon in Creo, open the Parameter dialog and edit the selection parameter

8. Save the start model.
9. Repeat this procedure for each start model used.

Your parameter definition will now be applied to each model that has been created with the modified start models.

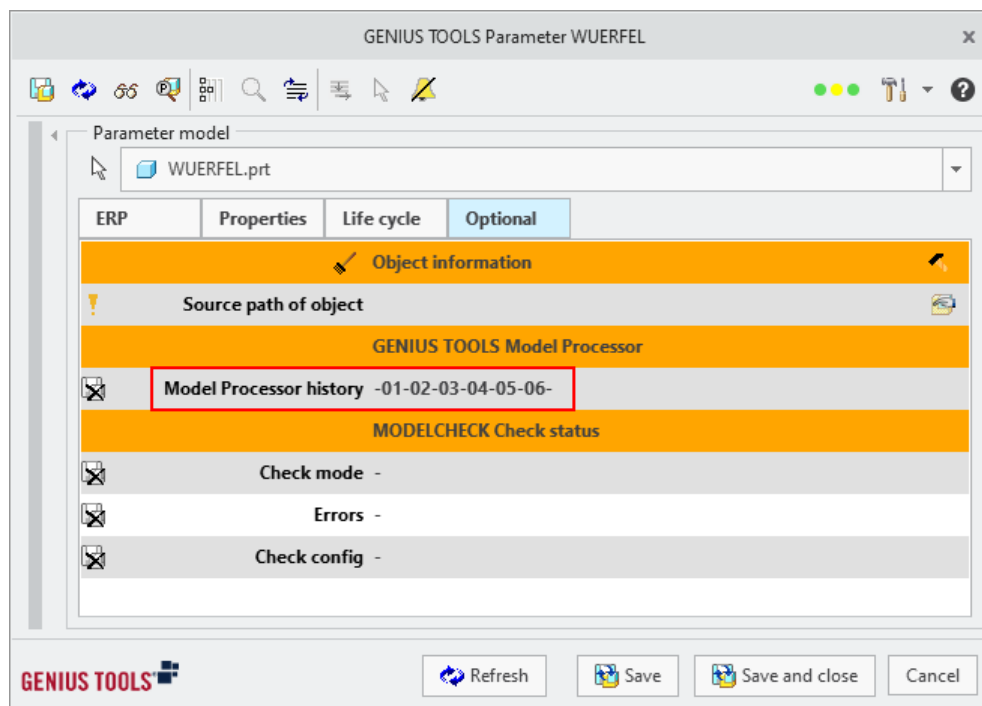
13.3.3.5 Using JavaScript with Parameter

This example demonstrates the practical use of JavaScript functions for managing parameters with the module `GENIUS TOOLS Parameter`.

Requirements

First, consider what goals are to be achieved with the JavaScript functions. To record this, the requirements are specified here. The required JavaScript functions are then derived from the requirements.

(a) When opening the module *GENIUS TOOLS Parameter*, check whether the opened model meets the current data requirements. For the data requirement to be recognized as current, a model must have reached revision level 6 through revision with the *GENIUS TOOLS Model Processor*.




(b) In addition, the unit for specifying the mass in drawings is to be checked: For mass specifications $< 1 \text{ kg}$, the mass has to be indicated in g instead.

Tip: This requirement can also be solved with relations. However, the implementation with JavaScript provides the advantage that the function is more flexible and does not require any regeneration time. A JavaScript function is executed only once, while relations are executed continuously.

Necessary JavaScript functions

First, functions are needed that define the content and second, functions that define the time of execution. In this example there are two functions that define the time and two functions that define the content. For clarity of the functions, the execution times are described here first and below the functions for the actions. Each function is separated from the next by a blank line. So the scope of a function can be seen directly.

Writing JavaScript functions

1. Open the model in which you want to apply the JavaScript code.
2. Open *GENIUS TOOLS Parameter*.
3. Open *GENIUS TOOLS Parameter Editor*.
4. Select the subparameter definition *gtp_sut_sub_optional.xml* to see the properties stored there.
5. Go into the Details view to the JavaScript section. Click the icon  to open the JavaScript Editor.

Tip: It is recommended to work with the JavaScript Editor as it provides access to predefined functions that can be used as templates. These functions help you maintain the formatting required for the JavaScript code to be executable, such as the position and number of braces.

Writing JavaScript functions: Defining points in time

6. Insert via *right click > Insert function > Function Bodies* the function `PostLoad` to define the execution time.
7. According to the defined requirements, two checks should take place ((a) and (b)) when opening *GENIUS TOOLS Parameter*. Therefore, the `PostLoad` function is needed to execute the function immediately after loading when the model is opened.

```
function PostLoad() {  
}  
  
function PreSave() {  
}
```

Writing JavaScript code: Defining contents

8. For the contentual execution of the requirements, both self-written and predefined functions are necessary. This is implemented via *right click > Insert Function > Empty Function*. Insert an *Empty Function* twice. You will receive templates into which the requirements will now be inserted.

```
function PostLoad() {
}

function PreSave() {
}

function fname() {
};

function fname() {
};
```

9. Define requirement (a):

- a. Name the function. `fname` becomes `MP_HISTORY_ALERT`.
- b. Describe the information to be searched for as a variable. Base this on how the value to be checked is created in *GENIUS TOOLS Parameter*:
`var CURR_MP_STATUS="-01-02-03-04-05-06-";`

Model Processor history -01-02-03-04-05-06-

- c. If the revision level of a model corresponds to this format, there is no effect. However, in a model that has not yet reached this revision level, a warning message should be issued to indicate the need for the user to revise the model. To achieve this distinction, insert an IF loop: `if() {}`
- d. Fill this IF loop:
 - i. Insert the query for the `MP_HISTORY` parameter: *right-click > Insert Function > getInputValue > getInputValue("MP_HISTORY");*
`getInputValue` queries the valid revision level with the parameter `MP_History`.
`!=` compares the result with the actual revision state with the variable `CURR_MP_STATUS` defined above.
 - ii. To make it clear that the model needs to be revised, the corresponding field in *GENIUS TOOLS Parameter* can be colored red. Insert the color change of the field in question:
Right click > Insert Function > setBgColorValue > setBgColorValue("MP_HISTORY", 10,100,200); and set as color *red* (255,0,0).
 - iii. Add the output of a warning message: *Right click > Insert Function > Print messages > alert*
 Here the message is created bilingual: `alert('Datenqualität veraltet ... Bitte GT MPU starten! \n Data quality outdated ... Please run GT MPU!');`
 The function checks if the opened model corresponds to the current status of the revision with the *GENIUS TOOLS Model Processor*: "-01-02-03-04-05-06-". The

Model Processor status must correspond to this scheme so that it is clear that all necessary updates have passed through this model. If this is not the case, this error message will be issued.

```
function MP_HISTORY_Alert(){
  var CURR_MP_STATUS = "-01-02-03-04-05-06-";
  if (getInputValue("MP_HISTORY") != CURR_MP_STATUS) {
    setBgColorValue("MP_HISTORY", 255,0,0);
    alert('Datenqualität veraltet ... Bitte GT MPU starten! \n Data quality outdated ... Please run GT MPU!');
  }
}
```

JavaScript function for requirement (a)

10. Define requirement (b):

- a. Name the function. `fname` becomes `MassDRW`.
- b. Create a new variable: `var MASS_Creo`
Describe the information to be searched for as a variable.
- c. Insert the query for the parameter and set it to `false` to not round the value according to the rules of the Creo configuration option `PARAM_DEC_PLACES`. Since the parameter refers to the active model of a drawing, insert the function `creoDrwActiveMdlNameGet()` and enter the name of the drawing:
`creoParameterValueGet(creoDrwActiveMdlNameGet(),"PRO_MP_MASS", false);`
- d. Include an `if-else` loop so that if one rounding rule does not apply, the other is executed:


```
if(){
}
else{
}
```
- e. Specify `if`:
Instead of using the Creo configuration option's rounding rules, now define your own rule to set *kg* as the default unit.
`var MASS_round = Math.round (MASS_Creo * 100) / 100;`
- f. Specify `if`:
If the value can be rounded according to this rule, the unit is set to *kg*:
`var MASS_unit = " kg";`
- g. Specify `else`:
Define a second rule to move the decimal place backwards. The indication of *0.001 kg* becomes *1 g*.
`var MASS_round = Math.round (1000 * MASS_Creo * 100) / 100;`
- h. Specify `else`:
If the value can be rounded according to this rule, the unit is set to *g*:
`var MASS_unit = " g";`

- i. Define a parameter that controls the display of the defined rounding rules:

```
var MASS_display = MASS_round.toString()+MASS_unit;
```

- j. Set the parameter to a value with the corresponding unit:

Use *right click > Insert Function > setInputValue* to add the function

```
setInputValue("MASS", val);
```

- k. Assign the variable `MASS_display` to the function.

```
function MassDRW(){
  var MASS_Creo = creoParameterValueGet( creoDrwActiveMdlNameGet() ,"PRO_MP_MASS", false);
  if (MASS_Creo >= 1) {
    //alert(MASS_Creo);
    var MASS_round = Math.round (MASS_Creo * 100) / 100;
    var MASS_unit = " kg";
  }
  else
  {
    var MASS_round = Math.round (1000 * MASS_Creo * 100) / 100;
    var MASS_unit = " g";
  }
  var MASS_display = MASS_round.toString()+MASS_unit;
  setInputValue("MASS_DRW", MASS_display);
}
```

11. Enter the functions `MP_HISTORY_Alert` and `MassDRW` in the functions for executing the time points.

`MP_HISTORY_Alert()` ; is executed once when *GENIUS TOOLS Parameter* is opened.

`MassDRW()` ; is executed once when opening and once when saving, so that the correct mass information is loaded (PostLoad) and the correct mass information is saved if the material has been modified with *GENIUS TOOLS Parameter* (PreSave).

```
function PostLoad() {
  MassDRW();
  MP_HISTORY_Alert();
}

function PreSave() {
  MassDRW();
}
```

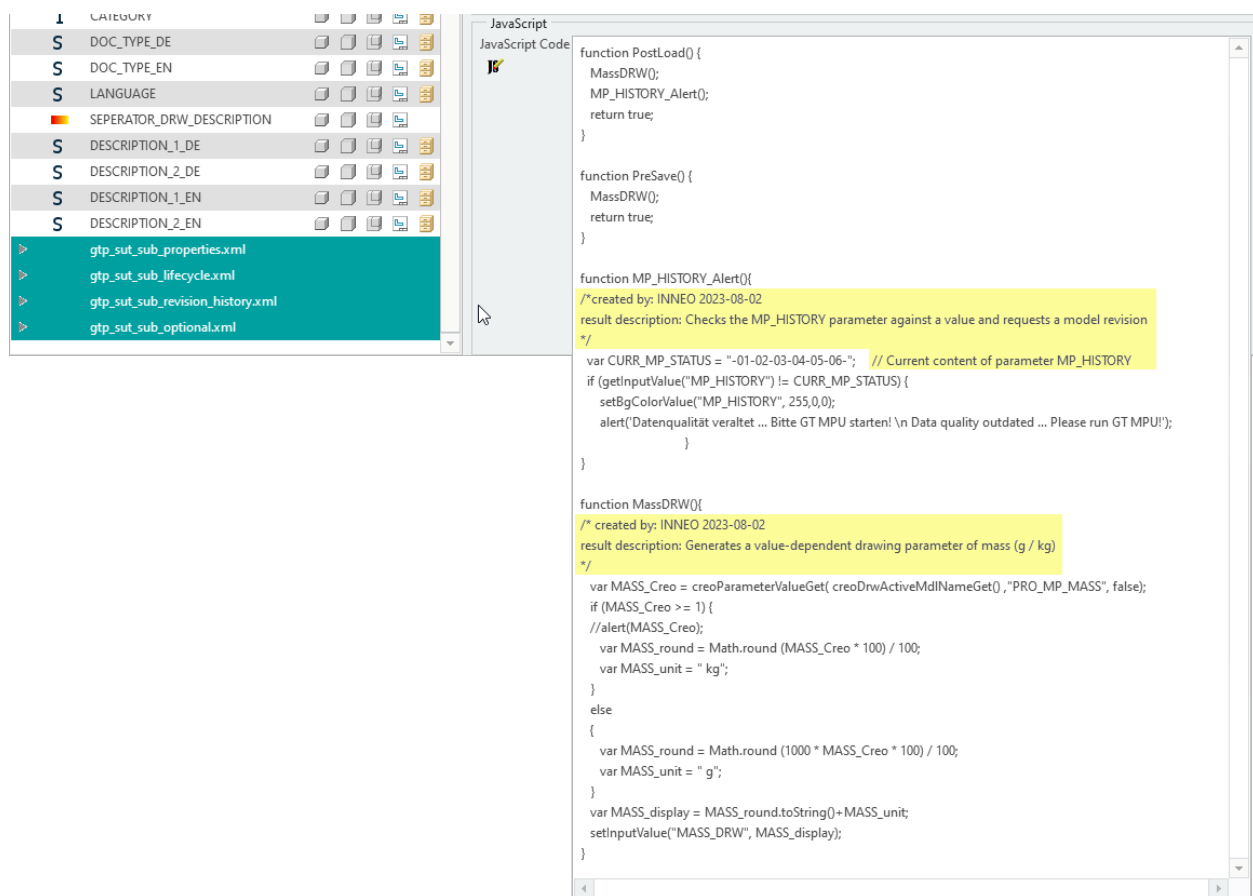
Please note: Function names are case sensitive. Make sure that the function name and the function call are written the same way.

12. At the end of these functions select via *right-click > Insert Function > return... > return true*; This function confirms that saving *GENIUS TOOLS Parameter* is allowed in any case. With `return false`; you would prevent that a model that does not correspond to the current revision level can be saved.

```
function PostLoad() {
    MassDRW();
    MP_HISTORY_Alert();
    return true;
}

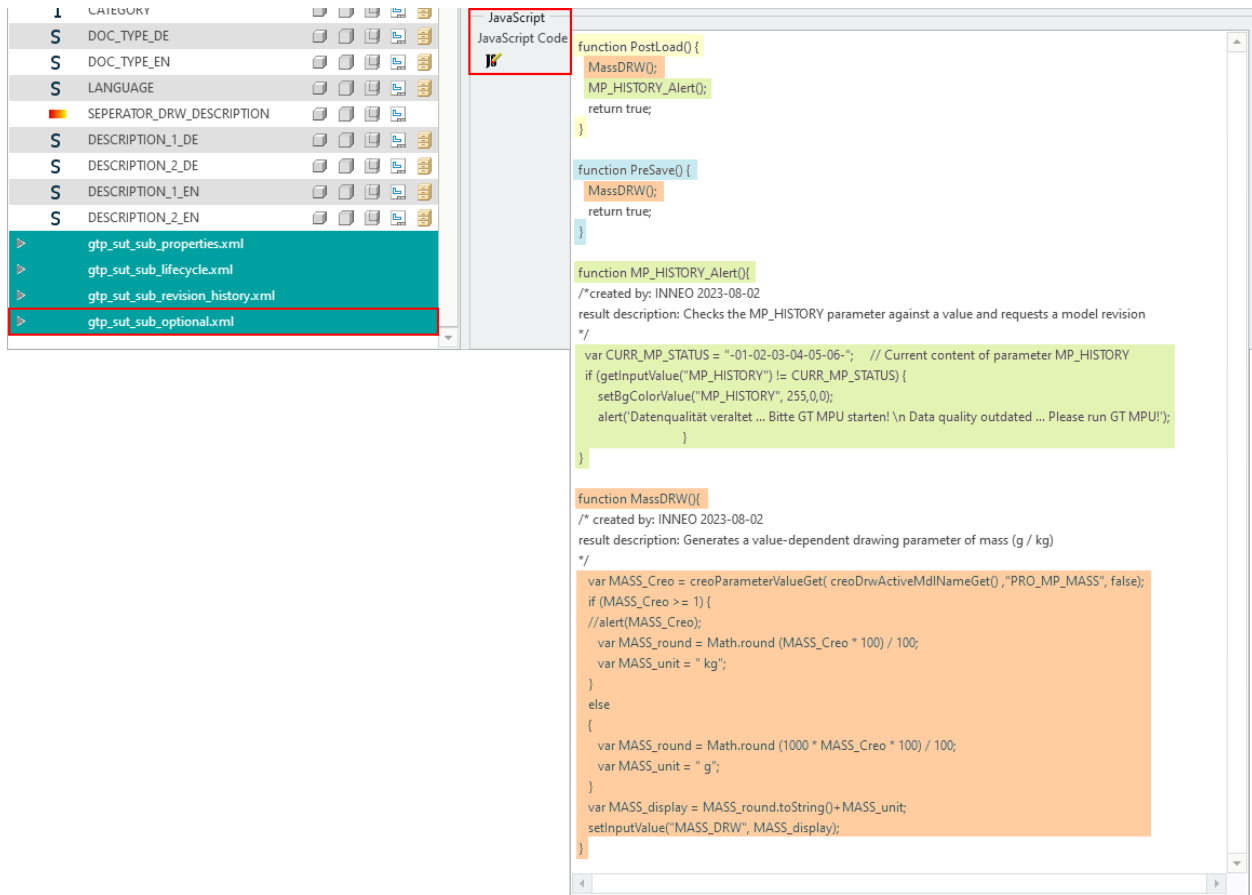
function PreSave() {
    MassDRW();
    return true;
}
```

13. Add comments to document intent, creator, date of creation, etc.



Entire JavaScript code

14. The final JavaScript code looks like this:



15. In the *JavaScript Editor*, click *Save*.

16. In the *GENIUS TOOLS Parameter Editor*, click *Save and quit*.

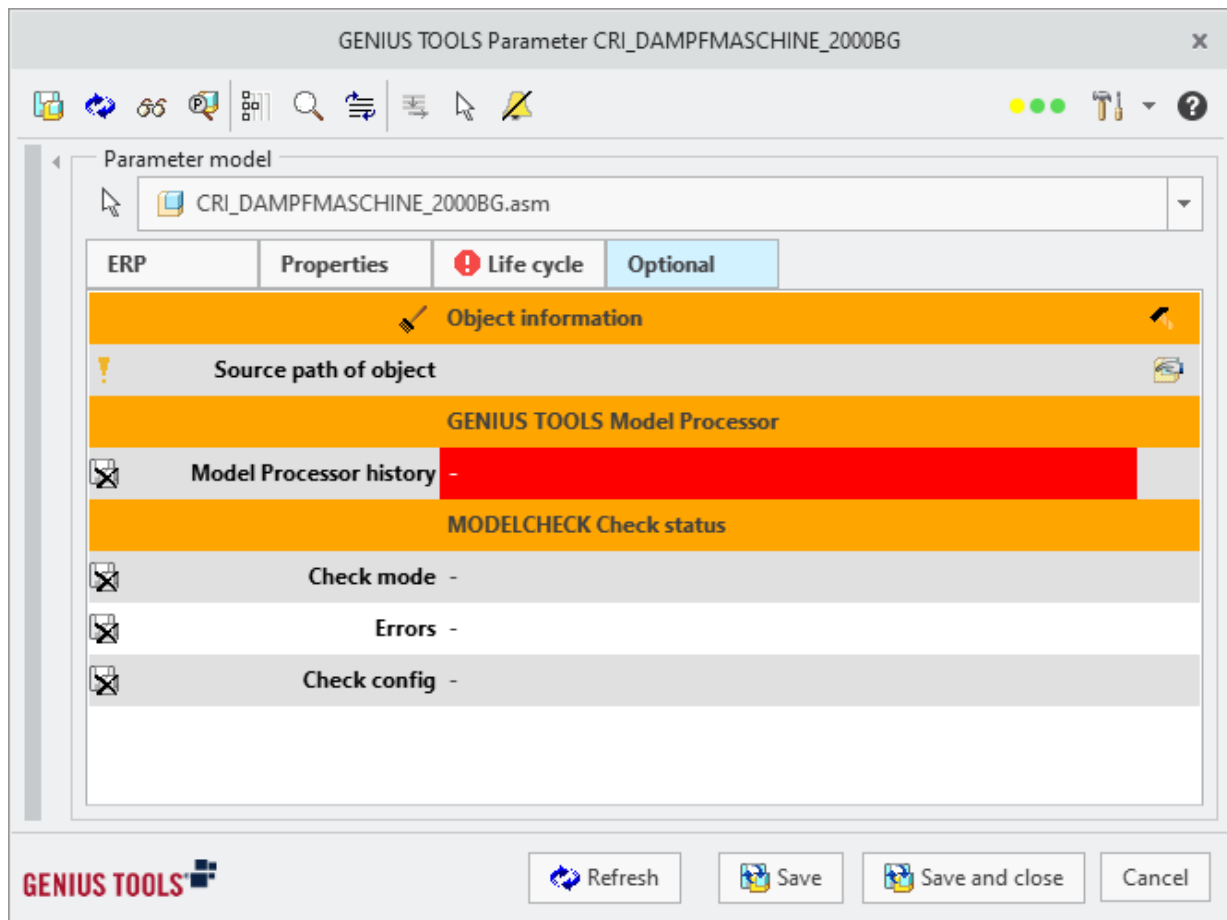
17. In the *GENIUS TOOLS Parameter*, click *Refresh* . Now the JavaScript code is applied.

Applying JavaScript functions

18. If *GENIUS TOOLS Parameter* is opened in a model and revision level 6 is not available according to requirement (a), this warning message is displayed. This message can be closed by clicking *OK*.



19. In *GENIUS TOOLS Parameter*, the line *Model Processor History* is then displayed with a red background.



20. You can fix this error by using the *GENIUS TOOLS Model Processor* to revise the model according to current company guidelines. Alternatively, you can ignore the error and save and close *GENIUS TOOLS Parameter*.

21. When you open *GENIUS TOOLS Parameter* from a drawing, the mass specification appears in the desired unit (Implementation of requirement (b)).

Mass (on drawing) 58.71 kg

Masseangabe bei ≥ 1 kg

Mass (on drawing) 62.28 g

Masseangabe bei < 1 kg

14 Quick Access

GENIUS TOOLS Quick Access is a database-dependent ring menu for Creo Parametric. It allows you to start normal and intelligent commands (mapkeys) with short mouse movements. This significantly speeds up the work with Creo Parametric.

Intelligent mapkeys have extended functionality and allow the use of variables, parameters and placeholders.

Quick Access is available in all Creo modes with the following functions:

1. using regular and intelligent mapkeys
2. launching GENIUS TOOLS for Creo and any Creo functions via short mouse movements
3. different usage scenarios⁴⁰⁸ :
 - central configuration (global)
 - user-specific configuration (local)
 - simultaneous central and user-specific configuration (mixed mode)
4. easy-to-use editor for homogeneous working environment
5. export and import of all created mapkeys with images and descriptions for easy data exchange

14.1 Usage

This section contains information on using GENIUS TOOLS Quick Access. It describes the general structure of the program.

Starting the program

Start GENIUS TOOLS Quick Access via the [<] key in a Creo session. (You may change the start key, see chapter [Changing the start key](#)⁴²³.)

Quick Access opens at the current mouse position.

Depending on the current Creo mode, different commands are displayed in the ring menu.



Place the action for opening on a button of a multi-button mouse. The Creo command is only active in the Creo environment.

Click outside the ring menu to close Quick Access without executing a command.

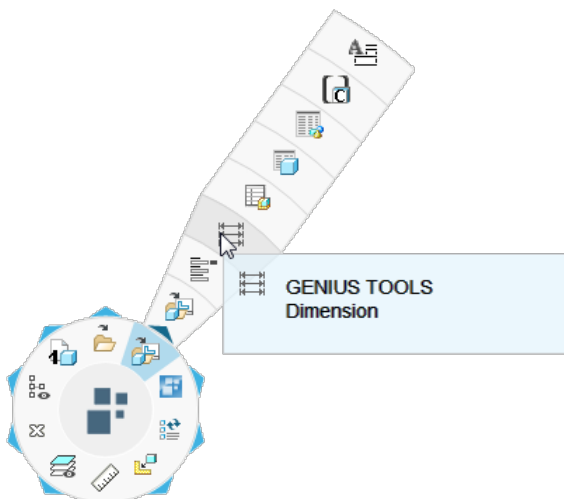
14.1.2 User interface

Quick Access displays command icons in two ways:

1. Single icons: Single icons are displayed directly in the menu ring. Clicking the icon starts the associated Mapkey.
2. Group icons: Group icons are indicated by a blue arrow. The displayed icon is the first icon of a group. Move the mouse over the blue arrow to expand a group.

Greyed-out icons cannot be used for the current selection.

If you are not sure about what an icon will execute, leave the mouse pointer over the icon. Tooltips will be displayed shortly.

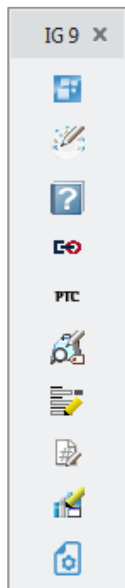


Leave the mouse pointer over an icon to display its tooltip

Undocking individual groups

Individual command groups can be undocked and are permanently visible as a toolbar. Click a group using either the middle (default) or right mouse button to undock it.

Undocked groups are not redrawn mode-sensitively. In addition, they are not greyed-out selection-sensitively.



*Permanently visible
toolbar*

14.1.3 Mode and selection dependence

Displayed command icons depend on the current Creo mode.

The following modes are available for commands in Creo Parametric and in Quick Access:

Mode (Acronym)	Description
NO MDL	No part opened in Creo Parametric
PRT	Part
SMT	Sheetmetal part
ASM	Assembly
DRW	Drawing
SEC	Sketch

Mode (Acronym)	Description
MFG	Sheetmetal assembly
DGM	Diagram
FRM	Frame
LAY	Layout

Selection-dependent mapkeys

Apart from the Creo modes, there is another dependency in Quick Access: Selection.

Each displayed icon represents two Mapkeys executing different command sequences:

1. without selection in Creo Parametric
2. with selection in Creo Parametric

This ensures executing the correct command sequences as Mapkeys can have a slightly different sequence depending on the selection in Creo. If a Mapkey is not deposited, the icon is greyed-out in this selection state.

14.1.4 Usage examples

14.2 Configuration

In this section you will find information on how to configure Quick Access with the Editor⁴¹¹ and for the display options of the Quick Access ring menu, i. e. different zoom levels⁴²⁵ and the use of custom images.

14.2.1 Different types of use

GENIUS TOOLS Quick Access can be customized for use within companies in different ways. Depending on the preferred way of working, it is possible to work with a global, a local or a mixture of both databases. Thus, there are three different ways to work with Quick Access:

1. Global configuration (as-delivered condition) 409

In a global database, all users access the same database. If one user makes changes to the database, these changes are also valid for all other users.


2. Local configuration 410

A local database is a copy of the global database. Each user gets his own local database, which he can freely modify and extend. Each user can view and edit only his own database.

3. Global and local configuration: Mixed mode (recommended mode of operation) 410

Working with global and local databases can be combined, e. g. by predefining one part of the ring menu globally and releasing the other part to the user for individual configuration.

Warning: GENIUS TOOLS Quick Access uses a different database for each Creo version. The databases are addressed via the variable `%GT_CREO_VERSION%`. If you use multiple Creo versions in your company, all databases must be adapted for the corresponding versions. Alternatively, the backward compatibility of a database must be tested by an administrator.

The following sections describe the most important configuration options for these types of use. In addition, further configuration options can be set. Additional information on the configuration options can be found in the section [Configuration](#)  593.

The ring menu groups 8 and 9 are predefined by the default settings of GENIUS TOOLS and can only be adjusted with a change in the configuration option `gtqa_change_closed_groups` afterwards.

Global configuration (as-delivered condition)

Global configuration means that all employees in the company work with a Quick Access configuration that is the same throughout the company. To allow all employees in the company to work with the global database managed by the administrator, set the following configuration options. No user other than the administrator is authorized to make changes.

– `gt_start_quick_access_editor`

Must be activated for administrator. Value: 1. For all other users, this configuration option must be deactivated. Value: 0.

Please note: If you work with the GENIUS TOOLS Startup TOOLS, `%GTFC_ADMIN%` is preset.

– `gtqa_command_file`

Refers to the file of the global Quick Access database.

`(%GT_RESOURCE_FOLDER%\quick_access\quick_access_%GT_CREO_VERSION%.db`

- **gtqa_command_icon_folder**

Refers to the global icon resource directory of Quick Access.

(%GT_RESOURCE_FOLDER%\quick_access\img_w20)

Local configuration

Local configuration means that each user works with his own local database (= copy of the global database). The administrator does not make any specifications in the ring menu and therefore there are no company-wide specifications for the use of Quick Access. Each user can and must freely configure his own Quick Access ring menu and customize it himself via the editor.

- **gt_start_quick_access_editor**

The Quick Access Editor must be executable. Value: 1

- **gtqa_command_file**

Refers to the user-specific file of the Quick Access database:

(%appdata%\INNEO\GENIUS_TOOLS\for_Creo\quick_access\quick_access_%GT_CREO_VERSION%.db)

- **gtqa_command_icon_folder**

Refers to the global icon resource directory of Quick Access.

(%appdata%\INNEO\GENIUS_TOOLS\for_Creo\quick_access\img_w20)

- **gtqa_local_command_group_split**

Must use the value "all" (default).

Global and local configuration: Mixed mode (recommended mode of operation)

We recommend using a combination of global and local configuration. You can make part of the commands available to all users via the global database and at the same time prevent users from making unintended changes to this database. Users who want to configure further commands via Quick Access in addition to the predefined commands can do so via their own local database.

- **gtqa_command_file**

Refers to the file of the global Quick Access database.

(%GT_RESOURCE_FOLDER%\quick_access\quick_access_%GT_CREO_VERSION%.db)

- **gtqa_command_icon_folder**

Refers to the global icon resource directory of Quick Access.

(%GT_RESOURCE_FOLDER%\quick_access\img_w20)

- **gtqa_local_command_file**

Refers to the user-specific file of the Quick Access database:

```
(%appdata%\INNEO\GENIUS_TOOLS\for_Creo\quick_access\quick_access_%
GT_CREO_VERSION%.db)
```

– **gtqa_local_command_group_split**

Defines how many Quick Access command groups may be customized by end users (0: only the first group can be edited to 7: all groups can be customized).

– **gtqa_local_icon_folder**

Refers to the user-specific icon resource directory of Quick Access.

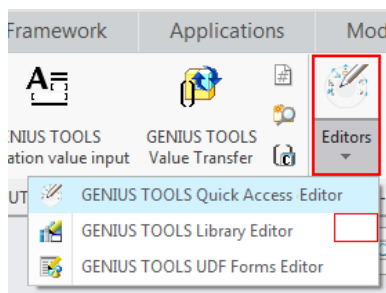
```
(%appdata%\INNEO\GENIUS_TOOLS\for_Creo\quick_access\img_w20)
```

14.2.2 Quick Access Editor

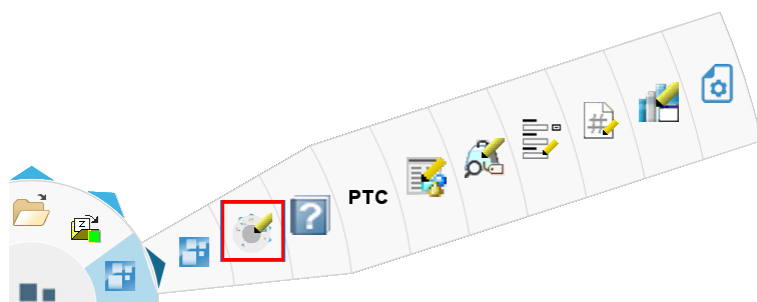
The editor is used to manage the commands and functions in GENIUS TOOLS Quick Access.

Starting the program

Start Quick Access Editor from the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).



Starting via the ribbon menu



Starting via Quick Access

Save database

Modifications made with Quick Access Editor are immediately saved to the Quick Access database.

A backup copy of the database is created once a day by default. The first change starts this process. You can disable the creation of the backup copy with the configuration option `gtqa_editor_create_db_security_copy_once_a_day`.

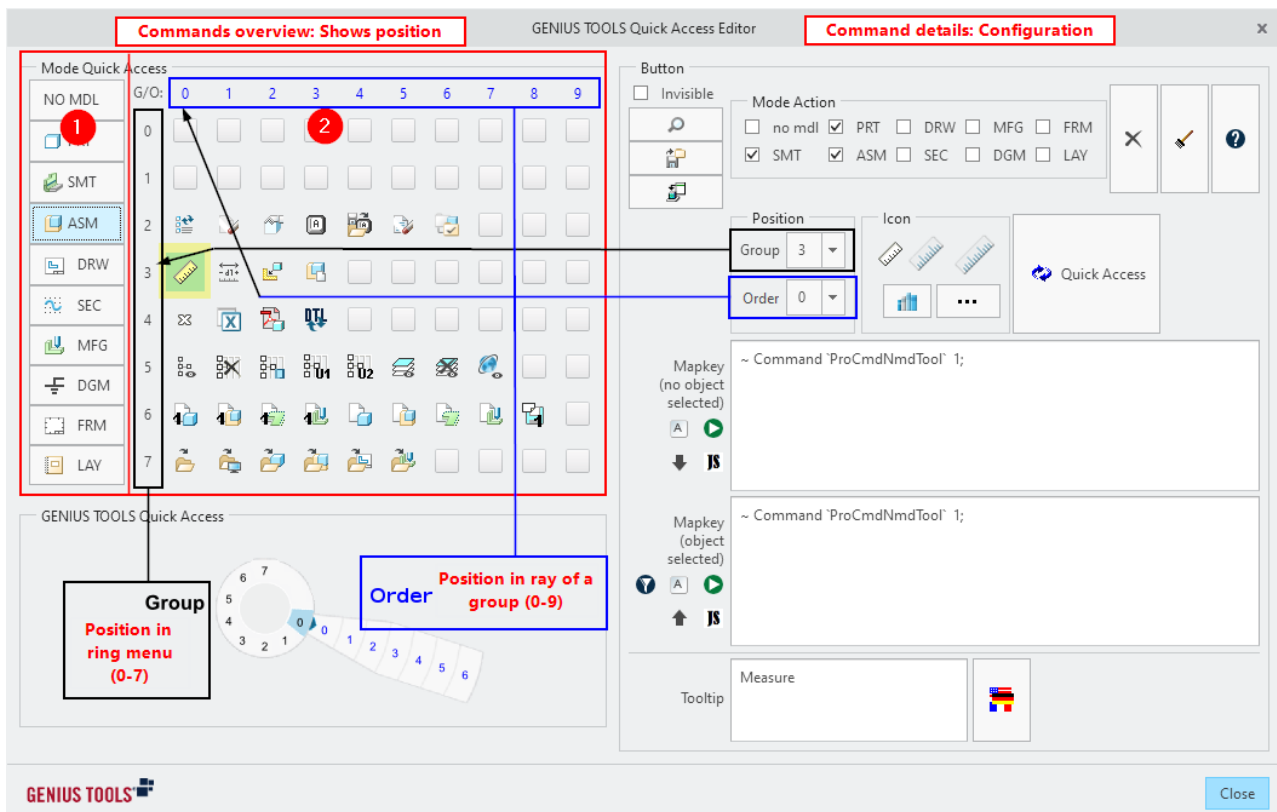
14.2.2.1 User interface

GENIUS TOOLS Quick Access Editor is divided into two areas:

1. **Command overview**⁴¹² (left): Displays commands of all groups in a specific mode.
2. **Command details**⁴¹³ (right): Configuration of individual commands.

14.2.2.2 Commands overview

The commands overview displays commands already created in the available Creo modes. Commands displayed in GENIUS TOOLS Quick Access always depend on the Creo mode. The commands overview shows all created commands in the available Creo modes. On the left side, the different Creo modes (1) are displayed. Click on a mode to see the command grid (2) for that mode.



Creo mode (1) and command grid (2) in the commands overview

Click on a command in the command grid (2) to display the configurable properties of the command in the Command details area on the right. Control the positioning of the commands there with *Group* and *Order*.

The different groups show the position of the command in the ring menu (0-7, vertical display). The position within a group is the order which is displayed horizontally in the command grid (0-9).

You can store 80 commands (8 groups x 10) with individual icons per mode. To increase the number of customized commands you can [adjust the number of groups](#)⁴⁰⁸ with the configuration option `gtqa_change_closed_groups`.

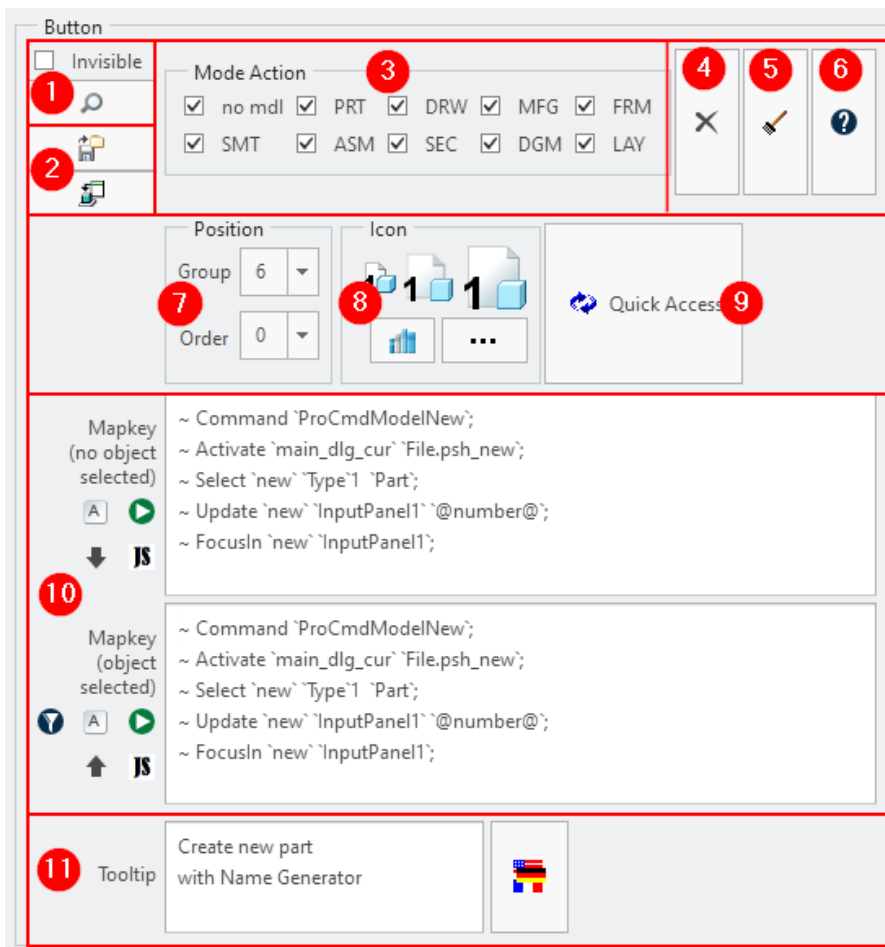
Warning: Pay attention to sufficient space for displaying the commands! Additional commands are displayed in all other modes via the ALWAYS mode. This may cause that not all modes can be displayed in the ring menu. A warning message will be displayed when saving overloaded groups.

The following Creo modes (1) are available:



Acronym	Description
ALWAYS	Always (This mode must be released via a config option <code>gtqa_always_option</code>)
NO MDL	No part opened in Creo Parametric
PRT	Part
SMT	Blech
ASM	Assembly
DRW	Drawing
SEC	Sketch
MFG	Sheetmetal assembly
DGM	Diagram
FRM	Frame
LAY	Layout

14.2.2.3 Command details

In the command details area you can configure a command.



1. **Visibility settings:** Deactivates the command without deleting it. The position is saved. To retrieve a command, click the magnifying glass icon. After a refresh of Quick Access, the command is available again.
2. **Import / export** ⁴²²
3. **Mode action:** Select the Creo modes in which a command is available in the Quick Access ring menu.
4. **Delete command**
5. **Clean up database:** Deletes all images not in use in the Quick Access database.
6. **Open help**
7. **Position:** Determines the position of the command in the Quick Access ring menu. Specify a group for each command as well as the order in the ring menu beam for this group. Commands with the same group and order are displayed one after the other.
8. **Image:** Select an icon for the display. You can select different sizes to support zoom levels for different monitor sizes and resolutions, see [Adapting zoom levels and icons](#) ⁴²⁵.
9. **Refresh Quick Access:** Reloads all commands. Updated commands are available instantly.

10. **Mapkey:** Defines a command⁴¹⁵, e. g. a mapkey,⁴¹⁶ with and without having selected an object.
11. **Tooltip:** Enter copy to get a monolingual tooltip. To store a multilingual tooltip, click on the flag icon . In the dialog that opens, a tooltip can be defined for each language required. Standard texts can be selected via the  button, see also the description of the standard text selection dialog.⁴⁹⁷

14.2.2.4 Creating commands

Quick Access command are created as follow:

1. Open the GENIUS TOOLS Quick Access Editor.
2. In the **Commands overview**⁴¹² select the Creo mode in which to create a new command.
3. Select a free field in the commands overview. Define the group and sorting in the area Position (7).
4. Select the Creo modes in which to use the command at *Mode Action* (3).
5. Select an icon (8) from the library or from your own files.
6. Specify the **sequence of commands**⁴¹⁶ (10), e. g. for a mapkey, with and without selection.
7. Enter a tooltip (11).
8. Refresh GENIUS TOOLS Quick Access. (9)

Tip: Store templates in the Quick Access database by making ready configured commands invisible (1).

14.2.2.5 Defining commands


There are different ways to create a command:

- Mapkeys: see **next chapter**⁴¹⁶
- Commands with starttags: Javascript functions, keyboard entries, opening web sites

Commands with start tags

You can define commands with the following start tags:

1. Javascript function: js:

Executes a Javascript function directly. Use the **JavaScript Editor**⁵⁷⁷, which opens with the key , to configure it.

Example: `js: alert("Hello engineer!")`

2. Keyboard entry: `key_input:`

Passes a string as keyboard input. Only letters and numbers are supported.

Example: `keyinput:abc`

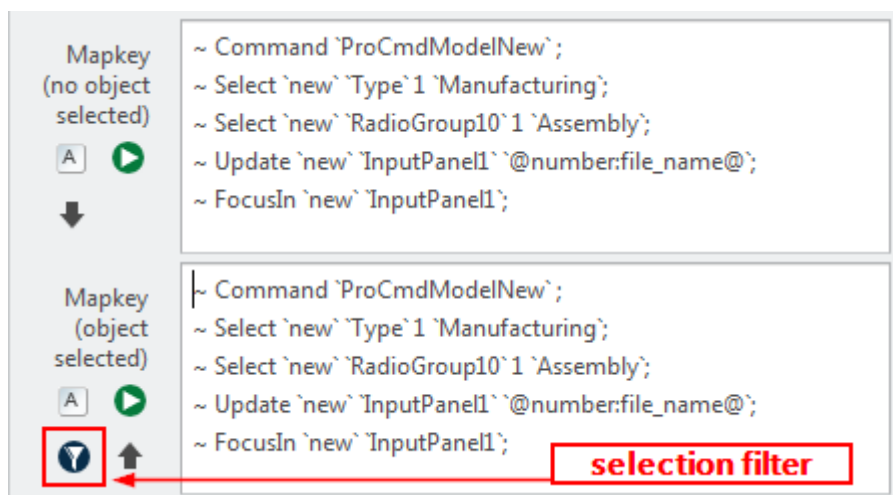
3. Open a web site: `browser:`

Passes a character string as URL to the Creo internal browser.

Example: `browser:www.inneo.de`

14.2.2.6 Defining mapkeys

For each command two mapkeys can be specified. They will be executed depending on the situation: with and without selection of an object. Icons are automatically greyed out if the corresponding situation-dependent mapkey is empty.



Mapkeys with and without selection in Creo Parametric

Enter the a mapkey into the two fields by either:




- creating a new mapkey⁴¹⁷
- copying a recorded mapkey⁴¹⁹
- inserting an existing mapkey⁴¹⁹ with %Acronym

For an example of recording a mapkey, see [Use cases](#)⁴²⁷.

We recommend entering a mapkey into each of the fields, even when they do not differ, as to have them always available.


Additional functions are available before the mapkeys:

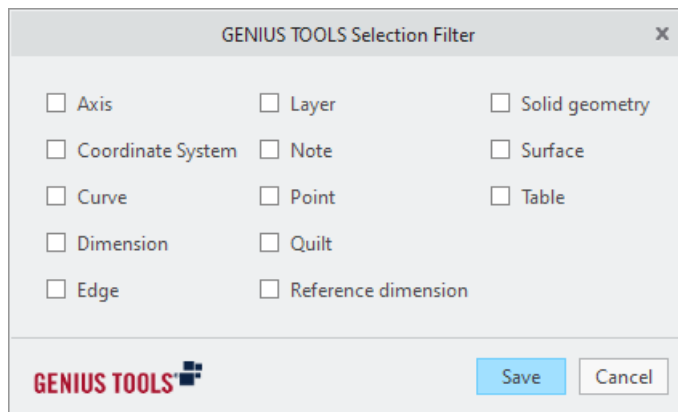
Icon	Description
	Shortens a mapkey.
	Plays the mapkey in the current Creo context.

Icon	Description
	Copy mapkey downward/upward.
	Opens JavaScript Editor ⁵⁷⁷ .
	Specifies the Creo objects for which the mapkey is executed on selection.

Please note: Starting a mapkey may close the editor.

Selection filter

The selection filter  defines the Creo objects for which the command should be applied when the object is selected.




Commands with a selection filter are only available in Creo Parametric with the proper selection (axes, layers, edges etc.). If the selection filter is empty, the mapkey is valid for any selected object.

Creating new mapkeys

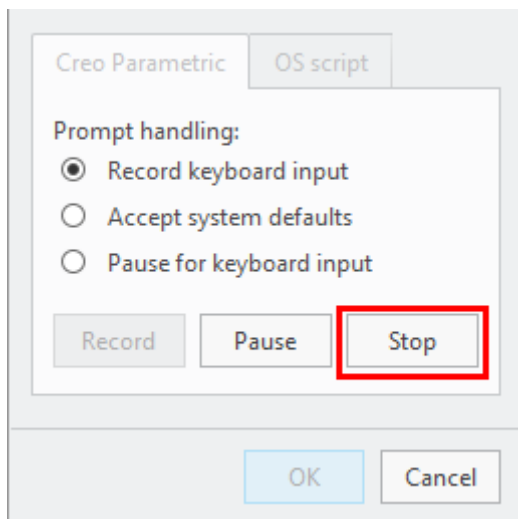
Mapkeys are created in the Creo-dialog *Record Mapkeys* and saved in a file. You can either copy the mapkey from that file into the Quick Access Editor manually or you can use the *Record Mapkey* function in Quick Access.

Creating new mapkeys with Quick Access

1. In GENIUS TOOLS Quick Access ring menu select the Record Mapkey command . This opens the Creo dialog Record Mapkeys.



2. Click *Record* in the record dialog.
3. Record the sequence of commands by clicking on all the necessary buttons.
4. Click *Stop* (Creo 3) or *Pause* (Creo 4 and later) in the record dialog, then click *OK* to end the recording.



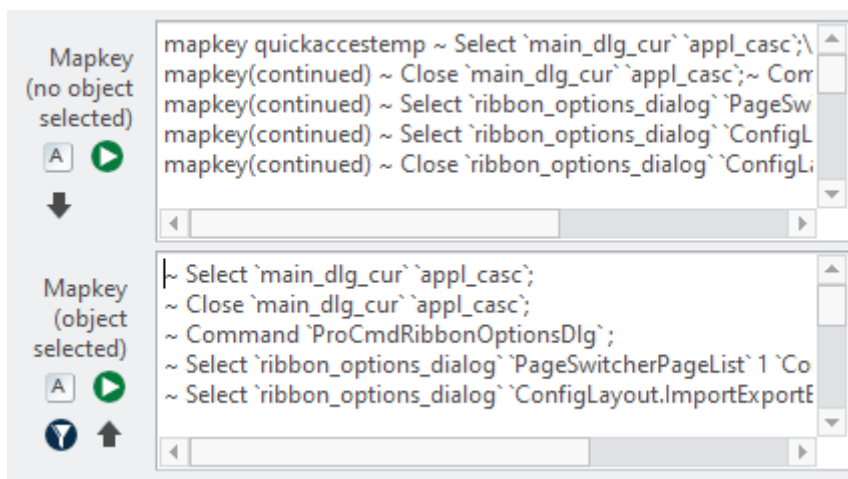
5. In GENIUS TOOLS Quick Access ring menu select *Save modified Mapkeys and open with Notepad*.





Result: The sequence of commands for this mapkey is saved in the file *config_quick_access_temp.pro* in the working directory.

Inserting recorded mapkeys from file

1. Open the file to which the mapkey has been saved by
 - clicking on the button *Save modified Mapkeys and open with Notepad* in the Quick Access ring menu or
 - by opening the file with a text editor, e.g. Notepad (Mapkeys can be created directly in Creo: *File > options > Environment > Mapkey Settings > Mapkeys > New*)
2. Copy the entire content of the file (Ctrl+C).
3. Open Quick Access Editor.
4. Insert the copied mapkey into the upper right field in the command details section (Ctrl+V).



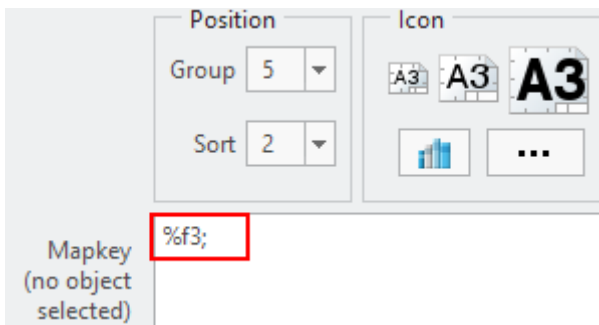
Copied content from file config_quick_access_temp.pro
(above) and shortened content (below)

5. Shorten the mapkey by clicking .
6. Test it by clicking the play symbol .
7. Copy the mapkey into the lower input field to use when an object is selected in Creo.
8. Close the editor. This refreshes Quick Access automatically.

Inserting existing mapkeys as acronym

Existing mapkeys are mapkeys that are already available in a Creo session, i. e. that are already inserted in a Creo configuration file (config.pro). Mapkey acronyms that are saved in a file in the working directory cannot be read into Quick Access.

You can insert existing mapkeys by entering the acronym for the keys as follow: %Acronym



Existing mapkey "Load A3 drawing frame" with acronym f3

You may insert several existing Mapkeys with %Acronym1; %Acronym2; one after the other. These Mapkeys are executed one after the other.

Use key_input: to emulate keyboard entries.

Example: Recording a mapkey

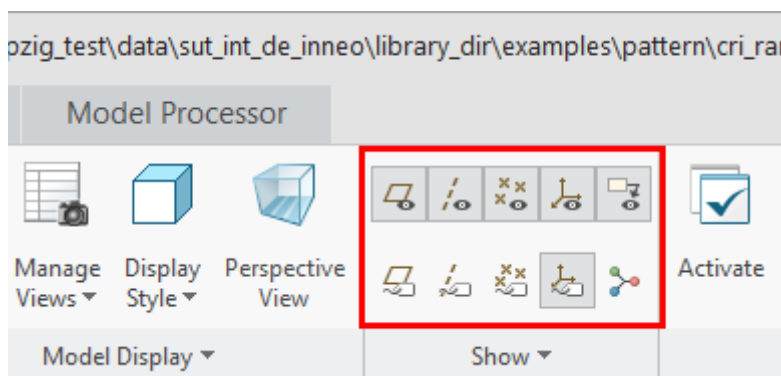
Example for extracting a Mapkey: Hide/show reference plane

Open GENIUS TOOLS Quick Access by pressing the [<] key.

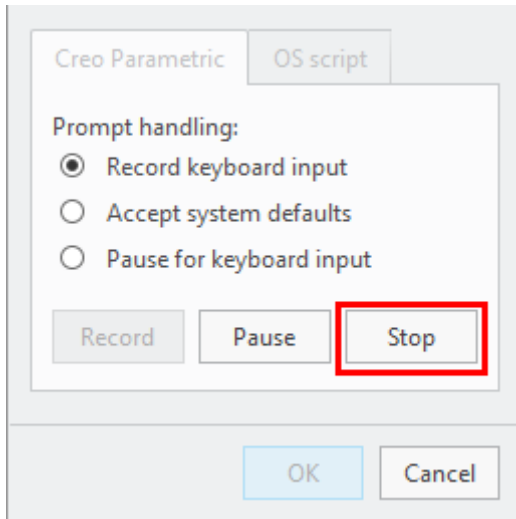
Select the *Record Mapkey* command, then click *Record* in the record dialog.



Activate or deactivate all datum display filters in the *Show* tab.

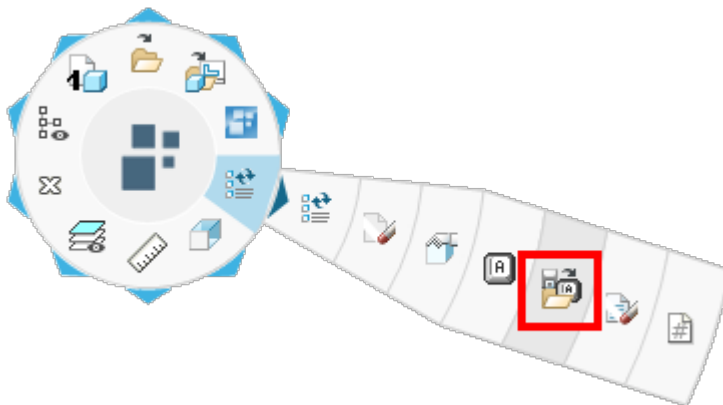


Click *Stop* (Creo 3) or *Pause* (Creo 4 and later) in the record dialog, then click *OK* to end the recording.



Open GENIUS TOOLS Quick Access again by clicking the [<] key.

Select *Save modified Mapkeys and open with Notepad*.

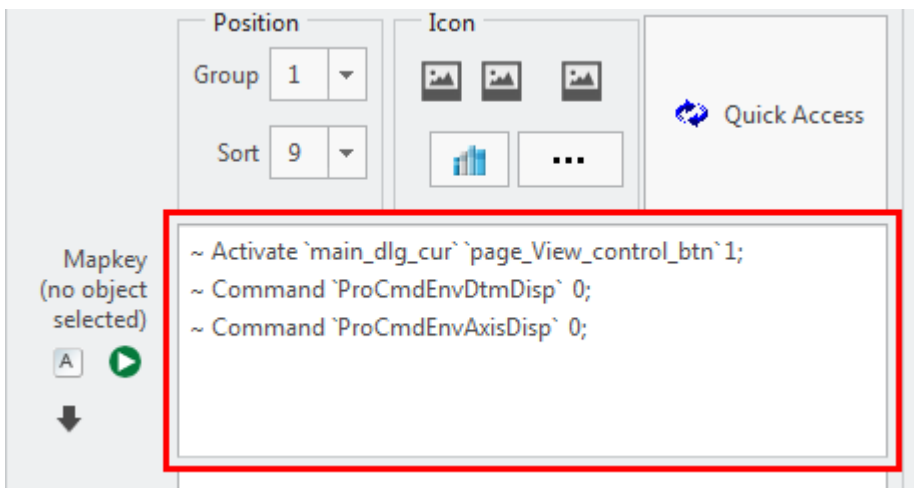


Copy the entire text from the editor (Ctrl+C).

Open GENIUS TOOLS Quick Access Editor.



Select an empty position in the grid or an existing command you wish to overwrite and insert the copied mapkey into the upper right field (Ctrl+V).

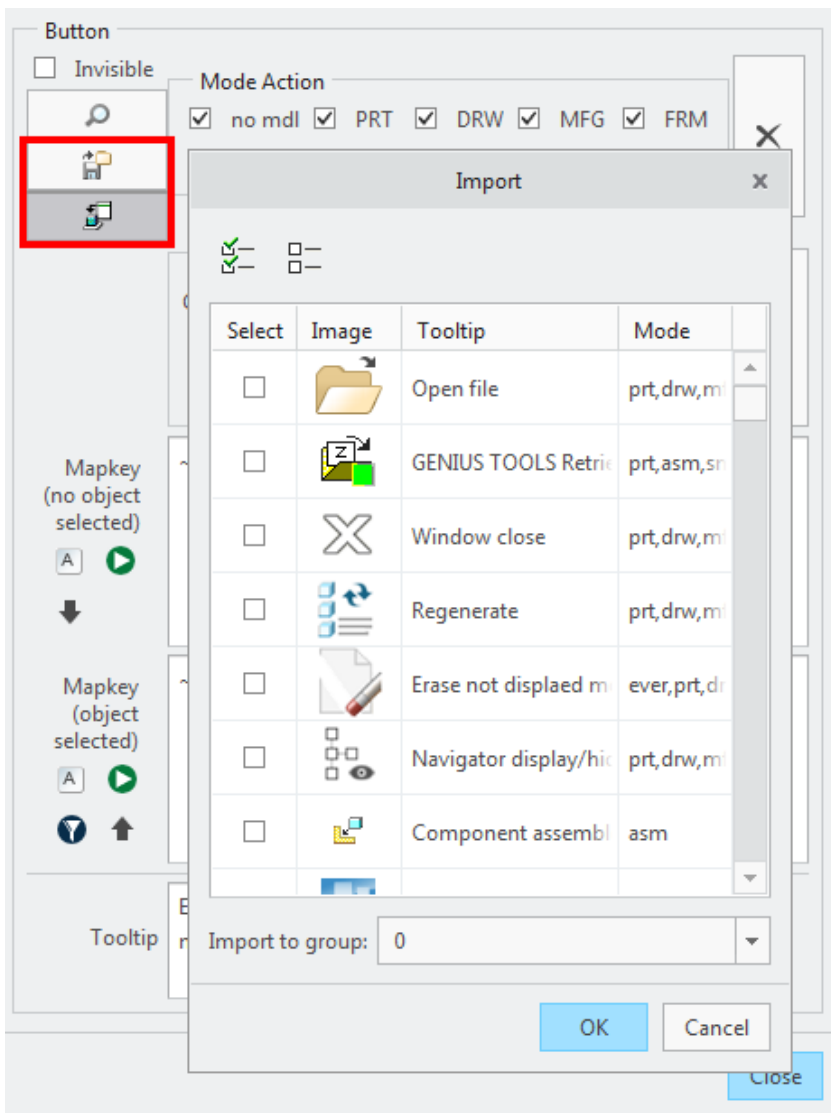


Shorten the Mapkey and test it, then copy it into the lower input field to use when an object is selected in Creo.

Close the editor. Quick Access is refreshed automatically on closing.

14.2.2.7 Export and import

Created commands can be exported into and re-imported from a QAEX file (Quick Access export file). Images, tooltips, the modes in which a command can be executed and the position are also transported. The position of the commands can be controlled with an import. Import only single commands into a group for this.



Commands can be re-imported into Quick Access

You can export and import QAEX files only to the current database. You can implement a cross-database exchange of commands via a separate folder, e. g. the Button Store folder in the GENIUS-TOOLS resource folder <GT_Resource_Folder>.

14.2.3 Changing the start key

GENIUS TOOLS Quick Access opens with the [<] key ("less-than" key) by default. This start key can be changed with the configuration option `gtqa_start_mapkey`.

Tip: We recommend placing the start key on the side of the keyboard where your hand is free, i.e. left-handed users should place the start key on keys on the right side of the keyboard, e. g. [,] or [.]

14.2.4 Inserting project images

You have the option to place user-specific images in the center of the Quick Access ring menu, e.g. for your company logo or to indicate which project a user is currently in.



Ring menu with the standard GENIUS TOOLS image



Ring menu with user-defined image

Inserting a project image

1. Create your image in PNG file format.
2. If you work with different **zoom levels**,⁴²⁵ place two additional image files with these additional abbreviations:
 - For medium-sized files: `<nameprojectimage>_30.png`
 - For large files: `<nameprojectimage>_40.png`

› parametric › configuration › gt_resource_folder › quick_access › img_background



`gtqa_quick_access_ice.png`



`gtqa_quick_access_ice_30.png`



`gtqa_quick_access_ice_40.png`

Image files in three different sizes for zoom levels

3. For users of GENIUS TOOLS Startup TOOLS: Create a directory with the name `img_background` in the GENIUS TOOLS resource folder of caddepot of the installation computer under `<nameofworking environment>\parametric\configuration\<GT_RESOURCE_FOLDER>\quick_access`.

Please note: Files outside the GENIUS TOOLS resource folder (*GT_RESOURCE_FOLDER*) are renewed at each update, e. g. files under *<operatingenvironment>\apps\gtfc\text\resource*.

4. Specify the path to the image file in the configuration option *gtqa_background_picture* (for example, by using [GENIUS TOOLS Configuration Utility](#)⁴⁵⁸).

You can use variables for the file or specify the following paths:

<GT_RESOURCE_FOLDER>\quick_access\img_background\<nameprojectpicture>
<GTfC_install>\text\resource\<nameprojectimage>.

When using these paths, the project image is searched for a size-dependent version:

- At zoom level 1.4 - 1.8, the medium sized file (*<image name>_30.png*) is used.
- For zoom level greater > 1.9, the large file (*<image name>_40.png*) is used.

14.2.5 Adapting zoom levels and icons

GENIUS TOOLS Quick Access supports multiple zoom levels to improve usability and readability on high-resolution monitors. For icons to be clearly visible you can upload images of different sizes (resolution) in the subfolders of the Quick Access resources directory.

Zoom level	Size of images	Subdirectory
1 - 1.3	20x20 Pixel	<i>img_w20</i>
1.4 - 1.8	30x30 Pixel	<i>img_w30</i>
From 1.9	40x40 Pixel	<i>img_w40</i>

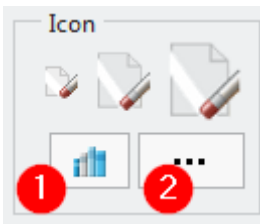
For best user experience the zoom levels 1.0, 1.6 and 2.1 are recommended.

Icons

Icons are used from the directories *img_w20*, *img_w30* and *img_w40* depending on the defined zoom level.

Assigning an icon to a command

Load icon images in the Icon area of the [command details](#)⁴¹³ using the *Load icon from Quick Access library* button (1) for images stored in the resource directory. Use the right button *Import icon from Quick Access Library* (2) to insert external icons.



Icons should at least have 20x20 pixels

External icons

Once an external source has been selected with Quick Access Editor, the image file will be automatically added to the Quick Access library.

Make sure that the size of external icons is 40x40 pixels. The other sizes are downsized for the proper dimensions and copied into the images directories (*img_w20-40*). Small icons are not `imagestyleclass="Default"` scaled up.

Creating icons

If you wish to copy icons into the images directories manually, make sure that identical icons with different resolutions always have the same name.

When creating your own icons pay attention to some specific characteristics:

- Export icons as PNG files. Check the export settings and avoid saving color values of transparent pixels.
- Take care of good recognition of the individual icons in different sizes. Modify the image composition if necessary.
- Pay attention to a strong foreground-background contrast.

Define zoom levels

Use the following configuration options.

gtqa_zoom

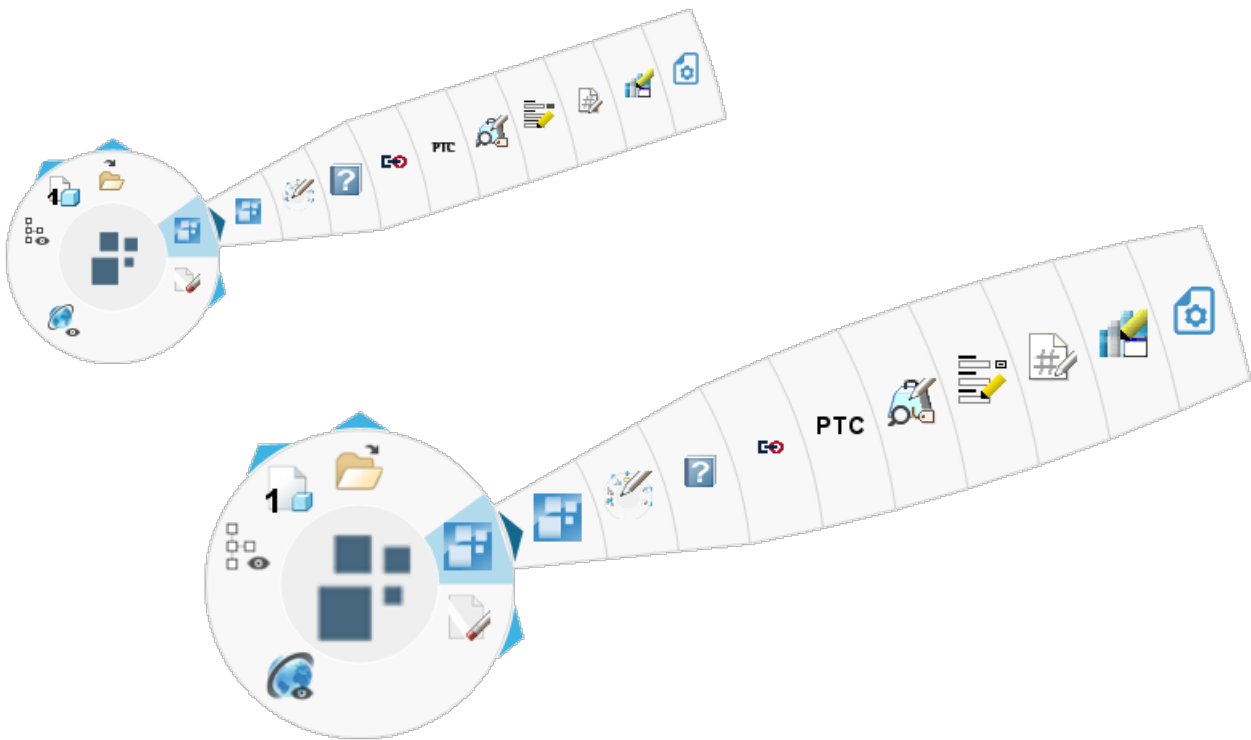
Set the zoom level in steps of 0.1.

gtqa_command_icon_folder / gtqa_local_command_icon_folder

Make sure the *img_w20* directories are specified.

gtqa_group_bow

Use this option to display an arc in groups. This enlarges the display area for icons.



Enlarged display area for icons (below)

14.2.6 Use cases

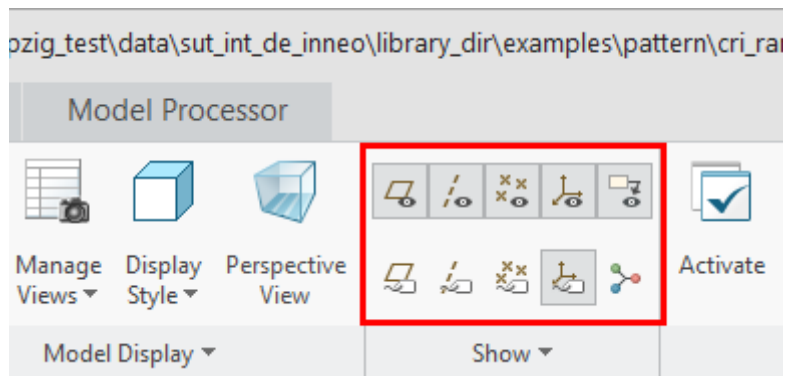
Recording a mapkey

This example shows how to create a mapkey and set it up in Quick Access. The mapkey to be created shows and hides reference planes.

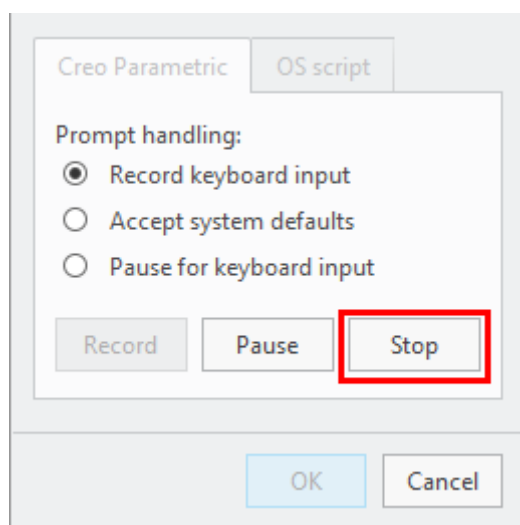
1. Open GENIUS TOOLS Quick Access by pressing the [<] key.
2. Select the *Record Mapkey* command, then click *Record* in the record dialog.



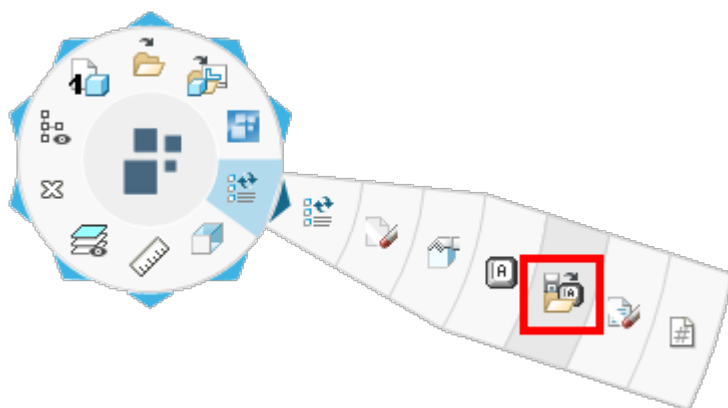
3. Activate or deactivate all datum display filters in the *Show* tab.



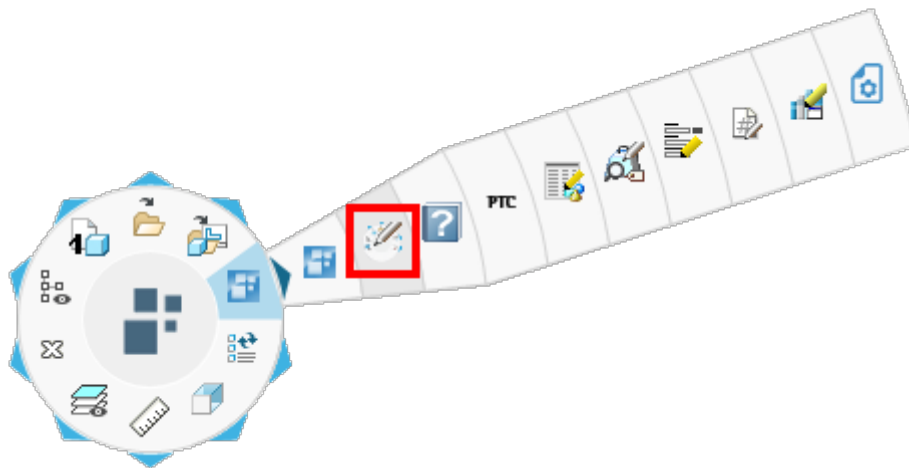
4. Click *Stop* (Creo 3) or *Pause* (Creo 4 and later) in the record dialog, then click *OK* to end the recording.



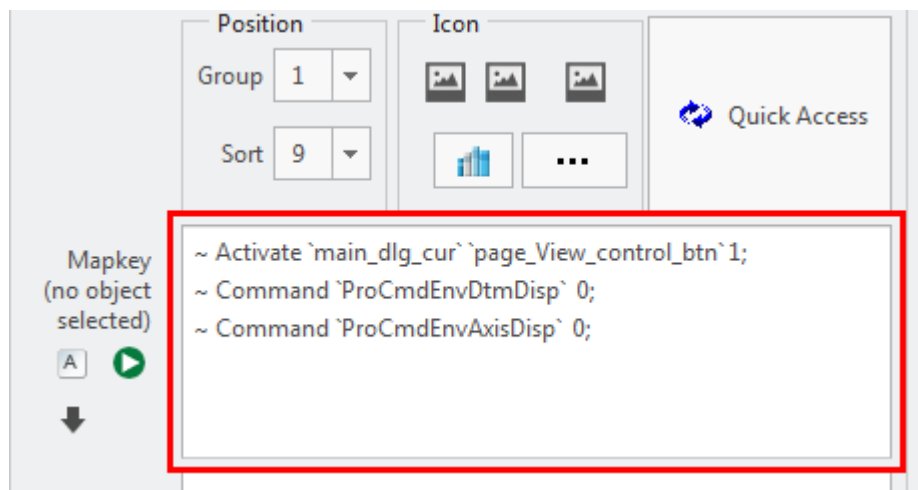
5. Open GENIUS TOOLS Quick Access again by clicking the [<] key.
6. Select the *Save changed* command to save the mapkey.
7. Open this file with an editor, e. g. Notepad.



8. Copy the entire text from the editor (Ctrl+C).
9. Open GENIUS TOOLS Quick Access Editor.



10. Select an empty position in the grid or an existing command you wish to overwrite and insert the copied mapkey into the upper right field (Ctrl+V).



11. Shorten the mapkey and test it, then copy it into the lower input field to use when an object is selected in Creo.

12. Close the editor. Quick Access is automatically updated when closed.

15 Value Transfer

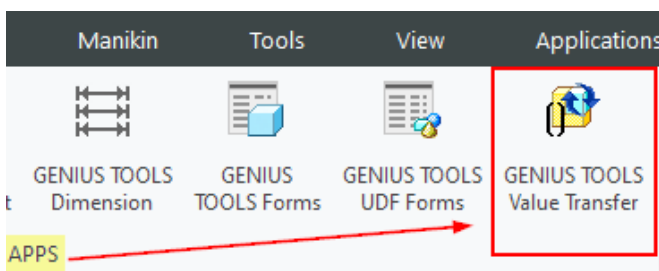
GENIUS TOOLS Value Transfer can be used in assembly mode to change multiple values in dimensions and parameters, as well as material definition files of subcomponents, in one step.

1. Search for subcomponents
 - with optional filters
 - display of search results in well-structured table format
2. Transfer parameter values
 - with control of the parameter values to be transferred by displaying the current parameter values for each component

By integrating *GENIUS TOOLS Value Transfer* into *GENIUS TOOLS Parameter*, it is possible to quickly transfer an assembly parameter (e.g. the project number) to submodels. In this case, the function for opening the Value Transfer dialog must be created for the project number parameter. The value transfer dialog will then be called from *GENIUS TOOLS Parameter* when the function of the parameter is clicked.

Starting the program: in assembly mode

Start *GENIUS TOOLS Value Transfer* from the ribbon menu in the *GENIUS TOOLS* tab or from *GENIUS TOOLS Quick Access*.



Starting via the ribbon menu



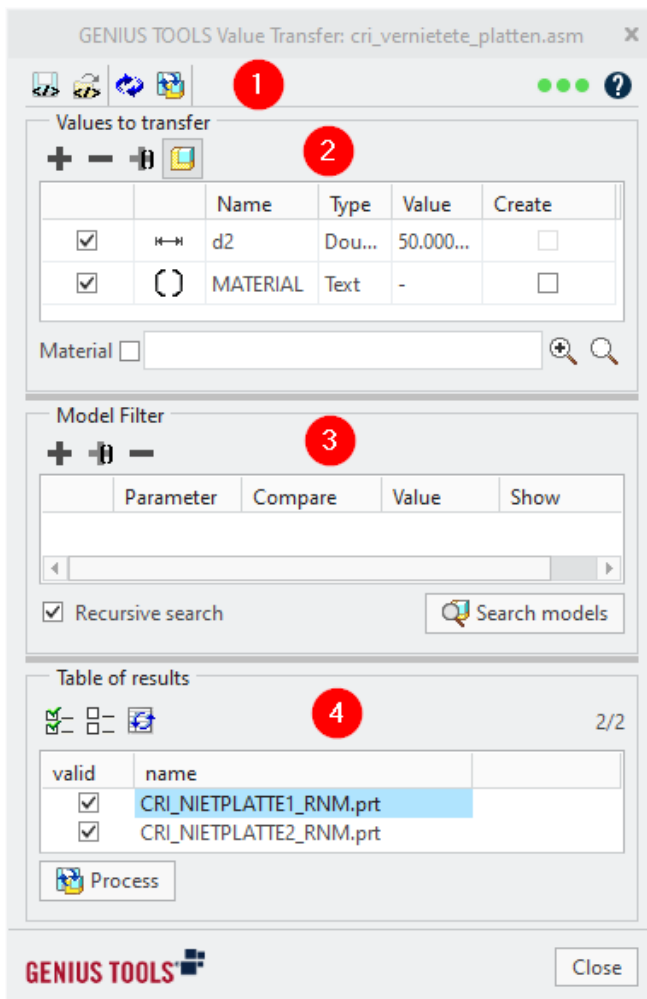
Starting via Quick Access with the key [<]

You can also start *GENIUS TOOLS Value Transfer*  in the following components:

- In GENIUS TOOLS Forms⁸⁵ from the command bar⁸⁸.
- In GENIUS TOOLS Parameter³²⁷ as a button in the input field of a parameter which has been equipped with the function Transfer parameter to assembly component³⁷⁰ without filtered values.

15.2 User interface







The user interface of GENIUS TOOLS Value Transfer consists of the following elements:



1. Command bar⁴³²
2. Transfer list⁴³²
3. Filter list⁴³²
4. Results table⁴³⁵

15.3 Command bar

The command bar displays frequently used functions. The following buttons are included:

Icon	Name	Description
	Save parameter and model search as XML	Saves all settings for parameters, dimensions and filters in an XML file.
	Load parameter and model search from XML	Loads settings for parameters, dimensions and filters from an XML file.
	Refresh	Analyzes the current assembly, evaluates parameter and dimension values and reapplies filters.
	Write parameter into model and regenerate	Transfers parameter and dimension values to sub-components and regenerates the entire assembly.
	Status indicator ³³⁸	Shows the current status for <i>loading</i> , <i>working</i> and <i>saving</i> with traffic light colors and opens the status dialog.
	Help	Opens the Help.

15.4 Transfer list

Parameters and dimensions, which should be transferred to sub-components, are specified in the transfer list. The entries in the list are displayed as column in the [results table](#) ⁴³⁵.

Creating parameters and dimensions

Add elements with the **+** button and select:

- **Parameter/Dimension (combined):** allows to enter free text. When transferring these values, the sub-components are searched for a dimension with that name. If such a dimension does not exist, an attempt is made to find a parameter with the same name. Otherwise, a parameter with that name is created and filled with the value.
- **Parameter:** allows to select an assembly parameter.
- **Dimension:** allows to select of dimensions from sub-components.

Adding an element to the transfer list

Transferring values

To transfer values to an assembly, use the assembly symbol (1). If an assembly is locked by your PDM system, the button is grayed out. Non-locked subassemblies and parts can still be processed.

Use the checkbox (2) in the first column above to specify which parameters and dimensions to transfer to subcomponents. Specify the values to be transferred in the columns *Name* (4), *Type* (5) *Value* (6).

(2)	(3)	Name (4)	Type (5)	Value (6)	Create (7)
<input checked="" type="checkbox"/>	[Symbol]	CUSTOMER	Text	ACME	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[Symbol]	PROJECT	Text	42.23	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	[Symbol]	DIAMETER	Text	40	<input type="checkbox"/>

Transfer list

Use the second column (3) to change the type (Parameter, Dimension or Combined).

Warning: Always check the type of parameters and dimensions! A wrong type may cause existing parameters or dimensions to not be overwritable.

Type of parameter	Description
Boolean parameter	<i>yes, true, 1, ja</i> (engl. yes) und <i>wahr</i> (engl. true) are interpreted as <i>correct</i> , everything else is interpreted as <i>wrong</i> .

Type of parameter	Description
Double, Integer	Allows the input of numbers.
Text	Allows the input of text.

If a parameter does not exist in a subcomponent, it can be created automatically by activating the checkbox *Create* (7) and the value will be transferred.

To delete an entry, click on the entry and then on the Minus button.

Changing materials

You can also change materials of sub-components with *GENIUS TOOLS Value Transfer*. Click on the *Material* checkbox (8) and select the required material using the magnifying glass buttons (9) with the material selection or from a model.

15.5 Filter list

Criteria for filters to search for in sub-components are specified in the filters list. When the filter criteria are met, the specific sub-component is displayed in the results list.

Select the checkbox *Recursive search* to include subassemblies. Use the checkbox *Show* to change the filtering. Then refresh the results table. Add parameters using the Plus buttons with free input or by using the Creo parameter selection dialog. Click the parameter name, then click the Minus button to remove a parameter.

Specify a comparative operator in the *Compare* column. Available are: *equal* (=), *not equal* (!=), *greater-than* (>), *less-than* (<), *contains*, *contains not* and *regular expression*.

Enter the comparative value into the last column *Value*. Confirm the filters list by clicking the *Search models* button to refresh the results table.

Warning: Always refresh the results table, as only the subcomponents displayed in it are refreshed.

Filtering with variables

Add an entry to the filters list and open the context menu for the entry. Select the desired variable, set the comparative operator and enter a comparative value.

Model Filter

+ -

	Parameter	Compare	Value	Show
<input checked="" type="checkbox"/>	CATEGORY	diverse	20	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	@mdltype@	equal	prt	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	MATERIAL	diverse	CuZn39Pb0,5	<input type="checkbox"/>

☒ Recursive search

Search models

Use variables (here: @mdltype@) to filter by specific model properties

The filters are case-sensitive. For variables, case is ignored for the *Equal* and *Not Equal* operators.

If the value of a filter parameter is empty and the comparison is set to *Equal*, all models are listed in which either this parameter does not exist or is empty. If only models with new parameter values in which the parameter exists are to be assigned, the check mark in *Create* must be removed.

Tip: Variables are always treated like text parameters.

You can find the variables available via the context menu in the [Variables overview](#) ⁶⁶⁶.

15.6 Table of results

The Table of results displays the filtered sub-components. The current values of the parameters from the parameters list are displayed in the individual columns.

Table of results

65/65

valid	name	() MATERIAL	<-> d10
<input type="checkbox"/>	CRI_ANSCHLUSSVERSCHRAUBUN...	CuZn39Pb0,5	6.000000
<input type="checkbox"/>	CRI_D433T1D3_2<CRI_DIN433_TEIL...	Stahl allg.	
<input type="checkbox"/>	CRI_D433T1D4_3<CRI_DIN433_TEIL...	Stahl allg.	

Process

Use the checkboxes in front of the components to select the subcomponents to which *GENIUS TOOLS Value Transfer* will be applied.

Parameters that are controlled by a relation or have access restrictions are marked with a yellow background. Dimensions that are controlled by a relation are also marked with a yellow background.

If a PDM system is used, the individual models can only be checked if they are unlocked. Models can be unlocked via the right mouse button menu. If several models are marked (via Shift and Control), several models can be unlocked.

By clicking the *Process* button, the parameter values specified in the parameter section are written into the sub-components.

Please note: Only active sub-models are considered when processing.

The parameter values in the results table can also be modified manually. Just click into a table cell and modify the value.

16 UDF Forms

With GENIUS TOOLS UDF Forms you can create UDF groups in models and manipulate individual properties later on.

GENIUS TOOLS UDF Forms is available in assembly, part, and drawing mode with the following features:

1. Creation of UDF groups in models
 - dependent or independent,
 - optionally with variable dimensions from lists and tables,
2. Use of UDF family tables - for form control,
3. Use of variable parameters (only feature parameters located on the first feature of the UDF group),
4. Subsequent editing of already created UDF groups,
5. Re-placing UDF groups with the same values of already created UDFs,
6. UDFs containing body references (starting with Creo 7) are supported.
7. Rules between properties can be defined with JavaScript.
8. Loading of external data into the mask is possible (e. g. EXCEL, CSV).

The definition and construction of a UDF form is done with a graphical editor (UDF Form Editor).

16.1 Fundamentals

With *GENIUS TOOLS UDF Forms* you can create UDF groups in models and manipulate individual properties later on.

Glossary

In this glossary, the terms are sorted by relevance.

UDF (generic term, Abbreviation for User-Defined Feature, built-in UDF group or saved UDF GPH file)

User-defined feature with selected single features and their associated properties for placing the UDF in a part or assembly. UDFs can be dependent or independent. Beyond Creo Parametric functionality, *GENIUS TOOLS UDF Forms* allows you to create and subsequently edit UDF groups in models.

UDF GPH File (<udfname>.gph)

Creo internal file format in which all the details of a UDF are stored on disk. *GENIUS TOOLS UDF Forms* does not allow the creation of GPH files, but the assembly, subsequent editing and replacement.

UDF-Group

A UDF inserted into a model and previously saved. It appears in the model tree of the model.

Target model

The target model is the model opened in Creo into which a UDF group is placed.

UDF Form

Graphical mask for assembly and subsequent manipulation of properties. For the creation in the target model the UDF file and the UDF Forms definition are necessary.

UDF Forms Definition

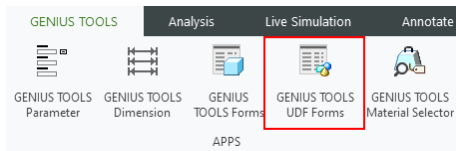
XML structure that contains the description of the mask for a UDF. This XML structure can be stored either in the file directory or in the Creo model. If a UDF is assembled multiple times, the UDF Forms definition is included once in the target model.

16.2 Usage

This section contains information on using GENIUS TOOLS UDF Forms. It describes the general structure of the program.

Starting the program: in part mode

Start GENIUS TOOLS UDF Forms from the ribbon menu in the GENIUS TOOLS tab or with GENIUS TOOLS Quick Access ([<] key).



Starting from the ribbon menu



Starting via <%GTQA%

GENIUS TOOLS UDF Forms only starts after selecting a UDF in the Creo main window or a UDF group in the model tree.

Forms and UDF Forms in the model

If a model contains a Form or UDF Form, a suitable Forms icon is displayed in the Creo Parametric main window. Click on the icon to open the form.



Form in the model

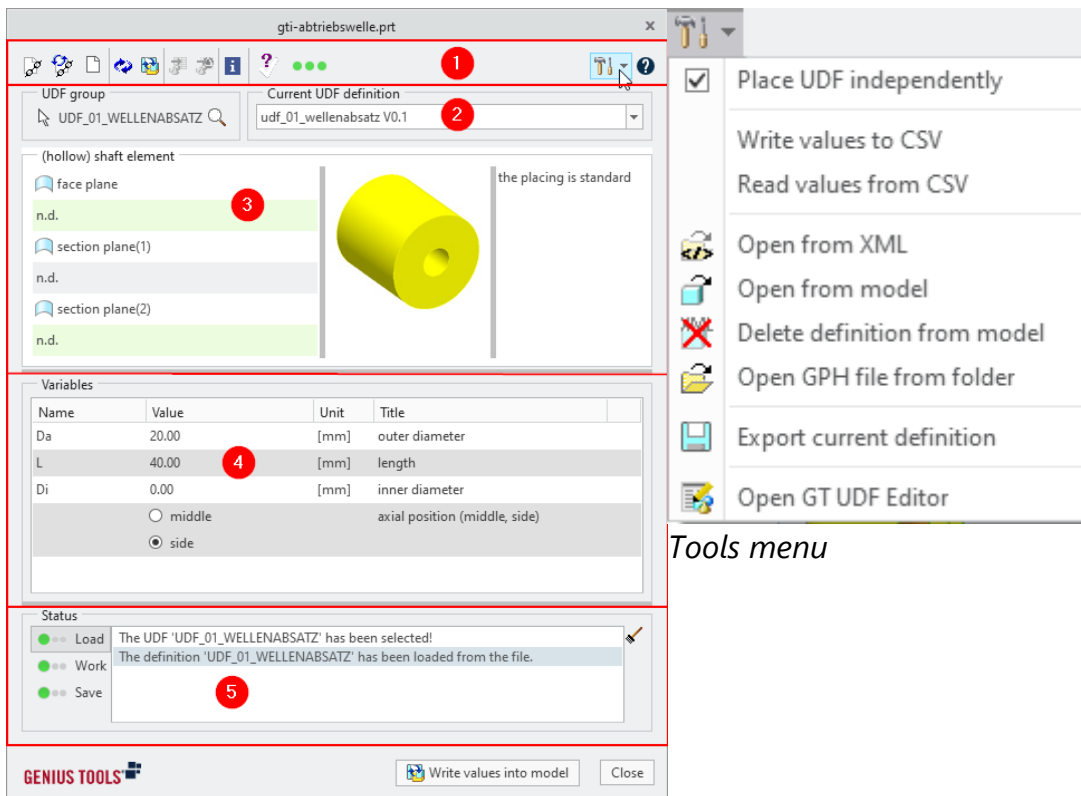


UDF Form in the model

16.2.1 User interface

The GENIUS TOOLS UDF Forms user interface consists of the following elements:




1. Command bar⁴⁴⁰ with tools menu⁴⁴²
2. UDF selection⁴⁴³
3. Placement references⁴⁴³
4. UDF Form⁴⁴⁵
5. Status indicator as short overview, a separate window for the status indicator with search function³³⁸ can be opened via the tool menu









Tools menu

16.2.2 Command bar

The following buttons are included in the command bar:

Icon	Name	Description
	Create with references	Creates a UDF in a model. The placement references are used.
	Repeated creation with references	Creates several UDFs in a row. The placement references are used.
	Create	Creates a UDF in the model. References must be selected during placement. The button can be hidden with <code>gtuf_show_create_without_references = 0.</code>

Icon	Name	Description
	Create new UDF group by Creo std. Dialog (values would not be transferred)	Integration of the Creo function for creating a new UDF in <i>GENIUS TOOLS UDF Forms</i> . You can create a new UDF in an open model and then edit it and enter values with <i>GENIUS TOOLS UDF Forms</i> without changing the user interface. This function is only available if the configuration option <code>gtuf_show_create_by_creo = 1</code> is set (Default: 0).
	Refresh	Updates the view after editing the UDF Form with the editor.
	Save values into model	Saves the entered values in the form into the current model and regenerates it.
	Apply to table	Links the active UDF to an already placed drawing table. (Only active in drawing mode)
	Apply to symbol	Fills drawing symbols with the values of the UDF. The drawing is regenerated. (Only active in drawing mode)
	Show info	Opens a stored, language-dependent help document.
	Run check function	Starts the JavaScript function CheckUI from the UDF definition to check the values entered in the form. (The CheckUI function must first be stored in the UDF object.)
	Status indicator ³³⁸	Shows the current status for <i>loading</i> , <i>working</i> and <i>saving</i> with traffic light colors and opens the status dialog.




Icon	Name	Description
	Tools	Opens the tool menu containing various supporting functions.
	Help	Opens the user manual.




Repeated creation with references

Use the function *Repeated creation with references* if you want to place a UDF several times in a model:

1. Activate the part in which you want to place UDFs.
2. Define constant references beforehand and then click on the button *Repeated creation with references*.
3. Select the missing references in the part. Repeat this step until the required UDFs have been placed.
4. Click with the middle mouse button to cancel the placing.

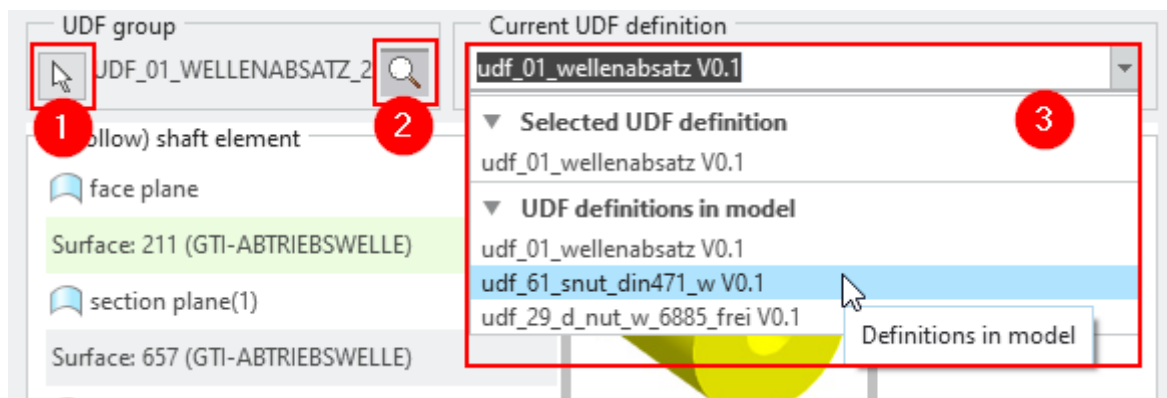
Tools menu

Icon	Name	Description
	Place UDF independently	If a UDF is placed dependently, changes to the values of the UDF form (the GPH file) will be applied. Define the default setting for the checkbox in the editor . ⁴⁵⁴
	Write values to CSV	Saves the current values in the form to a CSV file.
	Read values from CSV	Reads values from a CSV file and applies them to the current UDF form.
	Open from XML	Opens a UDF group from an XML file (UDF definition).
	Open from model	Opens a UDF definition from the current model.
	Delete definition from model	Deletes a UDF definition from the current model.

Icon	Name	Description
	Open GPH file from folder	Opens a UDF file from the file system.
	Export current definition	Saves the definition of the current UDF group as XML file.
	Open UDF Editor	Opens the GENIUS TOOLS UDF Forms Editor to edit the UDF definitions.

16.2.3 UDF selection

The UDF selection consists of three elements:



UDF selection

Select UDF groups in a part, assembly or drawing directly in Creo or in the model tree with the arrow icon (1).

Use the magnifying glass icon (2) to highlight the currently selected UDF in models.

The drop-down list (3) shows the UDF definitions in the model. Open the list and choose between the UDF definitions.

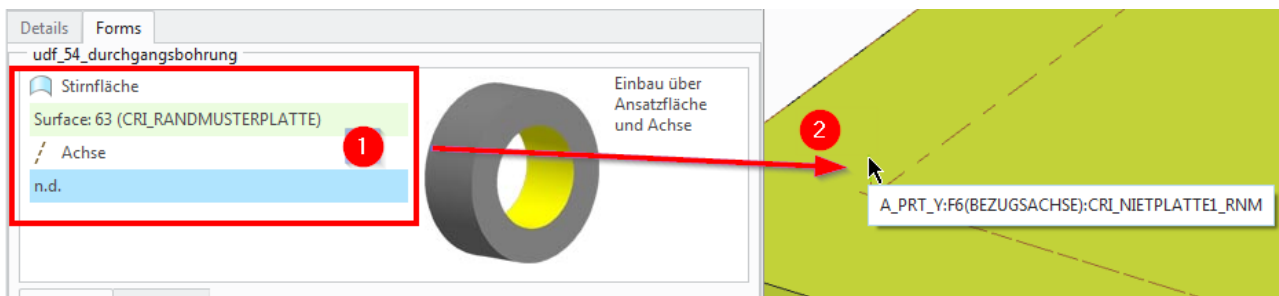
Only one UDF definition is saved in the model if a UDF group is placed several times.

16.2.4 Placement



Placement references

This section displays the placement references of the selected UDF group on the left side of the dialog. The right side contains a description and a preview image, which can be configured in the editor.

Click on a colored field (1) and select a placement reference in the model (2).

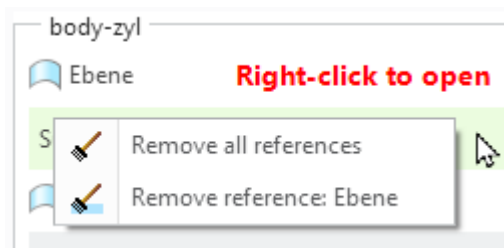


Selecting the reference Axis in the model

Use the functions *Create with references*  and *Repeated creation with references*  as soon as you have defined consistent placement references for multiple installations.

Tip: UDFs can be placed in assembly mode by enabling the required part.


Use the context menu to delete placement references.



Context menu

Placement references for bodies

In Creo 7.0 and later versions a body could be a placement reference, if the UDF has been created in Creo 7.0. If a UDF form requires only one body placement reference, this reference is filled in automatically with the active body. If the UDF shape has multiple body references, the bodies must be selected. If a body reference is to create a new body, this intent can be predefined in the UDF Form Editor.



 aussenring

Body: New Body

To reach the "New Body" setting interactively, cancel the body reference selection with the middle mouse button.

Place UDF dependently

A UDF can be placed dependent or independent. If a UDF is placed dependently, subsequent changes to the values in the UDF form (i. e. the GPH file) are applied to the UDF. This functionality can also be created in Creo.

Enter your choice in the checkbox *Place UDF independently* of the **tools menu**  ⁴⁴². The default for the checkbox is activated. This can be changed in the **editor**  ⁴⁵⁴.

16.2.5 Form section

The form displays variables of the currently selected UDF group. The form elements are arranged in a table with the following columns:

- Name of the variable
- Current values of the variables. Depending on the configuration, different input fields are displayed. Click on a displayed value to activate input fields.
- Units of the variables
- Title: the localized description
- Tools: Displays value tables or Javascript functions from appear in the Tools column, if available.

Both, the visibility as well as the order of the columns can be changed in the *Description* tab of [UDF Forms Editor](#)⁴⁶⁴.

Input field types

Depending on the UDF group used, different types of control elements are available in the form.

Warning: Check the Creo configuration option `show_dim_sign`, as GENIUS TOOLS Forms respects it.

`show_dim_sign=no`: Entering a negative value changes the direction vector of a dimension. The value becomes positive.

`show_dim_sign=yes`: When a negative value is entered, the value remains negative.

Input field

Length

Input fields accept any string for input. Restrictions (e.g. numbers only) can occur depending on the parameter type.

Selection field

Length
Width
300 mm

Selection fields (drop-down lists) contain a selection of possible inputs. They are predefined through the configuration.

Radio button

Length

☒ 600 mm

☐ 300 mm

☐ 350 mm

Radio buttons display predefined selectable values. Only one of the values can be selected.

Binary checkbox

Length ☒ y: 600; n: 300

Checkboxes allow to choose from two options. This can be Yes/No decisions, for example. Always two values are deposited in the configuration for checkboxes. With the checkmark set, the first value is used. Checkmark not set applies the second value.

Value table



Value tables control the presetting of form elements. Click on the icon to open the selection dialog.

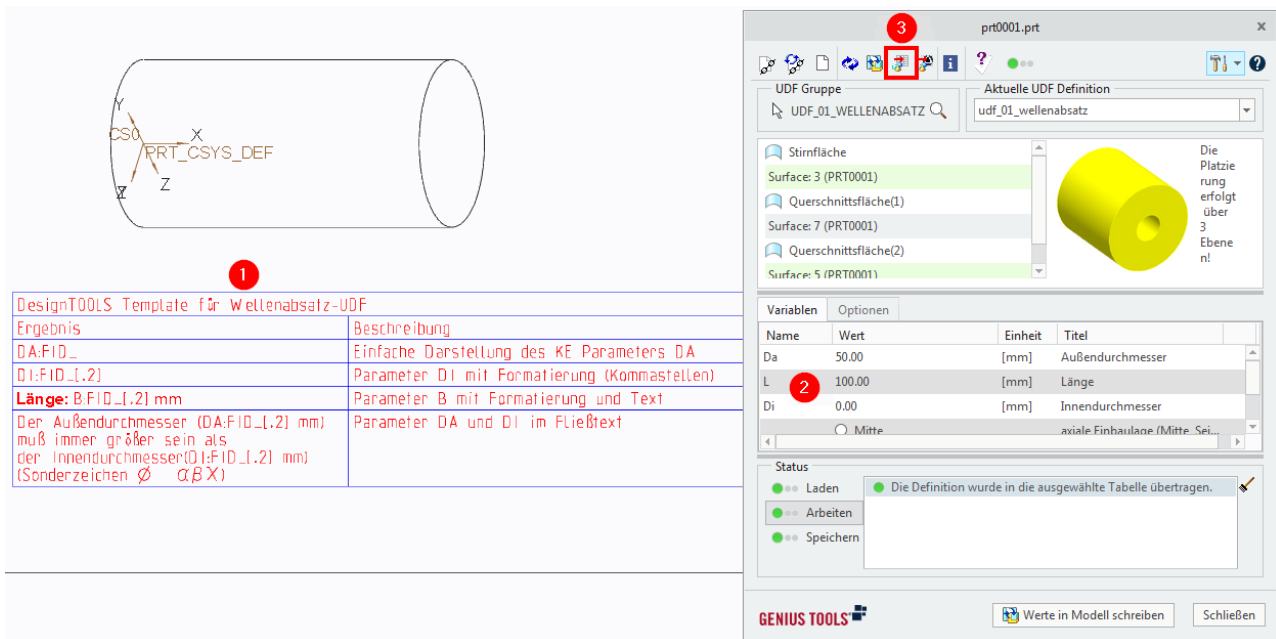
16.2.6 Use cases

In this section you will find use cases related to GENIUS TOOLS UDF Forms.

Linking UDFs to drawing tables

To link UDFs to tables, proceed as follows:

1. Place a UDF in a part.
2. Create a drawing.
3. Insert a table in the drawing.
4. Enter the required variables of the UDF in the table in the following notation:
VariableName:FID_[decimal places]
5. In UDF Forms, click the button *Apply to Table*.

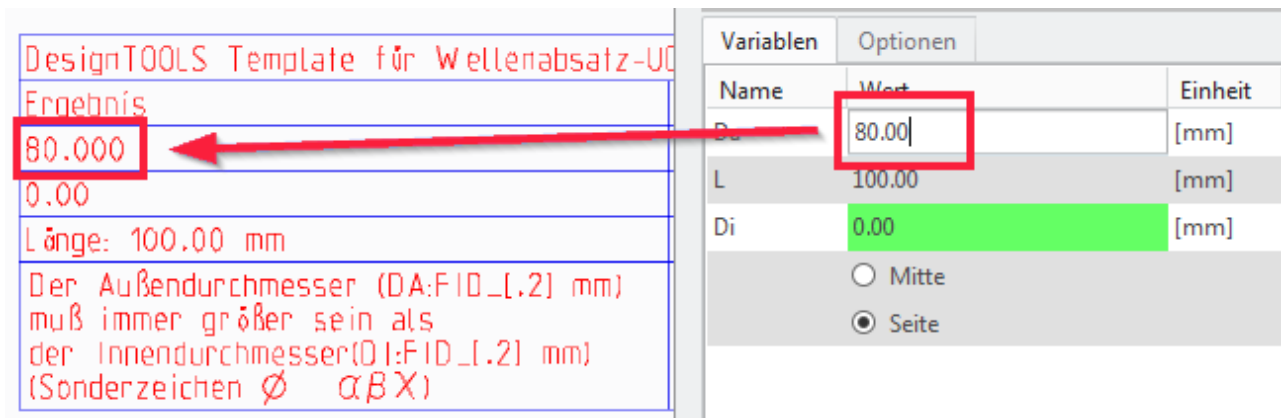


Create a table (1)

Enter the variables from the UDF (2)

Click on "Apply to table" (3) and select the table

Select the table created in the drawing. The UDF is now linked to the drawing table. The table is automatically updated each time the UDF is changed.



If the UDF is changed, the table is updated automatically

Linking UDFs to drawing symbols

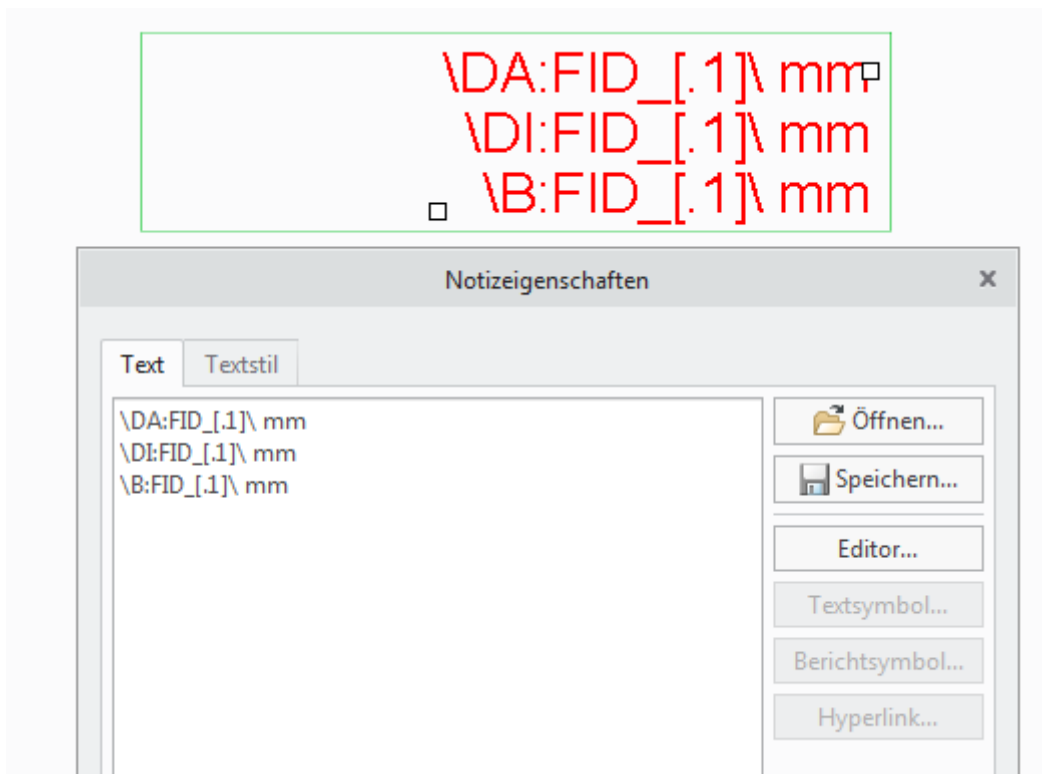
To link a UDF to an icon with variable text, proceed as follows:

Preparing the symbols

Create a separate symbol for each UDF!

Enter the variables of the UDF in the following notation as a note in the symbol:

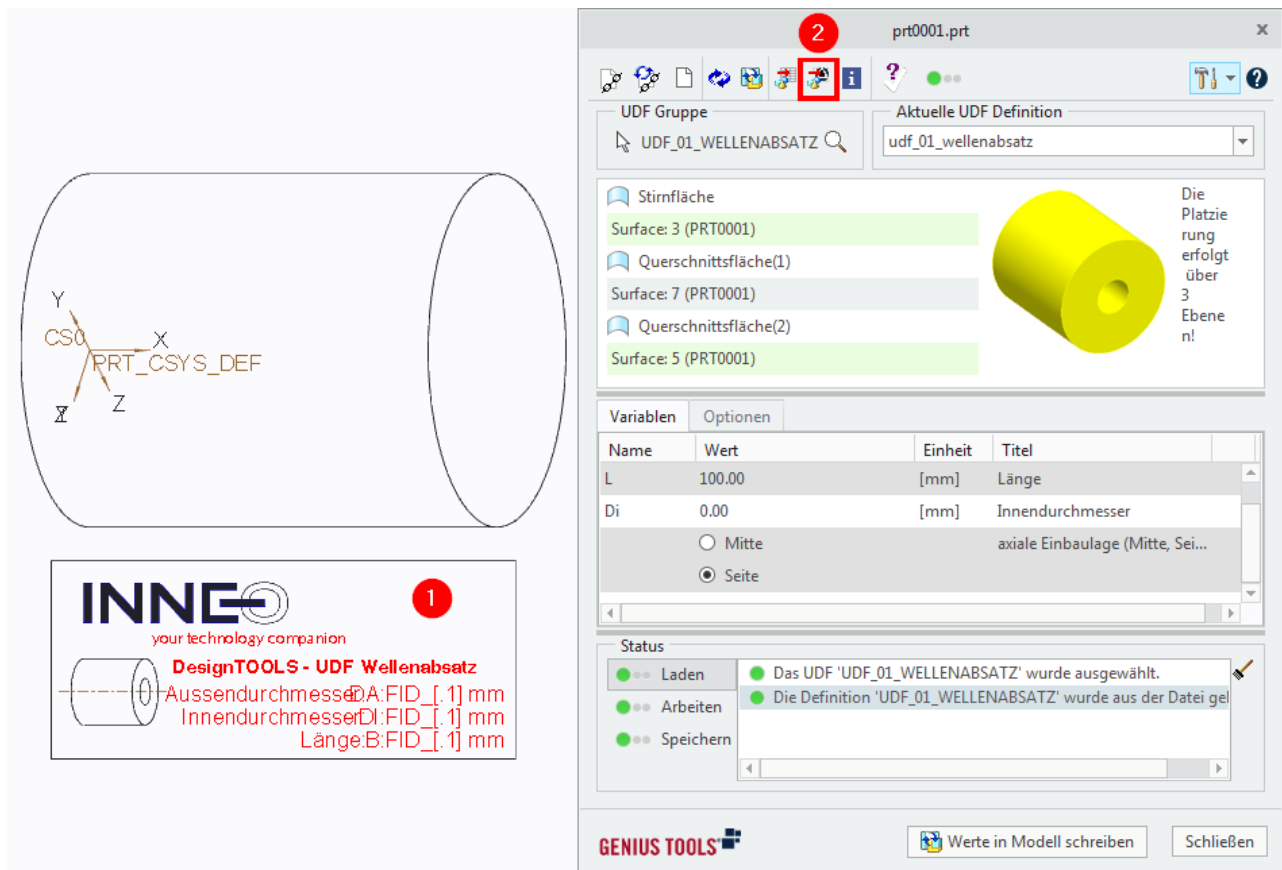
\VariableName:FID_[decimal places]\



Create a note with the desired variables in the symbol

Linking the UDF with the symbol

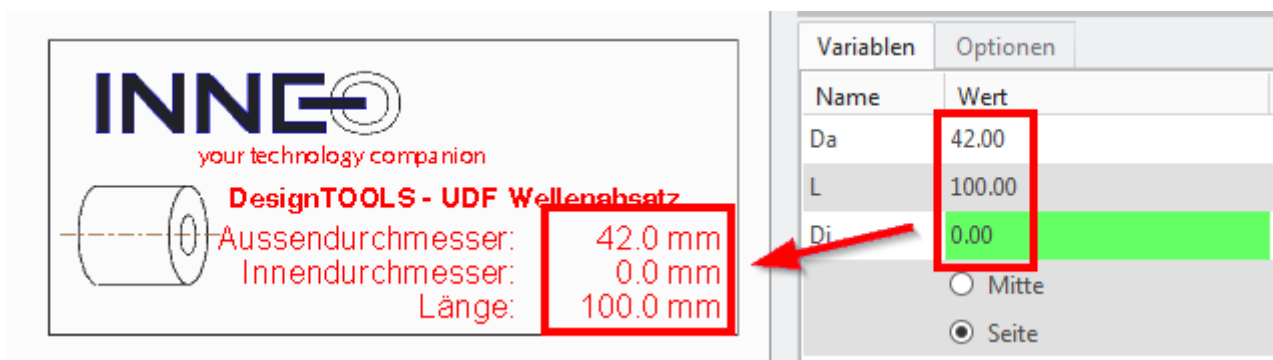
1. Place a UDF in a part.
2. Create a drawing.
3. Insert the appropriate symbol for the UDF in the drawing.
4. In UDF Forms, click the button *Apply to symbol*.



Place the prepared symbol (1)

Click on "Apply to symbol" (2) and select the placed symbol

Select the placed symbol in the drawing. The UDF is now linked to the drawing symbol. The symbol is updated automatically, each time the UDF is changed.



If the UDF is changed, the symbol updates automatically

16.3 Configuration

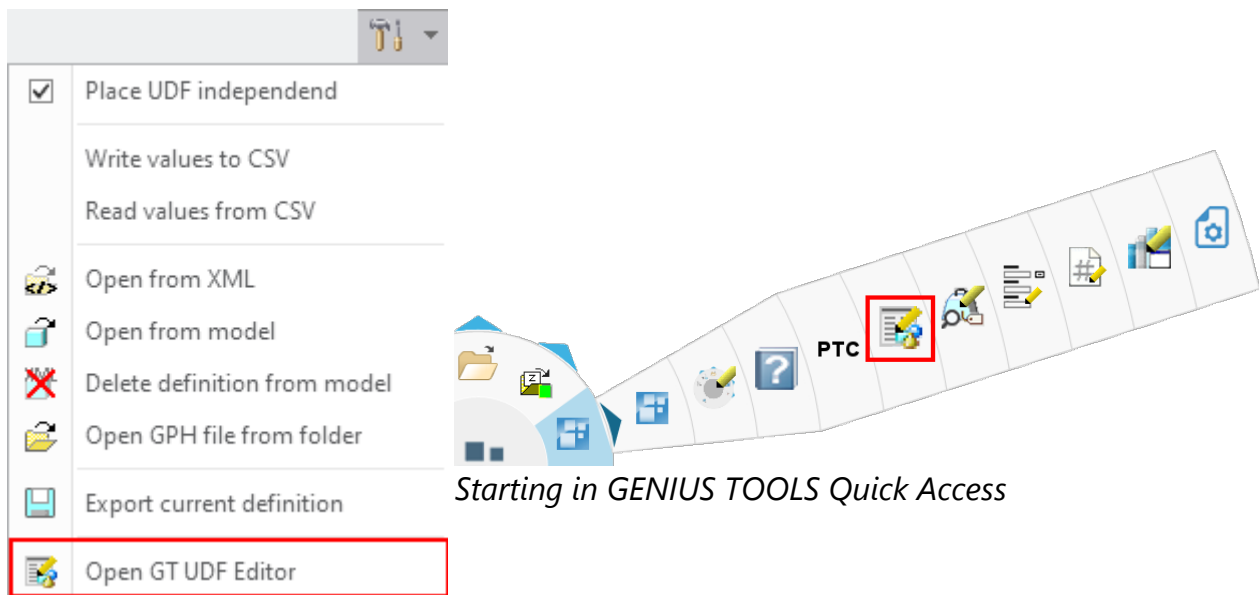
This section contains further information on configuration and structure of GENIUS TOOLS UDF Forms.

16.3.1 UDF Forms Editor

With GENIUS TOOLS UDF Forms UDF groups are created and edited. No changes are made to the UDF. All settings are stored in the XML structure of a UDF definition.

Starting the program

Start the editor from the tools menu of GENIUS TOOLS UDF Forms or via GENIUS TOOLS Quick Access (key[<]).

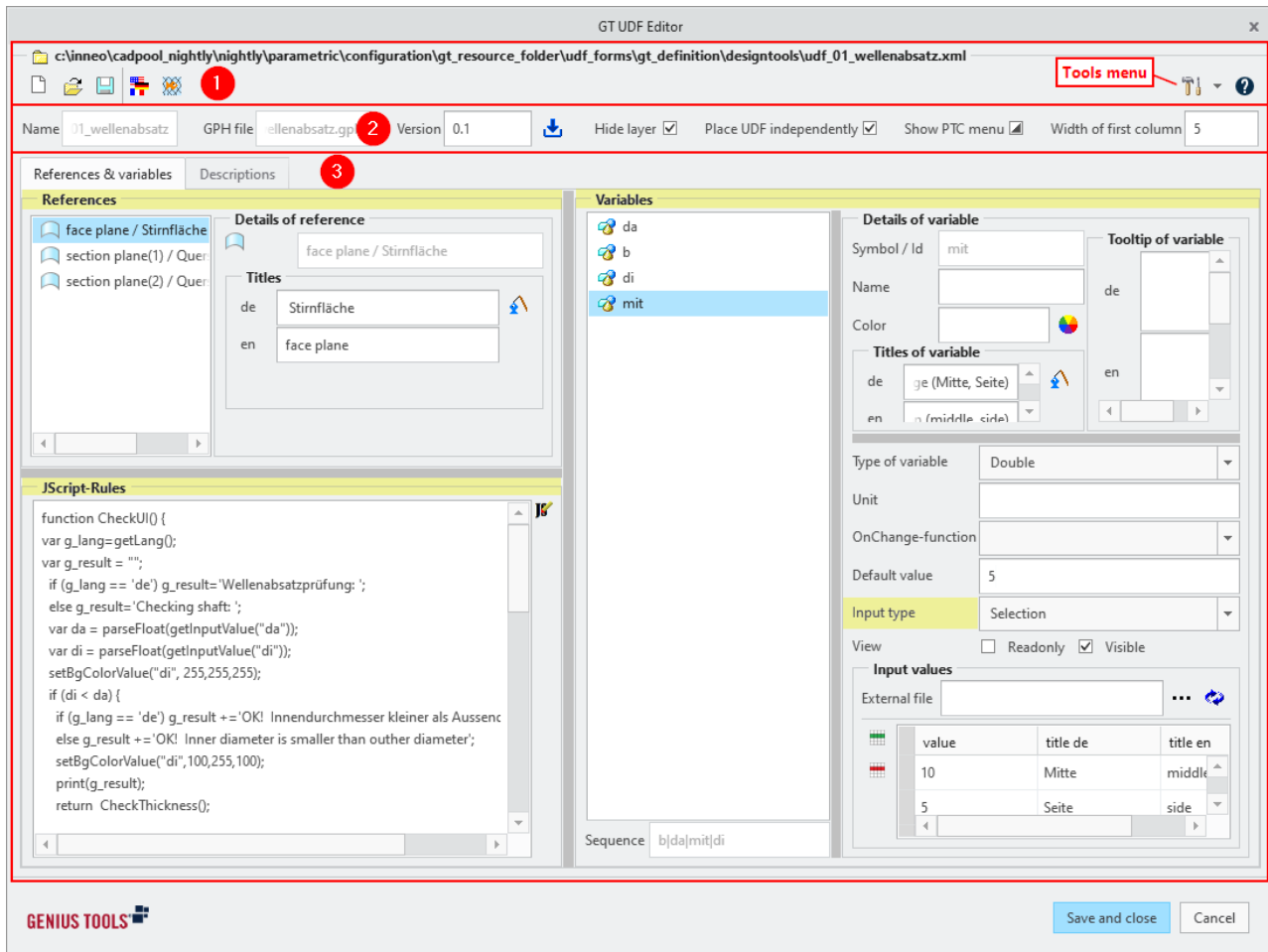


Starting in GENIUS TOOLS Quick Access

Starting from the tools menu

16.3.1.1 User interface

The user interface of GENIUS TOOLS UDF Editor consists of the following elements:










1. Command bar⁴⁵¹ with tools menu
2. Details of selected UDF⁴⁵³
3. Tabs:
 - References & Variables⁴⁵⁴ with the segments: References⁴⁵⁵ – JavaScript-Rules⁴⁵⁵ – Variables⁴⁵⁷ – Field types⁴⁵⁹
 - Descriptions⁴⁶³

16.3.1.2 Command bar

The following buttons are included in the command bar:

Icon	Name	Description
	Create new definition	Creates a new UDF definition from the

Icon	Name	Description
	from GPH file	selected UDF in XML format.
	Open from XML file	Opens a UDF definition from the file system.
	Save	Saves the current UDF definition.
	Display variant list	Displays a list of available variants of the current UDF.
	Edit Languages	Opens the dialog for managing the languages of a UDF definition.
	Automatic standard texts	<p>Opens the dialog for automatically setting standard texts for all variables and references. The table is also divided into these two areas.</p> <p>The identifier of the GPH file is used as the key for the translation for the references and the name for the variables.</p> <p>See Set standard texts for multiple elements⁴⁹⁷</p>
	Tools menu	The tools menu contains various supporting functions.
	Help	Opens the help.

The current location of the UDF definition is displayed above the command bar. If an internal UDF definition is edited, the footer turns yellow.



The footer changes color and indicates that an internal UDF definition is being edited.

Warning: Make sure to edit the correct definition!

Tools menu

Supporting functions are started from the tools menu.

Open a definition according to selected UDF model: Opens an external UDF definition for the model.

Create a new definition according to a selected UDF: Creates a UDF definition from the current selected UDF.

Open from GPH file: Opens a UDF definition from a UDF file.

Open from model: Opens a UDF definition from the current model.

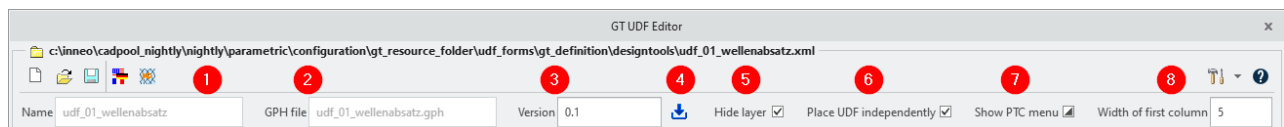
Save definition as XML file: Saves the current configuration to an external XML file.

Write definition into the model: Saves the current configuration into the UDF definition in the current model.

Delete a definition in the model: Deletes the selected UDF definition from the current model.

16.3.1.3 Details of UDF

The segment UDF Details contains the following information.



1. Name of UDF definition

Shows the name of the selected XML file, which defines the form of the UDF.


2. Name of GPH file

The GPH file stores the UDF.

3. Version

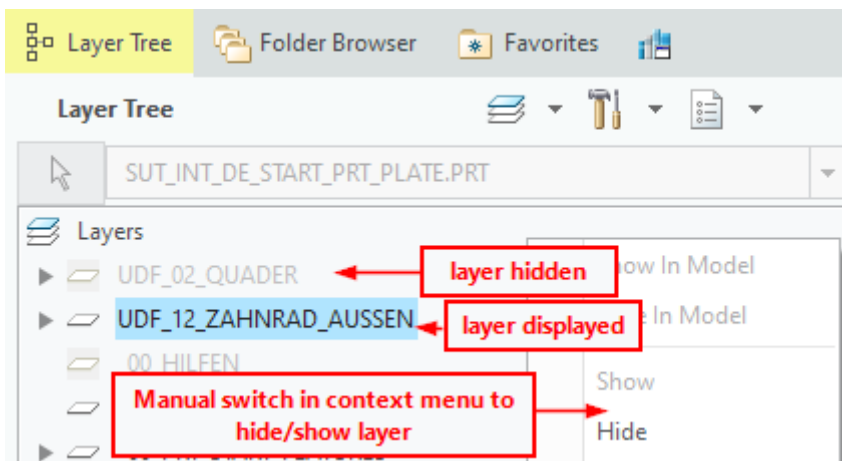
Displays the current version of the UDF definition (XML).

4. Update

The button  opens the dialog that updates the UDF definition with the current GPH file, see %TARGETTITLE%>⁴⁶⁵.

5. Hide layer

A slide with the same name is created for each built-in UDF. All auxiliary elements are located on this slide. If you create geometries with a UDF, these are displayed, but not the auxiliary elements required for the construction (default setting with check mark set). If you create auxiliary elements with UDF, then it is recommended not to set the check mark, so that the slide is displayed and you can see the implemented UDF.



6. Place UDF independently

Define the default setting for this checkbox in the tools menu of the GENIUS TOOLS UDF Forms dialog.

- Place UDF independently (default): Changes to the UDF form (the GPH file) will not be applied to the UDF.
- Place UDF dependently: Changes to the UDF form (the GPH file) will be applied to the UDF after updating.

7. Show PTC menu

This checkbox is used to determine whether the PTC menu for selecting the assembly properties is displayed: *On* ☒ / *Off* ☐ / *Conditionally On* ☒. *Conditionally on* ☒ is selected by default. With this function, the menu for selecting assembly properties is displayed only if Creo Parametric decides that it is required for the particular UDF.

8. Width of first column

Defines the width of the first column in the Variables area. Default is 5. Enter -1 if you want the first column to enlarge automatically when the dialog window is enlarged.

The first column can be defined in the Description tab under [Column order](#)⁴⁶⁴.

Variables		
Name	Value	Unit
Da	50.00	[mm]
L	100.00	[mm]
Di	0.00	[mm]

16.3.1.4 References and variables tab

The References and Variables tab consists of the sections:

- References:⁴⁵⁵ for managing the placement references of a UDF.
- Javascript rules⁴⁵⁵: executable code for a UDF form

- List of available variables⁴⁵⁷ of the UDF with details and field types
- Instances: This section is only displayed if a UDF contains variants or instances. Use the drop-down list *Function after instance change* to select a JavaScript OnChange function. Use the *Visible* check box to show or hide the variant selection in UDF Forms.

References

The placement references of a UDF are managed here.

The references of a UDF are displayed on the left. This references cannot be edited, since UDFs are not changed by the editor.

If references in a UDF are named in the *English text / translation text* format (with spaces before and after the slash), the reference titles are automatically filled in.

Reference details are displayed on the right. Here, localized titles of the references are defined for the UDF form. Select a reference and enter the language-dependent text directly in the input fields. Click the flag icon to edit the localized titles, alternatively.

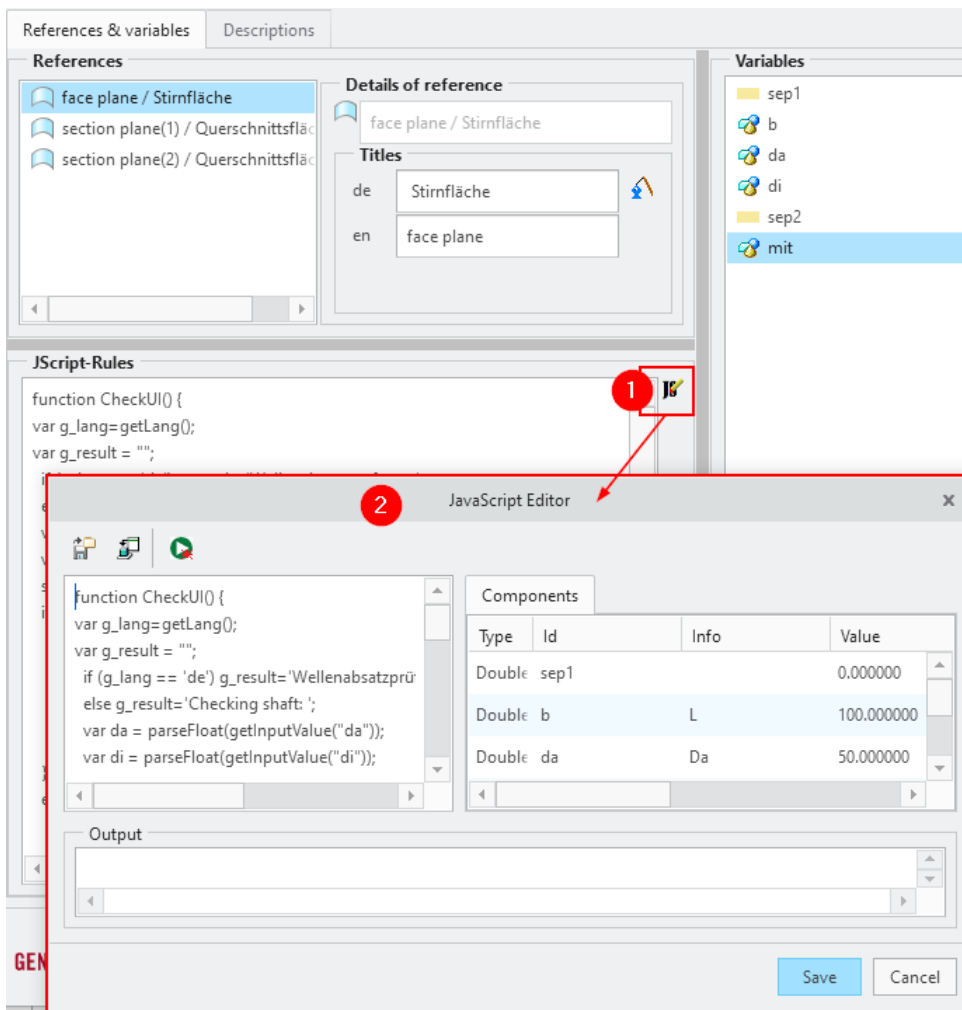
UDF references

1. Select a reference
2. Enter the localized reference directly or
3. Select a standard text via (Select default text⁴⁹⁷)
4. For body references only, a pre-selection can be made whether to create a new body.

JavaScript in UDF Forms

In the section *JScript Rules* in the tab *References and Variables*, executable JavaScript code for a UDF form is stored.

Enter JavaScript directly into the input field or use the *JavaScript Editor*⁵⁷⁷. Click on the JavaScript icon to open the editor. Go [here](#)⁵⁸² for an explanation of the JavaScript functions and short examples.




JavaScript symbol (1) opens the editor (2)

Tip: Use JavaScript to adjust the visibility and background color of separators.

Time of execution

JavaScript functions can be executed at different times. You can add as many as required to your JavaScript code.

Time of execution	Function
After loading a UDF group	PostLoad
After loading a value table (CSV/XML file)	PostLoadFromFile
Before saving a UDF group	PreSave
After saving a UDF group	PostSave

Time of execution	Function
After clicking the button <i>Check values</i> in the UDF Forms dialog	CheckUI
After changing a variable value	OnChange – is activated in the <i>variables details</i>  ⁴⁵⁹ area

The names of *PostLoad*, *PostLoadFromFile*, *PreSave* and *CheckUI* are fixed. Functions of the type OnChange can have any name.

Please note: Use the JavaScript function *creoMapkeyAddToStack* only as a PostSave function, because it executes mapkeys and mapkeys usually close windows.

Variables

The *Variables* section manages the available variables for a UDF. UDF parameters as well as dimensions can be variables.

Please note: Parameters from a UDF are also displayed under Variables. If parameters are to be controlled with UDF Forms, they must be created at the first feature of the UDF (GPH file). The parameters are written to the UDF group in the model later.

Creating variables and separators

In the variable list (1) on the left, two types of objects are created: Variables and Separators.

- Use separators to logically organize the variables in the UDF Form, for example as section headings.
- Variables come directly from the UDF or are first defined in the UDF Form. Variables are needed to customize the UDF using the form area, but can also be used in JavaScript as auxiliary variables for calculation.

You can use drag-and-drop to reorder the variables. The original variable sequence of the GPH file is displayed below the variable list in the Sequence field (2).

Variables

- 1: Variables list (sep1, b, da, di, sep2, mit)
- 2: Sequence field (b|da|mit|di)
- 3: Symbol / Id (mit)
- 4: Name
- 5: Color
- 6: Titles of variable (de: axiale Einbaulage (Mitte, Seite), en: axial position (middle, side))
- 7: Tooltip of variable
- 8: Type of variable (Double)
- 9: Unit
- 10: OnChange-function
- 11: Default value (5)
- 12: Input type (Radio)
- 13: View (Readonly, Visible)
- 14: Input values table

Annotations:

- Drag and drop for ordering
- Open context menu with right-mouse click

value	title de	title en
10	Mitte	middle
5	Seite	side

Variables area in the References & Variables tab

Variable details

Click on a variable to display its properties in the details area.

3. Symbol: The internal name of a variable or separator. The symbol can be changed for manually created objects.

4. Name: Defines the name of a variable. The name is displayed in the third column of the form.

5. Color: Defines the background color of the UI element. The color can be selected via the [color dialog](#)⁵⁹⁰.

6. Titel: Defines the title of a variable or separator. The title is displayed in the fourth column of a UDF form. Standard texts can be selected in the [default text dialog](#)⁴⁹⁷.

7. Tooltip: Defines localized tooltips. These are displayed when the cursor is placed over the variable or separator in UDF form.

8. Type of variable: Defines the data type of a variable. Text, Integer, Boolean and Double are available.

9. Unit: Defines the unit of a variable.

10. OnChange-Function: OnChange functions are executed when the value of a form element is changed automatically (e. g. by value tables) or manually by leaving the input field or pressing Enter. The available options are CheckUI, PreSave, CheckThickness, see chapter [Explanations and examples](#)⁵⁸².

Please note: Avoid endless loops through functions.

11. Default value: The value that is displayed by default in the input field of the form.

Defines the value of a variable that is entered into the mask when the surface is reloaded and updated. This value is also entered for value tables. Behind a value of a value table there is always a set of variables with one value each. When saving the UDF definition, the system checks whether the default value of the value table matches the default values of the variables of the associated record. Instances are not taken into account!

12. Input type: Defines the type of field for a variable: Text field, Selection, Radio (radio button), Value table and Extended Selection, see chapter [Selection fields and value tables](#)⁴⁵⁹. Boolean parameter will automatically be displayed as a checkbox.

13. View: Set a variable that users should not be allowed to change to *Readonly* and to *Visible* if the variable is to be displayed to users.

14. Input values: Define all possible values for the UDF Form or import values from an external file, see chapter [Selection fields and value tables](#)⁴⁵⁹. This area appears only for the input types Selection, Radio, Value Table and Extended Selection.

Selection fields and value tables

In the drop down menu Input type in the Variables area of GENIUS TOOLS UDF Forms Editor you can select different field types in forms:

Text field, selection, radio (option field), value table and extended selection field. With the exception of text fields, the values to be available for selection in the UDF form are specified here.

Selection fields and radio buttons

Selection fields and radio buttons are controlled by a table, whose values are defined here or can be imported from an external file.

Input type: Selection

View: ☐ Readonly ☒ Visible

Input values

External file: [Field with file explorer icon]

value	title de	title en
10	Mitte	middle
5	Seite	side

Manual entry

- Use the buttons on the left to add (green) and remove (red) rows.

Import values from external file

- The values from the CSV file are saved in the UDF form.
- When the form is opened, the software checks for the linked CSV file and adopts any updates.
- Select the required file in the field *External file* with the button (...). By default the file explorer opens in the folder `%GT_RESOURCE_FOLDER%\udf_forms\gt_data` by default. You can define another folder with the option `gtuf_external_data_folder` from version 8.0.2. on.
- If you want to upload changes to a CSV file while working in the UDF Forms Editor, click the refresh button (*Updates the value table*).

Create external file

The CSV file has to contain the selection values in the first column and the labels in the other columns. The header cell of the first column is ignored. The header cells of the other columns have to contain the two-digit codes for the corresponding languages.

	A	B	C
1		de	en
2	10	Mitte	middle
3	5	Seite	side

The CSV file can be created in the system folder *gt_data* or another folder (`gtuf_external_data_folder`). Up to version 8.0.2. the CSV file is searched in the working directory.

Reading the external file

As of version 8.0.2. there are several ways to link an external CSV file for selection fields.

1. Filename with extension, no path: the file must be located in the folder defined in the `gtuf_external_data_folder` configuration option. Default is `%GT_RESOURCE_FOLDER%\udf_forms\gt_data`.
2. Relative to the folder defined in the `gtuf_external_data_folder` configuration option. Use this if the CSV file should be located in another folder, e. g. together with XML

files. Example: the files *din13.csv* and *din13.xml* are located in %
`GT_RESOURCE_FOLDER%udf_forms\gt_definition\company`. The specification of the
 external file is then `..\gt_definition\company`

3. Absolute path: this should not be used, since changes cannot be transferred.

Value tables

Value tables control several variables simultaneously. Each table line is a single variant. Use the arrow buttons on the right to move rows or columns.

mit:	da:Da	di:Di
Mitte	40	30
Seite	60	50

Use the arrow buttons on the right to move rows and columns in the table.

- The values from the CSV file are saved in the UDF form.
- When the form is opened, the software checks for the linked CSV file and adopts any updates.
- Select the required file in the field *External file* with the button (...). By default the file explorer opens in the folder `%GT_RESOURCE_FOLDER%udf_forms\gt_data` by default. You can define another folder with the option `gtuf_external_data_folder` from version 8.0.2. on.
- If you want to upload changes to a CSV file while working in the UDF Forms Editor, click the refresh button (Updates the value table).

Create external file

To create the header rows for a CSV file, first create the required columns in the value table. Then, export the table as a CSV file . The exported file contains the required table header.





	A	B	C
1	mit:	da:Da	di:Di
2	mit	da	di
3	Mitte	40	30
4	Seite	60	50

Read external file

There are three ways to specify an external CSV file, see section under [Selection and option fields](#)⁴⁶⁰.

Editing value tables

The following buttons are available for editing.

Icon	Name	Description
	Import value table from file	Imports a value table from the following file formats: <ul style="list-style-type: none"> – xls (Excel 97-2003) – xlsx (Excel 2003-2016) – csv (Comma-separated values) – txt (text files with UTF-16LE/Unicode formatting)
	Export value table to file	Exports a value table to the formats: <ul style="list-style-type: none"> – xls (Excel 97-2003) – xlsx (Excel 2003-2016) – csv (Comma-separated values)
	Edit value table in spreadsheet software	The value table is opened in a spreadsheet program (depending on the client computer) and can be edited. Then it is reimported.
	Sort selected column	Sorts the rows by the active column. Three sorting orders are available: <ul style="list-style-type: none"> – 0-Z – Z-0 – initial
	Add/Remove row	Adds a new line under the current cursor position. To remove, select a row and click <i>Remove row</i> .
	Add/Remove column	Adds a new column to the right of the current cursor position. To remove,


Icon	Name	Description
		select a column and click <i>Remove column</i> .

Please note: Any changes to a value table made when editing the table in a spreadsheet software (deleting, moving or editing rows or columns) are adopted as-is in GENIUS TOOLS UDF Forms.

16.3.1.5 Description tab

In the Description tab, general information for a UDF form is stored.

Titles

Define the localized names of the UDF. They will be displayed in the UDF Form. Standard texts can be used via the button  (Description of the standard text selection dialog⁴⁹⁷).

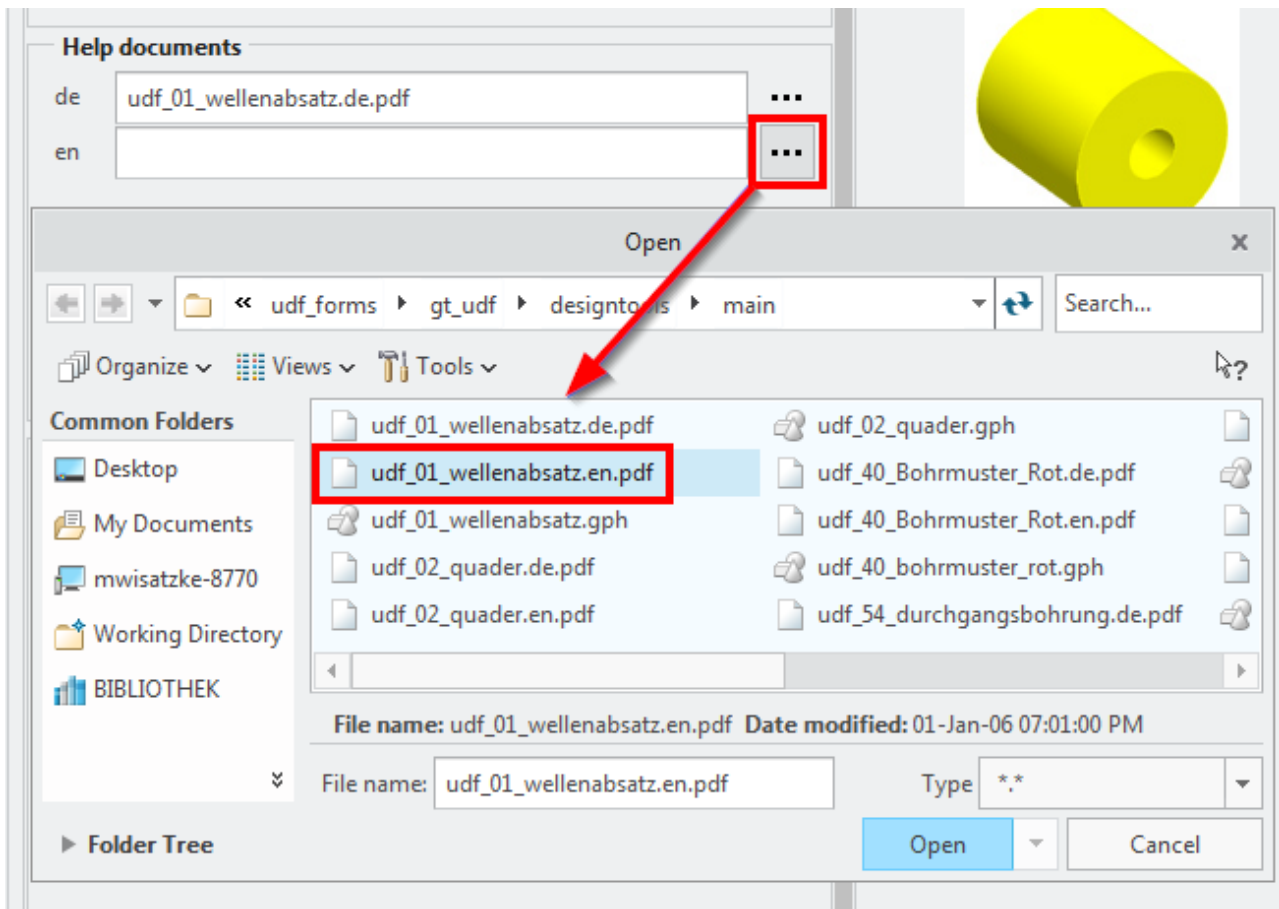
Help documents

Add language-dependent documents here. Enter the file path directly into an input field or select a file using the file selection.

Help documents have to be accessible and openable on client computers! The applications to use are specified by the client computer.

Supported path formats:

- absolute paths
- UNC paths
- HTTP paths
- If you enter the file name only, the help document is searched in the group directory and in the directory configured as `gtuf_data_folder`.



Open the file selection and select a help document

Descriptions

Enter a localized description of the UDF here. The description is displayed in the UDF Form dialog next to the placement references, depending on the Creo language.

Images

Preview images are configured in this area. Since images can contain localized information, they can be stored individually for each language.

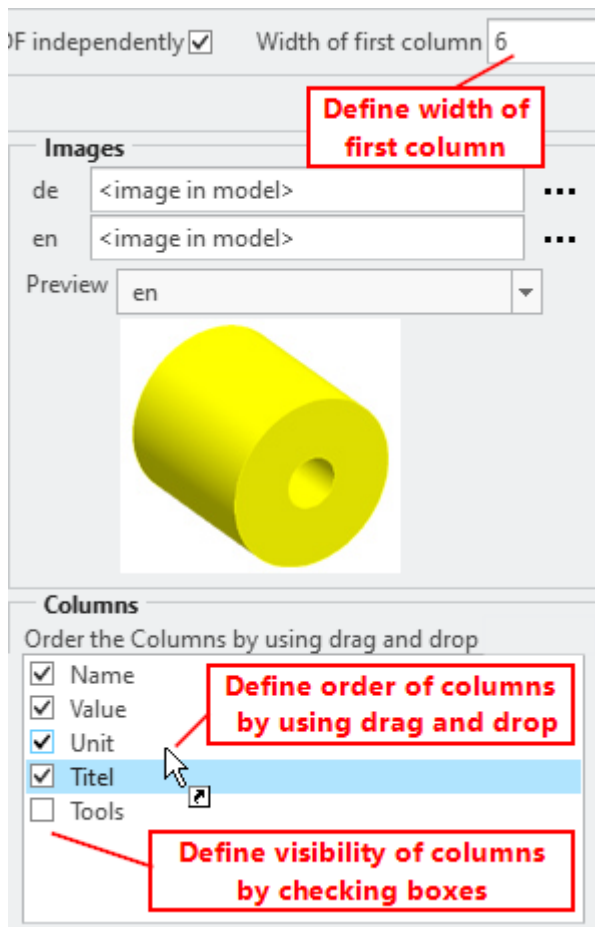
Enter the file path directly into an input field or select a file using the file selection.

Use the drop-down list *Preview* to switch between the localized preview images.

Order of columns

In UDF Forms Editor you can arrange the columns in the *Variables* area of a UDF Form by moving them per drag-and-drop and hide them by unchecking the boxes.

For the first column you can define the width in the upper Details area⁴⁵⁴.



The Tools column displays value tables⁴⁶¹ and Javascript functions⁴⁶⁵, if available.

16.3.2 Updating UDF definitions

The GPH file on which a UDF is based is being modified several times during the development process.

Creo and therefore also GENIUS TOOLS UDF Forms always uses the current GPH file.

The UDF definition used by the module GENIUS TOOLS UDF Forms is read from the directory and then no longer matches the GPH.

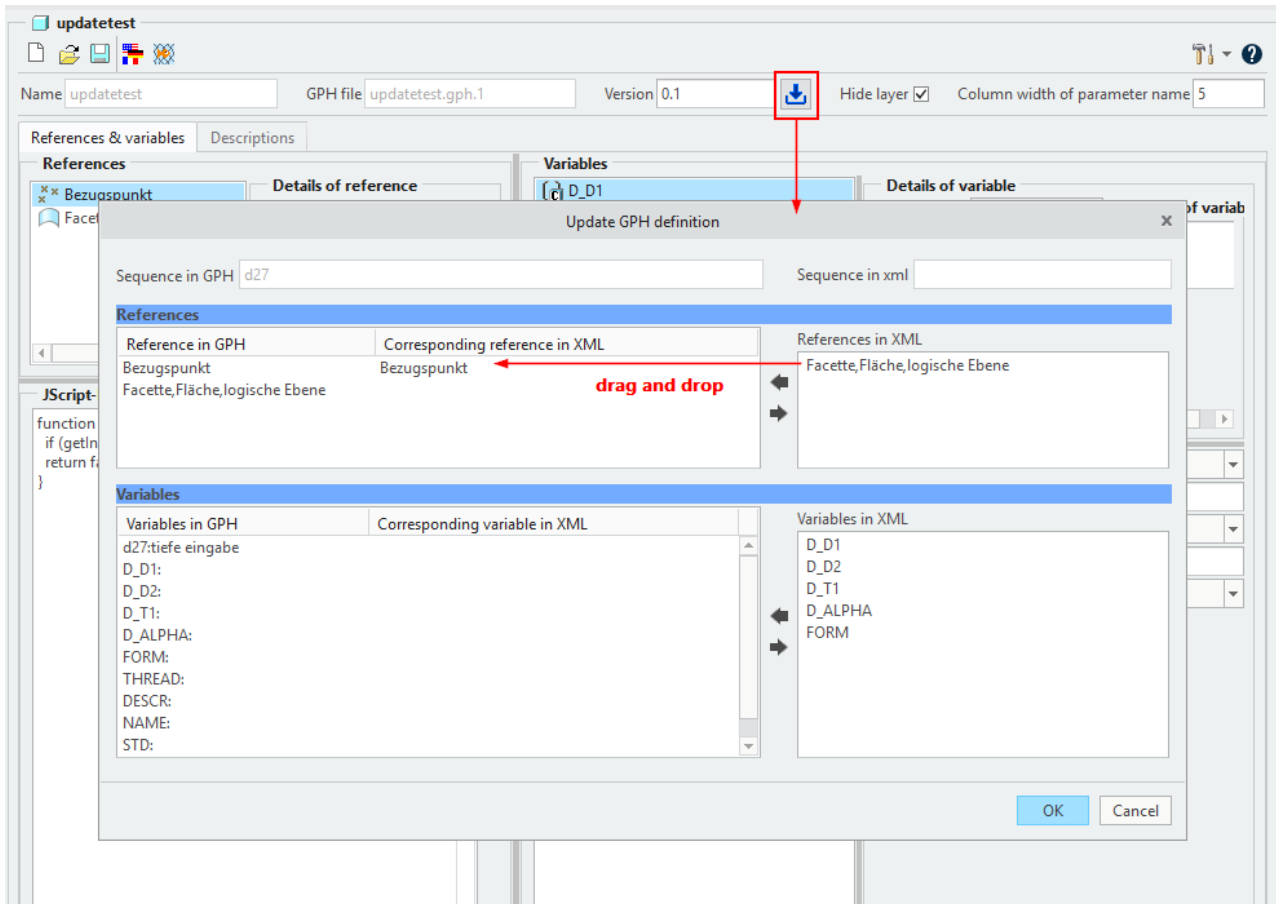
If a new UDF definition is created for the current GPH, the descriptions, help documents, JavaScript code, etc. are no longer included in the definition.

To solve the problem, there is now an update mechanism in GENIUS TOOLS UDF Forms Editor.

In this context, a version is stored in the UDF definition. When creating the UDF, the parameter `UDF_VERSION` is added to the created group. Several UDF definitions of the same name are now also supported in the model.

Warning: GPH files of older versions must not be in the session during the update process!

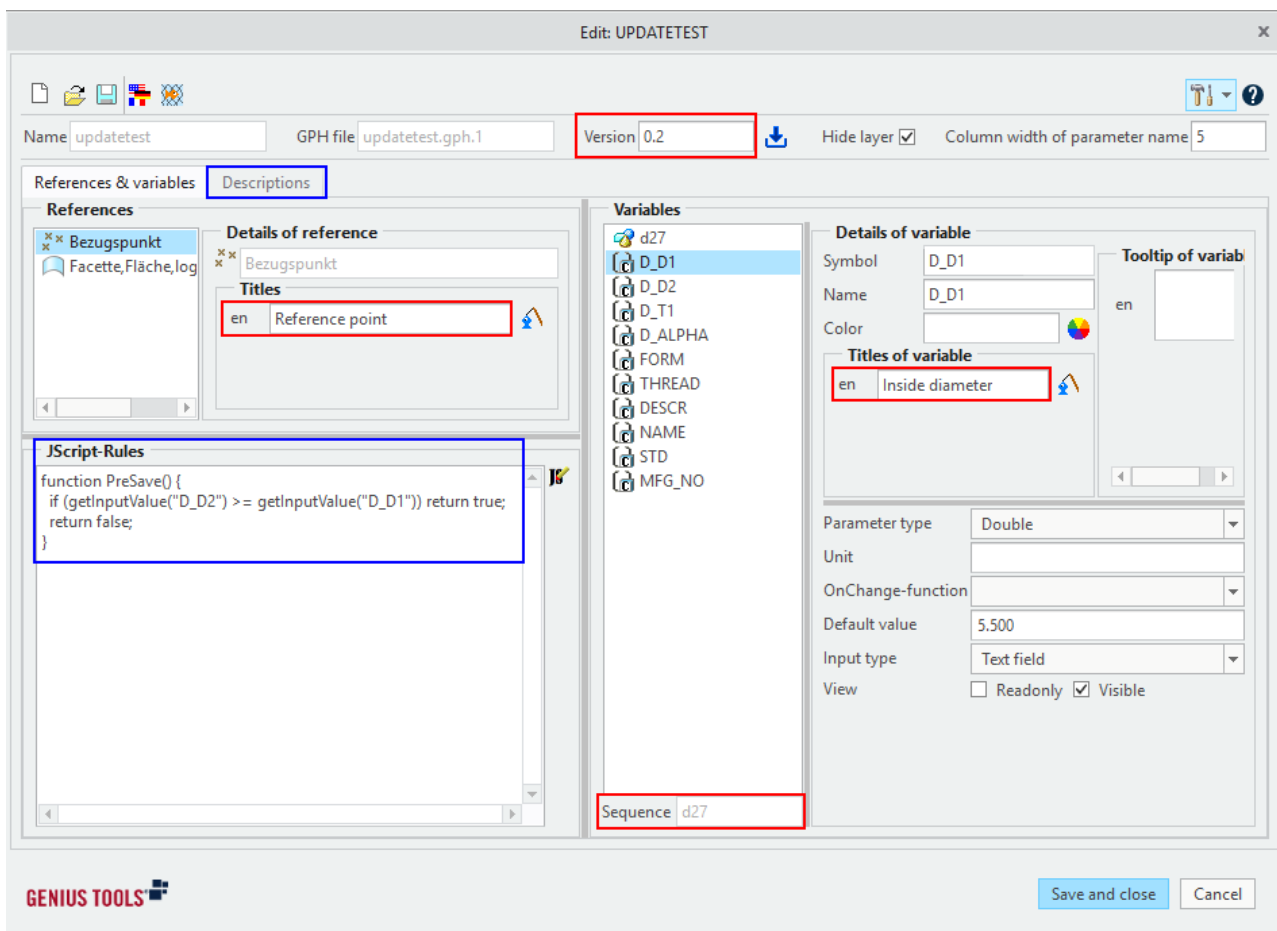
The update mechanism is started in the editor:



Update Dialog

In the update dialog, the individual references and variables of the current GPH are now compared with the references and variables of the UDF definition. The program creates an automatic assignment. Further assignments can be created using DragAndDrop. The assignment can also be done with the arrow keys. The table cells of both areas must be selected.

After saving, the version of the UDF definition is automatically incremented by 1, references that are no longer needed are removed and former variables from the sequence are also removed. Descriptions, help, JavaScript code etc. are retained.

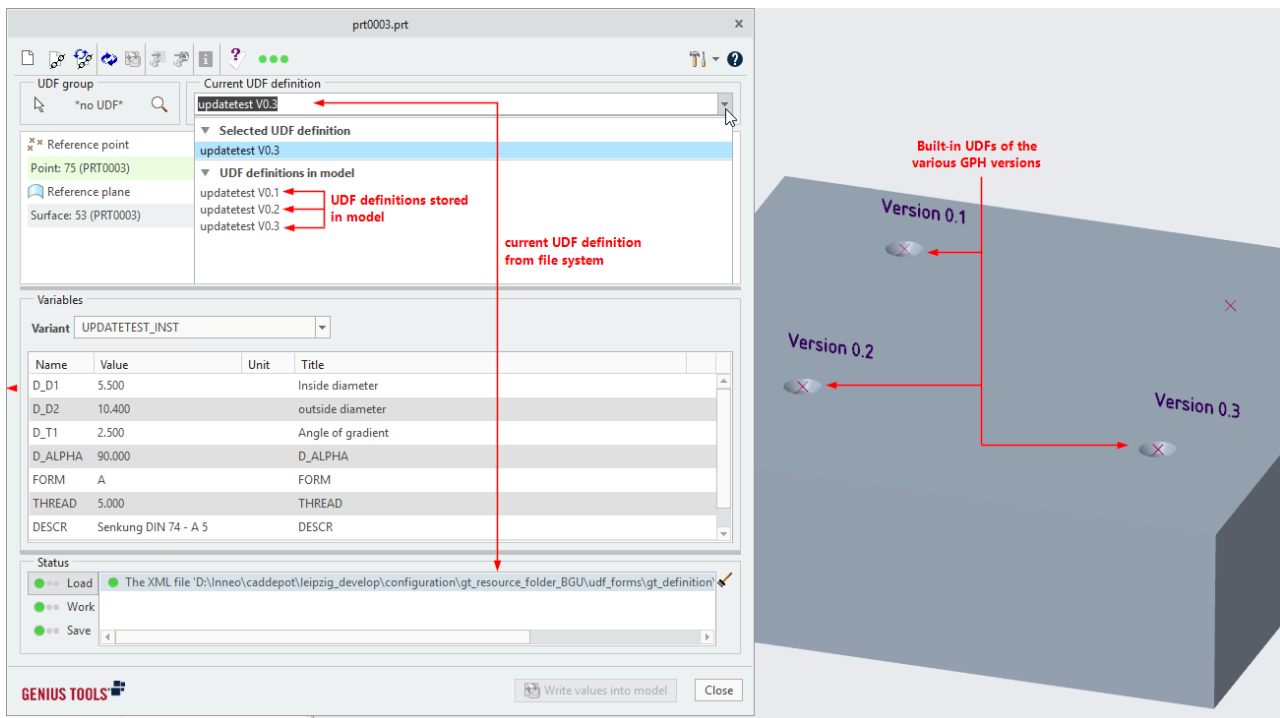


The screenshot shows the 'Edit: UPDATETEST' window. At the top, there are fields for 'Name' (updatetest), 'GPH file' (updatetest.gph.1), 'Version' (0.2), 'Hide layer' (checked), and 'Column width of parameter name' (5). Below this is a 'References & variables' section with two tabs: 'References' and 'Descriptions'. The 'References' tab is active, showing a list of references on the left and 'Details of reference' on the right. The 'Details of reference' shows a title 'en Reference point'. Below the references is a 'JScript-Rules' section with a code editor containing a function PreSave(). To the right of the references is a 'Variables' section with a list of variables and 'Details of variable' on the right. The 'Details of variable' shows a title 'en Inside diameter', a symbol 'D_D1', a name 'D_D1', a color picker, a parameter type of 'Double', a unit, an on-change function, a default value of '5.500', an input type of 'Text field', and view options 'Readonly' (unchecked) and 'Visible' (checked). At the bottom right, there are 'Save and close' and 'Cancel' buttons.

Result of the update

The blue areas show the adjusted or accepted values and the green areas show the values retained.

Model with UDFs in three different versions



After the update, UDF groups of older versions can also be edited with the corresponding UDF definitions.

This way, dimensions and parameters can be adjusted. The current GPH is always used for exchanging instances or creating a new UDF group.

17 Utilities

GENIUS TOOLS Utilities comprises several individual applications. Utilities contains the following applications which can be used in different modes.

Please note: Many functions are only available with subscription licenses for GENIUS TOOLS for Creo. Consult the list in chapter [License-dependent functions](#).²²

Function	Mode			
	Creo Standby	Assembly	Part	Drawing
3D Note Form ⁴⁷⁰		X	X	
Close All Other Windows ⁴⁷⁷	X	X	X	X
CS Assembler ⁴⁹³		X		
Component Parameter ⁴⁷⁹		X		
Configuration Utility ⁵⁵⁸	X	X	X	X
Copy Component Parameter To Substitution Component Parameter ⁴⁸⁵		X		
Create Search.pro ⁴⁸⁷	X			
Create Tolerance Table ⁴⁹¹				X
CS Assembler ⁴⁹³		X		
Default Text Editor ⁴⁹⁵	X			
Export Table 1:1 to Excel ⁵⁰⁶		X		X
Export Table to CSV ⁵⁰⁷				X
Export Table to Excel ⁵⁰⁹		X		X
Export Points ⁴⁹⁸		X	X	
Extend Relations ⁵¹⁹		X	X	

Function	Mode			
	Creo Standby	Assembly	Part	Drawing
Extended Dimensions Functions ⁵²⁴		X	X	
Full Backup ⁵²⁷	X	X	X	X
Load Save Converter ⁵³³	X			
Open Base Model ⁵³⁵		Features with reference models		
Open / Create Drawing ⁵³⁶		X	X	
Select Surfaces by Color ⁵³⁷			X	
Show Information ⁵³⁸	X			
Show Pitch ⁵⁴²		X	X	
Toggle Symbol Variants ⁵⁴⁵				X

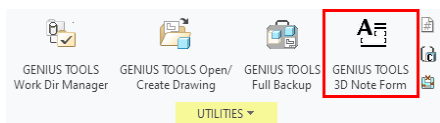
Also refer to the chapter GENIUS TOOLS for Creo ⁵⁹³ configuration.

17.1 3D Note Form

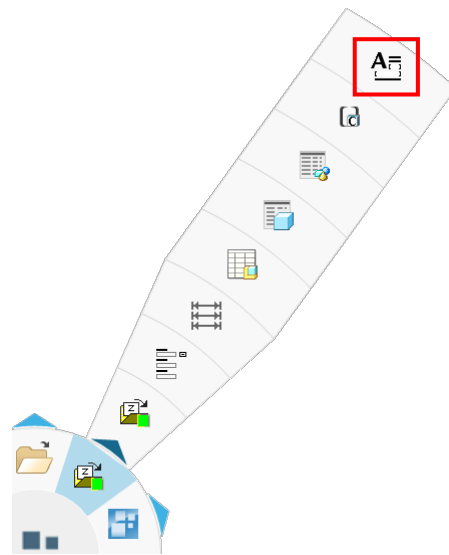
The component *3D Note Form* allows you to modify dimension and parameter values of all 3D notes of a model quickly and comfortably in one dialog box, i. e. without having to open each 3D note separately.

Starting the program: in assembly mode

3D Note Form can only be started in assembly mode, or if notes exist in a part. Start *3D Note Form* via the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key)



Starting via the ribbon menu



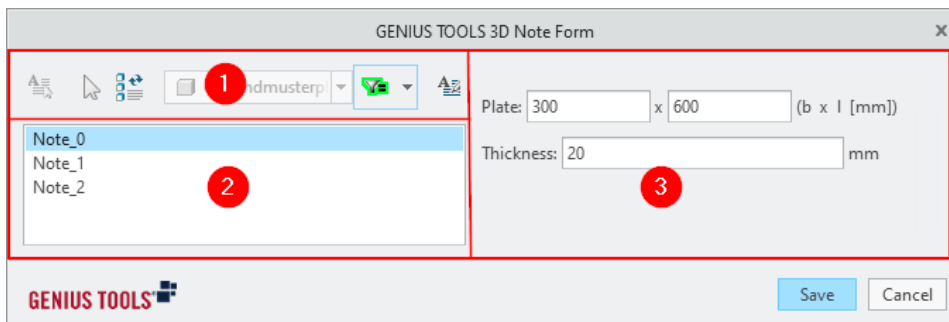
Starting via Quick Access

Hiding the 3D Note Form button

You can use the `gtu_start_3d_note_form` configuration option to disable the display of the *3D Note Form* button in the GENIUS TOOLS menu ribbon. (Default is 1 = On)

17.1.1 User interface

The user interface of the *3D Note Form* consists of the following areas:



1. Command bar






The command bar includes the selection list. In assembly mode this drop-down menu displays the most previously selected assemblies and parts.

2. Notes list

3. Current note

Command bar

The following buttons are included in the command bar:

Icon	Name	Description
	Note selection	Allows to select notes directly in the Creo main window.
	Model selection	Enables direct selection of models in the model tree or in the graphics window. Note: The function is only available in assembly mode!
	Save and regenerate	Saves the current note and regenerates the model with the current values.
 <input checked="" type="checkbox"/> Hide notes without input panels <input checked="" type="checkbox"/> Hide notes from holes	Filter	Hides notes with input panels and/or notes from holes (green) or displays them.
	Text editor	Opens and closes the dialog for text editing of a note.

Configuration options for the dialog box

gtu_3d_note_form_filter_hole_notes

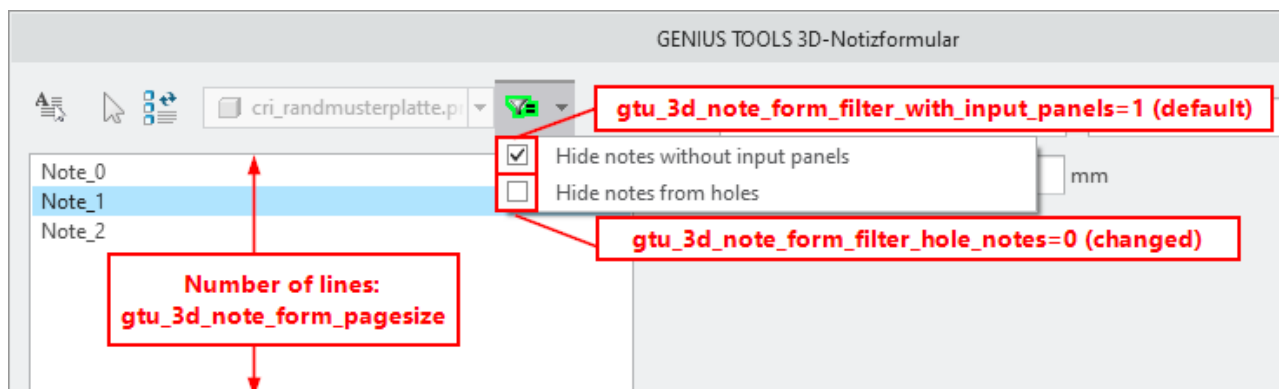
Defines whether notes on holes are hidden when the dialog box is opened (1, filter is enabled) or not (0). Default value: 1

gtu_3d_note_form_filter_with_input_panels

Defines whether notes without input panels are hidden when the dialog box is opened (1, filter is enabled) or not (0). Default value: 1

gtu_3d_note_form_pagesize

Defines the number of lines shown per page. Default: 15



17.1.2 Displaying and editing notes

Notes are displayed with static and dynamic content. Static contents of a note are texts. Dynamic contents like parameters or dimensions are displayed as editable input fields. If you use static texts, they can be changed in the *3D Note Form text editor*⁴⁷².

Enter parameters and dimensions into notes in following notation to make them editable with *3D Note Form*:

Parameter: &ParameterName

Dimensions: &DimensionName

Modify the values in the input fields as desired. After saving and regenerating the new values are transferred to the model.

Please note: If notes are longer than 15 lines, they are displayed in page mode. In this case, a note is split up and two buttons to switch pages are displayed.

Notes with calculations

The *3D Note Form* supports arithmetic operations and mathematical functions.

They differ in notation from default arithmetic operations in Creo. To use an arithmetic operation the expression must begin with the equals sign (=).

Example: "=12-d4" or "=Math.pow(d2,3)"

Please note: Dimension units will be ignored in calculations, only the values are used.

Mathematical function	Description	Input example
+	Addition	=d27+5
-	Subtraction	=12-d4
/	Division	=d42/23
*	Multiplication	=d31*3
Math.sqrt(x)	Square root	=Math.sqrt(9)
Math.pow(x,y)	Power (x to the power of y)	=Math.pow(d2,3)

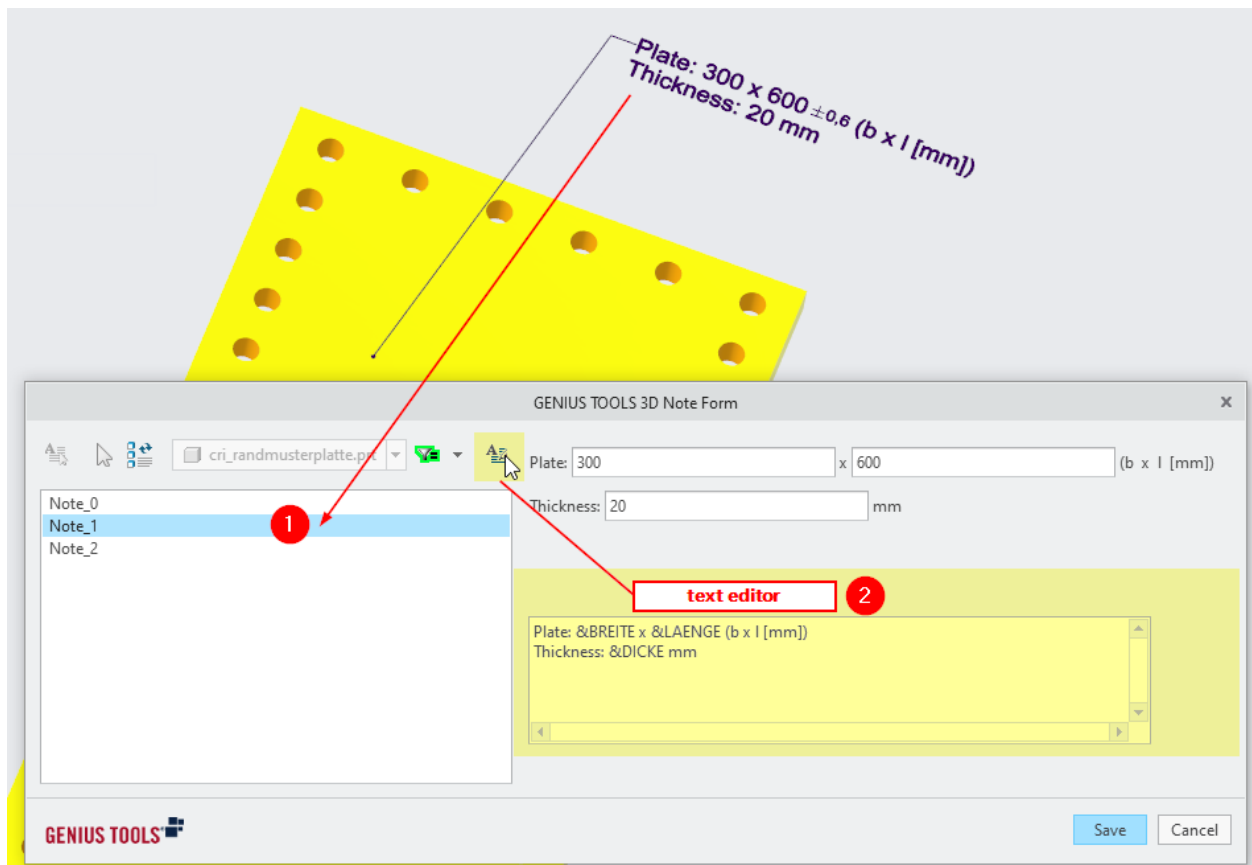
Mathematical function	Description	Input example
Math.abs(x)	Absolute value	=Math.abs(-5)
Math.round(x)	Commercial rounding	=Math.round(2.565)
Math.ceil(x)	Round up on next integer	=Math.ceil(3.6)
Math.floor(x)	Round down on next integer	=Math.floor(3.4)

Please note: Calculation formulas will not be saved.

17.1.3 Usage example

This example illustrates working with notes in the 3D Note Form with static text, dynamic content and calculations. Various notes are stored in this part.

1. Open the *3D Note Form*. Select the note *Note_1*.
2. Go to the text editor.



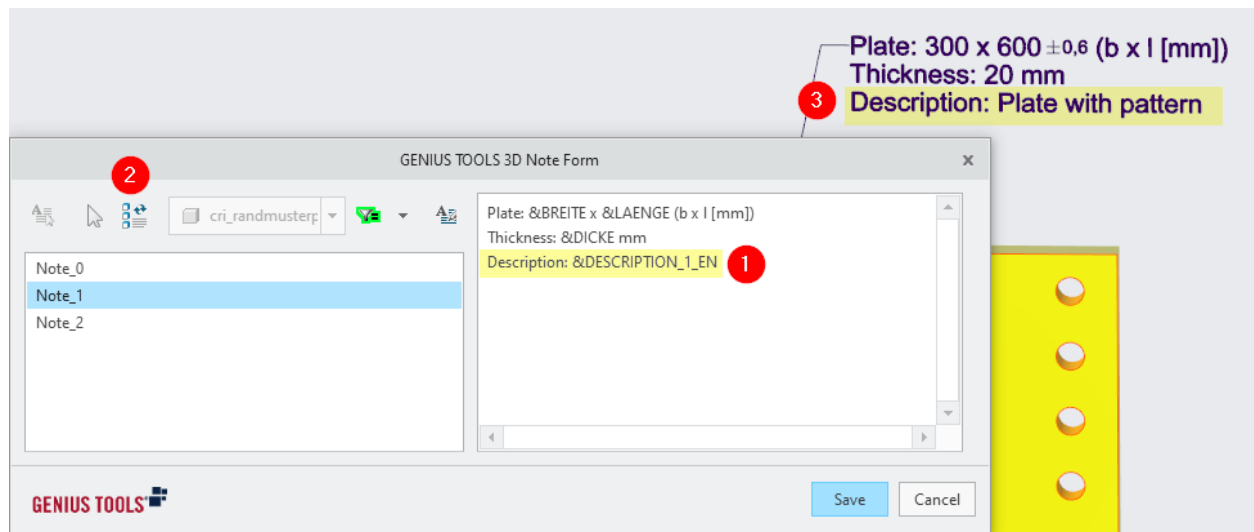
Display of the values of all 3D notes in the note form

Editing a note

In the text editor you will see the breakdown of the displayed note according to this scheme:

Note	Text deposited	Content of the note
Plate:	&WIDTH x &LENGTH (w x l [mm])	300 x 600 (w x l [mm])
Thickness:	&THICKNESS mm	20 mm

1. Add a new line in the text editor:
Description: &DESCRIPTION_1_EN
2. Confirm the entry with *Save and Regenerate model*.
3. The result is applied to the note.




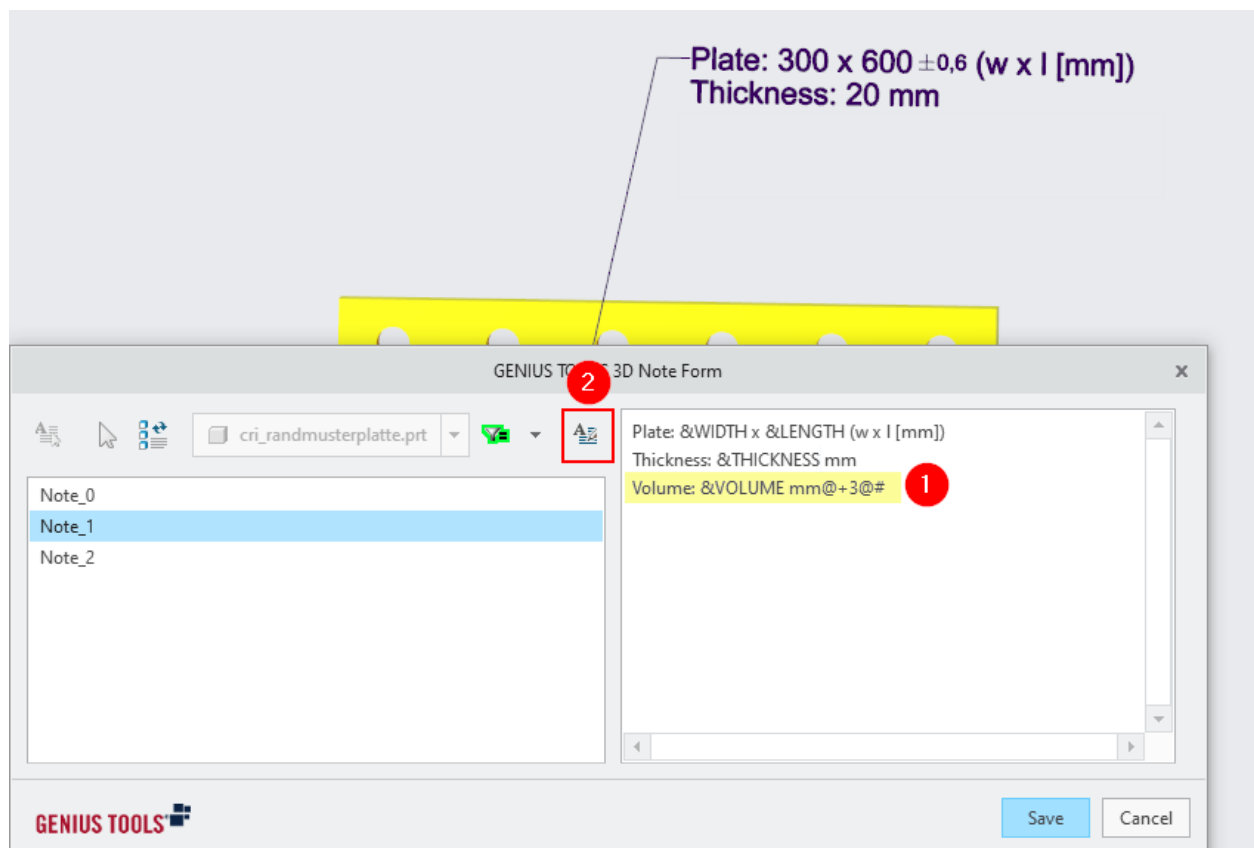
Defining note content via calculation

The `VOLUME` parameter is needed for the next step. Create this parameter if it does not already exist.

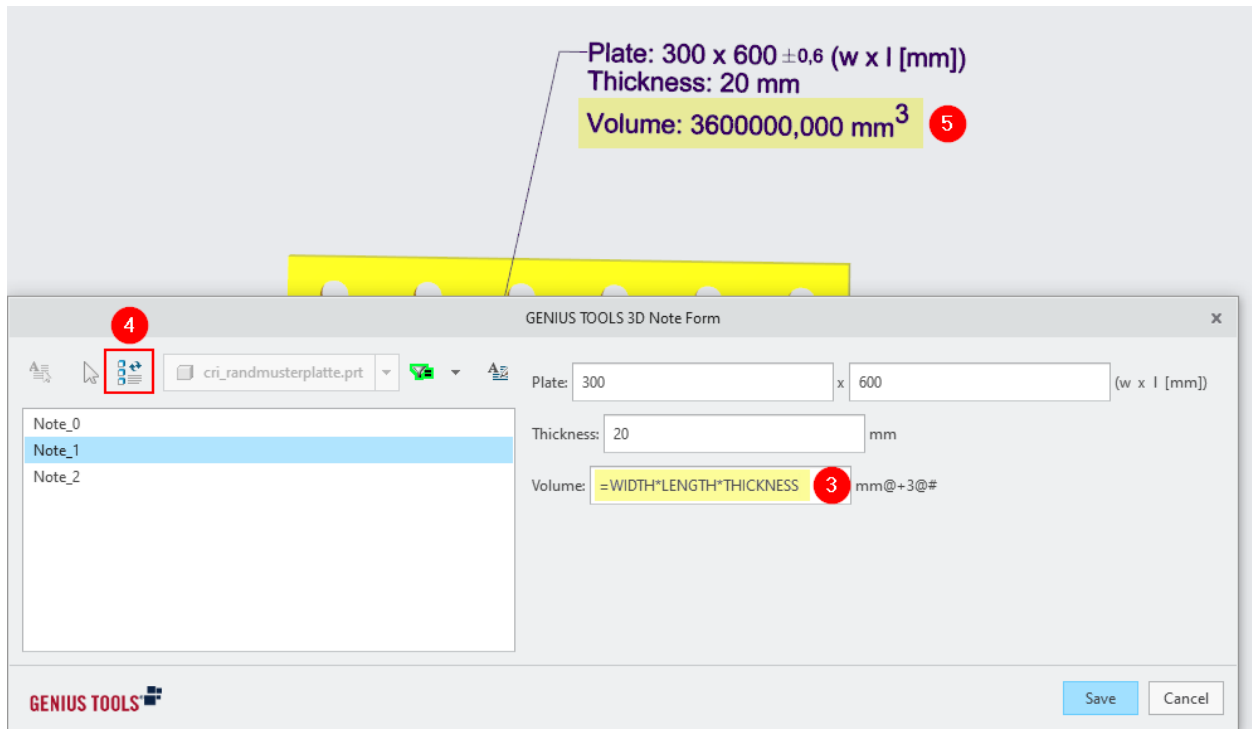
1. Add the following line to the note:

Volume: `&VOLUME mm@+3@#`

2. Exit the text editor by clicking the text editor button  again. Confirm the saving of the changes made with Yes.



3. If you now want to fill this parameter with the 3D note form, enter $=\text{WIDTH}*\text{LENGTH}*\text{THICKNESS}$ in the input field.
4. Confirm the input with Save and Regenerate. The volume is automatically calculated and transferred to the parameter.
5. The result is then displayed in the note.




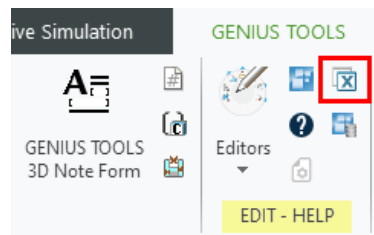
17.2 Close All Other Windows

This function closes all Creo windows except the current and the main window.

Please note: This function is only available with a subscription license for GENIUS TOOLS for Creo.

Starting the function: in all modes

The function *GENIUS TOOLS Close all other windows*  is located in the segment EDIT-HELP in the GENIUS TOOLS ribbon menu and can also be opened via GENIUS TOOLS Quick Access ([<] key).



Start via the ribbon menu



Start via Quick Access

Configuration

You can define the availability of the function with the configuration option `gtu_start_close_all_windows` (Default is 1=Active).

You can define, if the models will be saved before closing the window with the configuration option `gtu_close_all_windows_save_changed_models`. (Default is 0=No automatic saving)

17.3 Command Control

With Command Control you can hide or disable Creo Parametric ribbon menu commands (all commands).

The utility is activated with the configuration option `gtu_start_command_control=1`.

The configuration option `gtu_command_control_configuration` defines the commands to be changed. The value of the option has the following syntax:

```
NAME_1:MODUS_1|NAME_2:MODUS_2|NAME_3:MODUS_3
```

NAME_x is a Creo Parametric Command Name.

MODUS_x can take on one of the three values: UNAVAILABLE, INVISIBLE, DISALLOW.

Example: The following configuration options hide the two commands "Automatic check in" and "Change working directory":

```
gtu_start_command_control=1
```

```
gtu_command_control_configuration=ProCmdMdlTreeWfChkInExp:INVISIBLE|ProCmdSessionChangeDir
```

Mode values

UNAVAILABLE: Command is grey and cannot be selected.

INVISIBLE: Command is hidden.

DISALLOW: Command is visible, but is not executed.

Creo Parametric Command Name

To find the name of a Creo Parametric Command you can record a mapkey with this command and copy the command out of the mapkey text. (See also [Use cases](#)⁴²⁷.)

Example: What is the name of the command "Change working directory"?

Recorded Mapkey:

```
mapkey cd @MAPKEY_NAMEChange directory;@MAPKEY_LABELWorking directory;\
mapkey(continued) ~ Close `main_dlg_cur` `appl_casc`;~ Command `ProCmdSessionChangeDir` ;\
mapkey(continued) ~ Trail `UI Desktop` `UI Desktop` `DLG_PREVIEW_POST` `file_open`;
```

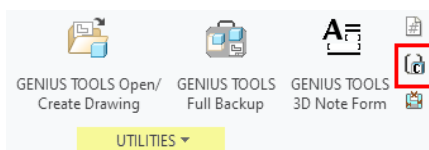
Result: The name for the command is „ProCmdSessionChangeDir“.

17.4 Component Parameter

GENIUS TOOLS Component Parameter creates component parameters in assemblies. A component parameter allows to assign and edit values for models of same name. With component models of same name it is possible to assign different component parameter values.

Starting the program: in assembly mode

Start *GENIUS TOOLS Component parameter* in assembly mode via the ribbon menu in the *GENIUS TOOLS* tab or via *GENIUS TOOLS Quick Access* ([<] key).



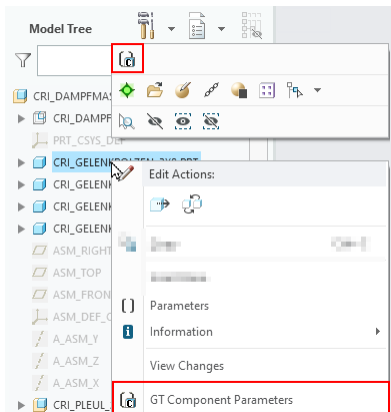
Starting via the ribbon menu



Starting via Quick Access

GENIUS TOOLS Component Parameter starts with an empty selection. Select a subassembly in the model tree to start with a selection.

The area for **editing component parameters**⁴⁸¹ can also be opened from the context menu of a model (right-clicking).

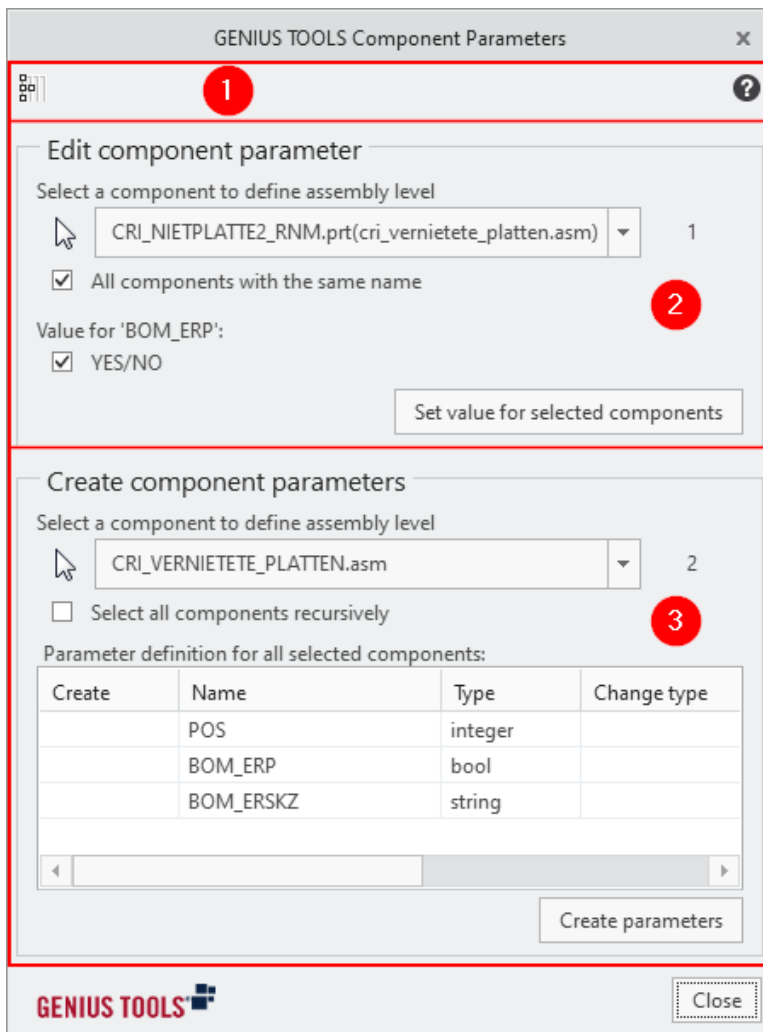


Hiding the button in the ribbon menu

If you wish to hide the *Component Parameter* button from the GENIUS TOOLS ribbon menu, set the configuration option `gtu_start_component_params` to 0. (Default is 1 = On)

17.4.1 User interface

The user interface of GENIUS TOOLS Component Parameter consists of the following elements:

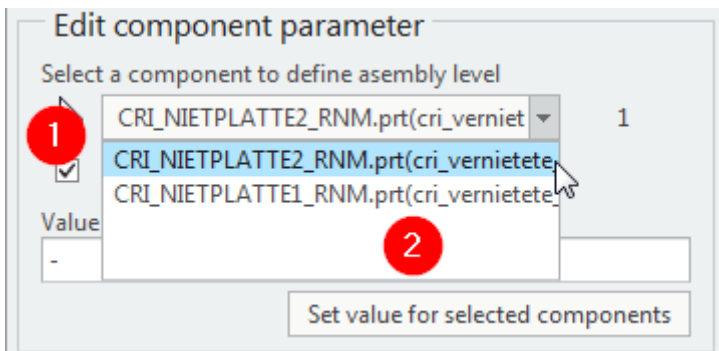


1. Command bar with button to show parameters in model tree and help.
Use the button left to display the model tree. The component parameters are automatically displayed as an additional column in the model tree. This can be configured individually for each parameter.
2. Configuration of the position number parameter⁴⁸¹
This area can be opened by right-clicking on a part in the model tree.
3. Area to create the component parameters⁴⁸³

17.4.2 Editing position number parameters

The parameter values for assembly elements are set in the editing area.

Use the model selection to select a subassembly or an element from it. Models already selected via the object selection are displayed in the list, too.



The model selection with object selection (1) and drop-down list (2).

The number of assembled instances of the selected element is displayed right to the model selection. Click on the number to open a new window displaying the IDs of the assembled elements.

The *All assembly elements with same name* option aggregates repeatedly assembled assemblies or parts. This way it is possible to pass the same position numbers to all elements.

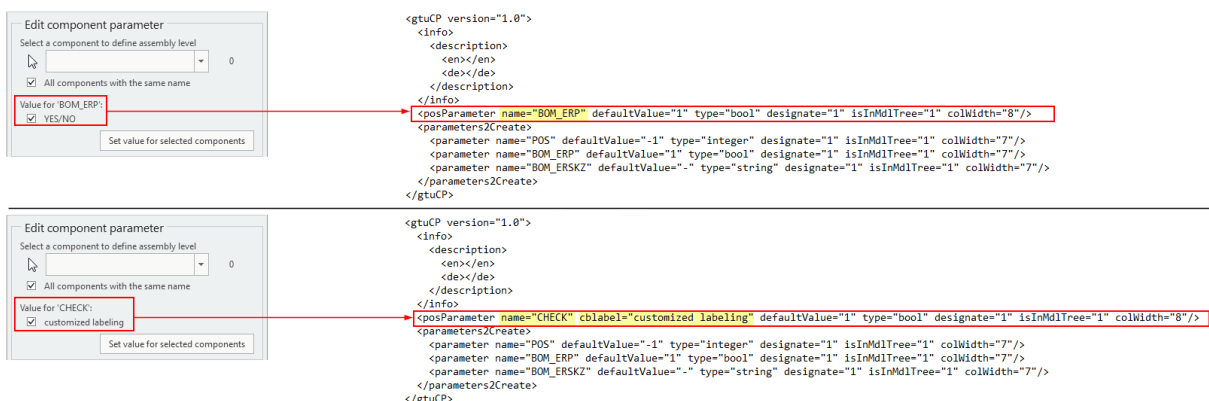
If the option is deactivated and multiple assemblies of an element have been detected, an individual value can be entered into the *Value for POS* table for each element.

Please note: The name (POS) for the component parameter for position numbers may vary due to the configuration on your system.

Boolean parameter

In the editing area you can select a boolean parameter. The display of this option is customizable. You do it with the help of the XML file *gtu_component_parameters.xml*. It is located under the path

<WorkingDirectory>\gtstarter\cadpool\INNEO\parametric\configuration\gt_resource_folder\utilities\component_parameters.



Definition of the label "Value for" in the XML file. Default setting (see above) and customization (see below).

17.4.3 Creating component parameters

The *Create component parameters* area is used to create parameters across an entire assembly or subassemblies.

Please note: You can create any parameters. Prior to this, the parameters have to be created in the configuration.

Select the desired editing layer via the model selection. The number of affected elements is displayed right to the model selection. Click on the number to open a new window displaying the IDs of the assembled elements.

Example

If a part on the layer below the main assembly is selected, the main assembly will be displayed. All elements on the same layer and in the same assembly (in this case, the main assembly) as the selected part will be affected by the edit.

Activate the *Recursive selection of all assembly elements* option to select all elements underneath an assembly. Deactivate the option to select only those on the first layer.

The table displays the component parameters to be created. Select or deselect individual parameters using the *Create* option.

Please note: Parameter properties can only be modified in the configuration of the XML file.

Click on *Create parameters* to create all selected parameters in the assembly elements.

17.4.4 XML configuration

The position number parameter and the component parameters to be created are specified in the *gtu_component_parameters.xml* file in the *component_parameter* subfolder of *gt_resource_folder*:

```
<gtuCP version="1.0">
  <info>
    ...
  </info>
  <posParameter    name="POS"
                   defaultValue="5"
                   type="string"
                   designate="1"
                   isInMdlTree="1"
                   colWidth="8"/>
  <parameters2Create>
    ...
  </parameters2Create>
</gtuCP>
```

The XML file to be used is specified via the *gtu_comp_file_name* configuration option.

Position number parameter

The following attributes are required for the position number parameter:

XML attribute	Description
name	Specifies the name of the position number parameter to be created.
defaultValue	The default value displayed in the text box.
type	Specifies the parameter type. Available are <i>string</i> , <i>integer</i> and <i>double</i> .
designate	Should the parameter be designated when used with Windchill (0: no, 1: yes).
isInMdlTree	Specifies whether the parameter should be displayed in the model tree when clicking the associated button.
colWidth	If <i>isInMdlTree</i> is active, this attribute determines the width of the column for this parameter in the model tree in characters.

Example

```
<posParameter      name="POS"
  defaultValue="5"
  type="string"
  designate="1"
  isInMdlTree="1"
  colWidth="8"/>
```

Warning: Only one position number parameter will be processed.

Component parameter

Also creation and initial filling of the component parameters is controlled via this XML file. Any desired target component parameters with different properties can be specified:

XML attribute	Description
name	Specifies the name of the parameter to be created.
defaultValue	The default value entered when creating the parameter.

XML attribute	Description
type	Specifies the parameter type. Available are <i>string</i> , <i>integer</i> , <i>double</i> and <i>bool</i> .
designate	Should the parameter be designated when used with Windchill (0: no, 1: yes).
isInMdlTree	Specifies whether the parameter should be displayed in the model tree when clicking the associated button.
colWidth	If <i>isInMdlTree</i> is active, this attribute determines the width of the column for this parameter in the model tree in characters.
isType2Change	If <i>isType2Change</i> is "1", the selection box in the interface for the type change is active. If the parameter already exists in the component and the parameter type is different from the default type, the parameter is deleted and a new parameter with this name, the correct type, the original value and the original parameter description is created again.

Changing the type of component parameters is necessary, for example, if parameters with an incorrect type were added via Assembly Report. The value of the parameter and the description of the parameter are retained.

Example


```
<parameters2Create>
...
<parameter
  name="CompParam"
  defaultValue="-"
  type="string"
  designate="0"
  isInMdlTree="1"
  colWidth="4"
  isType2Change="1"/>
...
</parameters2Create>
```

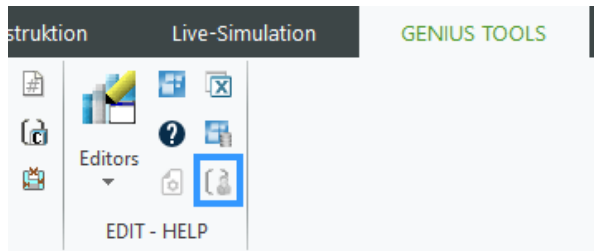
17.5 Copy Component Parameter To Substitution Component Parameter

This function copies all component parameters from the master representation to the substitution component representation in simplified representations.

Please note: This function is only available with a subscription license for GENIUS TOOLS for Creo.

Starting the program: in assembly mode (simplified representation)

The button  can be found in the GENIUS TOOLS ribbon menu on the segment EDIT-HELP, if an assembly is opened as simplified representation, which was created by substitution.

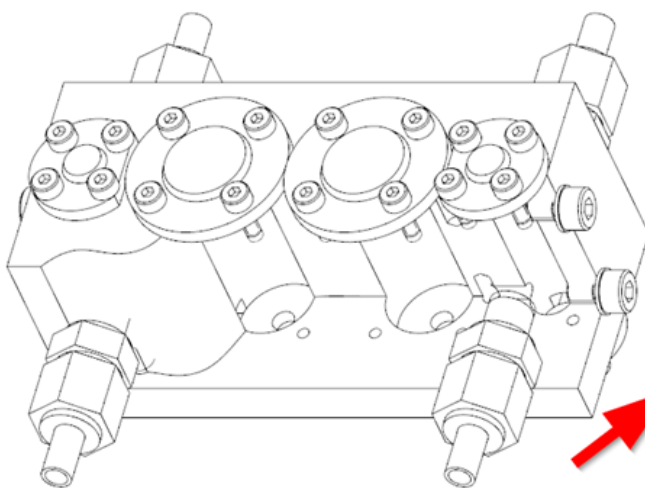


Start via the ribbon menu

Procedure

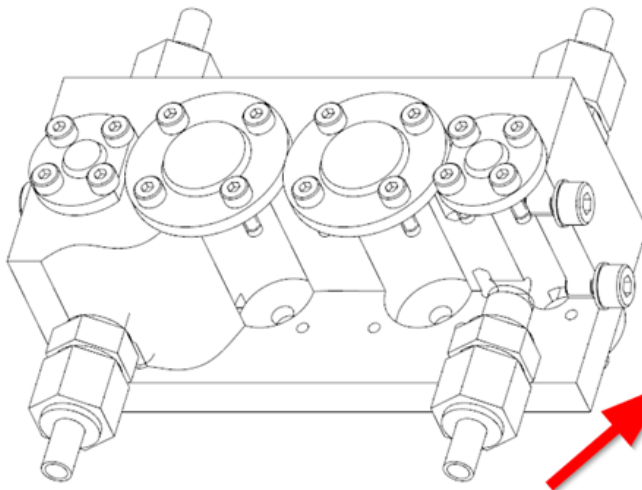
The substituted components will be enhanced by the component parameters from the component from the master representation at function runtime. If the parameters are not in existence yet, they will be created.

If you use the simplified representation inside a drawing for a report (Pro/Report), all component parameters are filled after executing the function. You can see the behavior described in the following images.



Artikel: 002021268		Baukastenstückliste		
Pos.	Anz.	Artikelnummer Halbzeugartikel-Nr.	Rev.	Benennung Modelldateiname
10	4	002021274	C	Anschlussverschraubung CRI_UEBERWURF_U_NIPPEL.ASM
35	2	002021222	C	ob.Schieberdeckel CRI_DECKEL_SCHIEB_2000.PRT
40	2	002021226	C	ob.Zylinderdeckel CRI_DECKEL_ZYL_2000.PRT
15	4	S002021239	B	Scheibe CRI_D433T103_2.PRT
45	2	002021262	C	unf.Schieberdeckel CRI_DECKEL_SCHIEB_U_2000.PRT
50	2	002021217	C	unf.Zylinderdeckel CRI_DECKEL_ZYL_U_2000.PRT
	1	0002021253	C	Zylinderblock 2000 CRI_ZYLINDERBLOCK_2000.PRT
25	32	S002021250	B	Zylinderschraube CRI_D912M2L6.PRT
30	4	S002021265	B	Zylinderschraube CRI_D912M3L5.PRT

Before using the function Copy Component Parameter To Substitution Component Parameter



Artikel: 002021268				Baukastenstückliste	
Pos.	Anz.	Artikelnummer Halbzeugartikel-Nr.	Rev.	Benennung Modelldateiname	
10	4	002021274	C	Anschlussverschraubung CRI_UEBERWURF_U_NIPPEL.ASM	
35	2	002021222	C	ob.Schieberdeckel CRI_DECKEL_SCHIEB_2000.PRT	
40	2	002021226	C	ob.Zylinderdeckel CRI_DECKEL_ZYL_2000.PRT	
15	4	S002021239	B	Scheibe CRI_D433T1D3_2.PRT	
45	2	002021262	C	unf.Schieberdeckel CRI_DECKEL_SCHIEB_U_2000.PRT	
50	2	002021217	C	unf.Zylinderdeckel CRI_DECKEL_ZYL_U_2000.PRT	
20	1	0002021253	C	Zylinderblock 2000 CRI_ZYLINDERBLOCK_2000.PRT	
25	32	S002021250	B	Zylinderschraube CRI_D912M2L6.PRT	
30	4	S002021265	B	Zylinderschraube CRI_D912M3L5.PRT	

After

Hiding the button in the ribbon menu

If you wish to hide the  button from the GENIUS TOOLS ribbon menu, set the configuration option `gtu_start_copyCParamToSubsCParam` to 0. (Default is 1 = On)

17.6 Create Search.pro

When opening an assembly (ASM file) you can only access parts that are saved in other directories, if these directories are written into a separate search path file.

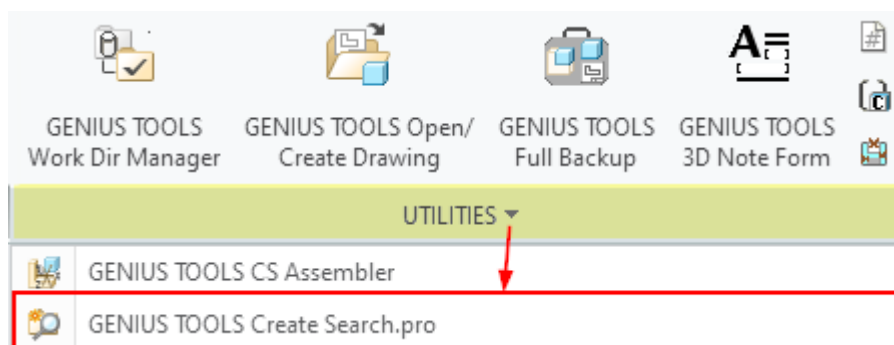
The GENIUS TOOLS function *Create Search.pro* creates and configures a project or assembly-specific *search.pro* file.

Search.pro

Search.pro is a file which includes the search paths for all parts of an assembly. It is especially useful in settings where a PDM system is not available.

Starting the program: in assembly mode

Start *Create Search.pro* via the ribbon menu in the GENIUS TOOLS tab.



A Save dialog opens after starting. After selecting a storage location the file will be saved and the *Create Search.pro* function is completed.

The content of the search path file depends on the configuration.

Hiding the button from the menu

If you wish to hide the *Create Search.pro* button from the GENIUS TOOLS ribbon menu, set the configuration option `gtu_start_create_search_pro` to 0. (Default is 1 = On)

Configuring search.pro file

gtu_create_search_pro_standard_save_name

Defines the suggested file name of the search path file in the Save dialog. Default: *search.pro*

gtu_create_search_pro_exclude_current_path

Defines whether the current working directory is included in the search.pro file (0) or not (1). Default: 1

gtu_create_search_pro_exclude_file

Defines a file containing search paths which are not to be included in the new search.pro file.

Tip: All paths that point to the standard library should be specified here. We recommend entering the file that is defined in the configuration option `search_path_file`.

gtu_create_search_pro_line_start

Defines characters which are inserted at the beginning of each line.

Tip: You should enter the config.pro format `search_path` here. Thus, the search.pro file can be loaded to the configuration settings of Creo.

gtu_create_search_pro_standard_save_folder

Defines the standard storage location.

0 – Creo standard directory (opens the File dialog)

1 - directory of the current object

2 – current working directory

Creating a mapkey for loading and reloading a search path file (search.pro file)

In order to guarantee a correct presentation of an assembly, the corresponding search path file should be loaded into the configuration settings of Creo before opening the


assembly file. You can automate this process by creating a mapkey which saves the sequences of commands.

This mapkey can be used as a key macro and can also be added to the Quick Access ring menu.

The following process describes how to load the search.pro file from the working directory. You can also directly copy and paste the mapkey from the section [Content of the mapkey file](#)⁴⁹⁰ below.

Procedure

For Creo to be able to read the search paths, the Creo configuration option `search_path` must be placed at the beginning of each line of the Search.pro file:

1. You are in standby mode of Creo.
2. Open the Configuration Utility editor from the GENIUS TOOLS ribbon.
3. Go to the configuration option `gtu_create_search_pro_line_start` and enter:
`search_path`
4. Close the dialog and save changes in the main window with  before closing Configuration Utility.
5. In the GENIUS TOOLS ribbon, click the *Reread Configuration* button.

Create a new search path file for the assembly in question or overwrite the old one:

6. In the GENIUS TOOLS ribbon menu, go to *Utilities* and click on *Create Search.pro* (See screenshot above).
7. In the following Save dialog, save the file in the working directory.

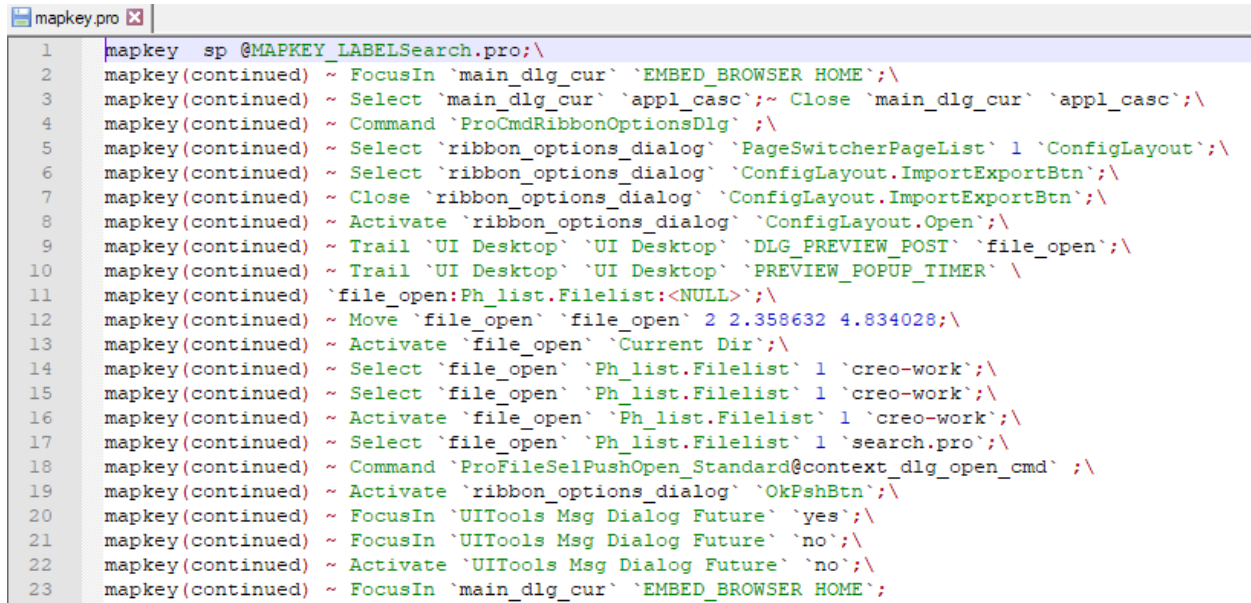
Record the mapkey as in step 8 to 16 or copy and paste the content from the section below:

8. In the Creo menu, go to *File > Options > Environment > Mapkey Settings*
9. In the *Mapkeys* dialog box, click *New*.
10. In the *Record Mapkeys* dialog box, enter a key combination and a name, e. g. "sp" and „Load Search.pro “
11. Click on *Record*.
12. In the Creo menu, go to *File > Options > Configuration Editor*.
13. Under Options, click *Import/Export > Import configuration File*.
14. Select the working directory and open the Search.pro file.
15. Click OK and in the following *Creo Parametric Options* dialog box, click *No*. (The settings should not be saved to a configuration file).
16. In the *Record Mapkey* dialog box, click the *Stop* button and click OK.

Save the mapkey as a file in the working directory:

17. In the *Mapkeys* dialog box, click *Save changed*.

18. In the following *Save* dialog, enter a file name, e. g. mapkey, and click OK.



```

1 mapkey sp @MAPKEY_LABELSearch.pro;\
2 mapkey(continued) ~ FocusIn `main_dlg_cur` `EMBED_BROWSER HOME`; \
3 mapkey(continued) ~ Select `main_dlg_cur` `appl_casc`; ~ Close `main_dlg_cur` `appl_casc`; \
4 mapkey(continued) ~ Command `ProCmdRibbonOptionsDlg` ; \
5 mapkey(continued) ~ Select `ribbon_options_dialog` `PageSwitcherPageList` 1 `ConfigLayout`; \
6 mapkey(continued) ~ Select `ribbon_options_dialog` `ConfigLayout.ImportExportBtn`; \
7 mapkey(continued) ~ Close `ribbon_options_dialog` `ConfigLayout.ImportExportBtn`; \
8 mapkey(continued) ~ Activate `ribbon_options_dialog` `ConfigLayout.Open`; \
9 mapkey(continued) ~ Trail `UI Desktop` `UI Desktop` `DLG_PREVIEW_POST` `file_open`; \
10 mapkey(continued) ~ Trail `UI Desktop` `UI Desktop` `PREVIEW_POPUP_TIMER` \
11 mapkey(continued) `file_open:Ph_list.Filelist:<NULL>`; \
12 mapkey(continued) ~ Move `file_open` `file_open` 2 2.358632 4.834028; \
13 mapkey(continued) ~ Activate `file_open` `Current Dir`; \
14 mapkey(continued) ~ Select `file_open` `Ph_list.Filelist` 1 `creo-work`; \
15 mapkey(continued) ~ Select `file_open` `Ph_list.Filelist` 1 `creo-work`; \
16 mapkey(continued) ~ Activate `file_open` `Ph_list.Filelist` 1 `creo-work`; \
17 mapkey(continued) ~ Select `file_open` `Ph_list.Filelist` 1 `search.pro`; \
18 mapkey(continued) ~ Command `ProFileSelPushOpen_Standard@context_dlg_open_cmd` ; \
19 mapkey(continued) ~ Activate `ribbon_options_dialog` `OkPshBtn`; \
20 mapkey(continued) ~ FocusIn `UITools Msg Dialog Future` `yes`; \
21 mapkey(continued) ~ FocusIn `UITools Msg Dialog Future` `no`; \
22 mapkey(continued) ~ Activate `UITools Msg Dialog Future` `no`; \
23 mapkey(continued) ~ FocusIn `main_dlg_cur` `EMBED_BROWSER HOME`;

```

Content of the mapkey file

19. Copy the contents of the file into a Config.pro file that is loaded when Creo is started.

Result

The key combination (here: "sp") is available as mapkey in every Creo session.

The mapkey can also be included in GENIUS TOOLS Quick Access. This is described in the chapter [Inserting existing mapkeys](#).⁴¹⁹

Content of the mapkey file for the key combination "sp"

```

mapkey sp @MAPKEY_LABELLoad search.pro;~ Select `main_dlg_cur` `appl_casc`; \
mapkey(continued) ~ Close `main_dlg_cur` `appl_casc`;~ Command
`ProCmdRibbonOptionsDlg` ; \
mapkey(continued) ~ Select `ribbon_options_dialog` `PageSwitcherPageList` 1
`ConfigLayout`; \
mapkey(continued) ~ Select `ribbon_options_dialog` `ConfigLayout.ImportExportBtn`; \
mapkey(continued) ~ Close `ribbon_options_dialog` `ConfigLayout.ImportExportBtn`; \
mapkey(continued) ~ Activate `ribbon_options_dialog` `ConfigLayout.Open`; \
mapkey(continued) ~ Trail `UI Desktop` `UI Desktop` `DLG_PREVIEW_POST` `file_open`; \
mapkey(continued) ~ Activate `file_open` `Current Dir`; \
mapkey(continued) ~ Trail `UI Desktop` `UI Desktop` `PREVIEW_POPUP_TIMER` \
mapkey(continued) `file_open:Ph_list.Filelist:<NULL>`; \
mapkey(continued) ~ Select `file_open` `Ph_list.Filelist` 1 `search.pro`; \
mapkey(continued) ~ Command `ProFileSelPushOpen_Standard@context_dlg_open_cmd`
; \

```

```
mapkey(continued) ~ Activate `ribbon_options_dialog` `OkPshBtn`\
mapkey(continued) ~ FocusIn `UITools Msg Dialog Future` `no`\
mapkey(continued) ~ Activate `UITools Msg Dialog Future` `no`;
```

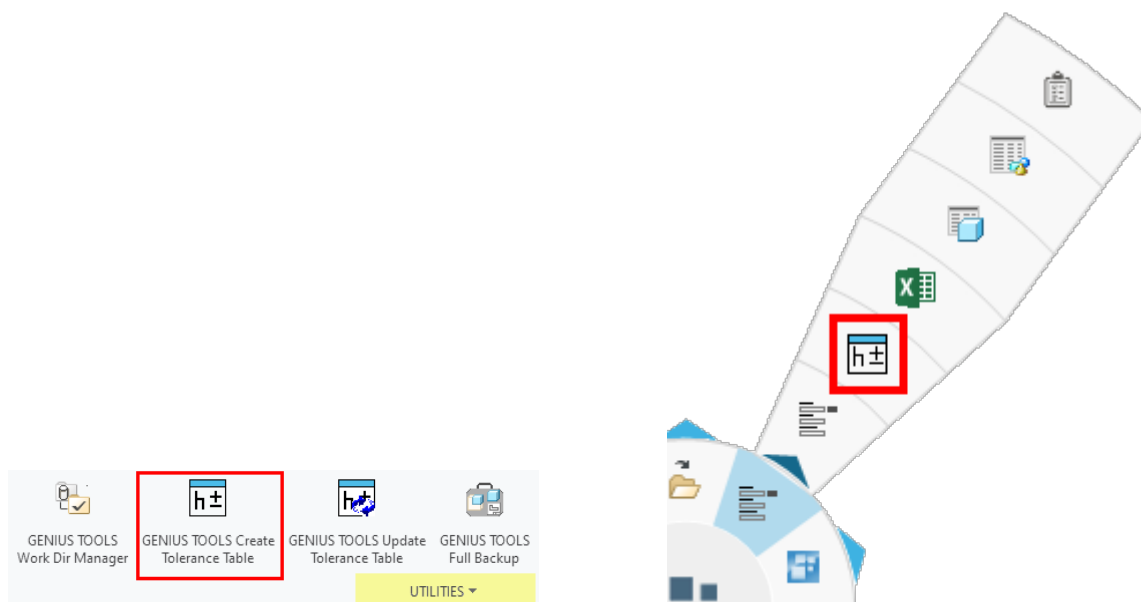
17.7 Create Tolerance Table

This function creates tables of dimensions in Creo Parametric drawing mode. The set tolerance dimensions are used.

Please note: Only fitting tolerances already displayed will be output.

Starting the program: in drawing mode

Create tolerance table is started in drawing mode via the ribbon menu or via Quick Access ([<] key).



Starting via the ribbon menu

Call-up via Quick Access

The function creates a tolerance table at a freely selectable position on a drawing. As a default the tolerance table will always be created for all views displayed on the drawing. It can be selected via the configuration whether views of the current sheet only or if all views should be considered.

The following table formats are supported:


Tolerance	Fitsize
Ø2.20 H6	0.006
	0.000
Ø2.20 H7	0.010
	0.000
Ø12.00 H7	0.018
	0.000

Form A: Dimensions

Tolerance	Minimum	Maximum
Ø2.20 H6	2.200	2.206
Ø2.20 H7	2.200	2.210
Ø12.00 H7	12.000	12.018

Form B: Minimum/maximum dimension

Update tolerance table

If a tolerance is changed or a new one is added, the tolerance table has to be recreated by pressing the button  in the ribbon menu.

Configuration of the display

You can hide the function in the ribbon menu with `gtu_start_tolerance_table`. (Default is 1=Shown)

The configuration option `gtu_tol_table_creo_insert` defines, whether the table is inserted without preview (0) or with preview (1).

Configuration of the table

gtu_tol_table_decimal_marker_follow_dtl

Changes the representation of numerical values from period to comma as decimal separator, if the DTL option `decimal_marker = COMMA` is set. Define, whether the DTL file should be read (1) or not (0). (Default is 1.)

gtu_tol_table_sort_order

Defines, whether the table is sorted in ascending (ASC) or descending (DESC) order.

gtu_tol_table_origin_at_bottom_right

Defines the origin and the growth direction of the table.

0 - The origin of the table is at the top left and it is built up to the bottom right.

1 - The origin of the table is at the bottom right and it is built up to the top left.

Please note: If the table is placed at the bottom of the drawing, reversing the growth direction is helpful. If the table is placed at the top edge, it makes sense to turn the growth direction downwards.

The selected growth direction of the table cannot be displayed in the preview (`gtu_tol_table_creo_insert=1`).

Configuration for use with ModelCheck

When using with ModelCheck, it may be necessary to change the font for the first line. Set the following option:

gtu_tol_table_hidden_font

Defines the font for the hidden table header. If no font is defined, the header will receive the drawing font. Default: Isofont

For more configuration options to define the table view refer to [configuration settings](#)⁶⁷³.

17.8 CS Assembler

CS Assembler is a tool for automatically adding a number of components to an assembly. CS stands for coordinate system.

Please note: CS Assembler is only available with subscription licenses for GENIUS TOOLS for Creo.

The target assembly and the component models have to contain specific coordinate systems. In the target assembly, the relevant coordinate systems are specified by component parameters. In the component models, the relevant coordinate systems are specified by a common name, which is defined in a configuration option.

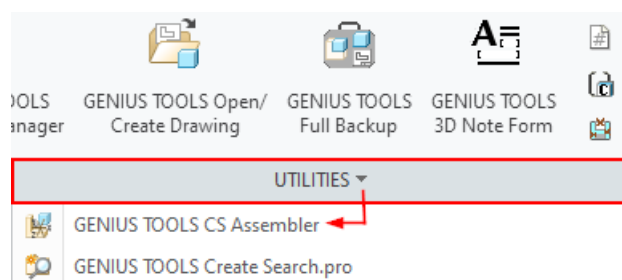
The assembling process works by making the corresponding coordinate systems coincident to each other.

The assembling instructions are defined in an XML file which assigns a target coordinate system in the assembly to each component model. For each component model, the file defines which target coordinate system or which target coordinate systems of the same name should be used.

Starting the program: in part mode

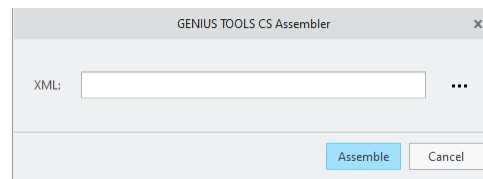
Open the assembly that you want to add components to.

Start *CS Assembler*  in GENIUS TOOLS ribbon menu in the *Utilities* tab.



Procedure

Select the XML file that contains the assembling instructions. The models specified in the XML file will be added to the assembly, if all requirements are fulfilled.



Requirements

1. In order to be able to use CS Assembler, the configuration option `gtu_start_csassembler` has to be active, that is, set to 1.
2. Each model that is to become a component of the target assembly has to have an assembling coordinate system, which has to have the name specified in the configuration option `gtu_csassembler_component_cs_name`.
3. The models to be assembled have to be located in the Creo search path.
4. The assembly that is to receive the components has to have a target coordinate system for each model to be assembled.
 - The target coordinate system can be identified by the component parameters *CLASSIFICATION* and *SUBTYPE*. These parameters are used to assign a target coordinate system to each component model in the XML assembling instruction. If there are multiple target coordinate systems in the assembly that fit the assembling instructions, the component model is added multiple times.
 - Alternatively, the name of the target coordinate system can be entered in the attribute *ics_name*.

This provides for several possibilities to find a coordinate system. You can either define *CLASSIFICATION* or define *ics_name* or use one of the following combination of attributes:

`classification` and `subtype`

`classification` and `ics_name`

You cannot combine `classification`, `subtype` and `ics_name`.

XML assembling instructions

The assembling instructions have to be available in a specified XML format and in UTF-8 encoding. This section describes the required XML format.

The root element `root` does not carry any additional information.

Each model to be assembled is defined by an element `assemble` with the following attributes.:

- `file`: file name of the model, with extension
- `classification`: value of the *CLASSIFICATION* parameter on the target coordinate system
- `subtype` (optional): value of the *SUBTYPE* parameter on the target coordinate system. You can use different values for the *SUBTYPE* parameter in order to use multiple target coordinate systems with the same *CLASSIFICATION*.
- `ics_name`: name of feature of the target coordinate system

`assemble` is an empty element.

Example

```
<?xml version="1.0" encoding="utf8"?>
<root>
  <assemble file="PRT001.prt" classification="001" subtype=""/>
  <assemble file="PRT002.prt" classification="002" subtype=""/>
</root>
```

Configuring coordinate systems: multi-level assembly and maximum assembly

gtu_csassembler_multi_level

Defines whether coordinate systems from newly assembled components should be used as target coordinate system inside the same assemble task (1) or not (0). Default: 1

gtu_csassembler_maximal_cs_count

Defines the maximal number of coordinate systems that can be assembled into a model. Default: 5000


Hiding the button from the menu

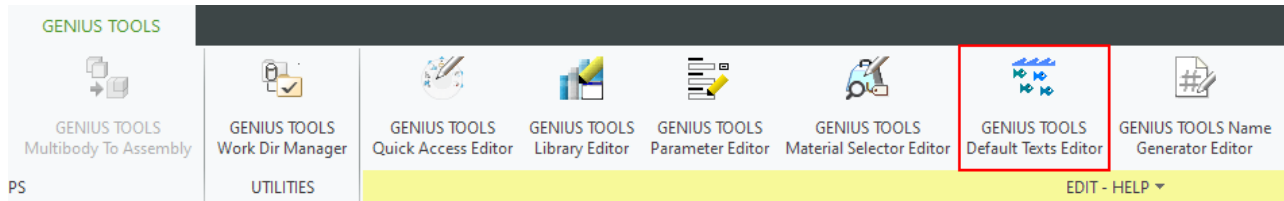
You can remove the function from the GENIUS TOOLS ribbon menu by setting the configuration option `gtu_start_create_search_pro` to 0. (Default is 1 = On)

17.9 Default Text Editor

Create a database of words which can be used in multiple GENIUS TOOLS components to fill dialog boxes with multilingual descriptions partly automated.

Starting the program: in Creo standby mode

Start *Default Text Editor*  in the GENIUS TOOLS ribbon menu.



GENIUS TOOLS Default Texts Editor starts with the database and table defined by the configuration.

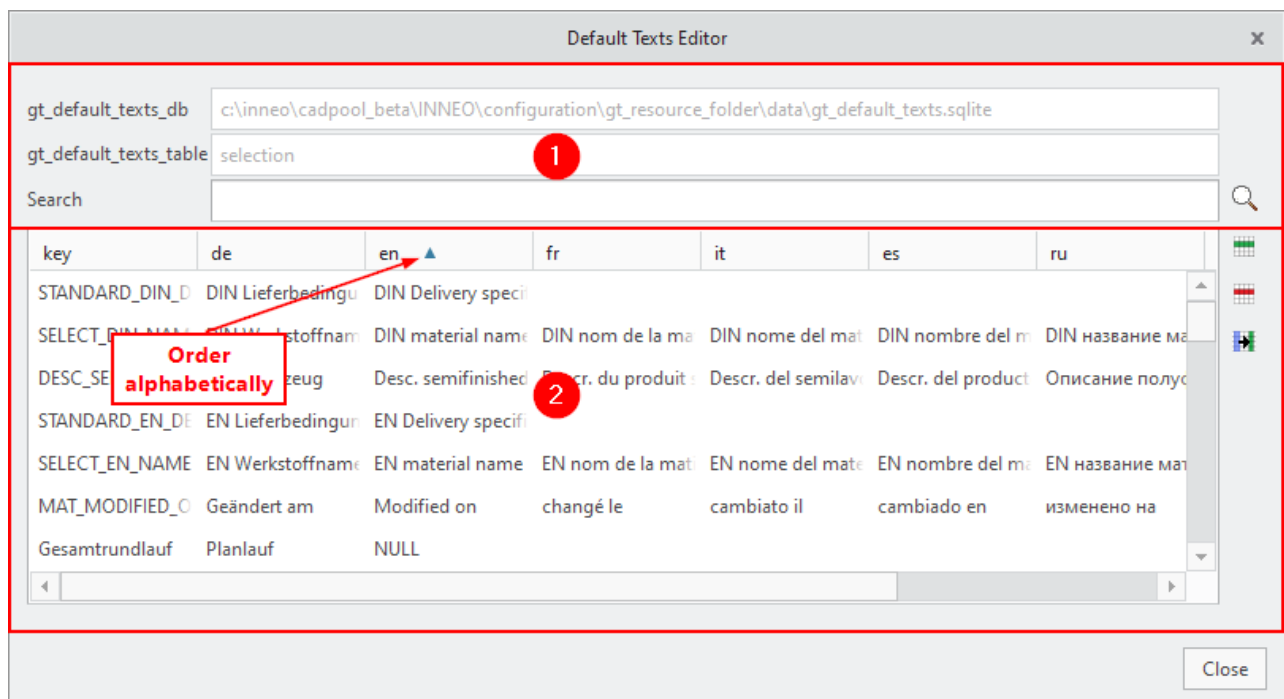
Configuration


Define the path to the database with the configuration option `gt_default_texts_db` and the table to be used from that database with `gt_default_texts_table`.




You can switch off the component so that no button appears in the ribbon menu with `gtu_start_edit_default_texts`.

17.9.1 Create default text

The user interface of GENIUS TOOLS Default Texts Editor is divided into two areas.




1. Configuration options and filters
 - use the search function  to filter down the list of editable elements
2. Tabular editor

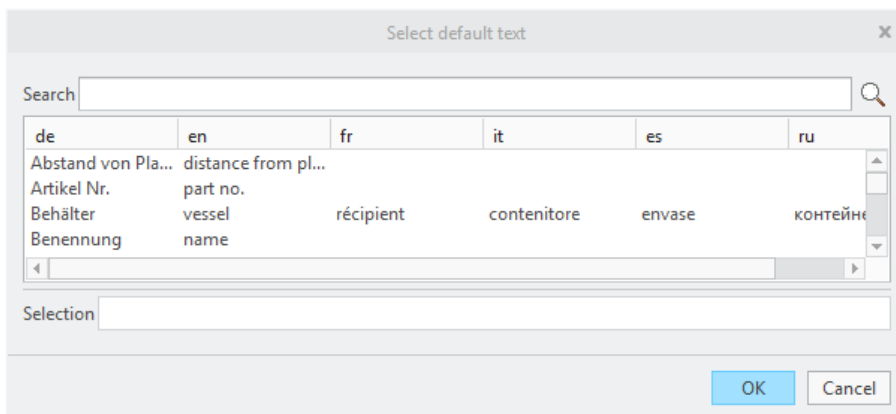
- edit words and phrases directly inside the table
- add elements/rows with the  button
- remove elements/rows with the  button
- copy values from one column to another with the  button

To add or delete languages (columns) the database must be changed accordingly.

Tip: You can change and add words while you [select standard text](#)⁴⁹⁷ in other GENIUS TOOLS components. Already set values will not be affected by these changes.


17.9.2 Select default text

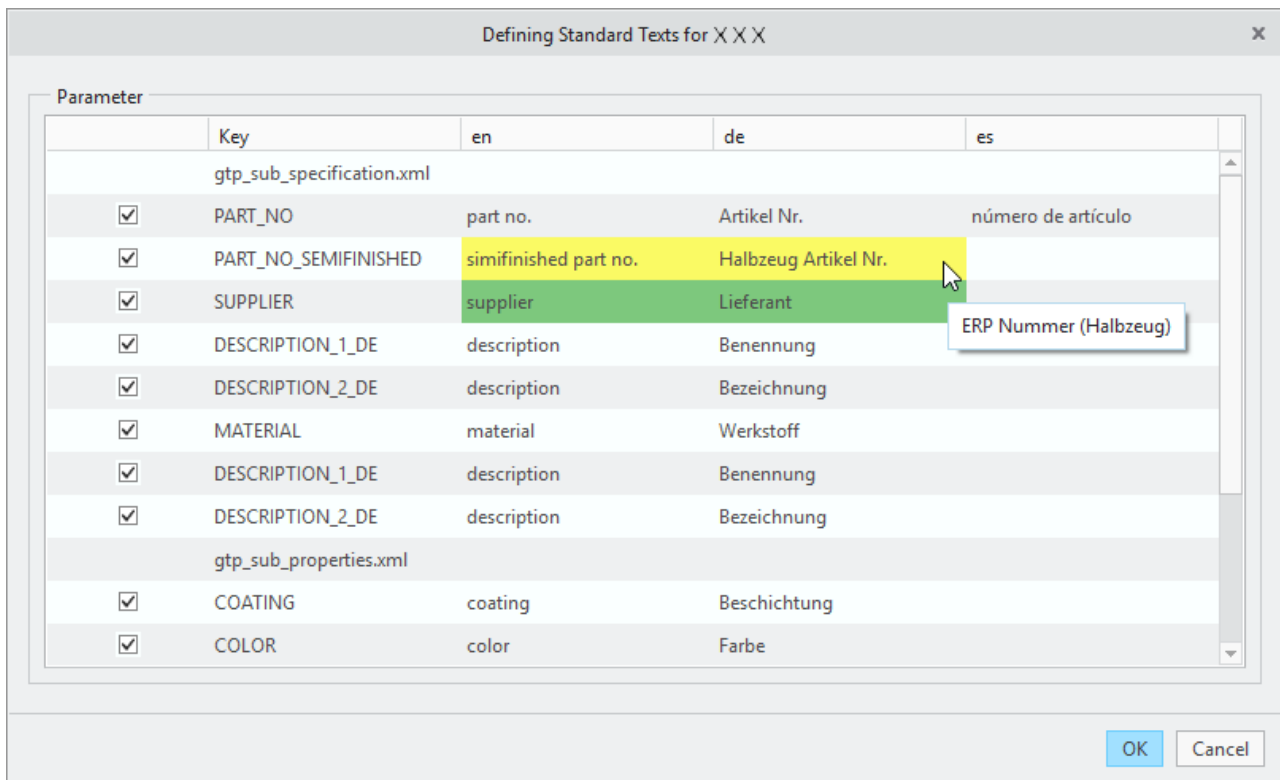
In the editors of many GENIUS TOOLS components it is possible to add multilingual text for a single element, e. g. parameter in GENIUS TOOLS Parameter or measurement in GENIUS TOOLS Forms. This function is opened with the fish button  which opens the dialog *Select default text*.



Select a standard text and click OK. The selected text is transferred to the language dependent fields. You can use the search field to filter the text list.

Select default text for multiple elements

In addition, standard texts for several elements can also be stored in some editors. The automatic setting of standard texts is called up via the button . The program collects all suitable elements and transfers them to the user interface for automatic setting of standard texts. An attempt is made to find a suitable translation on the basis of a keyword in the database. The result is displayed in a table. The table lists only those elements for which the program has found a suitable key.



New values cannot be entered here.

The values in the column *Key* depend on the module. For example, GENIUS TOOLS Parameter contains the names of the parameters and separators.

The language columns depend on the displayed columns in the calling editor.

The background colors of the individual cells have the following meaning:


- white: The translation of the element in the editor matches that in the database.
- green: The element has no translation in the editor yet.
- yellow: The translation of the element in the editor is different from the translation in the database. The translation of the editor appears as a tooltip.

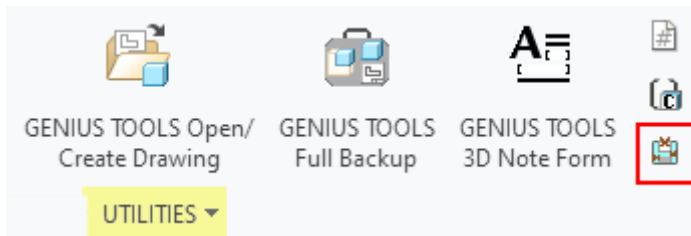
If the dialog is confirmed with OK, all checked lines are transferred to the editor.

17.10 Export Points

Thus functions allows you to export single points or an array of points as well as dynamically created points on curves (X-Y-Z-values) to an ASCII file, which can either have the format of a Creo point file (PTS file) or a user defined template.

Starting the program: in part and assembly mode

The button  for the *Points* function is in GENIUS TOOLS ribbon menu in the section UTILITIES and is available in part and assembly mode.

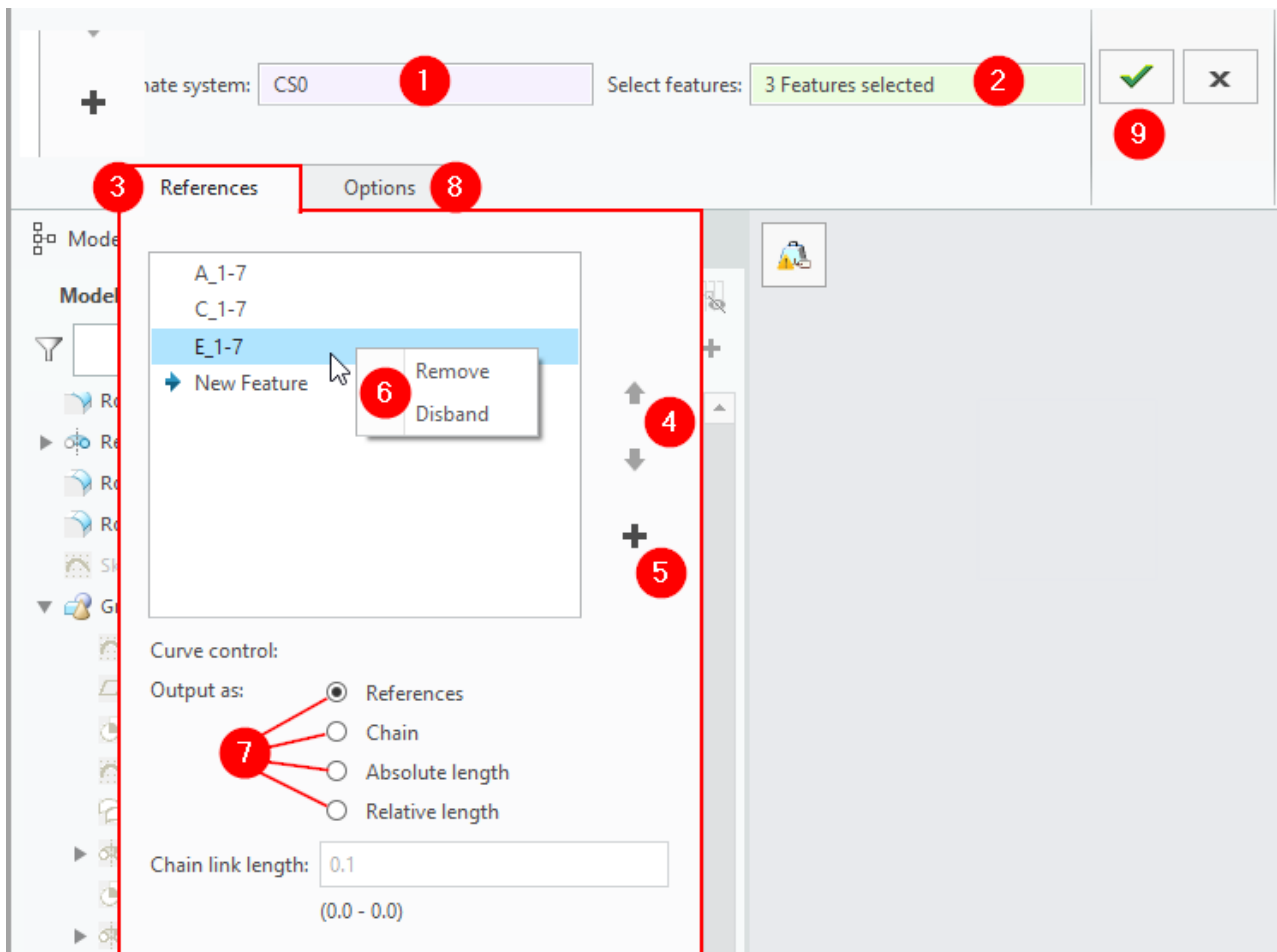


Configuring display of button

Use the configuration option `gtu_start_points` to hide the button in the GENIUS TOOLS ribbon menu. (Default is 1=On/not hidden)

17.10.1 Exporting points

With GENIUS TOOLS Points, you can export single points, curves and point groups as PTS files with different settings.



Dialog GENIUS TOOLS Points with opened References tab

Steps

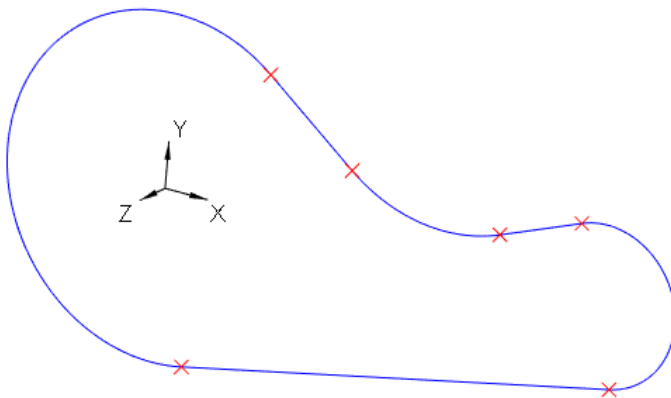
1. Click in the field Select coordinate system (1).
2. Select a coordinate system that serves as the base for the coordinates of the points. The name of the coordinate system is displayed – in above example: CS0.
3. Click in the field *Select features* (2).
4. Select Creo features such as points, curves or groups of points. You can also select single-level patterns. Multilevel patterns are not supported. (In above example: Features A_1-7, C_1-7 and E_1-7.)
 - The quantity of selected features is displayed in the main window in the *Select Features* field (2)
 - All names of selected features are displayed in the tab *References* (3).
3. In the *References* tab (3):
 - Check the order of features and adjust if necessary by using the arrows on the right (4).

- Click *Add points by name* (5) to select points by their name if the exported points conform to certain naming conventions.
 - You can delete features or disband groups of points by using the context menu by right-clicking on a feature (6).
4. Define the output options of curves (7). (For more information see [next chapter](#)⁵⁰¹.)
 5. In the tab *options* (8) choose an export format. (For more information see [Export options](#)⁵⁰².)
 6. Confirm the dialog by clicking the green check mark button (9).

17.10.2 Settings for points on curves

In the segment *Curve control* (7) in the *References* tab you can select the method of how to define points on a curve. There are four possibilities.

References: Start and end points of each segment of the curve are outputted.



Chain: The value of "segment length" is the distance between two points, not the segment length of the curve.



Absolute length: The value of "segment length" is the segment length of the curve that defines the coordinates of the points.

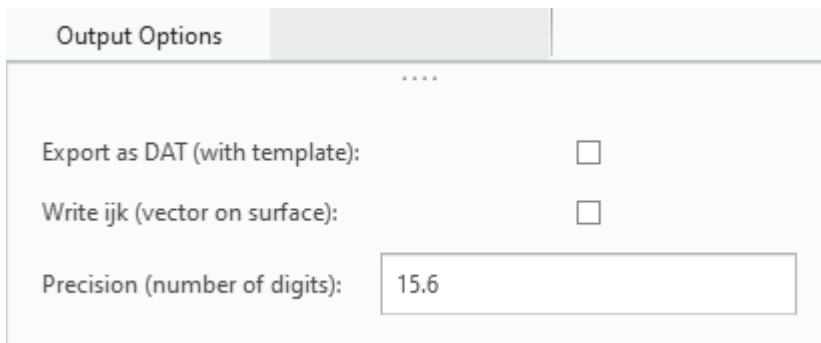
Relative length: The value of "segment length" is the relative segment length of the curve that defines the coordinates of the points.

Configure preselection

You can change the preselection of the curve control with the configuration option `gtu_points_curve_output_type`: 0 = References, 1 = Chain, 2 = Absolute length, 3 = Relative length (Default: 0)

17.10.3 Export options

Settings in the options tab define the output format of the points values file.



Options tab in GENIUS TOOLS Points

Export as DAT (with template): exports data of points values into a template.

Write ijk (vector on surface): The ijk-values describe the perpendicular vector (normal) if a point through a surface. The surface that is next to the point would be used automatically.

Precision (number of digits): defines the total number of digits and the digits after the decimal point of each point value.

- Input format ".X": number of digits after the decimal point (e. g.: .4 = four digits after the decimal point)
- Input format "X.X": number of total digits including the number of digits after the decimal point (z. B.: 10.4 = 10 digits of which four after the decimal point)

Configure preselection in the options tab

You can change the preselection of the *Export as DAT* checkbox with the configuration option `gtu_points_write_use_templates`. (Default is 0=off)

You can change the preselection of the *Write ijk* checkbox with the configuration option `gtu_points_write_ijk`. (Default is 0=off)

You can set the predefined value in the *Precision* field with the configuration option `gtu_points_precision`. (Default value is 15.6)

Output formats and examples

1. Standard output

No template and no ijk values

File format: Creo point file *.pts

Example of an output with default precision of 15.6:

```

96.090000    -0.320000    50.000000
130.690000    -5.660000    50.000000
123.440000    -33.600000    50.000000
89.940000     -36.160000    50.000000
82.610000     -8.000000     50.000000

```

2. Output with ijk values

The perpendicular is added to a surface at the position of a point.

File format: *.dat

Example of an output with default precision of 15.6.

```

96.090000    -0.320000    50.000000    0.000000    1.000000    0.000000
130.690000    -5.660000    50.000000    0.000000    1.000000    0.000000
123.440000    -33.600000    50.000000    0.000000    1.000000    0.000000
89.940000     -36.160000    50.000000    0.000000    1.000000    0.000000
82.610000     -8.000000     50.000000    0.000000    1.000000    0.000000

```

3. Output with template

You can customize the output file by using a template. See [next chapter](#)⁵⁰³ on how to create it.

File format: *.dat

17.10.4 Creating a template

You can set up a template by defining the following configuration options.

gtu_points_write_use_template

Defines whether a template should be used for outputting points values to a DAT file (1) or not (0). Presets the checkbox *Use templates* in the *Options* tab in GENIUS TOOLS Points. Standard: 0

gtu_points_write_filename

Defines the filename (without extension). It would be extended by .pts (without template) or with .dat (with template). Should the template use another extension this could be set by gtu_points_write_template_extension.

gtu_points_write_template_extension

Defines the file extension for a file with template. If the option is not set "dat" is used. The option has no impact on writing a file without a template.

gtu_points_write_template_names

Defines whether point names should be written to the output file. Default: 0

gtu_points_write_template_names_fill_up_length

Fills up the point name to a defined number of characters. Default: 20

gtu_points_write_template_modelnames

Defines whether model names should be written to the output file. Default: 0

gtu_points_write_template_filenames_fill_up_length

Fills up the model name to a defined number of characters. Default: 20

gtu_points_write_template_header

Defines a header for the DAT file.

gtu_points_write_template_footer

Defines a footer for the DAT file.

gtu_points_write_template_line_left

Defines the start of a line in the DAT file.

gtu_points_write_template_line_right

Defines the end of a line in the DAT file.

gtu_points_write_template_names_split

Replaces the defined string in the file name by a space character: Default: _

gtu_points_write_template_seperator

Defines the separator that is used for separating the values. Standard is space " ".

gtu_points_autoselect_points_with_rule

Points would be preselected at module startup and can be filtered by name. The preselection at assemblies would be done over all parts and subassemblies. The definition has to be done in RegExp (e.g. ^.*\$).

17.10.5 Example template

A .csv file with a defined head should be created.

set Options:

```
gtu_points_autoselect_points_with_rule=^.*$
```

All points get preselected.

```
gtu_points_write_filename=@mdl@
```

The file should be named after the model name (not the chosen coordinate system).

```
gtu_points_write_template_extension=csv
```

The file extension should be .csv.

```
gtu_points_write_template_modelnames=1
gtu_points_write_template_names=1
gtu_points_write_template_filenames_fill_up_length=1
gtu_points_write_template_names_fill_up_length=1
```

The model names and point names should be written but not blank padded.

Die Modellnamen und Punktenamen sollen geschrieben und nicht aufgefüllt werden.

```
gtu_points_write_template_header=File name,Point name,X,Y,Z,
```

The option write its value as a header to the csv file.

```
gtu_points_write_template_seperator=,
```

The cells should be separated by Comma.

```
gtu_points_write_use_template=1
```

The use of the template is activated at module start.

Outcome:

```
File name,Point name,X,Y,Z,
cri_rl-2.asm,APNT0,28.000000,75.500000,-43.000000,
cri_rl-1.asm,APNT0,28.000000,75.500000,43.000000,
cri_loetnippel.prt,PNT0,65.500000,60.000000,-40.000000,
cri_loetnippel.prt,PNT0,105.500000,60.000000,-0.000000,
cri_loetnippel.prt,PNT0,65.500000,60.000000,40.000000,
cri_loetnippel.prt,PNT0,-28.000000,75.500000,-43.000000,
cri_loetnippel.prt,PNT0,28.000000,75.500000,-43.000000,
cri_loetnippel.prt,PNT0,-28.000000,75.500000,43.000000,
cri_loetnippel.prt,PNT0,28.000000,75.500000,43.000000,
cri_dampfmaschine_2000_skel.prt,PNT0,11.000000,6.062178,
3.500000,
cri_dampfmaschine_2000_skel.prt,PNT1,-11.000000,-6.062178,
3.500000,
cri_dampfmaschine_2000_skel.prt,PNT2,11.000000,35.857312,
0.000000,
cri_dampfmaschine_2000_skel.prt,PNT3,-11.000000,23.732956,
0.000000,
cri_dampfmaschine_2000_skel.prt,PNT4,28.000000,30.734190,
0.000000,
cri_dampfmaschine_2000_skel.prt,PNT5,-28.000000,35.410727,
0.000000,
cri_dampfmaschine_2000_skel.prt,PNT6,28.000000,-2.338269,-
1.350000,
cri_dampfmaschine_2000_skel.prt,PNT7,-28.000000,2.338269,-
1.350000,
```


17.11 Export Table One-To-One to Excel

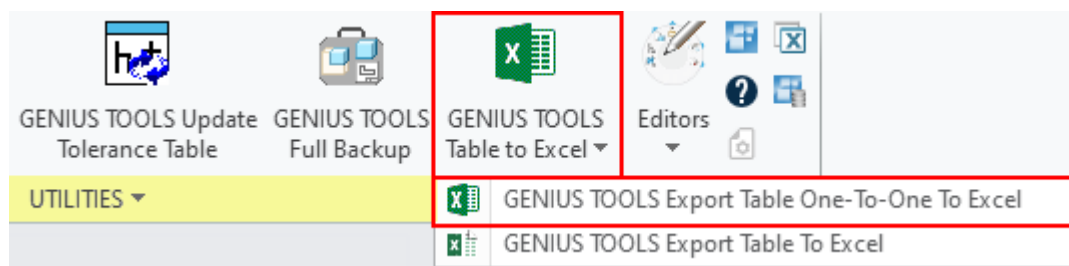
The function creates an Excel file without making use of a template, i. e. tables are always exported exactly as displayed. The Excel file without macros (XLSX file) can be generated from an assembly report (in assembly mode) or a drawing table (in drawing mode).

Warning: The Excel export requires Excel 2010 or later on the executing computer. Make sure that no Excel instance is running when you export a table.

Starting the program: in drawing and assembly mode

Export table One-To-One to Excel via can be started:

- For report tables: via the tools menu  of the dialog window **GENIUS TOOLS Assembly Report** ³⁰.
- For drawing tables: in drawing mode via the GENIUS TOOLS ribbon menu.



Starting via the ribbon menu in drawing mode

After starting the function, select a table in your current drawing. Merged cells will not be considered, the contents of the individual cells will be adopted as-is.

Configuring export settings

The Excel file is saved by default in the first level of the working directory with the name of the assembly or the drawing. The name can be changed.

Further export settings can be set with these configuration options:

gtu_table_to_excel_report_folder

Defines the directory in which Excel reports from *GENIUS TOOLS Export table One-To-One to Excel* and *GENIUS TOOLS Export table to Excel* are stored.

gtu_table_to_excel_autosave

Defines whether the Excel table is written to the defined report folder without a dialog to save (1). Default: 0

gtu_table_to_excel_open_export

Defines whether the excel file will be opened after the export (1) or not (0). (Default: 1)

Configuring export settings of drawing tables

For the export of drawing tables the following additional configuration options can be set:

gtu_table_to_excel_autosave

Defines whether the Excel table to be created is written to the defined report directory without a save dialog (1). Default: 0

gtu_table_to_excel_report_folder

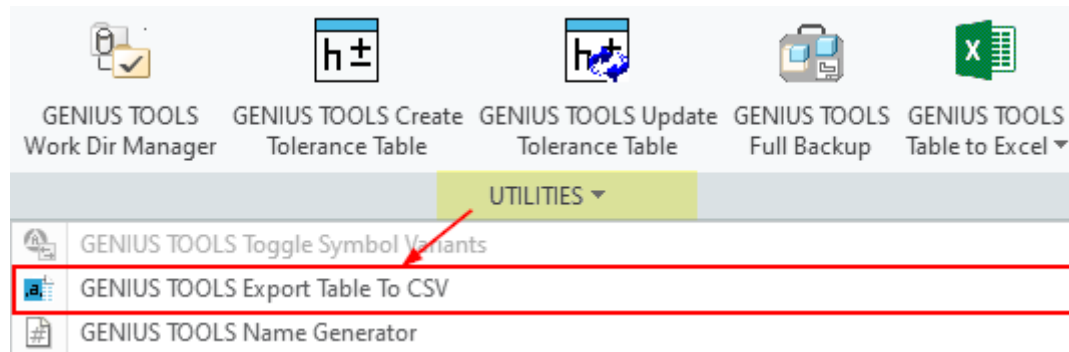
Defines the directory where Excel reports are saved from *GENIUS TOOLS Export Table One-to-One to Excel* and from *GENIUS TOOLS Export Table to Excel*.

17.12 Export Table to CSV

GENIUS TOOLS Export Table to CSV exports a selected drawing table, or a table template with data from a drawing. In both cases, the result is written to a CSV file.

Program start: in drawing mode

Start *GENIUS TOOLS Export Table to CSV* via the GENIUS TOOLS ribbon menu in the tab UTILITIES when having opened a drawing.



Create report

You can choose from two sources to retrieve the data you want to export: from a TBL file (as a template) or from a table that has already been placed on a drawing.

Selecting a table source

- **Report from file:** adds a template table (TBL file) to the current drawing which is filled using the table rules and exported to a CSV file. After completion of the export, the table is removed from the drawing.
- **Report from drawing:** exports the selected drawing table to a CSV file. The table is not changed.

Configuring export settings

Both options can be adapted in the following configuration options. (See screenshot.)

1. Name of the table, only when selecting "From file": **gtu_table_to_csv_table_template**
Specifies the TBL file that will be used as default for data export. Default path:
`%GT_RESOURCE_FOLDER%/utilities/int-table2csv.tbl`
2. Export folder: **gtu_table_to_csv_export_folder**
Specifies the directory to which the exported CSV file will be written.
3. Export file: **gtu_table_to_csv_export_file**
Defines the name of the exported CSV file. Can contain variables. Default: file name of model
4. Separator: **gtu_table_to_csv_export_sep**
Defines the separator between the values. Default: ; (semicolon)

Furthermore, you can define the way the CSV file will be encoded:

gtu_table_to_csv_write_file_as_utf8

Defines whether the CSV file is written as ASCII (0) or UTF8 (1). Start value: 1

gtu_table_to_csv_write_file_as_utf8_with_bom

Defines whether a CSV file written with UTF8 is additionally encoded with Byte Order Mark (BOM) (1) or not (0). Initial value: 0. For this, above option `gtu_table_to_csv_write_file_as_utf8` must be set to 1.

Deactivating the function

Use the configuration option `gtu_start_table_to_csv` to turn off the button display that starts the program in the ribbon menu. (Start value: 1 = On)

17.13 Export Table to Excel

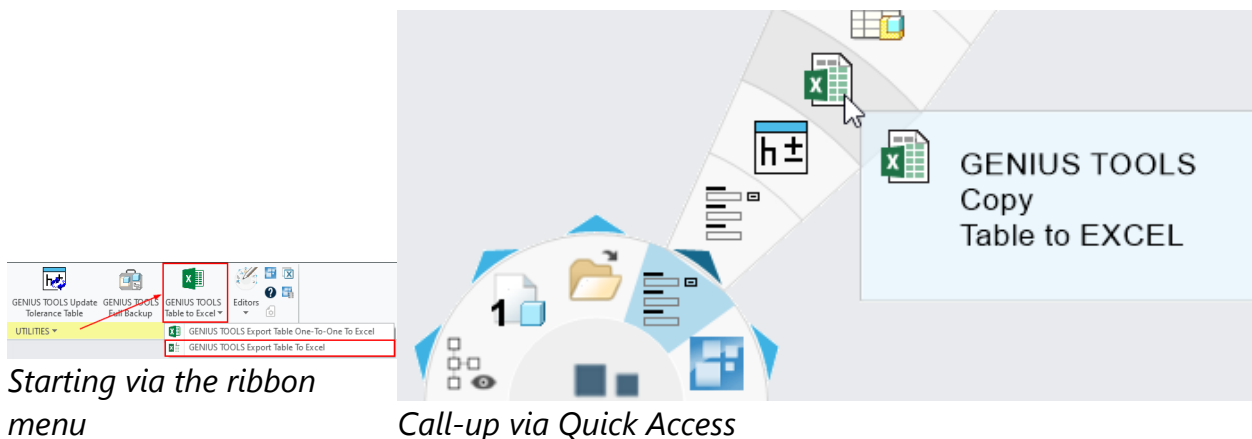
The function *Export Table to Excel* exports a report table, e. g. a Creo drawing table into an Excel file (with or without macros). The data to be exported is specified by using an Excel template.


Warning: The Excel export requires Excel 2010 or later on the executing computer. Make sure that no Excel instance is running when you export a table.

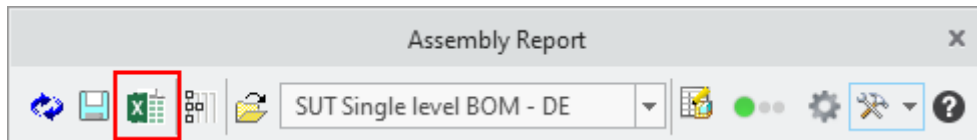
Starting the program: in drawing and assembly mode

Start GENIUS TOOLS Export Table to Excel in the different modes.

- For drawing tables (drawing mode):
via the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).



- For report tables in GENIUS TOOLS Inspect¹³² and Inspect Revision¹⁷⁰ (drawing mode):
in the command bars of the dialog windows
- For report tables in GENIUS TOOLS Assembly Report³⁰ (assembly mode):
with the button  in the command bar of the dialog window



Configuration

The configuration option `gtu_table_to_excel_open_export` defines, whether the Excel file opens after export (1) or not (0). (Default: 1)

17.13.1 Creating reports

Follow the dialog, to start the export of drawing or other report tables.

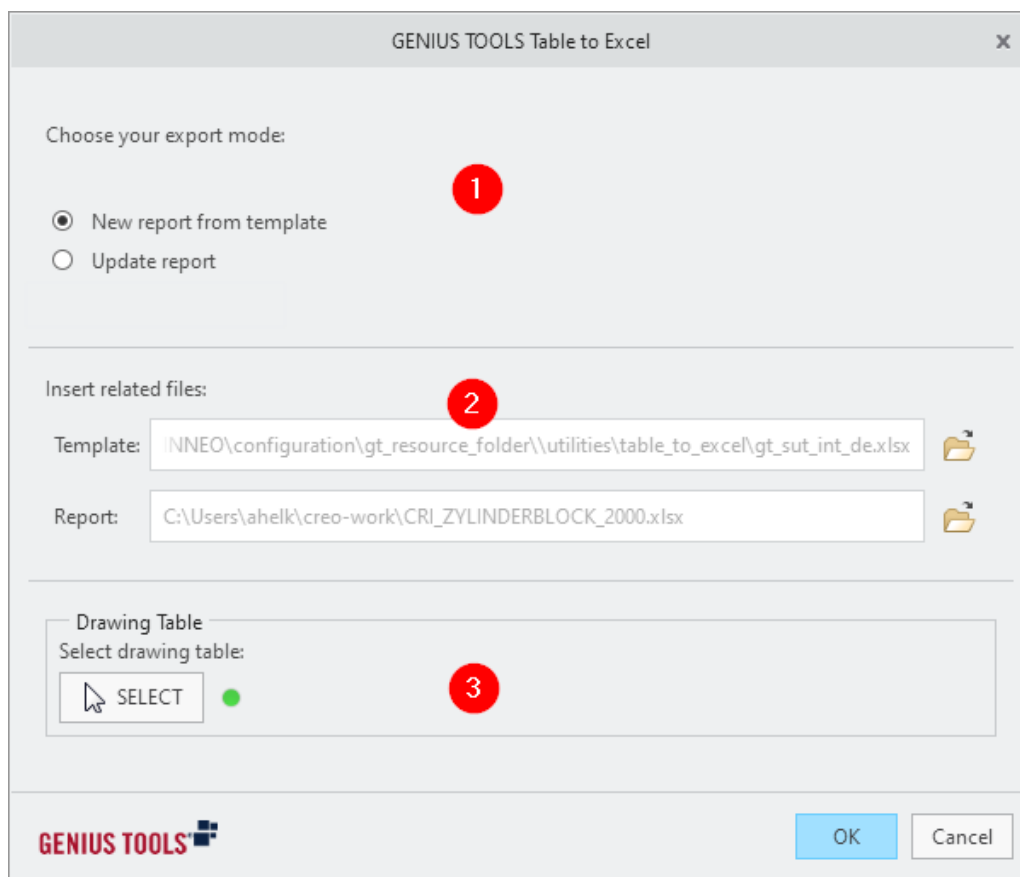


Table to Excel dialog

1. Choose export mode

Choose from the three options:

New report from template: Starts the function *Copy table to Excel*. This function fills an Excel template with the contents of a drawing table and saves the new Excel document.

Update report: This function updates an existing Excel document, whose values have been defined by a comment code. To update you only need to select the Excel document to be updated.

Please note: An existing Excel file can only be updated if the cell comments have not been removed (manually, or automatically via the configuration options).

Please note: This function is called *Comment* until Excel 2019, from Microsoft 365 onwards it is called *Note*.

Hiding the update function: By setting the configuration option *gtu_table_to_excel_erase_replaced_comments* to 1 you can hide the Update report function. The first segment of the dialog (Choose export mode) disappears. The default settings is 0 (do not hide Update function).

2. Select files

Template: Select a configured [Excel template](#)⁵¹² for the report. The default paths vary according to the various GENIUS TOOLS components, see section below.

Report: Enter directory of the report to be created.

3. Select drawing or report table

This segment differs in the various GENIUS TOOLS components:

- in drawing mode:

Click *Select* and select a drawing table in Creo.

- in GENIUS TOOLS Inspect:

For inspection symbols: Select a [configuration file \(XML\)](#)¹⁴⁹.

For revision / snapshot histories: Select a [configuration file with history \(XML\)](#)¹⁷².

- in GENIUS TOOLS Assembly

Select a [report definition \(XML\)](#)²⁶.

- reports from different GENIUS TOOLS components:

Several selection fields will open, if you select a template that contains more than one component acronym, e. g. *gti* and *gti_rev*, see [Export data from several GENIUS-TOOLS-components](#)⁵¹⁷.

Configuring default paths

Default paths in the dialog can be changed in the various GENIUS TOOLS components as follows.

For the template file

- in drawing mode:

`<gt_resource_folder>\utilities\table_to_excel`

Can be changed with `gtu_table_to_excel_template_folder`.

- in GENIUS TOOLS Inspect:

For inspection symbols: `%gt_resource_folder%\inspect\gti_inspection_template_de_en.xlsx`

Can be changed with `gti_excel_template`.

For revision histories: `%gt_resource_folder%\inspect`

Can be specified with `gti_revision_folder`.

- The default template `gti_revision_template_de_en.xlsx` can be changed with `gti_revision_excel_template`.

- in GENIUS TOOLS Assembly:

`%gt_resource_folder%\assembly\gt_assembly.xlsx`

Can be changed with `gta_export_template`.

For the output file

- In drawing mode:

Can be set with `gtu_table_to_excel_report_folder`.

- in GENIUS TOOLS Inspect, for inspection characteristics:

Can be set with `gti_excel_export_path`.

- in GENIUS TOOLS Inspect Revision:

Can be set with `gti_revision_excel_export_path`.

- In GENIUS TOOLS Assembly:

Can be set with `gta_export_path`.

gtu_table_to_excel_run_check

Specifies whether to check whether an Excel session is open. Default: 1 (yes)

17.13.2 Create template

Exporting a report to Excel requires an Excel template in XLSM or XLSX format (Excel file with and without macro functions). This function is available in the components GENIUS TOOLS Assembly Report and GENIUS TOOLS Inspect.

An export template is an excel table, where certain cells contain comments that consist of a component acronym and a fill command. By using various component acronyms you can create templates with data from different GENIUS TOOLS components.⁵¹⁷

The report output format is the same as the template format but can be changed.

17.13.2.1 Step-by-step guide


Create a separate excel file for each report template.

Design table

1. Open a new Excel file with an Excel version 2010 or higher.

2. Design a table by formatting cells and rows and fill in the headings as needed. Be careful to distinguish between the header and the report area:

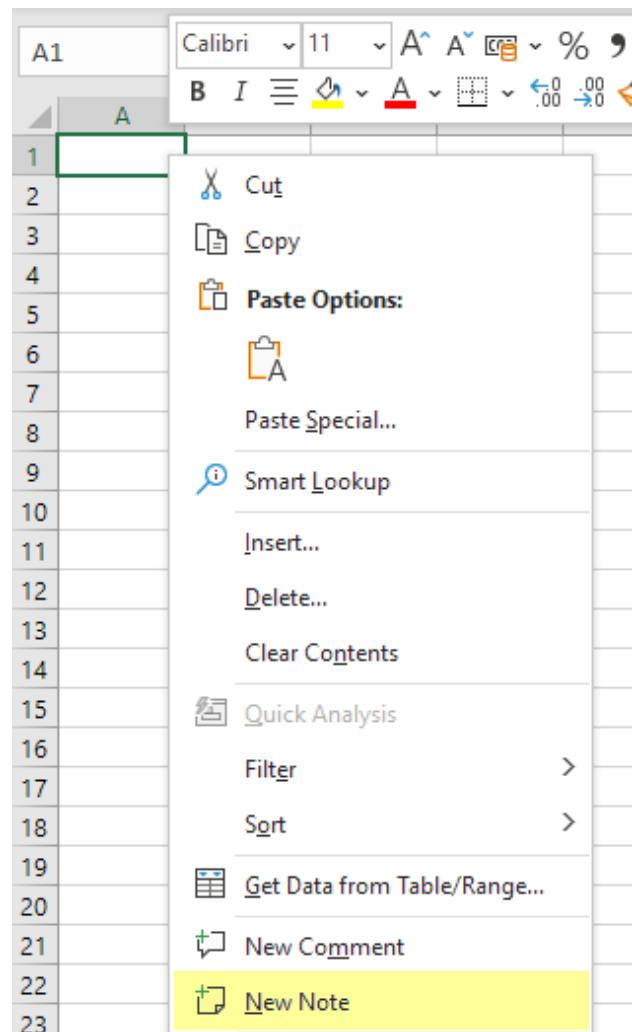
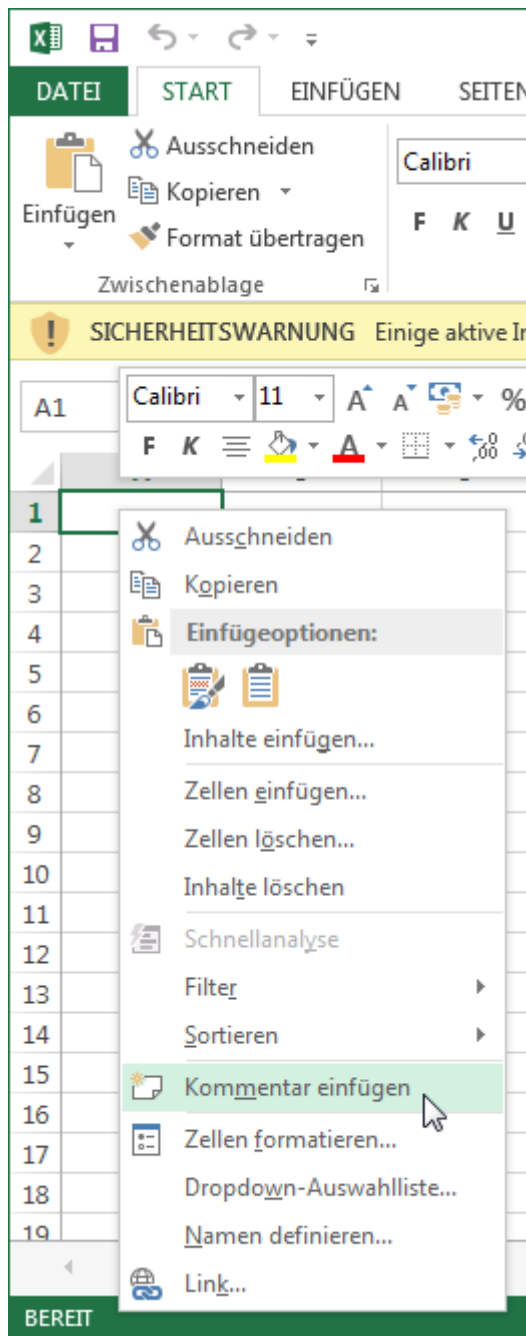
- head parameters and variables fill a single cell
- report parameters fill a column

	A	B	C	D	E	F	G	H
1	Datiname / filename:							STK-Typ / BOM type:
2	Benennung / Description 1:							Baukasten / Single level
3	Bezeichnung / Description 2:							Datum / Date:
4	Artikelnummer / Part number:							
5						That's IT.		
6	Position	Qty.	Part number Part no. Semifinished	Rev.	Description 1	Description 2 Description semifinished	Material Coating / Coloring	Mass Mass Σ
7								
8								
9								

Creating comments

3. Create comments by using the Excel context menu. The text of the comment determines the value that will be copied to a single cell or the cells of a column.

Please note: This function is called *Comment* until Excel 2019, from Microsoft 365 onwards it is called *Note*.



Content of comments

4. The first line of a comment is the editor name and will be ignored. The second line will be analyzed and fills the cell.

Nummer	Beschreibung	Verknüpft mit
Number	Description	Linked to
	Miller, Max	
	gti_rev:var_num	

5. The following content is possible in comments:
- for a single cell: head parameters and variables,

- for a column: report parameter

Head parameters

Head parameters are written into a single field.

6. Enter the text of the comment in a cell of the head area

- for GENIUS TOOLS Assembly Report: in the notation *gta:%head*%* (Replace * with the position of the head parameter in the editor.)

Example: Head parameter 1 = *gta:%head1%*, head parameter 2 = *gta:%head2%* etc.

A	B	C	D
Name			GTA: gta:%head1%
Description 1			
Article number			

Head parameters with numbering for GENIUS TOOLS Assembly Report

- für GENIUS TOOLS Inspect / Inspect Revision: Variables without component acronym
Beispiel: *%DRAWING_NO%*

A	B	C	D	E	F	G
Zeichnung-Nr. / Drawing no.:		-			GTI: %curmod:DESCRIPTIO N_1_DE%	
Benennung / Description 1:		-				
Bezeichnung / Description 2:		-				

Head parameter for GENIUS TOOLS Inspect

Report parameters

Report parameters fill cells in a column. The length of the Excel report depends on the number of rows displayed.

Report parameters must consist of the name of the component (acronym) and the fill command.

Components	Acronym	Fill command
GENIUS TOOLS Assembly Report	gta:	%ruleParameter% to give out position parameter %col*% for each column, e. g. %col1%
GENIUS TOOLS Inspect	gti:	See table. ¹⁴⁶
GENIUS TOOLS Inspect Revision	gti_rev:	See table. ¹⁷⁹

7. Enter the comment text (2) in the first row (1) below the heading.

Denomination 1	-
Denomination 2	-
IDNR	-
Filename	-

Revision data		Variable Text	
Revision	Content	Description	Linked to
1	Michaelis, Markus: gti_rev:rev_revision	2	

8. Define a report parameter column by column.

- für GENIUS TOOLS Assembly Report:
`gta:%ruleParameter%` as well as `gta:%col*%` for each column, e. g. `gta:%col1%`. Without the display of the position parameter, the counting of the columns starts at 0.
- für GENIUS TOOLS Inspect / Inspect Revision:
 Use the acronyms and commands as shown in the table of step 5.

Variables

Variables fill a single cell with general information.

9. Use the available variables of GENIUS TOOLS for Creo if you wish, e. g. date and time information as well as Creo object information, in the notation `@variable@`. See [Usage of variables](#). ⁰⁸⁵

Datum / Date:	GTA: @date@
---------------	----------------

Please note: The variables `@feat_id@`, `@selmdl@` and `@selmdlpath@` are not supported!

10. Save the template

- for GENIUS TOOLS Assembly Report in:
`%gt_resource_folder%\assembly`
- for GENIUS TOOLS Inspect / Inspect Revision in:
`%gt_resource_folder%\inspect\`

17.13.2.2 More options for templates

Output drawing table rows

Output one or more drawing table rows.

T:RowNumber (row) :ColumnNumber (column) :NumberFollowingRows:NumberHeaderRows (for split tables)

Always pay attention to the table orientation in Creo Parametric (the table direction in the table properties)! The first cell of a drawing table is always the initial point of the table.

Examples

T:1:1 - Copies the cell 1:1 (first row: first column (the counter always starts at 1) into the Excel cell.

T:1:1:all - Copies the cell 1:1 and all cells 1 of the following rows (2:1, 3:1...) to the associated Excel table rows (each in the column below).

T:1:1:3 - Copies the cell 1:1 and the first cell of the three following rows (2:1, 3:1, 4:1 - if so many exist) to the following Excel table rows.

T:1:1:all:3 - Copies all cells of position 1 after the first three rows to the associated Excel table rows.

Output empty cells

Furthermore, it is possible to copy empty cells that function as placeholders into the report table. In this case, only the frame is taken over.

`empty:NumberOfHeaders:NumberOfValueLines:NumberOfFootlines`

Examples

`empty:1` - Copies a line with a border from the Excel template.

`empty:6` - Copies 6 lines with border from the Excel template.

`empty:2:4` - Copies 2 lines with border from the Excel template.

`empty:1:all:2` - Copies all cells except the last two rows with borders from the Excel template.

17.13.2.3 Export data from several GENIUS-TOOLS-components

By using the component abbreviations you can create templates with different areas or spreadsheets. This is useful in GENIUS TOOLS Inspect, for example.

Fill fields and columns from different GENIUS TOOLS components by creating comments with the corresponding component abbreviations (gta, gti, gti_rev). Otherwise, set up the template according to the general procedure.

If you select a template with several component abbreviations, the selection fields belonging to the components will open in the lower area.

Example for the common export of report tables

- of inspection characteristics from GENIUS TOOLS Inspect and
- Revision histories from GENIUS TOOLS Inspect Revision

1. creation of several Excel tables with different component abbreviations, here: gti and gti_rev.

Report data from GENIUS TOOLS Inspect with the component abbreviation gti

Report data from GENIUS TOOLS Inspect Revision with the component abbreviation gti_rev

	A	B	C	D	E	F	G	H	I	J	K	L
1	Zeichnungs-Nr. / Drawing no.:			-								
2	Benennung / Description 1:			-								
3	Bezeichnung / Description 2:			-								
4	Dateiname / File name:			-								
5	No.	Sheet	Grid	Main type	Subtype	Nom. dim.	off size	Min. dim.	Max. dim.	Tol. standard	Tol. table	Tol. class
6					Michaelis, Markus: gti:tpc_main							
7												
8												

Inspect Revision +

Report data from GENIUS TOOLS Inspect with the acronym gti

	A	B	C	D	E	F	G	H	I	J	K	
1	Zeichnungs-Nr. / Drawing no.:		-									
2	Benennung / Description 1:		-									
3	Bezeichnung / Description 2:		-									
4	Dateiname / File name:		-									
5	Revision data			Variable texts								
6	Revision	Creo_ID	Number	Description	Link	Location	Symbol Number	Sheet	Grid	Maintype	Subtype	Nominal d
7					Michaelis, Markus: gti_rev:var_descr							
8												

Inspect

Revision

Report data from GENIUS TOOLS Inspect Revision with the acronym gti_rev

2. Save the file as gti_gtir_template.xlsx

3. Result: When selecting this file as a template, the dialog box opens with selection fields for Inspect and Inspect Revision.

The screenshot shows a dialog box titled "GENIUS TOOLS Table to Excel". It contains several sections for configuring the export process. The "Choose your export mode:" section has three radio buttons: "New report from template" (selected), "Update report", and "One-to-One". The "Insert related files:" section has two text boxes: "Template:" with the path "C:\Users\ahelk\Desktop\gti_gtir_template.xlsx" and "Report:" with the path "C:\Users\ahelk\creo-work\cri_gti_zyylinderblock_2000.xlsx". The "Inspect" section has a "Choose configuration:" dropdown menu with "gti_inspection.xml" selected. The "Inspect Revision" section has a "Choose history:" dropdown menu with "gti_example" selected. At the bottom, there is a "GENIUS TOOLS" logo and "OK" and "Cancel" buttons.


17.14 Extend Relations

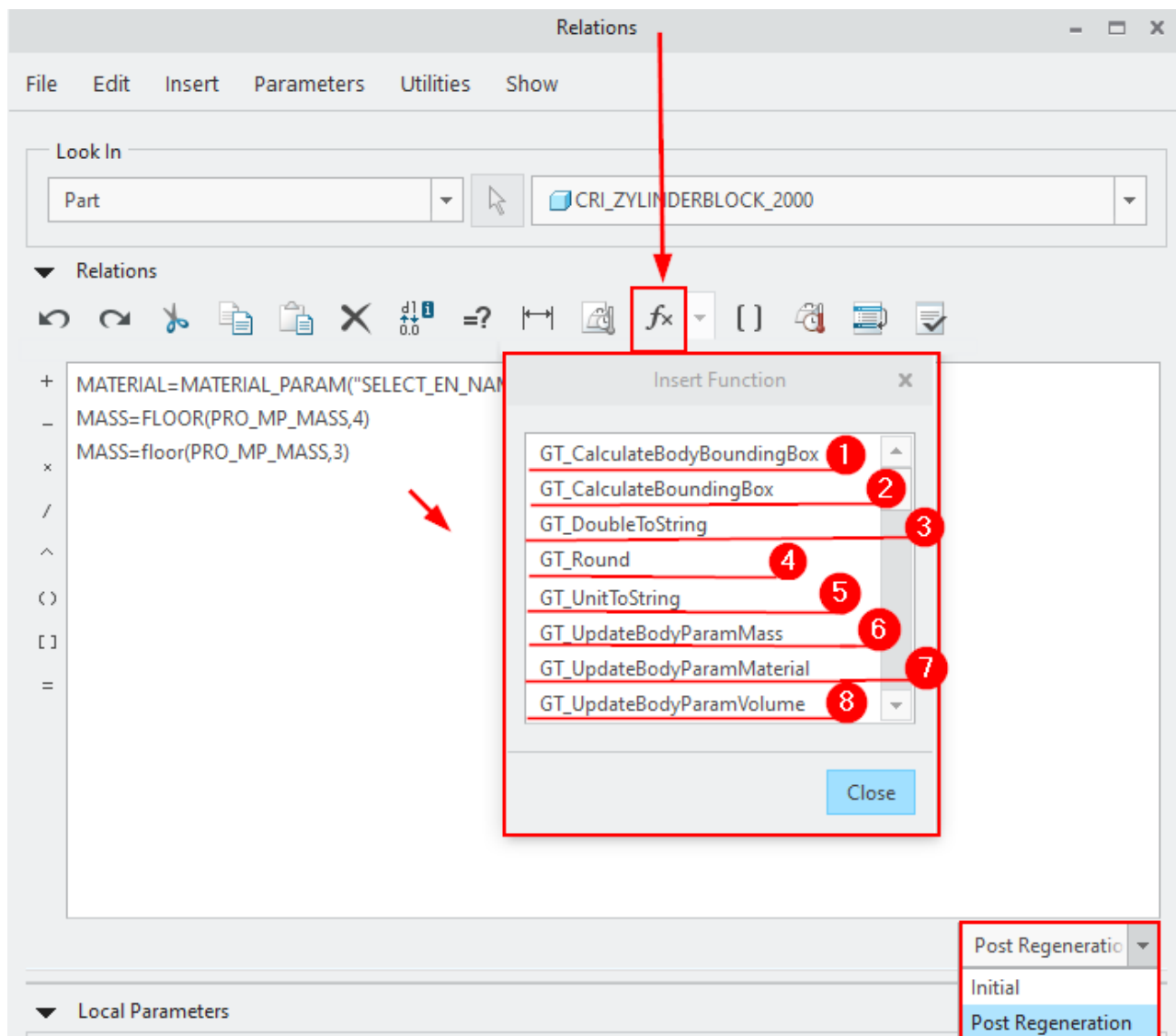
Extend Relations is a set of extra functions to model relations that define parameters for models and – in Creo 7.0 – also for bodies.

The additional functions are executed at (initially) or after (post) regeneration guaranteeing that all values are up-to-date.

Please note: The function *Extend Relations* is only available with subscription licenses for GENIUS TOOLS for Creo.

Adding functions: in part and assembly mode

The additional functions can be found in the tab *Tools* in the Creo's *Relation* dialog. In this Creo dialog, click . Take care to select the proper kind of relations – you can choose between *Initial* relations and *Post Regeneration* relations (Below: Post Regeneration). This selection allows you to control the timing of the execution of the relations and thus the regeneration time. *Initial* relations are executed whenever changes are made, but before the model is regenerated, while *Post Regeneration* relations are executed only after the entire model is regenerated.



Functions that extend relations in the Creo dialog Relations > Insert Function

Functions

These additional functions are used for parameters that define bounding boxes (functions 1 and 2) or bodies (functions 5, 6, 7) and for rounding values. Functions that generate values for bodies (1, 5, 6, 7) are available in Creo 7.0 and later versions.

1. GT_CalculateBodyBoundingBox("BodyParameter","optFormat")

A parameter is generated and updated for each body in a model with the values of a bounding box.

"optFormat" is optional: Write in the notation "x: %.Numberf y: %.Numberf z: %.Numberf"

Example: `GT_CalculateBodyBoundingBox("DIMENSION", "x: %.1f y: %.1f z: %.1f")`

2. GT_CalculateBoundingBox("ModelParameter","optFormat")

A parameter is generated and updated for the model with the values of the bounding box of the model..

"optFormat" is optional: Write in the notation "x: %.Numberf y: %.Numberf z: %.Numberf"

Example: `GT_CalculateBoundingBox("DIMENSION", "x: %.1f y: %.1f z: %.1f")`

3. GT_DoubleToString()

Function rounds parameter values and converts the received numbers into strings. See [Rounding](#). ⁵²³

4. GT_Round()

Function rounds parameter values. See [Rounding](#). ⁵²³

5. GT_UnitToString()

Function converts the unit of a parameter into strings.

Example: `MASS_UNIT = GT_UnitToString("MASS")`

6. GT_UpdateBodyParamMass ("BodyParameter",optMultiplier,optDecimalPlaces)

A body parameter with the value of the body mass is created and updated for each body.

The entries *optMultiplier* and *optDigitsAfterComma* are optional. If not defined *optMultiplier* = 1.0 and *optDigitsAfterComma* is not rounded.

Example: The multiplier 0.001 is used to convert the default setting "kg" to "g".

Input

`GT_UpdateBodyParamMass ("MASS", 0.001, multiplies the body mass by 0.001
2)`

Result

rounds to 2 decimal places
writes the result into the parameter "MASS"

7. GT_UpdateBodyParamMaterial("BodyParameter", "MaterialParameter")

A parameter is generated and updated for each a body with the value of the material parameter of the body's material.

A body parameter with the value of the body mass is created and updated for each body.

Example: `GT_UpdateBodyParamMaterial("MATERIAL","SELECT_EN_NAME")`

8. **GT_UpdateBodyParamVolume ("BodyParameter",optMultiplier,optDecimalPlaces)**

A body parameter with the value of the body volume is created and updated for each body.

The entries *optMultiplier* and *optDigitsAfterComma* are optional. If not defined *optMultiplier* = 1.0 and *optDigitsAfterComma* is not rounded.

Example:

Input

Result

`GT_UpdateBodyParamVolume("VOLUME")` writes the result into the parameter "VOLUME"

Configuring display

Use the following configuration options to change the standard display of the functions in the *Insert function* dialog.

gtu_start_relationExtension

Defines whether all commands of the *Extend Relations* utility are shown (1) or not (0).
Default: 1.

gtu_relationextension_calculateBoundingBox

Defines whether the commands *GT_CalculateBodyBoundingBox()* and *GT_CalculateBoundingBox()* are shown (1) or not (0). Default: 1.

gtu_relationextension_doubleToString

Defines whether the rounding command *GT_DoubleToString* is shown (1) or not (0).
Default: 1.

gtu_relationextension_round

Defines whether the rounding command *GT_Round* is shown (1) or not (0). Default: 1.

gtu_relationextension_unitToString

Defines whether the converting command *GT_UnitToString* is shown (1) or not (0). Default: 1.

gtu_relationextension_updateBodyParamMass

Defines whether the body mass command *GT_UpdateBodyParamMass* is shown (1) or not (0). Default: 1.

gtu_relationextension_updateBodyParamMaterial

Defines whether the body material command *GT_UpdateBodyParamMaterial* is shown (1) or not (0). Default: 1.

gtu_relationextension_updateBodyParamVolume

Defines whether the body volume command *GT_UpdateBodyParamVolume* is shown (1) or not (0). Default: 1.

Rounding

The functions *GT_Round* and *GT_DoubleToString* round numerical values according to IEEE 754. *GT_DoubleToString* converts a number into a string.

GT_Round(Value, RoundDigits)

Enter the name of the parameter (e.g. LENGTH) or a numerical value in the parentheses of the function as well as the decimal places to be rounded.

Examples for parameter LENGTH=17.3845:

Input	Results
A=GT_Round(17.3845,1)	17.400000
B=GT_Round(LENGTH,1)	17.400000
C=GT_Round(LENGTH,2)	17.380000
D=GT_Round(LENGTH,3)	17.385000

GT_DoubleToString(value, "%[WIDTH][.PRECISION][f]")

Function rounds parameter values according to IEEE 754 and converts the obtained numbers into strings. Enter in the parenthesis of the function:

Value: name of the parameter or numerical value

[WIDTH]: This number specifies the minimum number of all characters of the string. Characters can be digits, commas and spaces. (Example: The number 17.38 has 5 characters).

- if this specification is less than or equal than the number of all digits and the comma, the specification has no effect on the representation of the string. (In the example: E, F)
- if this specification is greater than the number of all digits and the comma, empty spaces are inserted at the beginning. (In the example: J, I, K)

[.PRECISION]: Number of digits after the comma

[f]: Formatdefinition:

- f: Floatpoint definition
- e: Exponential definition
- g: Shortest definition (f or e)

Examples of parameters LENGTH=17.3845:

Input	Result	Explanation
A=GT_DoubleToString(LENGTH,"%f")	17.384500	Number as floatpoint with pending zeroes (6 places)
B=GT_DoubleToString(LENGTH,"%e")	1.738450e+01	Number as exponential with pending zeroes (6 places)
C=GT_DoubleToString(LENGTH,"%12f")	17.384500000000	Number with WIDTH of 12
D=GT_DoubleToString(LENGTH,"%1f")	17.4	1 : One digits after dot
E=GT_DoubleToString(LENGTH,"%2f")	17.38	2 : Two digits after dot
F=GT_DoubleToString(LENGTH,"%7.2f")	17.38	7 : WIDTH of 7 and 2 digits after dot 2 : Two places after dot
H=GT_DoubleToString(LENGTH,"%1e")	1.7e+01	1 : One digits after dot
I=GT_DoubleToString(0.000001,"%g")	1e-07	Shortest definition

G is not generated because it is a PTC constant (gravitational constant 9.8)

17.15 Extended Dimension Functions

In the dialog *Extended Dimension Functions* a selected dimension can be quickly increased or decreased by a defined value. The modification is made on the nominal value.

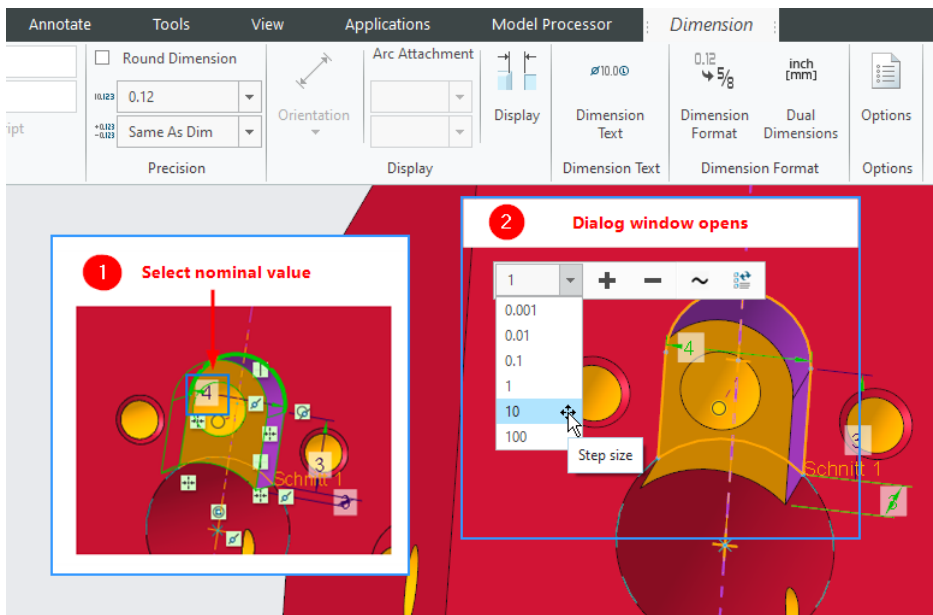
Please note: This function is only available with a subscription license for GENIUS TOOLS for Creo.

Opening the function: in part and assembly mode

The following conditions must be fulfilled:

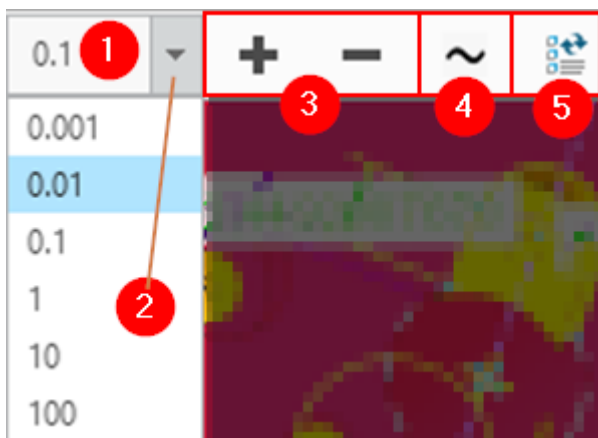
- a dimension must be selected,
- the selected dimension must not be a measured dimension,
- the selected dimension must not be controlled by a relationship.

The dialog opens above the selected value.



Please note: If the dimension is set by hand in the *Dimension* tab in the *Value* group, this can collide with the advanced dimension functions.

User interface



Dialog Extended Dimension Functions

1. Current step size
2. Dropdown window for selecting step size
3. Increasing/ decreasing the value by one in the selected step size
4. Round to the digit of the selected step size
5. Regenerate model


Configuring the function

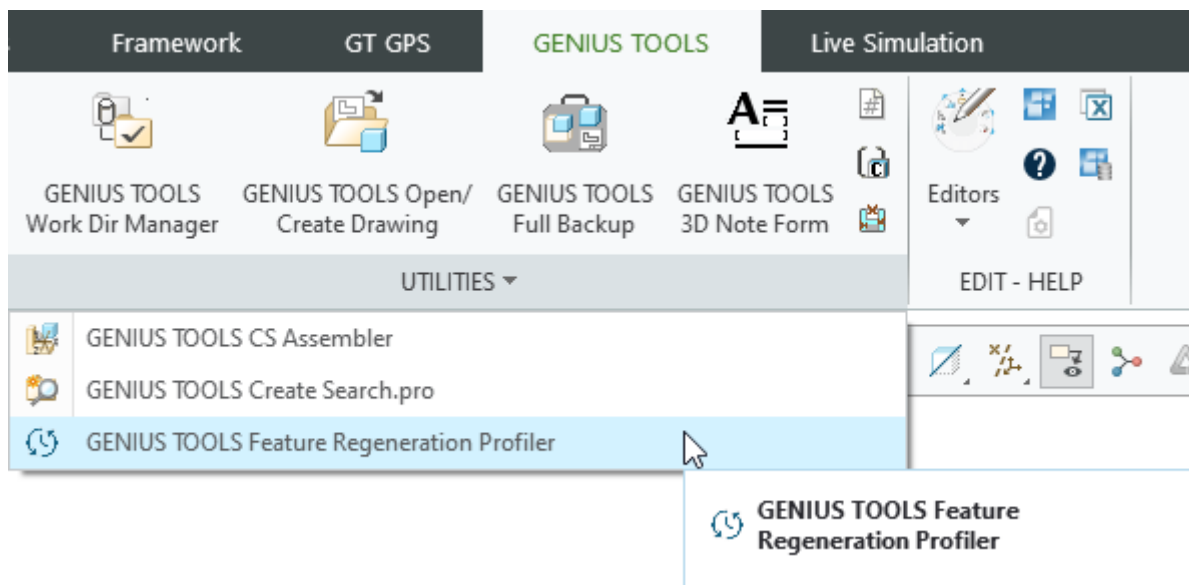
Use the configuration option `gtu_start_extendedDimensionFunctions` to hide the display. (Default is 0=Off/hidden)

17.16 Feature Regeneration Profiler

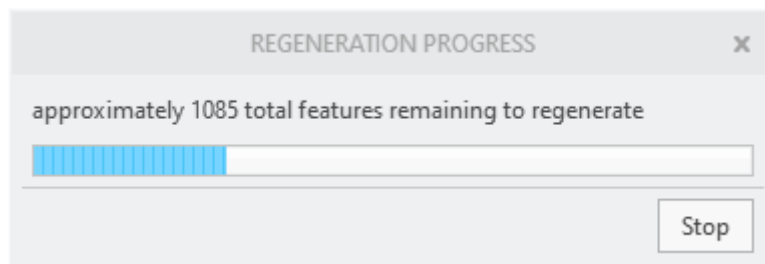
GENIUS TOOLS Feature Regeneration Profiler allows you to regenerate your models and simultaneously display the times for the regeneration of each feature.

Starting the program

The button for the GENIUS TOOLS Feature Regeneration Profiler  feature is below the GENIUS TOOLS menu ribbon in the UTILITIES segment.



Clicking on the button starts the regeneration directly. While the regeneration is running, an interface with the progress and the remaining features to be regenerated is displayed.



Information about the regeneration

The Creo message log displays information about the regeneration of each feature.

Each feature is listed in a line according to the following scheme:

Model name - Feature ID | Number of regenerations | Time in seconds (0.001 → 1 m s).

At the end of the message log, the total time of the regeneration is indicated.

```

RO-CRI_RL-1-0002 - 12 | 1 | 0
RO-CRI_RL-1-0002 - 2 | 1 | 0
RO-CRI_RL-1-0002 - 4 | 1 | 0
RO-CRI_RL-1-0002 - 45 | 1 | 0.003
RO-CRI_RL-1-0002 - 6 | 1 | 0
RO-CRI_RL-1-0002 - 9 | 1 | 0
RO-CRI_RL-2-0001 - 12 | 1 | 0
RO-CRI_RL-2-0001 - 2 | 1 | 0
RO-CRI_RL-2-0001 - 4 | 1 | 0
RO-CRI_RL-2-0001 - 45 | 1 | 0.006
RO-CRI_RL-2-0001 - 6 | 1 | 0
RO-CRI_RL-2-0001 - 9 | 1 | 0
RO-CRI_RL-2-0002 - 12 | 1 | 0
RO-CRI_RL-2-0002 - 2 | 1 | 0
RO-CRI_RL-2-0002 - 4 | 1 | 0
RO-CRI_RL-2-0002 - 45 | 1 | 0.004
RO-CRI_RL-2-0002 - 6 | 1 | 0
RO-CRI_RL-2-0002 - 9 | 1 | 0
complete regeneration time | 3.169

```

*Message log of a completed
regeneration*

17.17 Full Backup


GENIUS TOOLS Full Backup allows you to back up an object and objects that dependent on it, as well as back up drawings with the same name and suppressed objects.

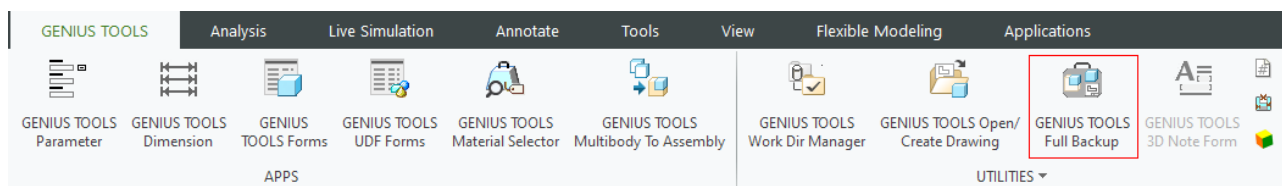
Thus, GENIUS TOOLS Full Backup extends the Creo Parametric function "create backup".

The objects to be saved can be saved in a directory or a ZIP file.


Please note: GENIUS TOOLS Utilities Full Backup is only available with subscription licenses for GENIUS TOOLS for Creo.

Starting the program: all modes

The function Full Backup  is found in Creo in the segment UTILITIES in the GENIUS TOOLS ribbon menu.



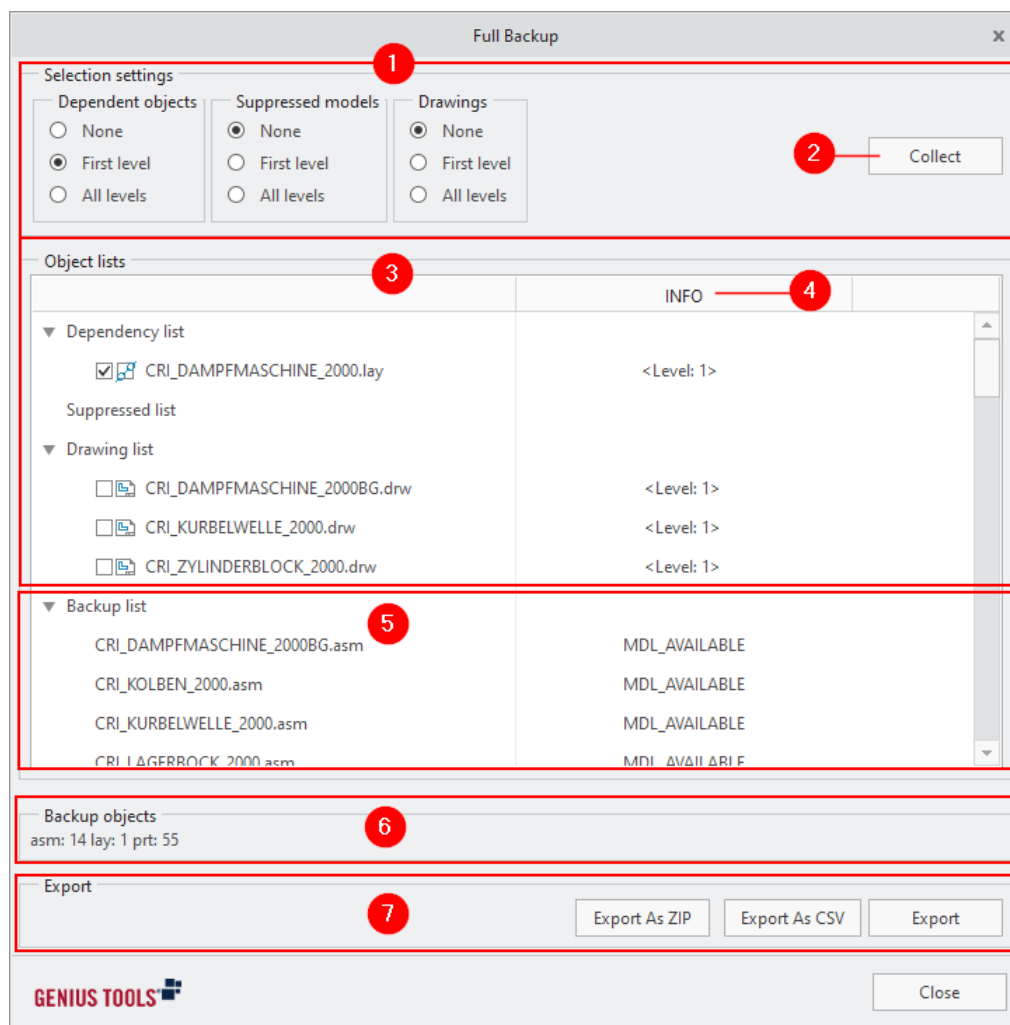
Hiding the button from the menu

Use the configuration option `gtu_start_fullbackup` to hide the GENIUS TOOLS Full Backup button  in the segment UTILITIES in the GENIUS TOOLS ribbon menu. (Default is 1= On)

17.17.1 Collect objects for backup list

You can decide whether and how dependent objects are added to the backup list and also include other objects.

Before the backup, you can select and deselect objects in the corresponding object lists.



GT Utilities Full Backup - Dialog

Decide in the selection settings (1) which elements to include in the backup list.

- None: Collects the current model and makes all drawings with the same name available for manual selection.

- First / All levels: GENIUS TOOLS Full Backup searches for dependent objects, drawings with the same name and suppressed models in different levels, see [next chapter](#)⁵³⁰ for details.

The *Collect* action (2) transfers to the object lists (3):

- all objects for the backup list: these are already checked.
- objects of the next level: are displayed with an empty checkbox and can thus be transferred manually to the backup list.
- details of the level are included in the info column (4).
- a model-parameter value defined by the configuration option `gtu_full_backup_column_parameter`.

All elements that are checked in the object list are transferred to the backup list.

The backup list (5) contains all backup objects to be backed up, sorted by their file extension:

- drawings (drw), assemblies (asm), parts (prt): are available for selection / deselection in the object lists
- drawing frames (frm) and layout (lay): are bound to the drawings in the backup list.

Objects that cannot be found are displayed in the *Failed Models* list.

Please note: the backup list may contain objects that are not available for selection, e. g. drawing frames.

The status bar (6) shows the number of objects in the backup list, listed according to the file extension.

The backup list can be exported (7):

- as a ZIP file: File name and path of the location are defined in the file selection dialog. Afterward fb__ and the current datum get added to the file name, e. g. *fb__ 17-06-2022_cri_kurbelwelle_2000*.
- as a CSV file: File name and path of the location are defined in the file selection dialog.
- in a folder: a subfolder is created with date and object name, e. g. *fb__ 17-06-2022_CRI_KURBELWELLE_2000*. All objects of the backup list are saved in this folder. If the folder is already existing an index is added.

Warning: After the commands *Export As Zip* and *Export*, the storage paths of all backup objects no longer point to their original directories. Before further editing with Creo Parametric, all objects must be removed from the session and must be called up again from their original directories. It is the same behavior as the standard Creo Parametric command *Save Backup*.

17.17.2 Choose levels of dependencies

GENIUS TOOLS Full Backup searches for dependent objects, drawings with the same name and suppressed models in different levels.




Drawings are analyzed to provide you with a checkbox for additional drawing models for the backup list.

Searching and analyzing dependent objects can take a long time for large assemblies or for data storage locations that have poor network connectivity. In these cases, it is recommended to select none as the search level and manually add the files that should be exported.

Overview of the selection settings:

Dependent objects

Search: None

Object to backup	Inclusion in backup list
	The current part. – If it is a variant, the generic is saved automatically. – If a drawing with the same name exists, it is offered for selection.
	The current assembly and all active sub-assemblies. – If it is a variant, the generic is automatically saved. This also applies to all sub-components. – If drawings with the same name (sub-models) exist, they are offered for selection.
	Manual selection for: the current drawing. All drawing frames and all drawing models that control the selected drawings are saved automatically. – Generics are automatically saved for all drawing models that are a variant. This also applies to all sub-components. – If drawings with the same name (sub-models) exist, they are offered for selection.



The current part.
– If it is a variant, the generic is saved automatically.
– If a drawing with the same name exists, it is offered for selection.



The current assembly and all active sub-assemblies.
– If it is a variant, the generic is automatically saved. This also applies to all sub-components.
– If drawings with the same name (sub-models) exist, they are offered for selection.



Manual selection for: the current drawing. All drawing frames and all drawing models that control the selected drawings are saved automatically.
– Generics are automatically saved for all drawing models that are a variant. This also applies to all sub-components.
– If drawings with the same name (sub-models) exist, they are offered for selection.

Search: First level

Object to backup	Inclusion in backup list
------------------	--------------------------

Dependent objects



All objects that control elements of the current part.



All objects that control elements of the current part or its subcomponents.



All objects that control elements of the drawing models or their subcomponents. (drawing frames and drawing models).

Search: All levels

Object to backup	Inclusion in backup list
---------------------	--------------------------



All objects that are dependent on parts of subsequent levels.



All objects that are dependent on parts or subcomponents of subsequent levels.



All objects that control elements of the drawing models or their subcomponents of the subsequent levels. (drawing frames and drawing models).

Suppressed objects

Object to backup	Inclusion in backup list
---------------------	--------------------------

Search: None



No suppressed objects of the current assembly are included.
- The suppressed objects would offered for selection.

Search: First level



All suppressed objects of the assembly that controls the part will be included.
- The second level of suppressed objects would offered for selection.



All suppressed objects of the current assembly are included.
- The second level of suppressed objects would offered for selection.

Suppressed objects



All suppressed objects of the assembly that controls the drawing will be included.

- The second level of suppressed objects would offered for selection.




Search: All levels



All suppressed objects of the found dependent objects are included in the backup list. The search depends on the selected search level for dependent objects.

Drawings

Search: None

Object to backup	Inclusion in backup list
	<p>The current part.</p> <ul style="list-style-type: none"> - If it is a variant, the generic is saved automatically. - If a drawing with the same name exists, it is offered for selection.
	<p>The current assembly and all active sub-assemblies.</p> <ul style="list-style-type: none"> - If it is a variant, the generic is automatically saved. This also applies to all sub-components. - If drawings with the same name (sub-models) exist, they are offered for selection.
	<p>Manual selection for: the current drawing. All drawing frames and all drawing models that control the selected drawings are saved automatically.</p> <ul style="list-style-type: none"> - Generics are automatically saved for all drawing models that are a variant. This also applies to all sub-components. - If drawings with the same name (sub-models) exist, they are offered for selection.



The current part.

- If it is a variant, the generic is saved automatically.
- If a drawing with the same name exists, it is offered for selection.



The current assembly and all active sub-assemblies.

- If it is a variant, the generic is automatically saved. This also applies to all sub-components.
- If drawings with the same name (sub-models) exist, they are offered for selection.



Manual selection for: the current drawing. All drawing frames and all drawing models that control the selected drawings are saved automatically.

- Generics are automatically saved for all drawing models that are a variant. This also applies to all sub-components.
- If drawings with the same name (sub-models) exist, they are offered for selection.

Search: First level

Object to backup	Inclusion in backup list
------------------	--------------------------

Drawings



Drawings with the same name are selected immediately. Additional drawing models are added to the backup list. If these have drawings with the same name, they are offered for selection.

Search: All levels




Drawings with the same name are selected immediately. Additional drawing models are added to the backup list. If these have drawings with the same name, they are offered for selection.

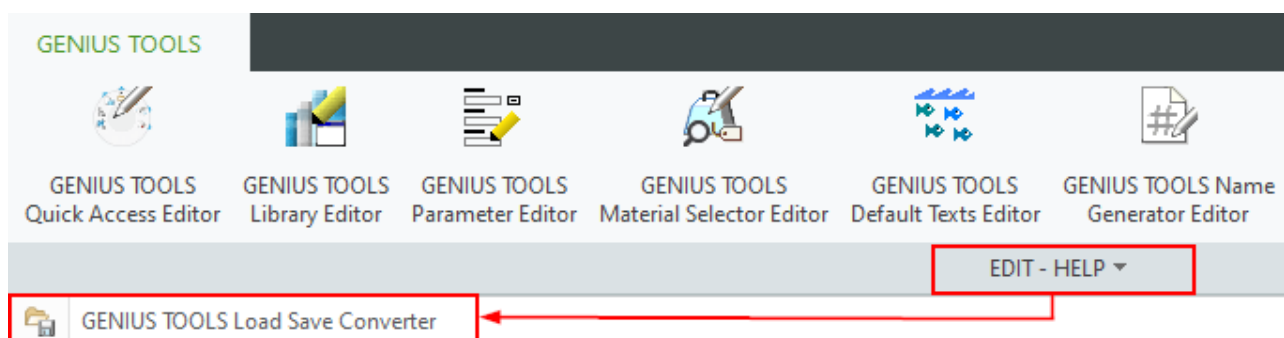
17.18 Load Save Converter

This functions allows you to convert objects, which have been created in older version of Creo, Wildfire or Pro/engineer, into the currently used Creo version. Also, you can convert to the student format.

Please note: GENIUS TOOLS Load Save Converter is only available with subscription licenses for GENIUS TOOLS for Creo.

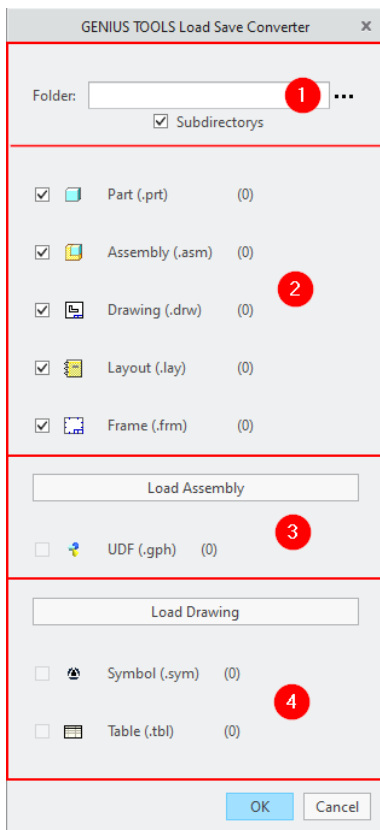
Starting the program: in Creo standby mode

The function Load Save Converter  is found in Creo standby mode in the segment EDIT-HELP in the GENIUS TOOLS ribbon menu.



Steps


1. When using the Load Save Converter function Creo objects are uploaded from a directory and saved again. This converts them in to the currently used Creo format.



*Dialog box of GENIUS
TOOLS Load Save
Converter*

2. Choose the directory (1) that contains the dated objects.
3. Check the object types (2) you want to convert.
4. If you want to convert UDF (user defined features), check the box in the segment *Load Assembly* (3). This uploads a stored assembly and converts the UDF that it contains.
5. If you want to convert symbols or tables, check the box in the segment *Load Drawing* (4). This uploads a stored drawing and converts the symbols and tables that it contains.
6. Click OK.
7. The conversion of the files is displayed in the message area.

Configuring display of button

Use the configuration option `gtu_start_loadSaveConverter` to hide the Load Save Converter button  in the segment EDIT-HELP in the GENIUS TOOLS ribbon menu. (Default is 1=On/not hidden)

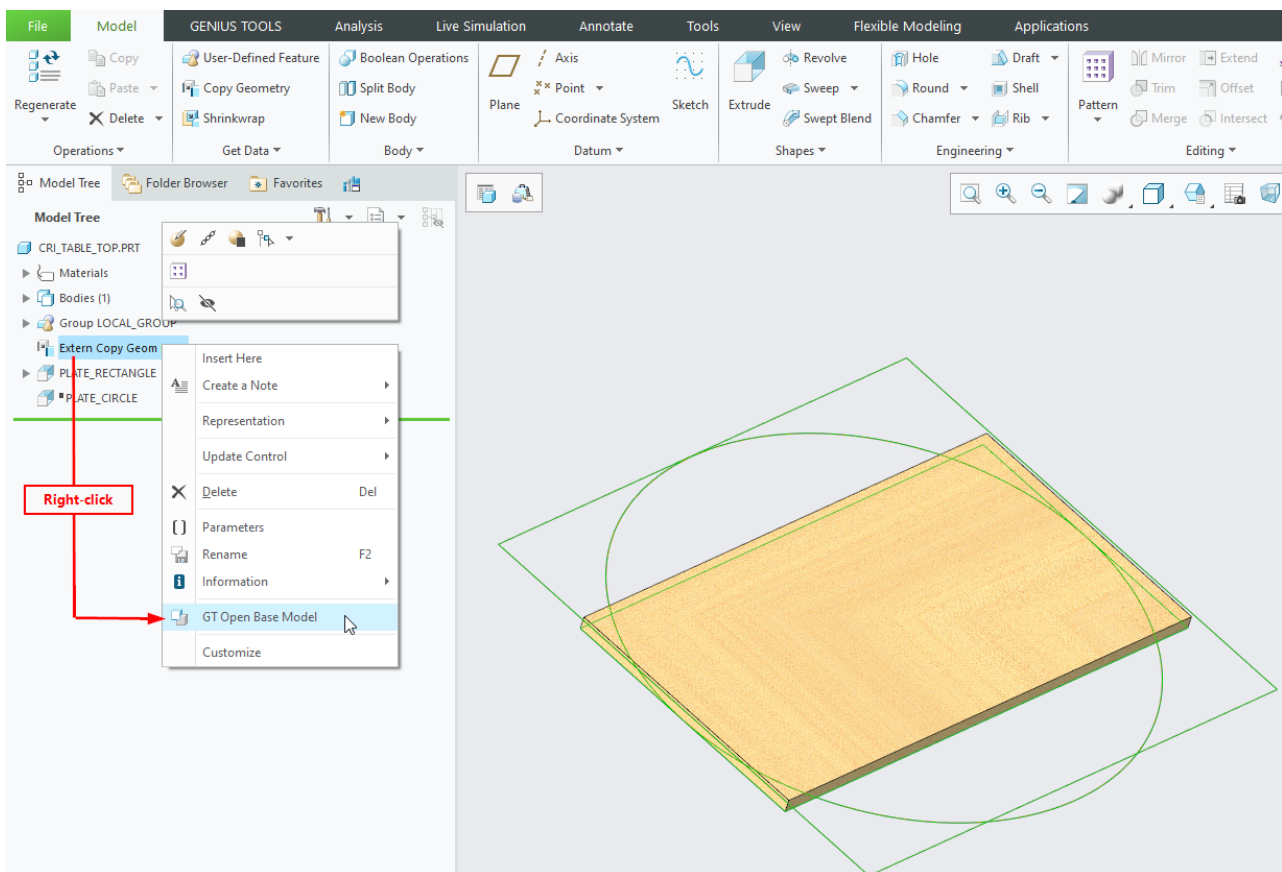
17.19 Open Base Model

This function allows you to quickly open geometric base models that are reference sources for features.

Please note: The function GENIUS TOOLS Open Base Model is only available with subscription licenses for GENIUS TOOLS for Creo.

Starting the program: in part and assembly mode

The function *GENIUS TOOLS (GT) Open Base Model* can be found in the context menu – which opens when right-clicking – on all features that have a reference source. (Here: "Extern Copy Geom")



GENIUS TOOLS Funktion GT Open Base Model in context menu

Configuring display

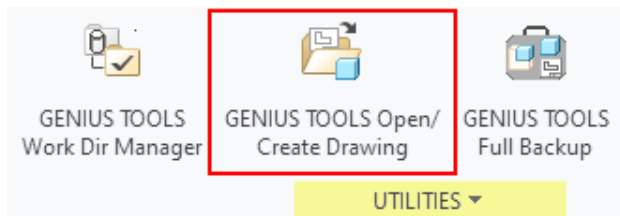
You can switch off the display of the function in the context menu with the configuration option `gtu_start_openGeomOrigin`. (Default: 1=On)

17.20 Open/Create Drawing

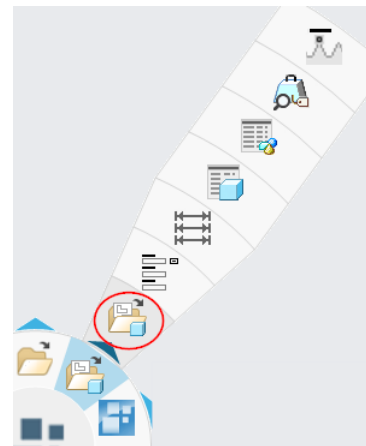
This function opens or creates a drawing of the current model depending on whether a drawing with the model name already exists.

Starting the program: in parts and assembly mode

Start the function via the ribbon menu in the GENIUS TOOLS tab or via GENIUS TOOLS Quick Access ([<] key).



Starting via the ribbon menu



Call-up via Quick Access

The function searches for a drawing named: <PREFIX><Model name><SUFFIX>.drw.*, or like defined in the configuration option `gtu_ord_drw_name`, in the default search paths of Creo Parametric.

Example: `gtu_ord_drw_name = @mdlname@`

The model name is used. (Default setting)

The configuration option `gtu_ord_pre_drw_name` defines the prefix and the configuration option `gtu_ord_post_drw_name` defines the suffix for the file names of the drawings.

Without Windchill with activated REST API or (`gtu_ord_use_pdm_server = 0`):

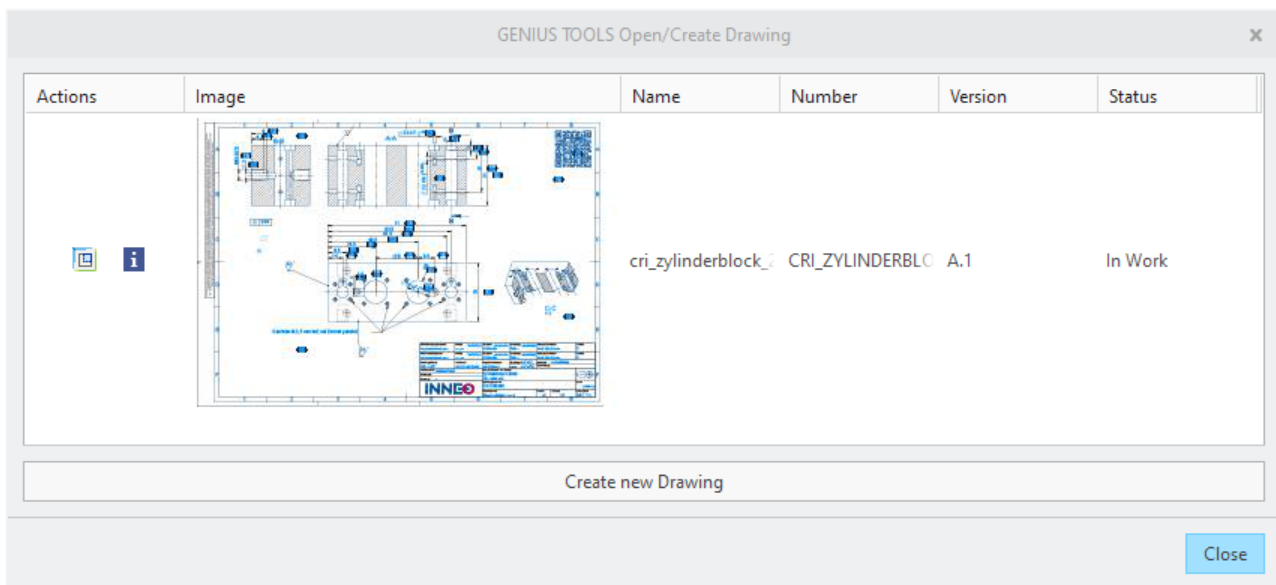
If a drawing with the defined name is found inside the search path, it would be opened. If no drawing is found, the dialog for the creation of a new drawing would be shown. Inside these dialog the file name and other options are already filled.

Example for the file name `gtu_ord_drw_name = $$repl$MO-$DW-$@mdlname@$$`

Within the filename, `mo-` is replaced by `dw-` for retrieval and creation.

With Windchill with activated REST API and (`gtu_ord_use_pdm_server = 1`):

If one or more drawings are found by the Windchill REST API, these would be shown for selection. If not, or by the dialog, the dialog for the drawing creation can be opened. In this dialog the file name and other options are already filled.



Selection of a drawing from Windchill

Example for the configuration `gtu_ord_drw_name = .` It would be used the WT-Number generator for the creation of a new drawing.

Configuration

Provide function in assembly mode before selecting a part

By default, the creates a drawing *after* a part has been selected in an assembly. Set the configuration option `gtu_ord_try_to_use_selected_part_if_inside_asm` to 0 to open the function first and select the part or parts in the 3D model in the second step, i. e. there is an additional dialog "Select 1 element".

Fill dialog with customized mapkey

The creation dialog for a new drawing could be filled / controlled with a mapkey. This could be defined at `gtu_ord_createdrw`. Inside the configuration option % -signs need to be doubled up.

Example: `gtu_ord_createdrw=%%mymapkey).`

Is `gtu_ord_createdrw` not defined, the Mapkey `%createdrw` would be run if defined.


Also refer to the [Configuration](#)⁵⁵⁰ section.

17.21 Select Surfaces by Color

Use this function to select surfaces of the same color or all uncolored surfaces. This works for both body surfaces and quilts. Selected surfaces are highlighted in the model and can thus easily be given a new color or be otherwise modified.

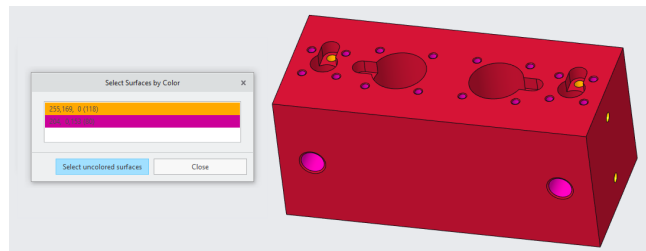
Please note: GENIUS TOOLS Select Surfaces by Color is only available with subscription licenses for GENIUS TOOLS for Creo.

Starting the program: in part mode

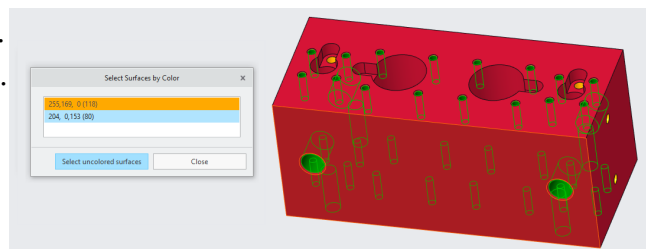
The button  for GENIUS TOOLS Select Surfaces by Color is available in part mode in the segment UTILITIES in GENIUS TOOLS ribbon menu.

Steps

1. Open the dialog Select Surfaces by Color.

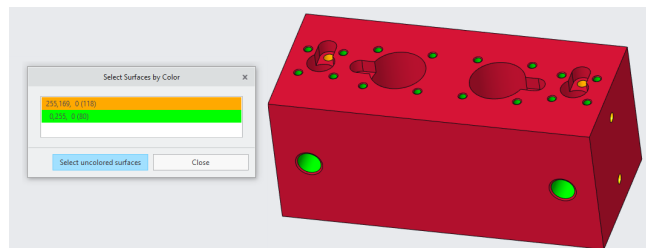


2. Select a color from the dialog. (Here: Pink). The line of the selected color turns light blue. Selected surfaces are highlighted in the model.




Please note: Surfaces with the color of the part (here: red) cannot be selected.

3. In the Creo dialog modify the selected surfaces as usual. (Here: a new color has been chosen in Creo > View > Appearances: green.)

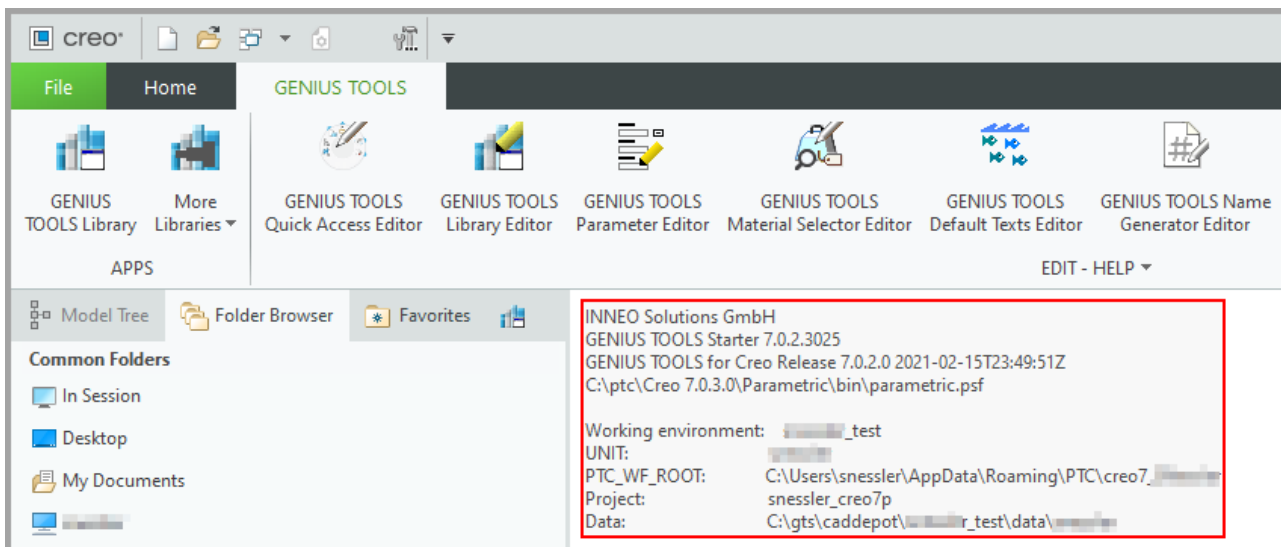


Configuring display of button

Use the configuration option `gtu_start_selectSurfaceByColor` to hide the Select Surfaces by Color button  in GENIUS TOOLS ribbon menu. (Default is 1=On/not hidden)

17.22 Show Information

Textual information can be customized and displayed in the Creo Parametric main window.



Warning: If the Creo configuration option `web_browser_in_separate_window=yes` is set, informative text cannot be displayed in the main window until Creo version 6. As of Creo version 7 icons can be displayed in a separate main window.

Configuration

The following configuration options set up the information on display.

gtu_ui_change_show_info

Switches the function on. (Default is 1=on)

gtu_ui_change_show_info_text

Defines the text to be displayed.

To create a line break, write `
` or `</br>`.

Variables, which are defined by %, are used differently from the standard way described under [Variables](#)⁶⁸⁵. Variables are searched and replaced in the following order. If no value is found in one step, the next step is tried:

- Creo configuration option
- environment variable
- GENIUS TOOLS configuration option
- the variable is not replaced

The replacement happens at GENIUS TOOLS for Creo start. Other variables are replaced at runtime (e.g. `$ENVIRONMENT_VARIABLES` or string replacements). Please also refer to [Configuration of the GENIUS TOOLS for Creo](#)⁶⁵⁴.

Example: `%GT_VERSION_STRING%

%PROE_START%

INNEO Solution GmbH`

gtu_ui_change_show_info_color

Defines the color of the letters. (hexadecimal color code)

gtu_ui_change_show_info_background

Defines the color of the background. (hexadecimal color code)

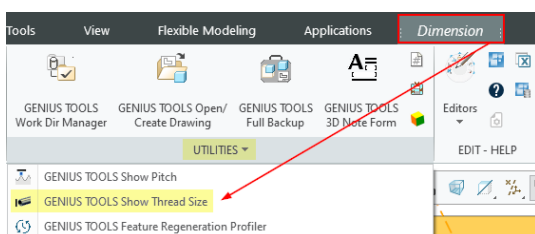
17.23 Show Thread Size

This function extends the displayed dimensions of a bore thread by the pipe thread size. For example, the dimension $\varnothing 15$ becomes G3/8 ($\varnothing 15$), so that the thread size from the file *iso.hol* is prefixed to the numerical value. This works only for bore threads (Type: Standard, Profile: Straight, Tapped).

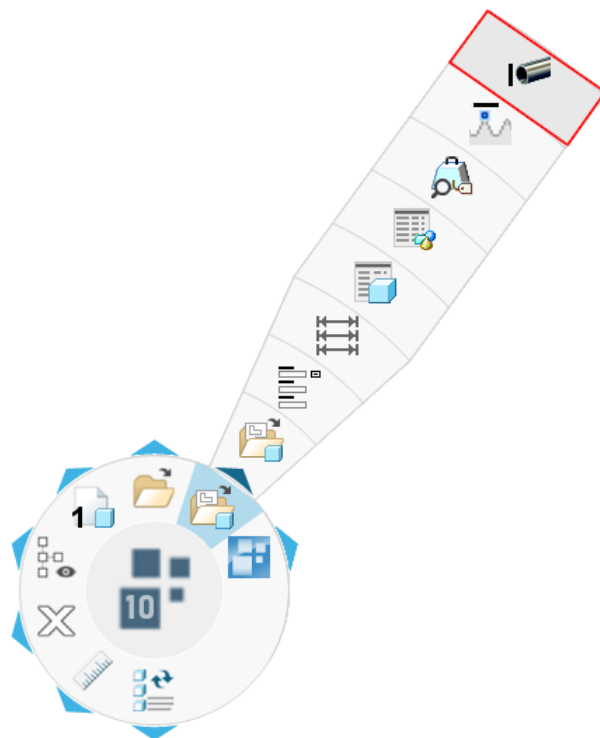
GENIUS TOOLS Show Pipe Thread Size is based on the same operation principle as GENIUS TOOLS Show Pitch⁵⁴².

Starting the program: in part mode (Dimension function)

Select a dimension of a hole thread on a part. Start *Show pitch* via the GENIUS TOOLS ribbon menu or via GENIUS TOOLS Quick Access ([<] key).



Start via the ribbon menu

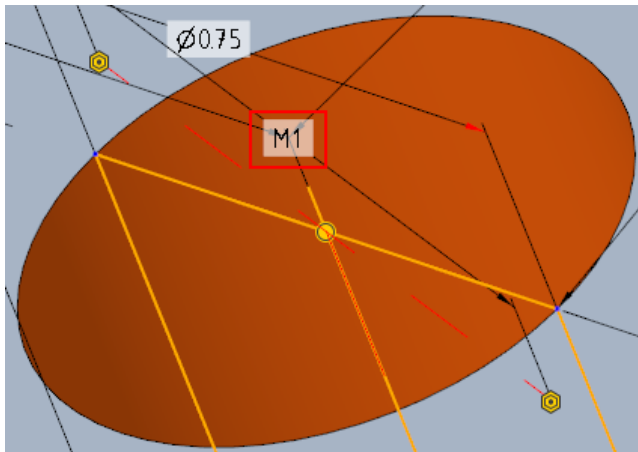


Start with Quick Access ring menu

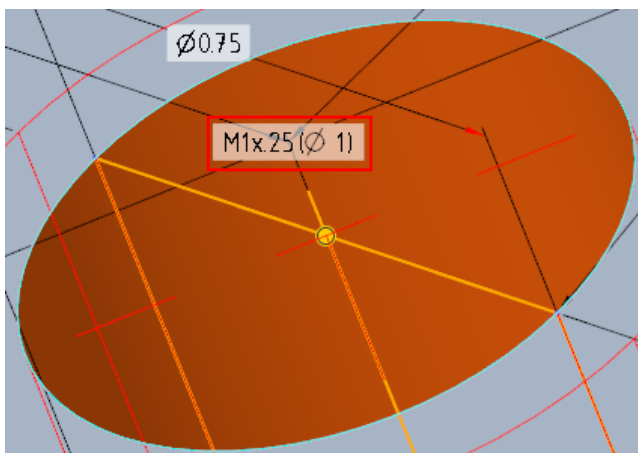
Procedure to show pitch

Precondition: If you start Creo Parametric in English, make sure that `gtu_show_thread_size_text_definition` is set to `SCREW_SIZE`. (See configuration options below.)

1. Select a dimension on which to display the size of the bore thread.



2. Open the function *Show Thread Size*.
3. The selected dimension is directly extended by the thread size.



Deactivating function

You can use the configuration option `gtu_start_show_thread_size` to remove the button from the ribbon menu. (Default: 1=on)

Configuring function

gtu_show_thread_size_check_param

Defines a language-depended feature parameter. If defined, it will be checked if the parameter exists. If the parameter does not exist, the function is unavailable.

gtu_show_thread_size_check_param_fallback

Defines a language-depended feature parameter. If defined, it will be checked if the parameter exists. If the parameter does not exist, the function is unavailable.

gtu_show_thread_size_text_definition

This value has to be customized for "Show Thread Size" to work with the set Creo language.

Please note: The older variable *feat_no* was to *feat_thread_id*, but is still recognized.

gtu_show_thread_size_text_definition_fallback

This value has to be customized for "Show Thread Size" to work with the set Creo language. The fallback is used if *gtu_show_thread_size_check_param* can not be found.

See also chapter [Configuring GENIUS TOOLS for Creo](#)⁵⁵⁰.

17.24 Show Pitch

This function places additional text on a dimension of a metric thread and links the value of the pitch of the thread with it. The value $\varnothing 10$ thus turns into *M10 x 0.75*, whereby the nominal value of the pitch is taken from the properties of the thread and the "M" for metric threads (thread type: ISO) is taken from the *iso.hol* file.

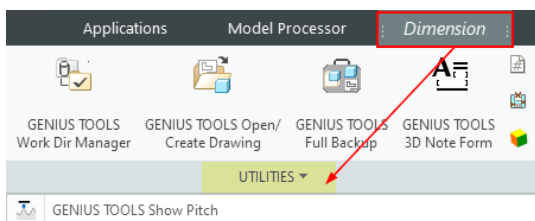
There are two ways of constructing a metric thread:

- separate feature *Cosmetic thread* for internal and external threads
- internal thread as integrated cosmetic thread within a hole feature

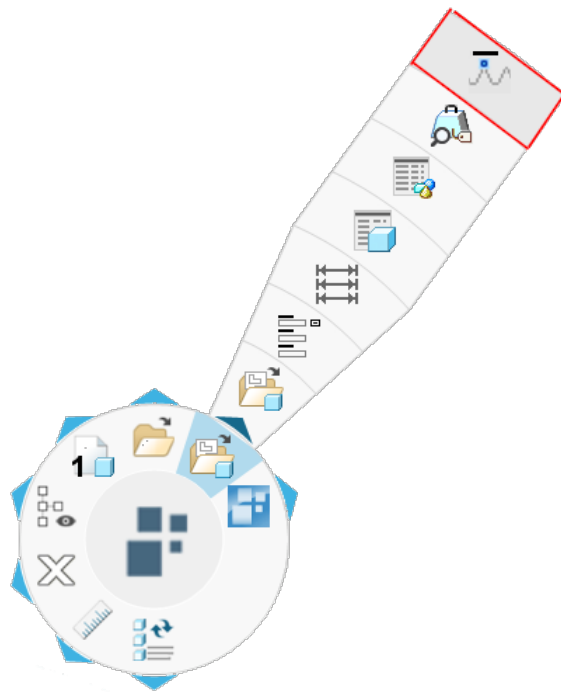
GENIUS TOOLS Show Pitch is based on the same operation principle as *GENIUS TOOLS Show Pipe Thread Size*⁵⁴⁰.

Starting the program: in part mode (Dimension function)

Select a dimension of cosmetic thread feature or a threaded hole in Creo. Start *Show pitch* via the GENIUS TOOLS ribbon menu or via GENIUS TOOLS Quick Access ([<] key).



Start via the ribbon menu



Start with Quick Access ring menu

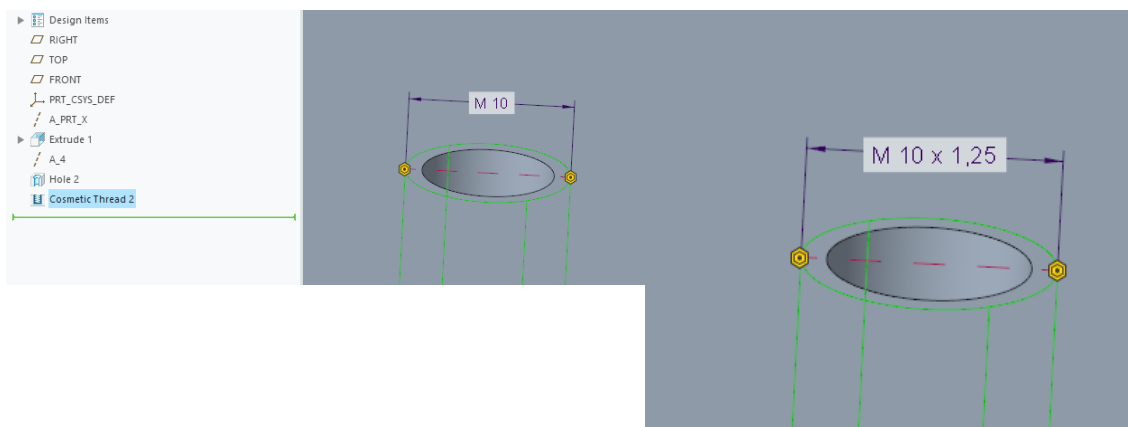
Procedure to show pitch

Precondition: If you start Creo Parametric in English, make sure that `gtu_show_pitch_text_definition` is set to `PITCH`. (See configuration options below.)

1. Select the dimension on which the pitch of a metric thread is to be displayed:
 - for internal threads: dimension of the thread's radius at the cosmetic element, here with $\varnothing 10$.
 - for external threads: dimension at the cylindrical feature
2. Open the function *Show Pitch*.
3. Special case: If multiple outer threads are contained within a volume feature, select a cosmetic thread feature.

Result: The function replaces the dimension by `M @D x &PITCH:@feat_thread_id@[.2]` and thus displays *M 10 x 1.00*

The displayed text can be configured (see below).



Display of the dimension before and after using the function Show pitch on an internal thread of a Cosmetic thread feature

Deactivating function

You can use the configuration option `gtu_start_show_pitch` to remove the button from the ribbon menu. (Default: 1=on)

Configuring function

gtu_show_pitch_text_definition

If Creo Parametric is started in English, this option has to be changed from `STEIGUNG` to `PITCH`. Default: `M @D x &STEIGUNG:FID_feat_thread_id@ [.2]` In a German installation the configuration has to be `STEIGUNG`.

The `feat_thread_id` variable must be used to insert the feature ID.

Please note: The older variable `feat_no` was to `feat_thread_id`, but is still recognized.

gtu_show_pitch_text_definition_fallback

This value has to be customized for GENIUS TOOLS Show Pitch to work with the set Creo language. The fallback is used if `gtu_show_pitch_check_param` can not be found. The default setting is `M @D x &PITCH:FID_@feat_thread_id@ [.2]`

gtu_show_pitch_check_param

Defines a language-dependent feature parameter. If defined, it is checked if the parameter exists. If the parameter does not exist, the function is unavailable. The default setting is `STEIGUNG`.

gtu_show_pitch_check_param_fallback

Defines a language-dependent feature parameter. If defined, it is checked if the parameter is existing. If the parameter does not exist, the function is unavailable. Default is `PITCH`.

See also chapter [Configuring GENIUS TOOLS for Creo](#) ⁵⁵⁰.

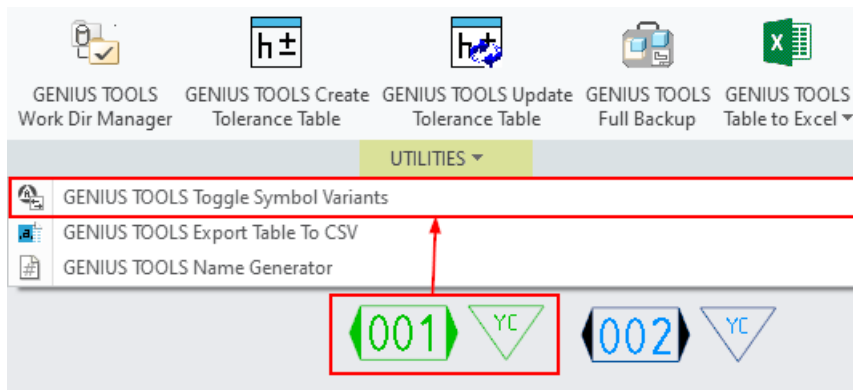
17.25 Toggle Symbol Variants

This function allows you to quickly switch between all variants of a grouped symbol. The order of the variants corresponds to the order in the SYM file that defines the symbol group.

Please note: The function *Toggle Symbol Variants* is only available with subscription licenses for GENIUS TOOLS for Creo.

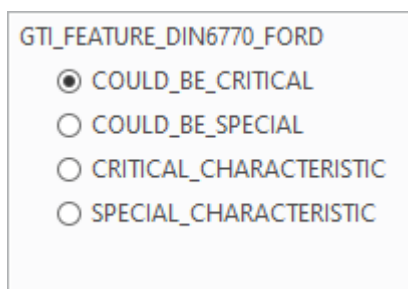
Starting the program: in drawing mode

The button  is located in the GENIUS TOOLS ribbon menu in the segment *Utilities*.



The following conditions must be fulfilled:

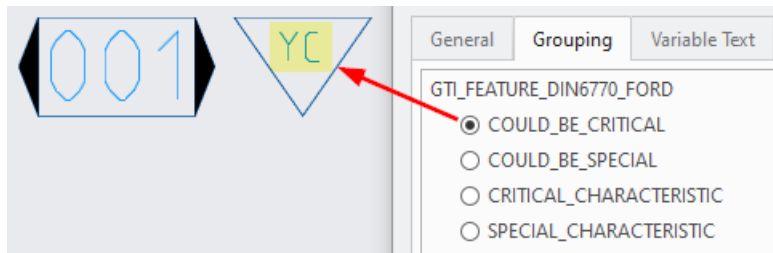
- a symbol must be selected,
- the selected symbol must only have one variant level, i. e. no sub groups. For symbols with sub groups the button is greyed.




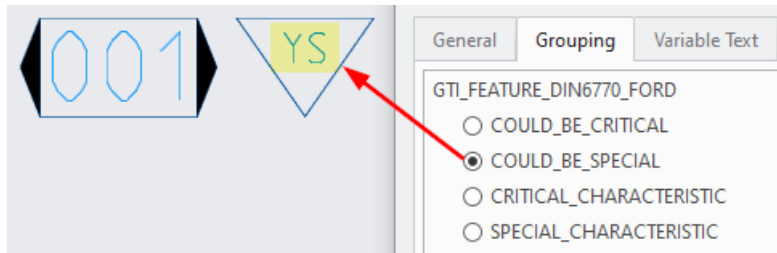
Symbol with one variant level


Example

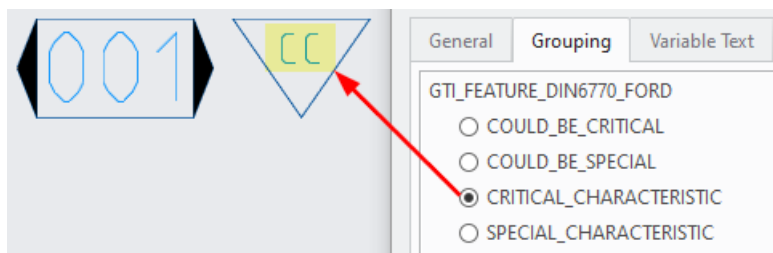
1. Initial state:



2. Click on  to select the next grouping variant:



3. By clicking again on  the next grouping variant is selected:



Configuring display of button


Use the configuration option `gtu_start_toggleSymbolGroups` to hide the Toogle Symbol Variants button in GENIUS TOOLS ribbon menu. (Default is 1=not hidden)

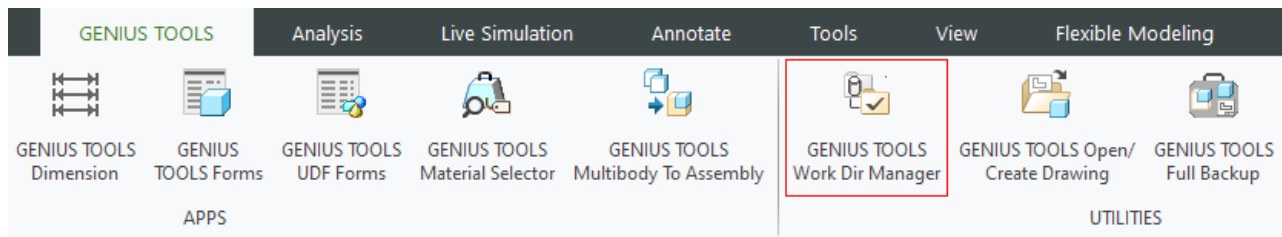
17.26 Work Dir Manager

With GENIUS TOOLS Work Dir Manager you can change the current working directory quickly. The function collects all used directories (outside of WT PDM) automatically during the regular working process.

Please note: GENIUS TOOLS Work Dir Manager is only available with subscription licenses for GENIUS TOOLS for Creo.

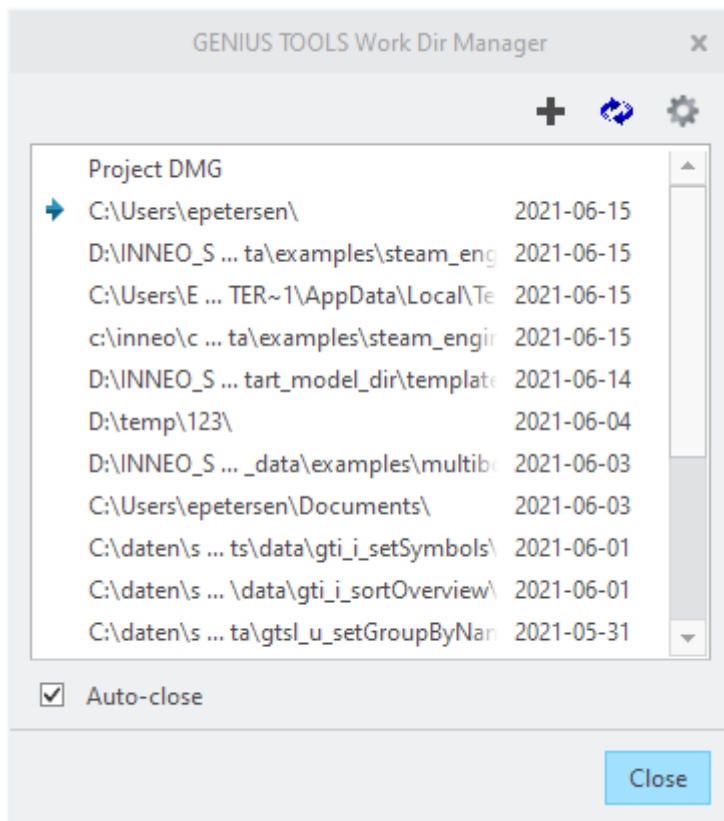
Starting the program: all modes

The function GENIUS TOOLS Work Dir Manager  is found in the segment UTILITIES in the GENIUS TOOLS ribbon menu.



Procedure

The dialog box lists the paths of all directories of the displayed models. These paths are automatically recorded.






GENIUS TOOLS Work Dir Manager Dialog

The current working directory can be changed by double-clicking on another path.

The paths are sorted by the last time stamp. Paths that are not used for a certain time, will be automatically deleted from the list. The time period is defined by the option `gtu_work_dir_manager_autodelete_after_days`. Paths that should not be deleted must be marked as favorites in the context menu.

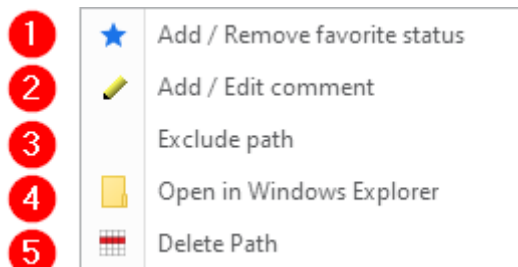
Command bar

Symbol	Name of function	Description
	Add path	Add new working directory. Newly added path is automatically set as the current working directory.
	Reload	Reloads Work Dir Manager
	Edit excluded paths	Opens the CFG file containing all excluded paths, i. e. those that have been permanently removed from being displayed in the Work Dir Manager window (see function 3).

If you check the *Auto-close* box, the dialog box will close automatically after clicking on a path.

Context menu

Open the context menu by clicking the right mouse button in the dialog window.




Context menu with right-mouse click

Function	Name	Description
1	Favorite status	Path will not be deleted.
2	Comment	A comment is shown instead of the path name. Delete the comment again by emptying the input field (empty comment).
3	Exclude path	Deletes the path permanently from the list, so that it will not be added in the future. Excluded paths are written into a CFG file which can be edited with the command bar function <i>Edit excluded paths</i> .
4	Open in Windows Explorer	Opens the selected path in Windows Explorer
5	Delete path	Deletes the path from the list until a model is opened which contains a file with this path.

All settings are saved in the directory defined by the configuration option

`gtu_work_dir_manager_save_path`. (Default: `%appdata%\INNEO\GENIUS_TOOLS\for_Creo\work_dir_manager`)

Configuring display of button

Use the configuration option `gtu_start_work_dir_manager` to hide the GENIUS TOOLS Work Dir Manager button  from the GENIUS TOOLS ribbon menu. (Default is 1=On/not hidden)

The configuration option `gtu_work_dir_manager_always_at_front=1` always switches the dialog to the foreground.

18 Configuring GENIUS TOOLS for Creo

GENIUS TOOLS for Creo can be customized on several levels, i. e. each additional function for Creo Parametric can be adapted differently for various departments (units), projects and users, which guarantees maximum flexibility.

The individual modules are controlled via configuration files (CFG files). These files can be located in different places on the network or on the local system. They are read in according to a defined logic when Creo Parametric is started. This allows you to implement many different configurations.

In this section you will find information about the [startup mechanism and the configuration concept](#)^[550] as well as an [overview of all configuration options of the individual modules](#)^[553] and their control via the graphical user interface of GENIUS TOOLS Config Utility^[558].

In addition, you will find information about the [JavaScript Editor](#)^[577], which can be called from various modules.

18.1 Configuration and start

The extension modules of GENIUS TOOLS for Creo are controlled in configuration files with the file extension *.cfg*, which can be created in different layers.

Configuration files

A configuration is stored in a main configuration file (*gt_main.cfg*) and in a file for module settings (*gt_modules.cfg*).

Information in the configuration files can be commented with a semicolon.

Main configuration file

The file *gt_main.cfg* contains

- start switches of the individual modules, e. g. *gt_start_assembly*
- start switches of editors for individual modules, e. g. *gt_start_library_editor*
- options for setting the project and unit directories, see [section below](#)^[553].
- further configuration options starting with *gt_*, e. g. *gt_lang*

```
*** gt_main.cfg ***
```

```
gt_conf_unit=default
gt_conf_project=project
gt_resource_folder=R:\cad\creo\gt_resource_folder
gt_licpath=7766@cadserver
```

Information for project and unit directory, GT resource folder and license server

```
gt_start_assembly=1
gt_start_library=1
gt_start_material=1
gt_start_name_generator=1
gt_start_parameter=1
gt_start_quick_access=1
```

Start switch for various modules of GTfC: access allowed

```
*** Not in use/Nicht verwendet: ***
```

```
;gt_start_library_editor=0
;gt_start_material_editor=0
;gt_start_name_generator_editor=0
;gt_start_parameter_editor=0
;gt_start_quick_access_editor=0
```

Commented start buttons for different editors of the modules

Example of a gt_main.cfg file

Module configuration file

The *gt_modules.cfg* file contains the configuration options of the individual modules, i. e. configuration options starting with *gt**, e.g. *gta_export_file* for GENIUS TOOLS Assembly Report.

```
*** gt_modules.cfg ***
```

```
gta_clear_pos_parameter=1
gta_default_file=R:\cad\creo\gt_resource_folder\gta_single_level.xml
gta_export_template=R:\cad\creo\gt_resource_folder\gta_single_level_de_en.xlsx
gti_excel_template=gti_inspection_template_de_en.xlsx
gti_revision_excel_template=gti_revision_template_de_en.xlsx
gti_start_file=gti_inspection.xml
gtl_downsync=0
gtl_detail_window_udf_forms_height=1024
gtl_home_db=home.db
gtl_img_size=40
gtl_show_object_names=0
gtl_start_db=home.db
```

Set configuration options for

gta - GENIUS TOOLS Assembly Report

gti - GENIUS TOOLS Inspect

gtl - GENIUS TOOLS Library

Example of a gt_modules.cfg file

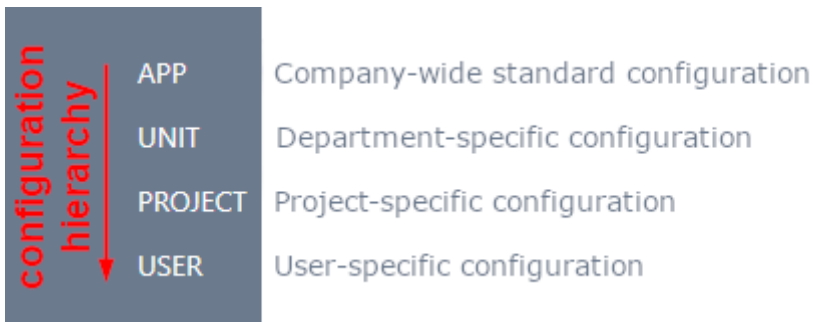
Configuration concept

For a multi-layer configuration, the configuration files *gt_main.cfg* and *gt_modules.cfg* can be stored at different locations in the network. Such a configuration should be used if employees of a project or department are to be given configurations that differ from other projects or departments.

GENIUS TOOLS for Creo provides four different layers for a multi-layer configuration. Which folders you use for this can be controlled through the use of environment variables.

The APP layer represents the company-wide standard configuration of the software. It can be extended, modified and adapted by separate settings in the layers UNIT (department-specific), PROJECT (project-specific) and USER (user-specific). The layers are arranged hierarchically in ascending order. A set value in a subsequent layer overwrites the set value

in the previous layer. For example, if a different directory is set in the user settings (USER) than under PROJECT, the user's directory is used.



Read-in of a multi-level configuration

This section describes the path specifications that apply to working without Startup TOOLS. If you work with Startup TOOLS, consult the section at the end of the chapter.

APP

All values under APP are the basic configuration. This configuration is overwritten each time GENIUS TOOLS for Creo is updated. Data from the default path will be overwritten with the Startup TOOLS during an update. An information dialog box appears with GENIUS TOOLS for Creo.

Default path: `<GTfCInstallationDirectory>\conf`

The environment variable `gt_conf_dir` can be used to redirect the first configuration directory (APP) to another defined path.

UNIT

Under UNIT department-specific configuration settings can be defined.

Default path: `<gt_resource_folder>\configuration\units\default`

The default software start path can be overwritten with the `gt_conf_unit` configuration option.

PROJECT

All configuration settings defined under PROJECT are a project-specific configuration.

Default path: `<gt_resource_folder>\configuration\projects\default`

The default software start path can be overwritten with the `gt_conf_project` configuration option.

USER

User-specific configuration settings are defined under USER.

Default path: `%appdata%\INNEO\GENIUS_TOOLS\for_Creo\configuration`

The default start path of the software can be overwritten with the configuration option `gt_conf_user`.

Define project and unit directory

If you want to change the start paths to the directories of the project and unit layers, the specifications must be made in the general configuration file (*gt_main.cfg*). The specifications should only be contained in the *gt_main.cfg* file of the APP layer.

Path changes are made using the configuration options *gt_conf_unit* and *gt_conf_project* mentioned above.

Startup mechanism

GENIUS TOOLS for Creo go through the following startup mechanism to determine the correct configuration.

1. Read configuration file *gt_main.cfg* coming from the APP layer.
2. Determine the to be used directories UNIT, PROJECT and USER from the *gt_main.cfg* file via the configuration options *gt_conf_unit* and *gt_conf_project* and *gt_conf_user*. (These specifications should be made at the APP layer).
3. Determine and set the resource directory to use from the *gt_main.cfg* files under APP, UNIT, PROJECT, and USER using the *gt_resource_folder* configuration option. (Determining and setting the resource directory follow the configuration concept.)
4. Determining the modules to be started via the configuration option *gt_start_** (start switch).
5. Reading in the configuration settings for the individual modules. Reading in the configuration of the individual modules also follows the configuration concept.
6. Start GENIUS TOOLS for Creo with the determined configuration.
7. Evaluating the GENIUS TOOLS licenses via the *gt_licpath* configuration option.

Reading in outdated CFG files

Up to version 6 of GENIUS TOOLS for Creo, there was a separate CFG configuration file for each module. These are still read in for compatibility reasons. By using the editors in the modules, CFG files are automatically transferred to the new configuration structure, i. e. only *gt_main.cfg* and *gt_modules.cfg* exist in this case.

Alternatively, GENIUS TOOLS Configuration Utility can be called. The first save process automatically adapts to the new structure.

The old CFG files are read into *gt_modules.cfg* in the following order.

Reading order	Module
1	Library (gt_library.cfg)
2	Forms (gt_forms.cfg)
3	Name Generator (gt_name_generator.cfg)
4	Quick Access (gt_quick_access.cfg)
5	Material (gt_material.cfg)
6	Parameter (gt_parameter.cfg)
7	Dimension (gt_dimension.cfg)
8	Utilities (gt_utilities.cfg)
9	Assembly Report (gt_assembly.cfg)
10	UDF Forms (gt_udf_forms.cfg)

Language configuration

The language of the user interface is defined by the Creo-language. It is German if you have a German Creo, else it is English.

There are different configuration options for the display of content, which determine these for each module. Some of the configuration options are preset with the language of the user interface and can be overwritten later by the configuration files.

Option	Default
gt_lang	UI language
gta_lang	UI language
gtf_def_lang	en, de (ger), fr
gtf_lang	UI language
gti_def_lang	en
gti_lang	UI language

Option	Default
gtl_def_lang	en
gtl_lang	UI language
gtm_db_def_lang	en
gtm_db_lang	UI language
gtp_lang	UI language
gtqa_db_def_lang	en
gtqa_db_lang	UI language
gtu_lang	UI language
gtuf_lang	UI language

The module contents are displayed in the defined language (`*_lang`). If this is not defined in the data to be displayed, the fallback language is used (`*_def_lang`), if available. If this is not available, or defined in such a way that it is also not available in the data to be represented, *en* is fallen back on.

An exception is the `gtf_def_lang` option, which can contain several language abbreviations. These are automatically added when a new form is created in GENIUS TOOLS Forms. The first specification (en by default) determines the fallback language.

For information on the effective range of the configuration options, please refer to [configuration settings](#)⁶⁹⁷.

Working with Startup TOOLS

If you are working with Startup TOOLS, you have installed GENIUS TOOLS Starter (GTS) and in this case you should use the corresponding GTS variables. This ensures that the configuration layers that apply in GENIUS TOOLS Starter largely correspond to the GTfC layers. The GTS environment variables are set by using GENIUS TOOLS Starter.

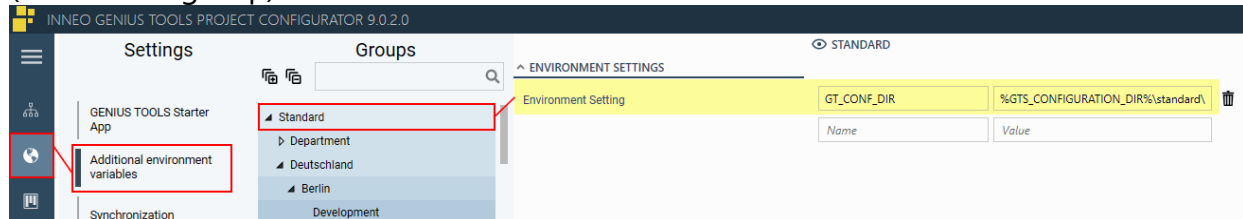
GTS environment variable	Description
GTS_CONFIGURATION_DIR	Points to the selected configuration directory, application-specific as of version 9.0. Path: <Caddepot>\<OperatingEnvironment>\<application>\configuration

GTS environment variable	Description
GTS_PROJECT_DIR	Environment variable for project-specific configurations. Points to the selected project directory, from version 9.0 application-specific. Path: <Caddepot>\<OperatingEnvironment>\<application>\configuration\projects\<ProjectName>
GTS_ROOT_DIR	Specifies the working environment. Path: <Cadpool>\<OperatingEnvironment>
GTS_UNIT_DIR_NAME	Contains the folder name of the last unit of a selected organizational structure or is empty if not working with units.

Procedure:**1. Reset variable for the APP layer**

In the delivery state of the Startup TOOLS, the startup layer is set to `%GTS_ROOT_DIR%\apps\gtfc\conf`. Reset the startup layer with the environment variable `GT_CONF_DIR` to:
`GT_CONF_DIR=%GTS_CONFIGURATION_DIR%\standard`.

The GTS environment variable `GT_CONF_DIR` can be created and defined in the GENIUS TOOLS Project Configurator as an environment variable in the global settings ("Standard" group).



Menu point *Settings* in GENIUS TOOLS Project Configurator

Result: As a result, the APP layer of GTFC is the same as the default configuration of GTS, i. e. the GTS configuration layer *Default* (global settings).

2. Create general CFG file in default directory

Copy the `gt_main.cfg` file from the `%GTS_ROOT_DIR%\apps\gtfc\conf` folder and place it in the default directory `%GTS_CONFIGURATION_DIR%\standard`.

The PROJECT layer and the USER layer of GENIUS TOOLS for Creo correspond to the configuration layer *Projects* and *User* of GENIUS TOOLS Starter. However, the UNIT layer must be adapted.

3. Customize entries for unit directory

In the delivery state of the Startup TOOLS, the unit path in the `gt_main.cfg` is specified as follows:

UNIT path with Startup TOOLS: `%GTS_CONFIGURATION_DIR%\standard`.

Change this to:

`gt_conf_unit=%GTS_CONFIGURATION_DIR%\units\%GTS_UNIT_DIR_NAME%`.

Result: As a result, the UNIT layer of GENIUS TOOLS for Creo corresponds to the last unit configuration layer of GENIUS TOOLS Starter.

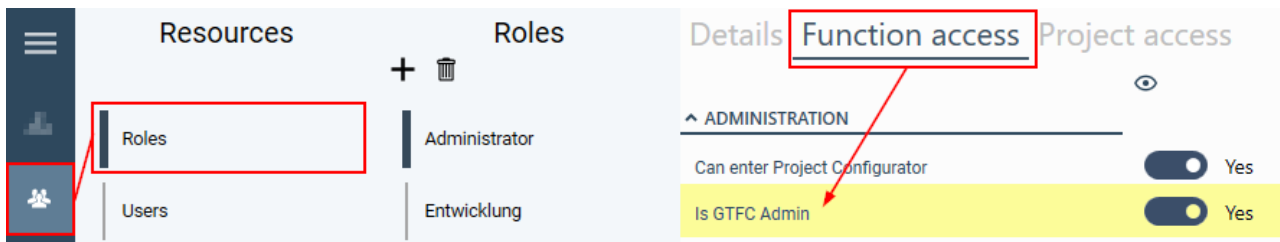
Startup TOOLS variables:

The following variables are created by using GENIUS TOOLS Starter.

The resource folder is defined by the Startup TOOLS with

`gt_resource_folder=%GTS_CONFIGURATION_DIR%\gt_resource_folder`.

For access to the editors of the individual modules, GENIUS TOOLS Starter provides the option that these can only be opened by the administrator. The variable `%GTFC_ADMIN%` is created for this purpose. The setting is made in the Function Access area of GENIUS TOOLS Project Configurator.



The general configuration file *gt_main.cfg* therefore contains several GTS variables.

```

;*** gt_main.cfg ***

gt_conf_project=%GTS_CONFIGURATION_DIR%\projects\%GTS_PROJECT_DIR_NAME%
;gt_conf_project=project
gt_conf_unit=%GTS_CONFIGURATION_DIR%\units\%GTS_UNIT_DIR_NAME%
;gt_conf_unit=%GTS_CONFIGURATION_DIR%\standard
gt_licpath=%GT_LIC_SERVER%
gt_resource_folder=%GTS_CONFIGURATION_DIR%\gt_resource_folder
gt_start_library_editor=%GTFC_ADMIN%
gt_start_material_editor=%GTFC_ADMIN%
gt_start_name_generator_editor=%GTFC_ADMIN%
gt_start_parameter_editor=%GTFC_ADMIN%
gt_start_quick_access_editor=%GTFC_ADMIN%
gt_start_udf_form_editor=%GTFC_ADMIN%

;*** Not in use/Nicht verwendet: ***

;gt_replace_character_if_not_found=
;gt_show_license_dialog=1
;gt_start_assembly=0
;gt_start_assembly_editor=0
;gt_start_assembly_editor=%GTFC_ADMIN%
;gt_start_configuration_utility=0
  
```

Example of a *gt_main.cfg* file with GTS variables


18.2 Configuration Utility

GENIUS TOOLS Configuration Utility allows you to maintain all configuration settings for GENIUS TOOLS for Creo centrally and define the desired locations. Doing so replaces manual editing of the configuration files^[50] (*gt_main.cfg* and *gt_modules.cfg*), a time consuming task.


With GENIUS TOOLS Configuration Utility all entries can be made in a graphical user interface. Adjustable value ranges are specified and incorrect settings are displayed.

Editing outside Configuration Utility is not recommended.

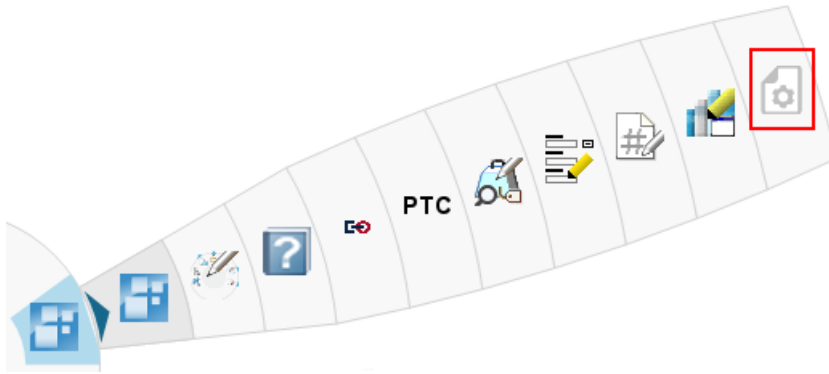
A special highlight is the possibility of using variants of values as comments within the graphical user interface. This makes it possible for administrators to very quickly test different configuration variants without having to manually enter the various settings.

If you want to test the effects of the changes made without restarting the program, click on the function *GENIUS TOOLS Reread Configuration*  in standby mode, see [Directly applying changes](#)^[51].

18.2.1 Starting the program

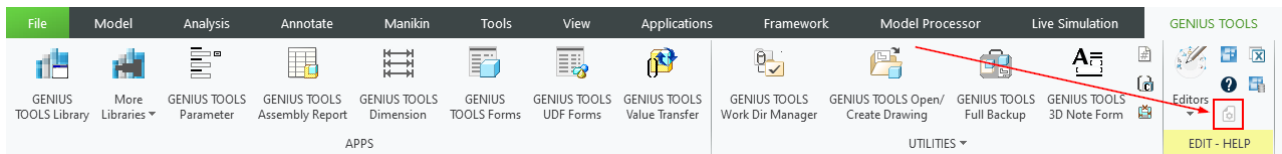
GENIUS TOOLS Configuration Utility can be accessed in various ways via the icon :

1. In all Creo modes: via Quick Access⁴⁰⁵:



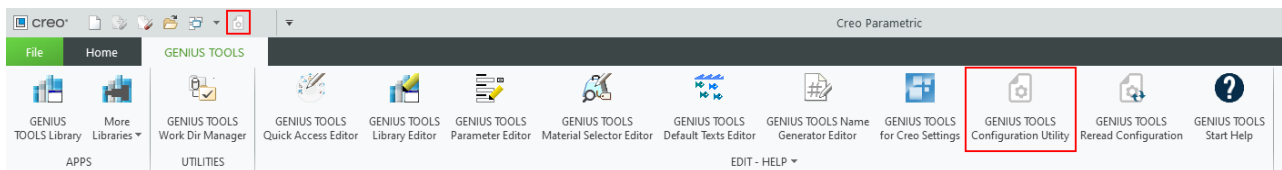
Start via GENIUS TOOLS Quick Access

2. In part, assembly and drawing mode: via the GENIUS TOOLS ribbon menu in the section EDIT-HELP



Start in part, assembly and drawing mode

3. In standby mode: via Creo header
4. In standby mode: via the GENIUS TOOLS ribbon menu in the group EDIT-HELP



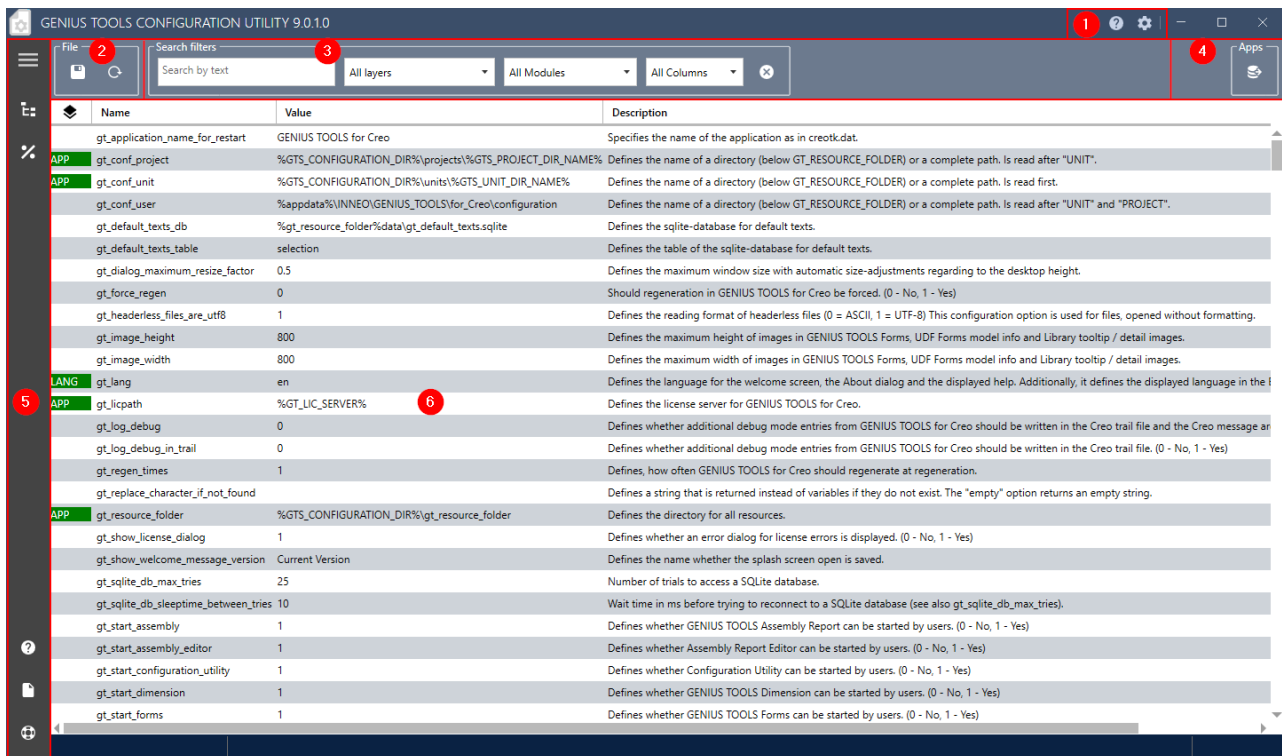
Start in standby mode

Please note: Do not start the Configuration Utility outside a Creo environment, otherwise the environment variables may not be read correctly.

Note the special procedure for saving changes and reading in a configuration during a Creo session, see [Saving changes](#)⁵⁶⁹.

18.2.2 User interface


The user interface of GENIUS TOOLS Configuration Utility consists of the following elements:



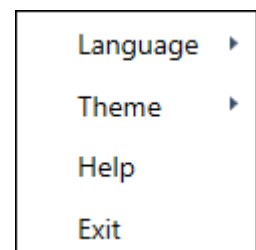
User interface of GENIUS TOOLS Configuration Utility

- Header: Help and user menu
- File: Saving changes⁵⁶⁹ and Reseting configuration options⁵⁷¹
- Search filter⁵⁶²
- Apps: Opens Database Version Control⁵⁷⁶ to check the Creo version status
- Sidebar
- Configuration overview⁵⁶¹: Double click on a line opens the editor for a configuration option.

User menu

The gear icon  opens the user menu with the functions:

- Language: German, English. The language can be changed at runtime.
- Theme: blue, light, dark
- Help: Opens this document.



Sidebar

The hamburger menu  is used to expand the functions:

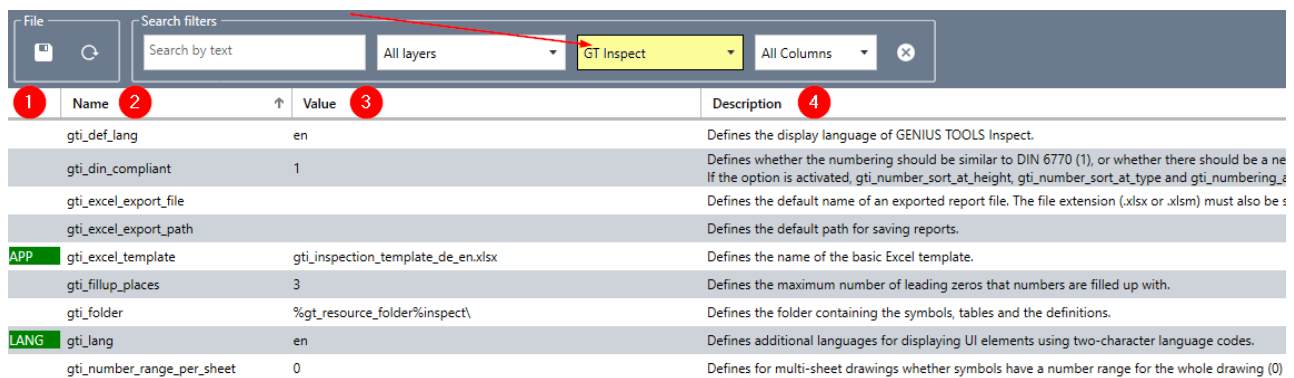
- Path display⁵⁷³ for the levels App, Unit, Project and User
- Environment variables⁵⁷⁴

- Help: Opens this document.
- Log Folder: Opens the folder where the GENIUS TOOLS Configuration Utility log files can be found.
- Support: Opens the INNEO support web page.

18.2.3 Configuration overview

The configuration overview displays all configuration options for the modules of GENIUS TOOLS for Creo.

Double-clicking on a line opens the [configuration editor](#)⁵⁶⁵, in which the desired values are entered for the layers APP to USER. If the preset configuration values have been changed, this can be seen at first glance by the green bar in the first column.




1	2	3	4
Name	Value		Description
gti_def_lang	en		Defines the display language of GENIUS TOOLS Inspect.
gti_din_compliant	1		Defines whether the numbering should be similar to DIN 6770 (1), or whether there should be a ne...
gti_excel_export_file			Defines the default name of an exported report file. The file extension (.xlsx or .xlsm) must also be s...
gti_excel_export_path			Defines the default path for saving reports.
APP gti_excel_template	gti_inspection_template_de_en.xlsx		Defines the name of the basic Excel template.
gti_fillup_places	3		Defines the maximum number of leading zeros that numbers are filled up with.
gti_folder	%gt_resource_folder%\inspect\		Defines the folder containing the symbols, tables and the definitions.
LANG gti_lang	en		Defines additional languages for displaying UI elements using two-character language codes.
gti_number_range_per_sheet	0		Defines for multi-sheet drawings whether symbols have a number range for the whole drawing (0)

Overview of configuration options for GENIUS TOOLS Inspect

The overview is divided into the following columns:

- 1. Layer** indicates configuration options with modified start values by an entry with green background color. The lowest valid layer in which the value was adjusted is displayed, see also [configuration concept](#)⁵⁶¹. Possible values are:
 - **LANG** LANG for *message.txt* files from a subfolder under %GTS_ROOT_DIR%\apps\gtfc\text
 - **ENV** ENV for environment variables
 - **APP** APP / UNIT / PROJECT / USER for the levels
- 2. Name** of the configuration option
- 3. Value** shows the value that is currently valid for the corresponding configuration option. Faulty values are highlighted with red font.

Name	Value
gti_number_sort_at_type	0
gti_numbering_all_sheets	1,5

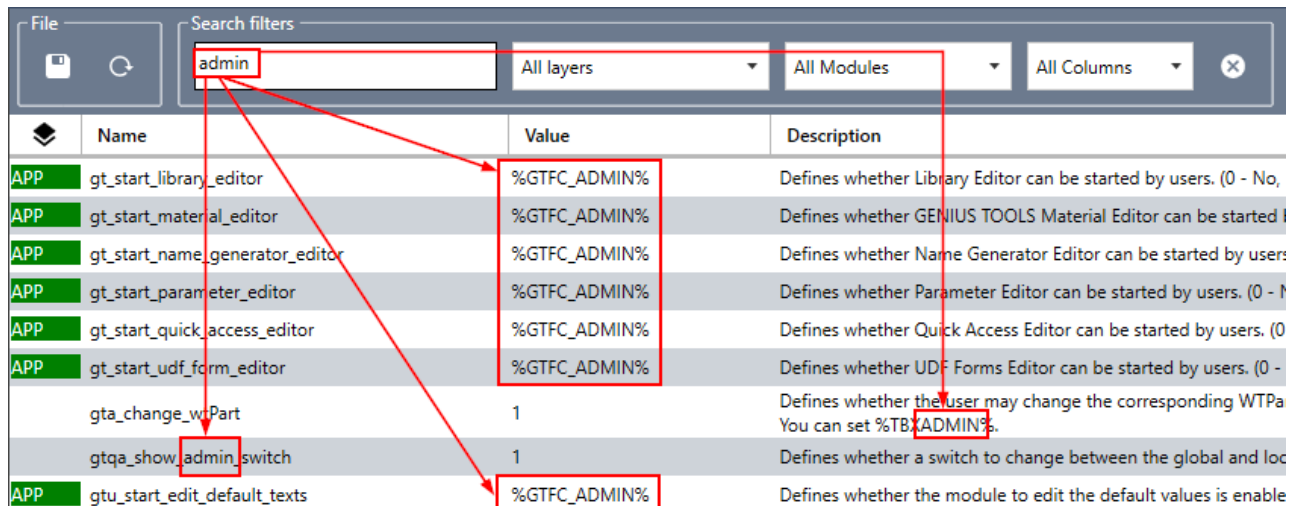
- 4. Description** contains explanatory text for the configuration option. Depending on the language setting, this is available in German or English. In other languages, the English version is used by default. The language can be changed within a session in the **user menu** .

18.2.4 Searching configuration options

You can use the search filter to filter configuration options. The filters can be combined in any way. Configuration options that match all filter rules are displayed as search results.

Searching with text

The text search allows searching the table for character combinations, words or word groups.

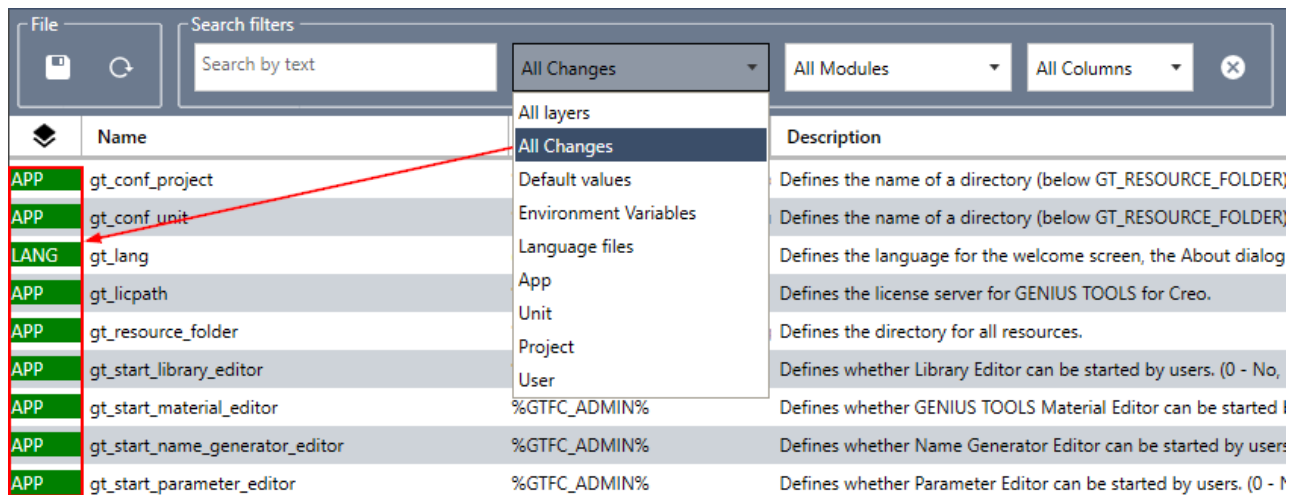


	Name	Value	Description
APP	gt_start_library_editor	%GTFC_ADMIN%	Defines whether Library Editor can be started by users. (0 - No,
APP	gt_start_material_editor	%GTFC_ADMIN%	Defines whether GENIUS TOOLS Material Editor can be started
APP	gt_start_name_generator_editor	%GTFC_ADMIN%	Defines whether Name Generator Editor can be started by users
APP	gt_start_parameter_editor	%GTFC_ADMIN%	Defines whether Parameter Editor can be started by users. (0 -
APP	gt_start_quick_access_editor	%GTFC_ADMIN%	Defines whether Quick Access Editor can be started by users. (0
APP	gt_start_udf_form_editor	%GTFC_ADMIN%	Defines whether UDF Forms Editor can be started by users. (0 -
	gta_change_wPart	1	Defines whether the user may change the corresponding WTPa You can set %TBXADMIN%.
	gtqa_show_admin_switch	1	Defines whether a switch to change between the global and loc
APP	gtu_start_edit_default_texts	%GTFC_ADMIN%	Defines whether the module to edit the default values is enable

Exemplary search for "admin"

Searching by layers

The drop-down menu *All layers* refers to the first column of the option table, where changed configuration options are displayed.



	Name		Description
APP	gt_conf_project		Defines the name of a directory (below GT_RESOURCE_FOLDER)
APP	gt_conf_unit		Defines the name of a directory (below GT_RESOURCE_FOLDER)
LANG	gt_lang		Defines the language for the welcome screen, the About dialog
APP	gt_licpath		Defines the license server for GENIUS TOOLS for Creo.
APP	gt_resource_folder		Defines the directory for all resources.
APP	gt_start_library_editor		Defines whether Library Editor can be started by users. (0 - No,
APP	gt_start_material_editor	%GTFC_ADMIN%	Defines whether GENIUS TOOLS Material Editor can be started
APP	gt_start_name_generator_editor	%GTFC_ADMIN%	Defines whether Name Generator Editor can be started by users
APP	gt_start_parameter_editor	%GTFC_ADMIN%	Defines whether Parameter Editor can be started by users. (0 -

Exemplary search for "All Changes"

All layers does not apply any filter.

All Changes shows only the lines where the value of the configuration option was changed (entry highlighted in green in the first column), regardless of the concrete level.

Default values displays only unmodified basic values.

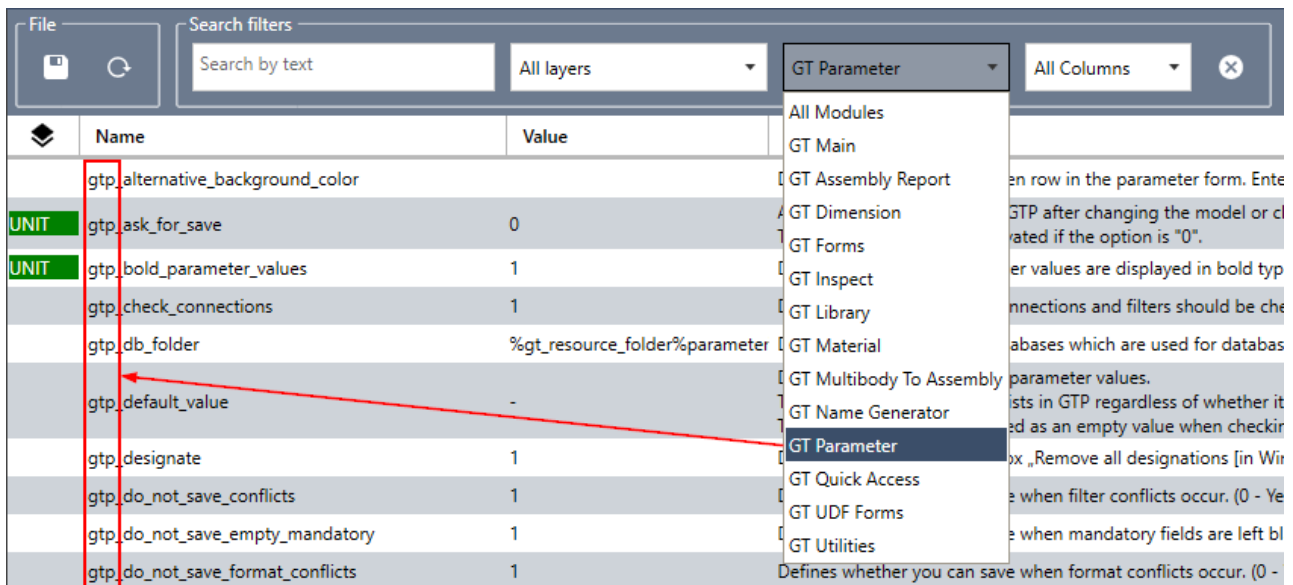
Environment variables shows only lines where the value of the configuration option was last set by an environment variable.

Language files only displays lines where the value of the configuration option was last set by a *message.txt* in a subfolder of *%GTS_ROOT_DIR%\apps\gtfc\text*.

App, Unit, Project, User displays only lines where the value of the configuration option has been set at the selected level.

Searching for modules

The filter of the third search menu shows only configuration options of the selected modules.



Exemplary search for "GT Parameter"

All Modules does not apply any filter.

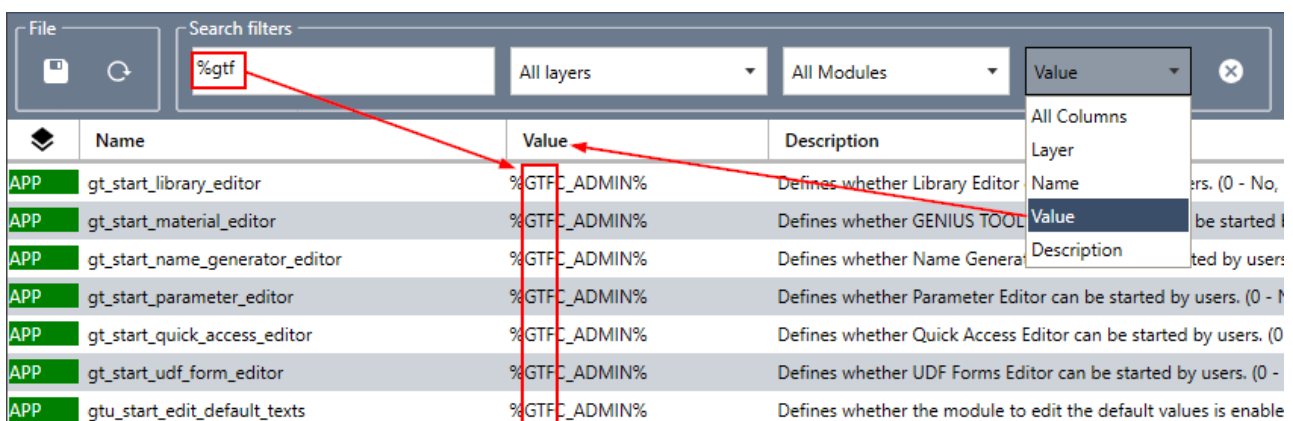
GT Main displays only configuration options that make general, module-wide settings. These start with the abbreviation *gt_* and are written to file(s) *gt_main.cfg*.

GT Assembly Report till GT Utilities displays the configuration options for the corresponding GENIUS TOOLS for Creo module. These can be identified by the respective abbreviation that precedes the name of the configuration option.

Sometimes, configuration options are not retrieved from the actual configuration file due to entry errors or other reasons. GENIUS TOOLS Configuration Utility can be used to detect such deviating entries and correct them if necessary, see [setting values](#)⁵⁶⁶.


Searching for columns

The column filter allows to limit the text search column by column. For example, character combinations can be searched only in the *Value* or *Description* columns - this significantly simplifies the search for frequently occurring characters.



Exemplary search for "gtf" in the column Value

Clearing filters

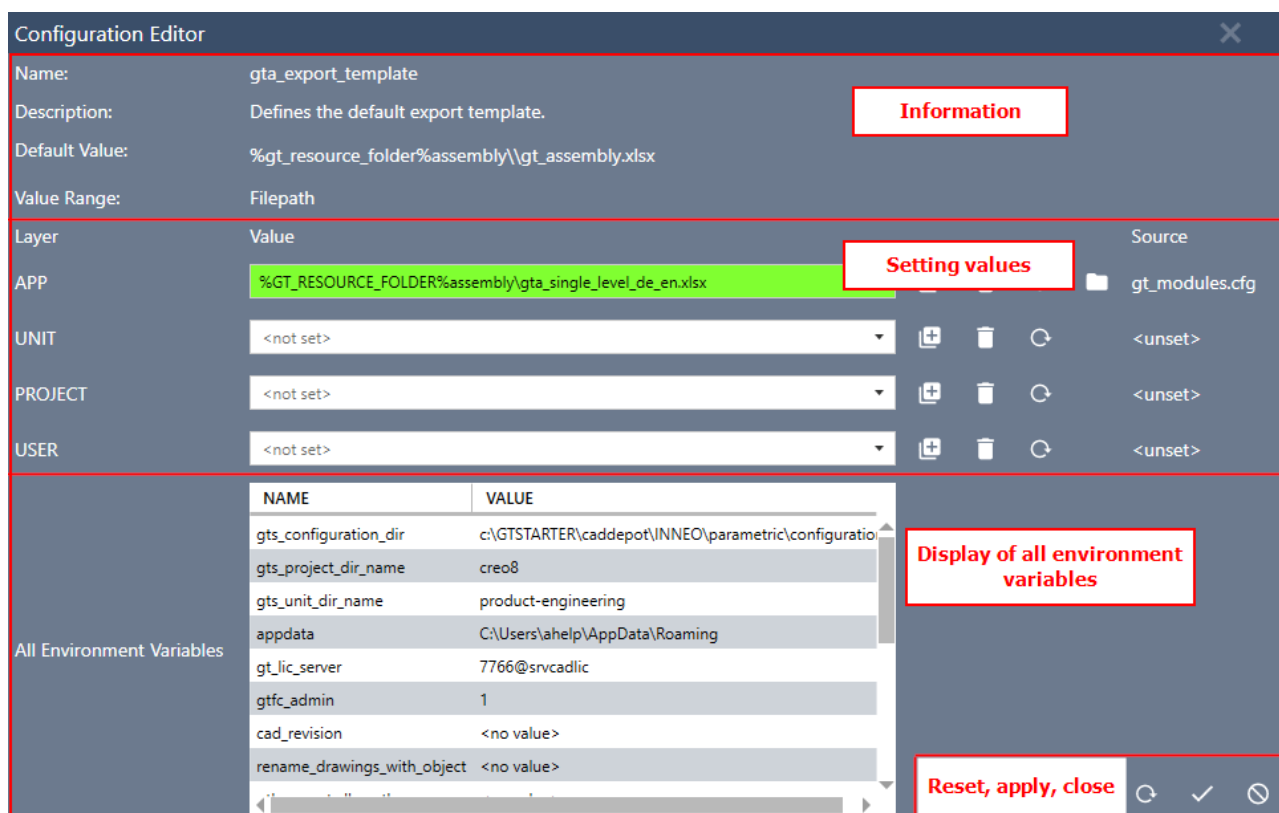
The *Clear all filters* button  resets all fields in the search filter area to their initial value.

18.2.5 Editing configuration options

Double-clicking on a line in the configuration overview opens the configuration editor. Here, individual configuration options for each layer can be viewed, changed, commented and deleted.

All changes made here are later written to the corresponding configuration file during the saving process ⁵⁶⁹.


Please note: Changes must be confirmed with  and then reloaded before they are valid, see [saving configuration options](#) ⁵⁶⁹.



Configuration Editor dialog window

Information: name, description, start value, value range of the selected configuration option.

Set values: values can be set, changed or deleted in the different layers.

All environment variables: The list is for information and does not allow any changes at this point. Changes are made in the window *Environment Variables* ⁵⁷⁴, accessible via  in the sidebar menu.

Reset, Apply or Close: Changes must be confirmed with ☒ and then reloaded before they are valid, see [saving configuration options](#)⁵⁶⁹.

Setting values

In this area, changes can be made to the configuration options on the four layers. (See also [Viewing layers](#)⁵⁷³.)

As soon as values are set in the configuration editor, the configuration editor is displayed as source.

Layer ¹	Value ²	³	⁴	⁵	⁶	Source ⁷
APP	1					gt_modules.cfg
UNIT	text					Configuration Editor
PROJECT	<empty value>					Configuration Editor
USER	<not set>					<unset>

1. Layer: App, Unit, Project and User


Configuration options can be set for each layer individually. The layers are arranged hierarchically in ascending order. A value set in a subsequent line overwrites the value set before it.

2. Value

A writable field in which a new value can be entered. Valid values are highlighted in green, invalid values are highlighted in red. Make sure that you only save configuration options when all lines are highlighted in green or white.

Existing values can be overwritten. You can also [desposit values as comments](#)⁵⁷².

Color	Meaning	Explanation	Impact
	green	valid value	The field was filled according to the value range or as environment variable ⁵⁶⁸ .
	red	invalid value	The field was not filled in according to the value range.
			Invalid values can be saved. Warning: Check whether you really want to save red highlighted values.

Color	Meaning	Explanation	Impact
	yellow	in process	With the 1st click into a white field, it turns yellow. As soon as you enter a value, the field turns green (or red in case of error).
	empty value	A field can be explicitly set as "empty", but this is not always useful, see note ⁵⁶⁷ .	Overwrites the start value and all previous layers. Warning: Check if you really want to save empty values.
white	default	No value set.	Has no effect on other layers.

All changed values are displayed again before saving the configuration, see [Saving changes](#) ⁵⁶⁹.

Please note: It may seem of advantage to set an empty value, e. g. so that the value of the previous layer is overwritten without setting an alternative. As a result, the configuration option is thus written to the corresponding configuration file, as opposed to being deleted. Check whether the affected application can and should process an <empty value> at the corresponding position.

3. Add comment

Button to enter the current value as a comment in the corresponding CFG file during the next saving process, see [writing comments](#) ⁵⁷².


4. Delete app layer

Deletes the current value of the respective layer. Comments are retained.

5. Reload app layer

Restores the original value of the configuration option. Comments are preserved.

6. Open path

If the value range expects a path or folder name, a folder icon  is displayed as soon as an entry has been made on the corresponding layer. A click on the folder symbol opens the entered path.

7. Source

Indicates from which source the current value of the respective layer was obtained. Usually this is a configuration file (*gt_main.cfg* or *gt_modules.cfg*) or the configuration

editor, if the corresponding option has already been opened once at runtime of GENIUS TOOLS Configuration Utility.

Please note: If a path is entered that is located in the *gt_resource_folder* directory, the *%gt_resource_folder%* variable is automatically inserted.

Overwriting layers

By means of the hierarchically structured layers (see [configuration concept](#)⁵⁶¹), you can define values that apply to different areas of your company. The last value set overwrites the layers described before it. To illustrate the principle of operation, here are some examples:

- **Language settings**, e. g. with the configuration option *gt_lang*
 - APP: The corporate language is English. On the layer APP English is entered as language.
 - UNIT: At the location Germany the company language is German, therefore the layer UNIT is additionally defined there with the language German.
 - USER: Within this location, there is a colleague whose native language is Italian and who would therefore like to operate Creo Parametric in Italian. This colleague therefore defines the language Italian for himself at the USER layer.
- **Location libraries**, e. g. with the configuration option *gtl_favorite_button1_liblink*
 - APP: The corporate library is stored on this layer.
 - UNIT: Depending on the subject area that the respective location deals with, there are separate location libraries that are preferred for working with. The corporate library is stored at the second position.
- **Individual working method**, e. g. with the configuration options *gtqa_tooltip_time* and *gtqa_zoom*
 - USER: Zoom of Quick Access / speed of tooltip display adjustable to suit individual working methods.

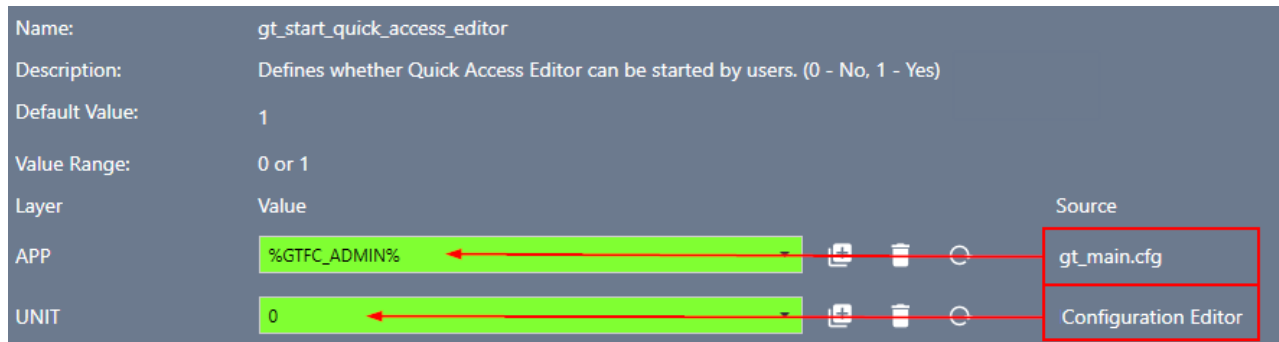
Using environment variables

Alternatively to an input that corresponds to the value range, you can set environment variables so that the validity of the respective configuration option depends on this variable. You maintain the environment variables without having to explicitly customize the configuration options.

The value of a configuration option was set at the app level by the *%GTFC_ADMIN%* environment variable and stored in the *gt_main.cfg* source file because it is a common configuration option with the *gt_start* abbreviation.

The configuration editor assumes with an even number of percentage characters that it is one or more matching environment variables, then evaluates the entry as correct and stores it in green.

On Unit layer the value was set to 0 with the configuration editor.



18.2.6 Saving changes

All value changes are saved in three steps.

1. Apply changes in the configuration editor
2. Save changes on the start page
3. Confirm changes in the save dialog

This sets the changes into the configuration files of the App, Unit, Project and User layers.

Please note: The set values should be saved globally. Make sure that you open GENIUS TOOLS Configuration Utility on the installation computer or that you are in Caddepot.

1. Applying changes in the configuration editor

In the lower right area of the configuration editor are three buttons for exiting. These buttons only affect the values. Comments cannot be deleted or restored with these commands. To edit comments, see [writing comments](#)⁵⁷².



Button for resetting, applying or closing in the configuration editor

1. Reload all (Reset)

Resets all values to the previously set values.
The configuration editor is not closed.

2. Save and close (Apply)

Changed values are written into the configuration overview on the start page and kept

for the saving process.

The configuration editor is closed.

3. Close without saving

Close window and do not apply changed values.

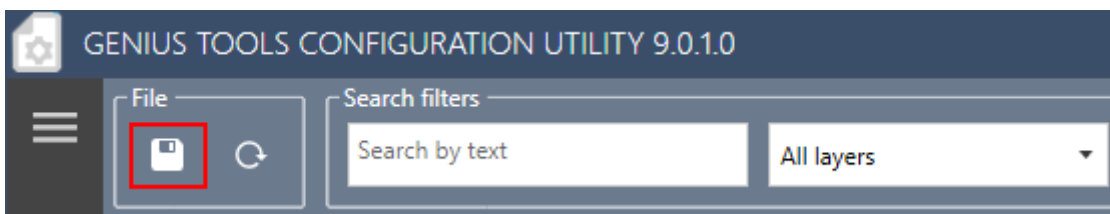
Another window opens where you need to confirm quitting without saving or you can cancel.

Comments will still be retained.

The configuration editor is closed.

2. Saving changes on the start page

Click on the Save icon on the start page to save any changes you have made. The save dialog opens.



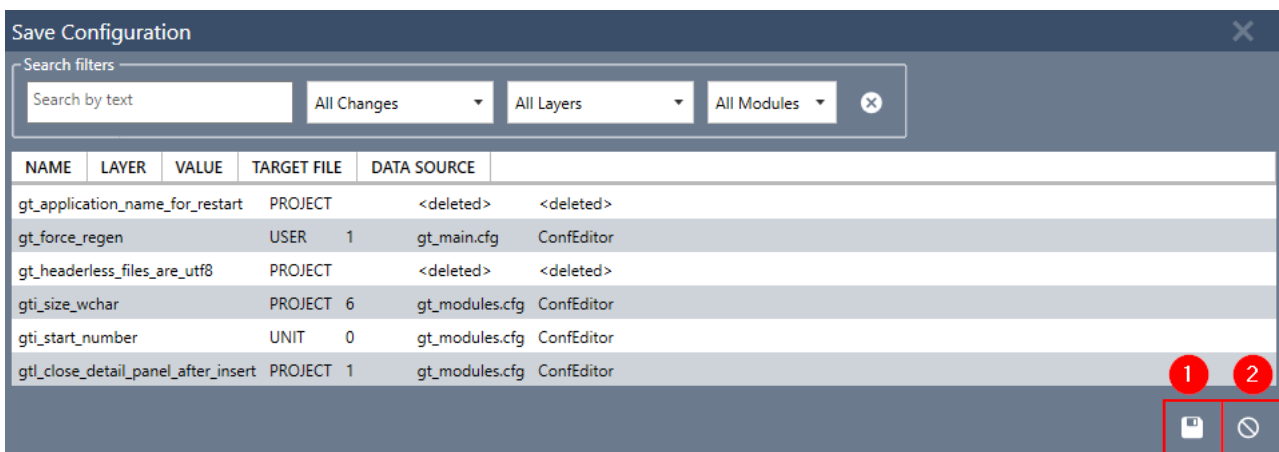
Button for saving the previous configurations on the start page of the user interface

3. Confirming changes in the save dialog

The dialog box *Save Configuration* gives you an overview of all the changes you have made. Here you can decide whether you want to finally confirm these changes.

Changes are shown according to layer and value. The data source indicates whether a configuration has been deleted (<deleted>), modified (*ConfEditor*) or comes from a specified source (e. g. *gt_main.cfg*).

Configuration options where comments have been modified are also listed without naming the comments. Similar to the configuration overview on the start page, the listing can also be filtered in the save dialog, see [Search filter](#)⁵⁶².



Overview of changed configuration options before saving

If you close the Save dialog without saving (2), the changes will still be kept.


To save the changes, click the file icon (1).

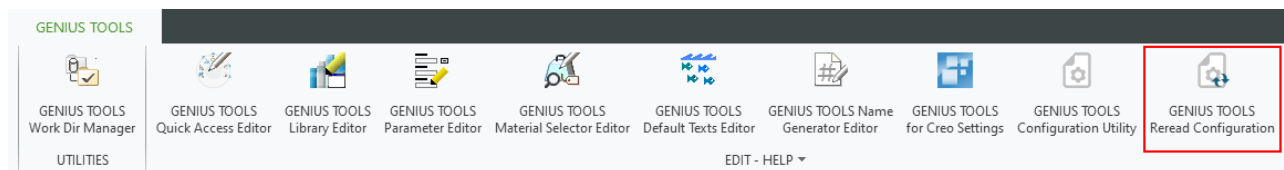
Result: The changed values and stored comments are saved in the configuration files of the corresponding layers. General settings are saved in *gt_main.cfg* files and module specific settings are saved in *gt_modules.cfg* files.

With each saving process, the old file is saved as a previous version in a ZIP file. If available, previous versions of the configurations can be viewed via the files stored there, so that any incorrect configurations can be undone at any time.

The changes are automatically applied the next time the Creo program is started. Alternatively, you can [apply the changes directly](#)⁵⁷¹.

18.2.7 Directly applying changes

If you want to test the effects of the changes made without restarting the program, click on the action *GENIUS TOOLS Reread Configuration*  in standby mode. This command is only available in standby mode so that all applications are closed for the reconfiguration.

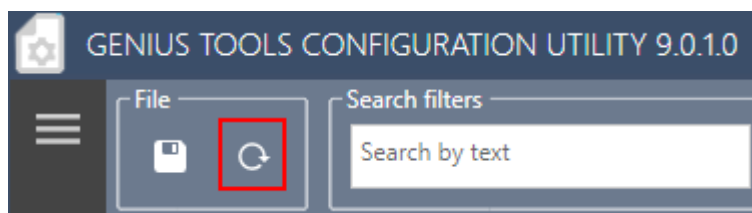


GENIUS TOOLS ribbon menu in standby mode

If you have renamed the auxiliary application *GENIUS TOOLS for Creo* in the file *creotk.dat* (or: *protk_gtfc.dat*), you must change the *gt_application_name_for_restart* configuration option to this name. This is necessary to enable the action *GENIUS TOOLS Reread Configuration*.

18.2.8 Resetting configuration options

To restore the previous configuration, click the button *Reload present configuration* on the start page. This resets all changes made in the configuration editor. The starting point is the last saved configuration.





Previously saved configurations cannot be undone with this command. To view previously saved configurations, you must load the saved ZIP file, see [Configuration files](#)⁵⁷³.

Please note: Changes to the environment variables are not reset during this process. These must be reset manually in the window *Environment variables*⁵⁷⁴. This ensures that any path changes to environment variables can be tested effectively.

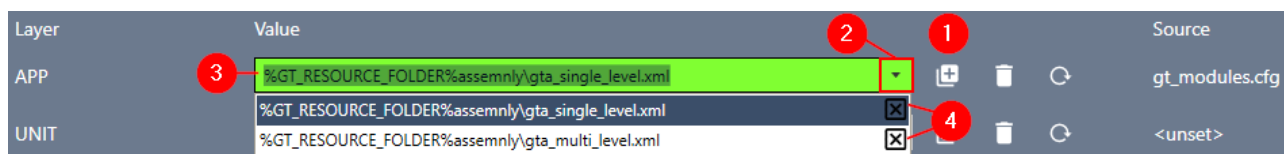
18.2.9 Using comments

By generating comments, you can use variants in the graphical interface and thus quickly test different configuration variants without manually entering the various settings.



Comments

- are commented values in a configuration file,
- have no influence on the GENIUS TOOLS for Creo modules,
- are saved when they are created,
- are separated from the storage process of the values and are available at any time, i. e. they are not deleted by the buttons  and  in the configuration editor.

You need the following elements in the configuration editor⁵⁶⁶ to use comments.



Set values area in the configuration editor


1. The button *Add comment* .
2. The drop-down menu behind the text field of the respective layer.
3. The text field of the respective layer in the column *Value*.
4. The *Delete comment*  button at the right edge of the individual lines of the respective drop-down menu.

Adding comments

A comment is created by a new entry in the text field. This overwrites the current value and thus it is lost. To retain the current value, it is advisable to copy it to the comment field beforehand so that it can be selected again afterwards.

Warning: If a comment is entered in the text field, the current value is overwritten and thereby lost.

Procedure if current value is to be retained:

1. Click on the *Add comment* button  (1). The drop-down menu (2) opens automatically. This shows you that the existing entry has been added as a comment.

2. Enter a new entry in the text field of the *Value* column (3).
3. Click the *Add Comment* button (1) to add the new entry as a comment.
4. Repeat the steps for further comments.

If the original value is to be set again, the *Reload app layer* button  can be operated.

Selecting comments


By selecting a comment in the drop-down menu of a layer, the content of the corresponding line can be set as a new value with one click. *Configuration Editor* is then automatically set as the source.

An exception arises, if the new value at the start of Configuration Utility on this layer was already set before by another configuration file and later overwritten by a value from another source. Then the source of the corresponding predecessor is selected.

Deleting comments


Expand the list of comments in the drop-down menu and click the *Delete Comment* button .

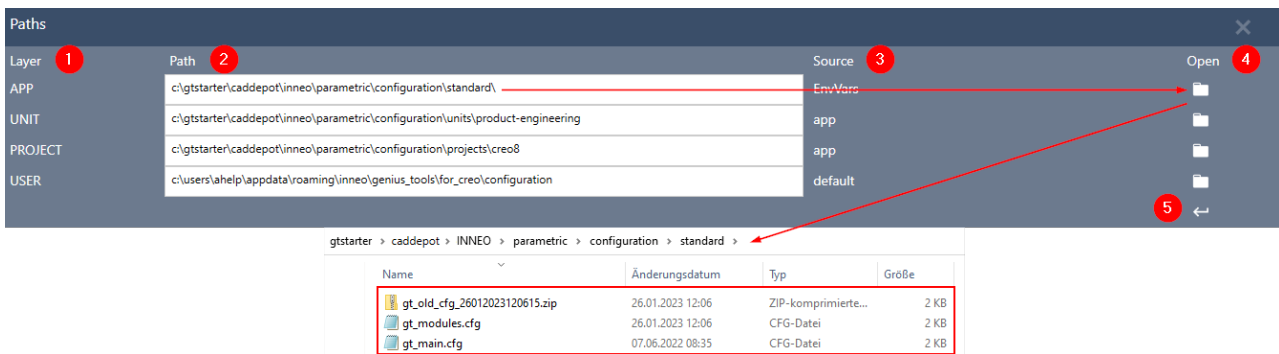
If this is also the current entry, it is automatically removed as well, i. e. the entry is considered an *empty value*.

Resetting the configuration  in the start page also resets the comments to the last saved state.

Please note: Newly created comments are not deleted by closing the configuration editor without saving.

18.2.10 Viewing configuration files and layers



The window *Paths* opens with the operation of the button  on the sidebar of the start page. It shows the complete paths (2) to the individual layers (1) and allows quick viewing and editing of the configuration files (4), i. e. the general configuration file *gt_main.cfg* and the module file *gt_modules.cfg* on the respective layer.




Open the current path for the App layer

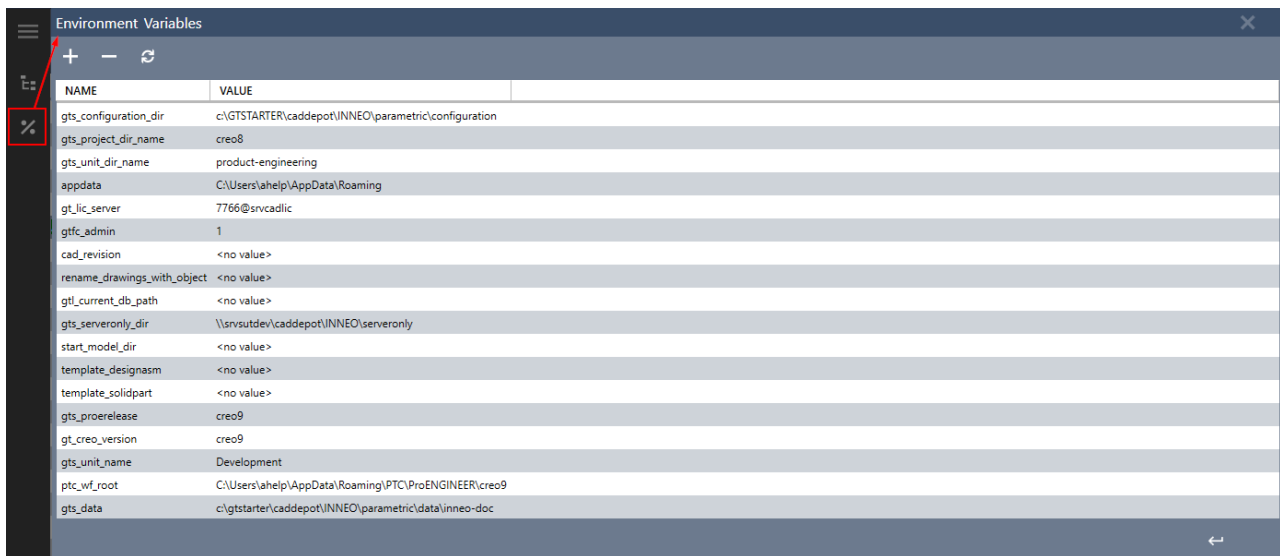
1. **Layer:** For information about the layers *App*, *Unit*, *Project* and *User*, see chapter [configuration and start](#) ⁵⁶⁰.
2. **Path:** Path specifications cannot be modified at this point.
The path specifications for the layers *Unit*, *Project* and *User* can be overwritten in the [configuration editor](#) ⁵⁶⁵ with the configuration options *gt_conf_unit*, *gt_conf_project* and *gt_conf_user*.
The path specification for the App layer is determined from the `%GT_CONF_DIR%` environment variable, see [configuration concept](#) ⁵⁶¹.
3. **Source:** Displays the layer in which the path specification was set. This corresponds to the green entries in the first column of the [configuration overview](#) ⁵⁶¹.
We recommend that, as in the above example, both the unit directory specification (defines the Unit layer) and the project directory specification (defines the Project layer) are defined once in the APP layer, i. e. in the *gt_main.cfg* file in the folder defined under APP.

Please note: If you are working with Startup TOOLS, the App layer should point to the default GENIUS TOOLS Starter folder, see chapter Configuration and start > [Working with Startup TOOLS](#) ⁵⁶⁵.

4. **Open:** The button  opens the folder containing the files *gt_main.cfg* and *gt_modules.cfg* as well as the packed folder *gt_old_cfg_%date%.zip*. after saving.
The ZIP file contains the previous configuration files of this layer, so that any misconfigurations can be undone at any time.
5. **Button** : Closes the window.


18.2.11 Environment variables



The window *Environment Variables* opens with the Percent button  on the sidebar of the start page.



Under *Name* all environment variables are listed which are noted as value in a configuration option. Any entry enclosed by percent signs is interpreted as an environment variable.

Environment variables can neither be created nor deleted in GENIUS TOOLS Configuration Utility.

You can change the values of environment variables to affect paths, for example, which can affect the structure of configuration files and options when you save them. Changed values are applied without a separate save. The changes become valid as soon as the button  is operated after closing the menu item Environment variables in the start display.

Tip: For test purposes, you can add or delete environment variables using the  and  and buttons. Environment variables are simulated with these.

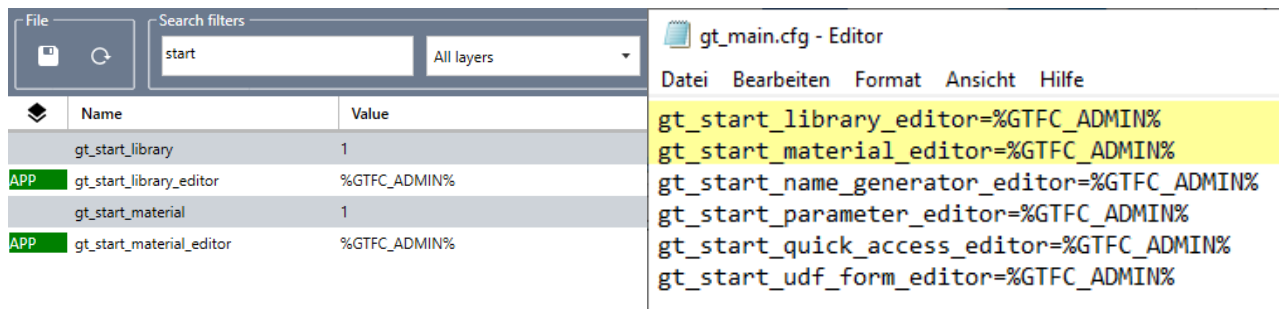
18.2.12 Use cases

18.2.12.1 Set access to programm editors

Access to the editors of the individual modules is controlled by the corresponding start switches. By default, the value is set to "1".

The variable `%GTFC_ADMIN%` is available to set the value depending on the user's authorization.

Example: The settings for GENIUS TOOLS Library and GENIUS TOOLS Material are set to `%GTFC_ADMIN%` at APP level (1) and thus written to the `gt_main.cfg` file located in the *standard* folder (2).



If you work with Startup TOOLS, you can set the variable `%GTFC_ADMIN%` in the GENIUS TOOLS Project Configurator in the menu item *Resources > Roles > Tab: Function Access > Administration: Is GTfC-Admin*. This sets the variable in the configuration options ("start button") of the following Editors: Library Editor, Material Editor, Name Generator Editor, Parameter Editor, Quick Access Editor, UDF Forms Editor.


18.3 Database Version Control

Database Version Control is an administration tool for updating the databases used by GENIUS TOOLS for Creo to the current version that fits the software version in use.

For information on how to use Database Version Control, please refer to [Usage](#)⁵⁷⁶.

For information on the database versions required by the current GENIUS TOOLS for Creo version, please refer to [Database and product versions](#)⁵⁷⁷.



Starting the program


Open Database Version Control from the user interface of the GENIUS TOOLS for Creo Configuration Utility by clicking the icon under *Apps*. 

18.3.1 Usage

Database Version Control checks all SQLite databases in a directory you select and determines whether they are up-to-date for use with GENIUS TOOLS for Creo.

To update your GENIUS TOOLS for Creo databases, proceed as follows.

1. Under *Database folder path*, select your database directory. 
To refresh the list of databases after switching directories, click *Reload*. 
2. Verify the following information in the list of databases:
 - a. Path, name and type of each database
 - b. Current version and new version. These two columns tell you whether the current version and the new version are identical or whether the database can be updated.

- c. Update. This setting determines whether each database will be processed when you start the update function.
Databases for which an update is available are checked by default.
3. Set the *Update* setting for all databases that you want to update.
4. Start the update function by clicking *Update*. 

18.3.2 Database and product versions

GENIUS TOOLS for Creo uses SQLite databases whose version has to fit the version of the software that you have in use. The database versions are numbered independently of the product versions. Not every product version requires a database update.

The following table provides an overview of the database versions required by the current GENIUS TOOLS for Creo version.

GENIUS TOOLS for Creo product version	library	name_generator	material	quick_access	stdTexts
6.0.0.0+	1.8	2.0	1.2	1.1	6.0.0.0

18.4 Javascript

JavaScript is supported in the GENIUS TOOLS components [Forms](#)¹¹⁶, [UDF Forms](#)⁴⁵⁵, [Parameter](#)³⁵⁵ and [Quick Access](#)⁴¹⁵. With the JavaScript Editor, you can develop and test JavaScript code for GENIUS TOOLS for Creo.

This allows you, for example, to check parameter and dimension values before transferring them to a model or to automate calculation and modification of such values.

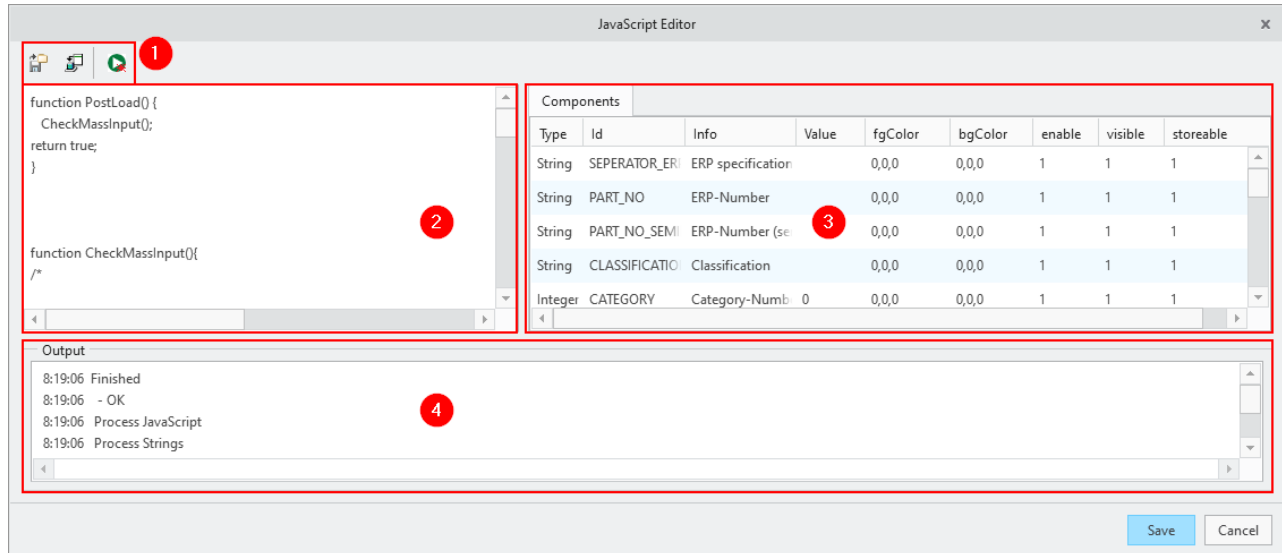
For a practical example of how JavaScript functions can be used, see the application example [Using JavaScript with Parameter](#)³⁹⁶.

18.4.1 JavaScript Editor

The Editor is always started from the respective GENIUS TOOLS for Creo module.

User interface

The JavaScript Editor user interface consists of the following elements:



1. Command bar
2. Input area
3. Component view
4. Output

1. Command bar

The following buttons are included in the command bar:

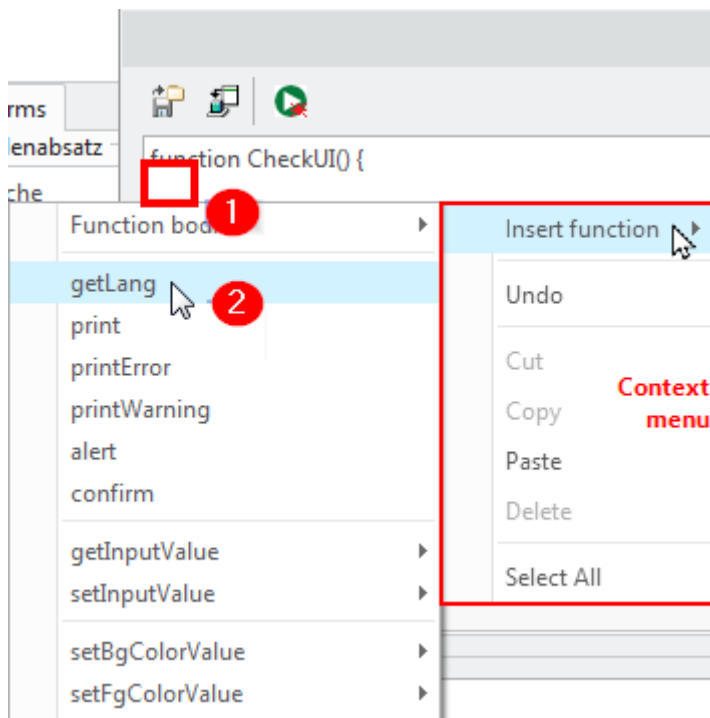
Icon	Name	Description
	Export	Exports the JavaScript as .js file.
	Import	Imports JavaScript from a .js file.
	Execute	Executes the JavaScript code with the given component values and displays the results in the Output dialog window (4).

2. Input area

JavaScript code is edited in the input area.

Use the context menu to insert code snippets for usage in GENIUS TOOLS modules. Right click on the position at which you want to insert a code snippet (1) and select the snippet from the menu (2).

Press CTRL+S to save your changes and close the editor.



Inserting a code snippet from the context menu

Add all code snippets to your program code and export them as .js file. Then open the file in a source code editor to use syntax highlighting and auto complete. Then reimport your source code.

In addition to the implemented functions you have the possibility to insert a function template via *Insert function > Empty Function*, which you can assign with your own function definition.

3. Component view

The component view displays all components available in the current context with their current properties, for example parameters of the current parameter definition or variables of the current UDF definition. Use the arrow icon ▲ to sort components alphabetically or by their values.

Click into a table cell to edit the values manually.

If JavaScript is executed in the Editor, the modifications are displayed in the component view. Execution in the Editor does not affect the real values in the model.

Properties overview

Type: Specifies the current type of a component (variable type).

ID: The name of a component allowing to address it via JavaScript.

Info: The title of a component.

Value: The current value of the component.

fgColor: Displays the current foreground color of the components as RGB values.

bgColor: Displays the current background color of the components as RGB values.

enable: Displays the editability of a component (0 - No, 1 - Yes).

visible: Displays the visibility of a component (0 - No, 1 - Yes).

storable: Displays the storability of a component (0 - No, 1 - Yes).

4. Output

In the Command bar, click *Execute* to receive outputs for the current JavaScript.

18.4.2 List of Javascript funtions

The following table lists the supported JavaScript functions in the individual GENIUS TOOLS modules.

Function	Forms	UDF Forms	Parameter	Quick Access
alert	X	X	X	X
CheckUI	X	X		
confirm	X	X	X	X
creoCurrentMdl NameGet	X	X	X	X
creoDrwActiveM dlNameGet	X	X	X	X
creoListAllMdlsl nSession	X	X	X	X
creoMapkeyAdd ToStack	X	X	X	X
creoMdlExtensio nGet	X	X	X	X
creoMdlMassGe t	X	X	X	X

Function	Forms	UDF Forms	Parameter	Quick Access
creoMdlNameGet	X	X	X	X
creoMdlOIDGet	X	X	X	X
creoParameterValueGet	X	X	X	X
creoParameterValueSet	X	X	X	X
creoSelectAsmComponentGet	X	X	X	X
Empty Function	X	X	X	X
excelOpen	X	X	X	X
excelCellValueGet	X	X	X	X
getEnable	X	X	X	
getEnvVar	X	X	X	X
getInputValue	X	X	X	
getLang	X	X	X	X
getStoreable	X		X	
getStringUI	X	X	X	X
getVisible	X	X	X	
openFileUI	X	X	X	X
PostLoad	X	X	X	
PostLoadFromFile	X	X		
PostSave	X	X	X	
PreSave	X	X	X	
print	X	X	X	X

Function	Forms	UDF Forms	Parameter	Quick Access
printError	X	X	X	X
printWarning	X	X	X	X
readFile	X	X	X	X
replaceVars	X	X	X	X
return	X	X	X	X
runHttpRequest	X	X	X	X
saveFileUI	X	X	X	X
setBgColorValue	X	X	X	
setEnabled	X	X	X	
setEnvVar	X	X	X	X
setFgColorValue			X	
setInputValue	X	X	X	
setStoreable			X	
setVisible	X	X	X	
writeFile	X	X	X	X

18.4.3 Explanations and examples

JavaScript function	Example
alert Outputs an alert box/dialog box with text. It has to be confirmed with "OK".	<pre>alert("Please check your input!")</pre>
CheckUI Will be executed on clicking the associated button of the current GENIUS TOOLS component.	<pre>function CheckUI() { if(getInputValue("InternalDiameter")) >= getInputValue("Diameter")) { return false; } }</pre>

JavaScript function**Example**

```
return true;
}
```

confirm

Outputs a dialog box with text. It has to be confirmed with Yes or No.

Returns the result of the query dialog (Yes=true/No=false).

```
var retConfirm = confirm("Do you want to continue?");
```

creoCurrentMdlNameGet

Reads the name of the current Creo model.

creoDrwActiveMdlNameGet

Returns the active model of a drawing.

creoListAllMdlsInSession

Returns the names of all models in a session.

```
function test() {
array = creoListAllMdlsInSession();
for (i = 0; i < array.length; i++)
print (array[i]);
}
```

creoMapkeyAddToStack

Loads a mapkey from JavaScript and executes it.

Thus mapkeys can be controlled depending on a value.

Note: Use only as PostSave function, because mapkeys usually close windows.

```
function PostSave() {
zd = getInputValue("ZYL_DIAMETER");
if (zd==111) {
creoMapkeyAddToStack("%cd;");
}
return true;
}
```

creoMdlExtensionGet

Reads the file name extension of the current model of the GENIUS TOOLS module.

creoMdlMassGet

JavaScript function**Example**

Reads the mass of the current model.

creoMdlNameGet

Reads the name of the active model from the context of the active GENIUS TOOLS module.

creoMdlOIDGet("ModelName")

Reads the object ID (OID) of a model whose name (including the file name extension) is passed to the function.

For example, the OID of a model can be used in the function **runHttpRequest** in order to access specific objects in a PDM system.

```
creoMdlOIDGet("my.prt")
Usage in an HTTP request:
var mdl = creoMdlNameGet();
var oid = creoMdlOIDGet(mdl);
var request =
runHttpRequest("http://pdm/Windchill/
servlet/rest/objects/"+ oid + "?%
24select=number", "", "1");
var obj = JSON.parse(request);
```

creoParameterValueGet(mdl,"Parameter Name" t/f);

Reads the value of a parameter of a model that has to be specified.

Use t or f (true and false) to determine whether floating point values should be rounded according to the Creo configuration option PARAM_DEC_PLACES (default: 6 digits) (t). If the values are not rounded (f), a maximum of 17 digits (total) will be output .

```
creoParameterValueGet (mdl, "NAME" f);
```

creoParameterValueGet(ParameterName);

Reads the value of a parameter of the current model.

```
var CurMod =
creoParameterValueGet ("DESCRIPTION_1_
EN");
alert('Name of the current model: ' +
CurMod + '');
```

creoParameterValueSet(mdl, ParameterName, Value);

JavaScript function

Writes a value into a parameter of the current model. The return value is the error code.

If the parameter is not in a family table and the function is applied to an instance, the parameter value is written into the generic model.

Example

```
var ret =
creoParameterValueSet("PART_NO",
"Value");

var ret =
creoParameterValueSet mdl, "PART_NO",
"Value");
```

creoSelectAsmComponentGet

Outputs the filename of an assembly component that has to be selected manually in the Creo window.

Empty Function

Creates a function template for defining your own functions. `fname` can be selected and replaced by double-clicking.

```
function fname() {

};
```

excelOpen

Opens an Excel spreadsheet specifying an absolute path and allows access to the individual cells from this Excel spreadsheet.

```
excelOpen("D:\\gstarter\\caddepot\\
user\\parametric\\data\\user\\
\\library_dir\\test_file.xlsx")
```

excelCellValueGet

From the Excel spreadsheet opened with **excelOpen**, this function gets the value from an Excel cell specifying sheet name (optional), column and row.

```
excelCellValueGet(excel, "1", "C",
"3")
```

getEnable

Reads out if a component can be edited.

getEnvVar

Reads the value of a Windows environment variable.

```
var login = getEnvVar("USERNAME");
```

getInputValue

JavaScript function

Outputs the entered value of a component (e. g. parameter, variable).

Example

```
var Offset = getInputValue("z");
```

getLang

Returns the interface language of the current GENIUS TOOLS module.

```
getLang();
```

getStoreable

Returns the storability of a component (Save option of a parameter).

getStringUI("request", "default_value" (optional))

Displays a field for free text entry. Multiple entries are separated with |. In that case, a drop-down menu will be displayed.

```
var text = getStringUI("Screw size", "M1.1x.25|M1.2x.25|M1.4x.3");
```

getVisible

Reads the visibility of a component.

openFileUI("*.extension", path (optional), title (optional), preselected_file (optional))

Opens a window for file selection specifying the file format and file name.

```
var filename = openFileUI("*.xlsx", "", "Select EXCEL calculation file", "");
```

PostLoad

Will be automatically executed after loading the current definition ((UDF) Forms definition).

```
function PostLoad() {
  ...
}
```

PostLoadFromFile

Will be automatically executed after loading a value table (CSV/XML-file). Executed after the function **PostLoad**, if it is defined.

```
function PostLoadFromFile() {
  ...
}
```

JavaScript function	Example
PostSave	
Will automatically be executed in a GENIUS TOOLS component after saving.	<pre>function PostSave() { ... }</pre>
PreSave	
Will automatically be executed in a GENIUS TOOLS component before saving.	<pre>function PreSave() { ... }</pre>
print	
Outputs a status message (green) in the status color code and in the Creo message log.	<pre>print("All changes were saved in the model.")</pre>
printError	
Outputs a error message (red) in the status color code and in the Creo message log.	<pre>printError("Values could not be written into the model!")</pre>
printWarning	
Outputs a warning message (yellow) in the status color code and in the Creo message log.	<pre>printWarning('There are ' + ErrorCount + 'errors in the calculation!')</pre>
readFile	
<p>Reads the contents of a file into a JSON object.</p> <p>var vals = readFile(filename, codex, obj)</p> <p>with:</p> <ul style="list-style-type: none"> – vals: JSON object of the file – filename: the complete path of the file – codex: the coding (0 - std ASCII, 1: UTF8, 2 UTF16 BE) – obj: an error object for better further processing (obj.code: Error code (0: no error) and obj.text: error text) 	<pre>function PostLoad() { var vals = readFile("c:\\temp\\test.txt", 0); var obj = JSON.parse(vals); if (obj.length > 0) { setInputValue("DESCRIPTION_1_EN", obj[0]); } }</pre>

JavaScript function**Example**

can be NULL.

replaceVars

The transferred data is replaced with a GENIUS TOOLS replacement function.

The transferred values are searched for in the following hierarchical order:

1. Creo configuration options (config.pro options)
2. Operating system variables
3. GENIUS TOOLS for Creo configuration options ⁵⁹³
4. GENIUS TOOLS variables ⁶⁸⁵, e. g. user entries ⁶⁸⁵
5. Name Generator ³⁰⁶

```
var t = replaceVars("@mdlpath@ %
TBXCONFIG%");
```

return

Returns a Boolean value. Is required for **CheckUI**, **PostLoad** and **PreSave** to return whether the function has been successfully completed.

```
return true;
return false;
```

runHttpRequest("URL","DATA",1/0)

Executes an HTTP request.

The function accepts the URL and URL parameters (DATA). An additional parameter USERNAME/PASSWORD takes the value 1 or 0 and defines whether the user should be asked for login information or not.

You can use the function **creoMdlOIDGet** in order to be able to access specific objects via their OID.

```
var site =
"http://ServerName/Folder/db_list_gt.
php";

var query =
"DB=pmm_tc.mdb&TABLE=namen&SFIELD=ger
man&FIELDS=german,english&SEARCH=ab*"
;

var ret = runHttpRequest(site,
query,0);
```

JavaScript function	Example
saveFileUI ("*.extension", path (optional), title (optional), preselected_file (optional))	
Returns the file name.	<pre>saveFileUI ("*.extension", "", "OK", "");</pre>
setBgColorValue	
Sets the background color for a GENIUS TOOLS component. Color specification in RGB values. If you do not pass RGB values, but only the component name, the color will be reset to the default row color (white or gray).	<pre>if (x > 1){ setBgColorValue ("z", 0, 234, 50); }</pre>
setEnabled	
Sets whether a component is editable.	
setEnvVar	
Writes the value in a Windows environment variable.	<pre>setEnvVar ("PROJECT_NO", "8004");</pre>
setFgColorValue	
Sets the foreground color of an input field in GENIUS TOOLS Parameter. Color specification in RGB values. If you do not pass RGB values, but only the component name, the color will be reset to the default foreground color.	<pre>setFgColorValue ("z", 255, 0, 0);</pre>
setInputValue	
Sets a component (e. g. parameter, variable) to a value.	<pre>setInputValue ("z", 30);</pre>
setStoreable	
Sets the storability of a component (Save option of a parameter).	

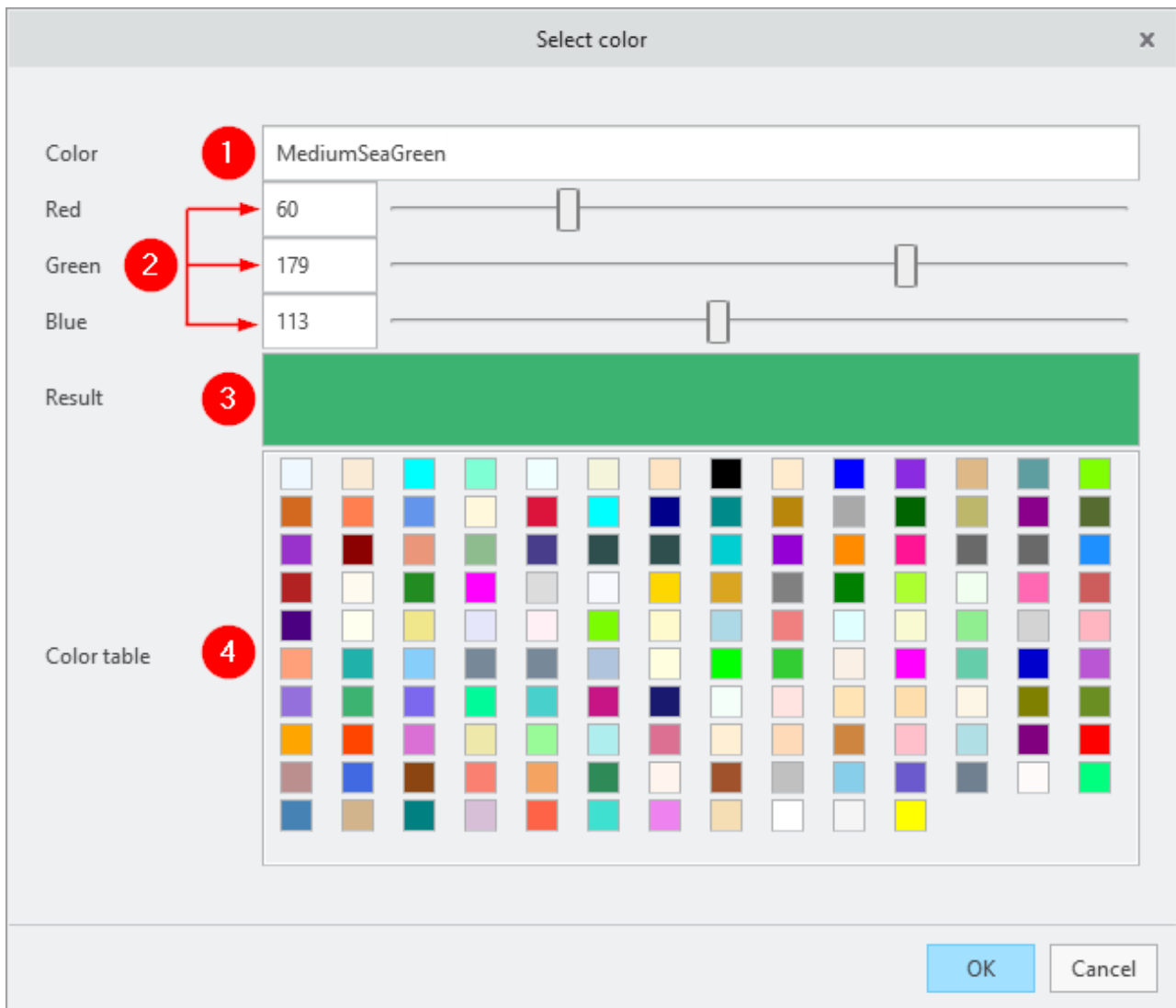
JavaScript function	Example
setVisible	
Sets the visibility of a component, for example a separator.	
writeFile	
Writes a JSON object to a file. var ret = writeFile(filename, jData, codex, obj) With:	<pre>function PostLoad() { var obj = {}; var line0= getInputValue("DESCRIPTION_1_EN"); var data = []; data[0] = line0; var jData = JSON.stringify(data); var ret =writeFile("c:\\temp\\ \\test.txt", jData, 0); }</pre>
<ul style="list-style-type: none"> – ret: "ok" if no error occurred – filename: the complete path of the file – jData: Json object string – codex: the coding (0 - std ASCII, 1: UTF8, 2 UTF16 BE) – obj: an error object for better further processing (obj.code: Error code (0: no error) and obj.text: error text) can be NULL. 	

18.5 Color selector

The color selection dialog is used for the simple determination of background colors in the different GENIUS TOOLS modules.

Predefined HTML standard colors as well as own colors can be used.

The color selection dialog is always called by an editor of one of the GENIUS TOOLS modules.



Farbauswahldialog


- 1 Color When the dialog is started, the color value of the calling GENIUS TOOLS module is written into this field.
 The following are supported:
 - predefined color names (e.g. Red)
 - comma-separated RGB values (e.g. 100,255,16)
 - colors in hexadecimal notation (e.g. #ffff10)
 When the dialog is loaded, the transferred color is analyzed and displayed.
 Changes in this field do not update the color display in the dialog.
 Confirming the dialog transfers the value in this field to the calling editor.
- 2 Red Determination of the color via RGB.

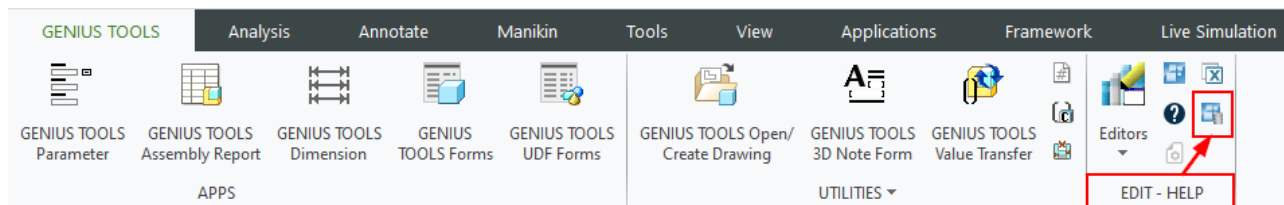
- Green In the input field, the individual color components in the range [0,255] can be defined.
- Blue The sliders also control the individual color components.
- These changes are immediately visible in the surface.
- 3 Result This field displays the resulting color.
- 4 Color table Table with predefined colors that can be selected with a mouse click.

18.6 Display Embedded Data

Embedded Data are information on a model that have been generated by GENIUS TOOLS components Assembly Report, Forms, UDF Forms and Inspect Revision. This data can be displayed and deleted in the current model.

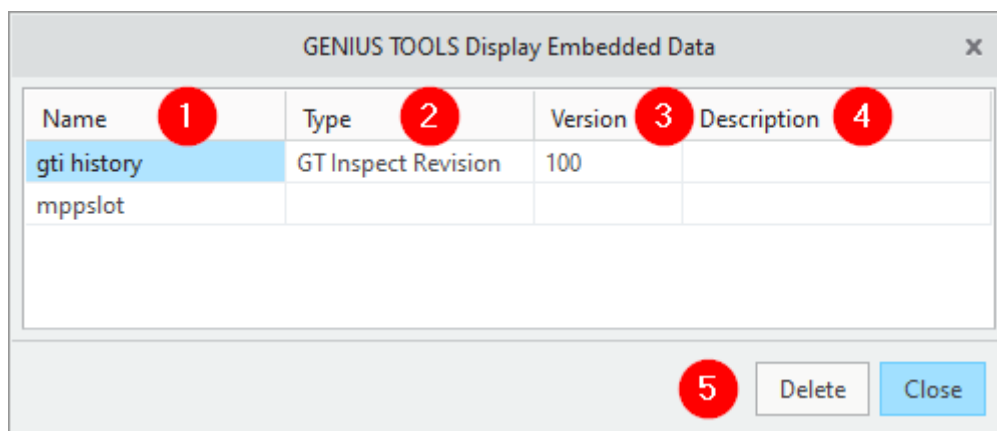
Starting the program: in part, assembly and drawing mode

The button  for the function is located in the segment EDIT-HELP in the GENIUS TOOLS ribbon menu.



18.6.1 User interface

The user interface of the *Display Embedded Data* dialog has four columns.



Overview of embedded data

1. File name (e. g. XML file)

2. Data type: specifies whether the data comes from one of the following GENIUS TOOLS for Creo components:
 - Assembly Report
 - Inspect Revision
 - Forms
 - UDF Forms
3. Version (if assigned)
4. Description text
5. Button to open the *Delete* dialogue

Delete dialog

In the first dialog, select a line for a GENIUS TOOLS for Creo-module by highlighting it and click *Delete* (5). This opens the *Delete* dialog containing the selected line:

Delete	Name	Type	Version	Description
<input type="checkbox"/>	gti history	GT Inspect Revision	100	

Delete embedded data

Tick the check box and confirm by pressing the *Ok* button.

18.7 Configuration Options

This is an overview of configuration options in the GENIUS TOOLS for Creo modules

This overview is created by an automated process.

Please note: Do not use environment variables such as `GT_RESOURCE_FOLDER` or `GT_CONF_USER` in Creo configuration files! These variables are only available after the start of the GENIUS TOOLS for Creo.

General Configuration Options

In the configuration file *gt_main.cfg* the settings for GENIUS TOOLS for Creo are administered.

gt_application_name_for_restart**Options** Any text**Default** GENIUS TOOLS for Creo

Specifies the name of the application as in creotk.dat.

gt_conf_project**Options** Path or foldername**Default** default

Defines the name of a directory (below GT_RESOURCE_FOLDER) or a complete path. Is read after "UNIT".

gt_conf_unit**Options** Path or foldername**Default** default

Defines the name of a directory (below GT_RESOURCE_FOLDER) or a complete path. Is read first.

gt_conf_user**Options** Path or foldername**Default** %appdata%\INNEO\GENIUS_TOOLS\for_Creo\configuration

Defines the name of a directory (below GT_RESOURCE_FOLDER) or a complete path. Is read after "UNIT" and "PROJECT".

gt_default_texts_db**Options** Path or foldername**Default** %gt_resource_folder%data\gt_default_texts.sqlite

Defines the sqlite-database for default texts.

gt_default_texts_table**Options** Table name**Default** selection

Defines the table of the sqlite-database for default texts.

gt_dialog_maximum_resize_factor**Options** Number between 0 and 1**Default** 0.5

Defines the maximum window size with automatic size-adjustments regarding to the desktop height.

gt_force_regen**Options** 0 or 1**Default** 0

Should regeneration in GENIUS TOOLS for Creo be forced. (0 - No, 1 - Yes)

gt_headerless_files_are_utf8**Options** 0 or 1**Default** 1

Defines the reading format of headerless files (0 = ASCII, 1 = UTF-8) This configuration option is used for files, opened without formatting.

gt_image_height**Options** Any number to up to 1200**Default** 800

Defines the maximum height of images in GENIUS TOOLS Forms, UDF Forms model info and Library tooltip / detail images.

gt_image_width**Options** Any number to up to 1200**Default** 800

Defines the maximum width of images in GENIUS TOOLS Forms, UDF Forms model info and Library tooltip / detail images.

gt_lang

Options Language abbreviation (en, de, es, ...)

Default de

Defines the language for the welcome screen, the About dialog and the displayed help. Additionally, it defines the displayed language in the External Model Data Viewer.

gt_licpath

Options Port@IP

Default 7766@localhost

Defines the license server for GENIUS TOOLS for Creo.

gt_log_debug

Options 0 or 1

Default 0

Defines whether additional debug mode entries from GENIUS TOOLS for Creo should be written in the Creo trail file and the Creo message area. (0 - No, 1 - Yes)

gt_log_debug_in_trail

Options 0 or 1

Default 0

Defines whether additional debug mode entries from GENIUS TOOLS for Creo should be written in the Creo trail file. (0 - No, 1 - Yes)

gt_regen_times

Options 1, 2 or 3

Default 1

Defines how often GENIUS TOOLS for Creo should regenerate at regeneration.

gt_replace_character_if_not_found

Options Any text

Default

Defines a string that is returned instead of variables if they do not exist. The "empty" option returns an empty string.

gt_resource_folder**Options** Path or foldername**Default** %TEMP%

Defines the directory for all resources.

gt_show_license_dialog**Options** 0 or 1**Default** 1

Defines whether an error dialog for license errors is displayed. (0 - No, 1 - Yes)

gt_show_welcome_message_version**Options** Any Text or empty**Default** Current Version

Defines the name whether the splash screen open is saved.

gt_sqlite_db_max_tries**Options** Any number**Default** 25

Number of trials to access a SQLite database.

gt_sqlite_db_sleeptime_between_tries**Options** Any number**Default** 10

Wait time in ms before trying to reconnect to a SQLite database (see also gt_sqlite_db_max_tries).

gt_start_assembly**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Assembly Report can be started by users. (0 - No, 1 - Yes)

gt_start_assembly_editor**Options** 0 or 1**Default** 1

Defines whether Assembly Report Editor can be started by users. (0 - No, 1 - Yes)

gt_start_configuration_utility**Options** 0 or 1**Default** 1

Defines whether Configuration Utility can be started by users. (0 - No, 1 - Yes)

gt_start_dimension**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Dimension can be started by users. (0 - No, 1 - Yes)

gt_start_forms**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Forms can be started by users. (0 - No, 1 - Yes)

gt_start_forms_editor**Options** 0 or 1**Default** 1

Defines whether the editor of the forms may be started.

gt_start_inspect**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Inspect can be started by users. (0 - No, 1 - Yes)

gt_start_inspect_3D**Options** 0 or 1**Default** 0

Defines whether GENIUS TOOLS Inspect 3D can be started by users. (0 - No, 1 - Yes)

gt_start_inspect_editor**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Inspect Editor can be started by users. (0 - No, 1 - Yes)

gt_start_inspect_revision**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Inspect Revision can be started by users. (0 - No, 1 - Yes)

gt_start_library**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Library can be started by users. (0 - No, 1 - Yes)

gt_start_library_editor**Options** 0 or 1**Default** 1

Defines whether Library Editor can be started by users. (0 - No, 1 - Yes)

gt_start_material**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Material can be started by users. (0 - No, 1 - Yes)

gt_start_material_editor**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Material Editor can be started by users. (0 - No, 1 - Yes)

gt_start_multibody_to_assembly**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Multibody To Assembly can be started by users. (0 - No, 1 - Yes)

gt_start_name_generator**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Name Generator can be started by users. (0 - No, 1 - Yes)

gt_start_name_generator_editor**Options** 0 or 1**Default** 1

Defines whether Name Generator Editor can be started by users. (0 - No, 1 - Yes)

gt_start_parameter**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Parameter can be started by users. (0 - No, 1 - Yes)

gt_start_parameter_editor**Options** 0 or 1**Default** 1

Defines whether Parameter Editor can be started by users. (0 - No, 1 - Yes)

gt_start_quick_access**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Quick Access can be started by users. (0 - No, 1 - Yes)

gt_start_quick_access_editor**Options** 0 or 1**Default** 1

Defines whether Quick Access Editor can be started by users. (0 - No, 1 - Yes)

gt_start_udf_form**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS UDF Forms can be started by users. (0 - No, 1 - Yes)

gt_start_udf_form_editor**Options** 0 or 1**Default** 1

Defines whether UDF Forms Editor can be started by users. (0 - No, 1 - Yes)

gt_start_utilities**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Utilities can be started by users. (0 - No, 1 - Yes)

gt_start_value_transfer**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Value Transfer can be started by users. (0 - No, 1 - Yes)

gt_version_string**Options** Any text**Default**

Shows current version of GENIUS TOOLS for Creo.

gt_visible_details_rows_set**Options** Any number >0**Default** 9

Height of the detail area as row count

gt_window_size_position_save**Options** 0 or 1**Default** 1

Defines whether the position and size of GENIUS TOOLS dialogs are saved and used again the next time. Also saves the status of the parameter model list.

GENIUS TOOLS Assembly Report

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Assembly Report are administered.

gta_autoload_folder**Options** Path**Default**

Defines the folder in which the currently selected definition is searched for in order to update it.

gta_change_wtPart**Options** 0 or 1**Default** 1

Defines whether the user may change the corresponding WTPart name if there is no match in the windchill database. You can set %TBXADMIN%.

gta_clear_pos_parameter**Options** 0 or 1**Default** 0

Defines whether all POS parameters created up to that point are reset to -1 or "-1" before the POS parameter is written.

gta_dec_places**Options** 0 to 6**Default** 2

Defines the number of decimal places displayed in tables in GENIUS TOOLS Assembly Record.

gta_default_file**Options** Filepath**Default** %gt_resource_folder%assembly\gt_assembly.xml

Defines the file name of the default report for Assembly Report.

gta_export_creo_position**Options** 0 or 1**Default** 1

Defines whether the position of a component should be exported to CSV or Excel in addition to the columns defined in the editor.

gta_export_file**Options** Filename**Default**

Defines the default name of an exported report file. You can use the GENIUS TOOLS for Creo variables. (For example: "%PART_NO%_%CAD_REVISION%_%CAD_CREATED_ON%") Please note: Do not use file extensions if you use Excel and CSV exports.

gta_export_path

Options Path or foldername

Default

Defines the default path for saving reports.

gta_export_position_value

Options 0 or 1

Default 1

Defines whether the position value of a component should be exported to CSV or Excel in addition to the columns defined in the editor.

gta_export_rownumber

Options 0 or 1

Default 1

Defines whether the row index should be exported to CSV or Excel in addition to the columns defined in the editor.

gta_export_template

Options Filepath

Default %gt_resource_folder%assembly\\gt_assembly.xlsx

Defines the default export template.

gta_export_type

Options 0 or 1

Default 1

Defines whether the component type should be exported to CSV or Excel in addition to the columns defined in the editor.

gta_fill_empty_with_default

Options 0 or 1

Default 1

Defines whether the component parameter for the index should be filled with its default value if the parameter does not exist in the model.

gta_lang

Options Language code (en, de, es, ...)

Default de

Defines additional languages to support multi-language titles in GENIUS TOOLS Assembly Report using two-character language codes.

gta_level_row_colors

Options Any text

Default

Defines the colors of table rows according to their level in the bill of materials. Separated by comma and given as hex values.

gta_open_export_csv

Options 0 or 1

Default 1

Defines whether the exported CSV is opened or not.

gta_save_xml_in_mdl

Options 0 or 1

Default 1

Defines whether the Assembly Report XML definitions are saved in models. (0 - XML definitions are saved externally in files, 1 - XML definitions will be written into the models)

gta_show_mdl_list

Options 0 or 1

Default 1

Defines whether the list of assemblies of the first level is visible or not.

gta_struct_insert_space

Options 0 to 10

Default 3

Defines the number of blanks that are prefixed per level in each entry.

gta_update_component_parameter_type**Options** 0 or 1**Default** 1

Defines whether component parameters are updated to the data type of the current definition.

GENIUS TOOLS Dimension

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Dimension are administered.

gtd_can_change_family_table_values**Options** 0 or 1**Default** 1

Defines whether users can change entries in family tables in GENIUS TOOLS Dimension.

gtd_dec_places**Options** 0 to 6**Default** 2

Defines the number of decimal places displayed in GENIUS TOOLS Dimension.

gtd_filter**Options** 0 or 1**Default** 0

Defines whether GENIUS TOOLS Dimension is started with active name filter (named dimensions). (0 - No, 1 - Yes)

gtd_filter_additive**Options** 0 or 1**Default** 0

Defines whether GENIUS TOOLS Dimension filters working additive or logical grouped. (0 - logical grouped, 1 - additive)

gtd_filter_value**Options** Floating-point number**Default** 0.001

Defines the start value for the Value filter.

gtd_filter_value2**Options** Floating-point number**Default** 5.000

Defines the start value for the second Value filter.

gtd_label_col_size**Options** Any number**Default** 10

Defines the width of the dimension names column in characters. A value less than 5 will be ignored.

gtd_regenerate_only_dimensions_model**Options** 0 or 1**Default** 0

Defines whether the root model (assembly) and all its sub models are regenerated (0) or only the currently selected model in GENIUS TOOLS Dimension with its sub models (1).

gtd_value_col_size**Options** Any number**Default** 5

Defines the width of the dimension values column in characters. A value less than 5 will be ignored.

GENIUS TOOLS Forms

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Forms are administered.

gtf_ask_before_changing_fam_table**Options** 0 or 1**Default** 0

Defines whether users will be asked before changes to entries in family tables are applied in GENIUS TOOLS Forms.

gtf_auto_replace_comma_for_float_values**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Forms also accepts numerical inputs with comma (,) as decimal separator (1) or not (0).

gtf_autoload_folder**Options** Path**Default**

Defines the folder that would be searched for the xml based on the value of gtf_autoload_parameter.

gtf_autoload_overwrite**Options** 0 or 1**Default** 0

Defines whether it should be looked for an external form only if no internal exists (0) or always (1).

gtf_autoload_parameter**Options** Any text**Default** WEBCODE

Defines the parameter which value should be used to post load external saved forms.

gtf_def_lang**Options** Language abbreviation (en, de, es, ...)**Default** en,de,fr

Defines the languages of element descriptions that are displayed in the Forms Editor.

gtf_default_folder

Options Path or foldername

Default %gt_resource_folder%forms

Defines the default folder for imports and exports of definitions and values.

gtf_descriptionwidth

Options Any number

Default 15

Defines the width of the name column of GENIUS TOOLS Forms in characters.

gtf_editor_multiselect

Options 0 or 1

Default 0

Defines whether GENIUS TOOLS Forms Editor "Add multiple items until cancel" is selected. (0 - is unselected, 1 - is selected)

gtf_external_data_folder

Options Path or foldername

Default %gt_resource_folder%forms

Defines the folder that is used for automatic CSV imports.

gtf_import_xml_use_name_if_no_id_defined

Options 0 or 1

Default 1

Defines if at import of values from xml the element name should be used if the id is missing.

gtf_lang

Options Language abbreviation (en, de, es, ...)

Default en

Defines the standard language of element descriptions which are displayed in Forms and the Forms Editor (standard is the current Creo language).

gtf_midwidth**Options** Any number**Default** 15

Defines the width of the middle column in Forms in characters.

gtf_namewidth**Options** Any number**Default** 10

Defines the width of the name column in Forms in characters.

gtf_open_with_selected_model**Options** 0 or 1**Default** 1

Defines whether the current model (0) or the selected model (1) should be used at the startup of GENIUS TOOLS Forms.

gtf_separator**Options** Any character**Default** ;

Defines the separator for reading CSV files.

gtf_show_regen_btn**Options** 0 or 1**Default** 0

Defines whether the regeneration from forms save process is deactivate able.

GENIUS TOOLS Inspect

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Inspect are administered.

gti_decimal_marker_follow_dtl**Options** 0 or 1**Default** 1

Defines whether the DTL setting "decimal_marker" is read, thereby starting the replacement of "." with "," in Inspect report tables (0 - No, 1 - Yes).

gti_def_lang**Options** Language abbreviation (en, de, es, ...)**Default** en

Defines the display language of GENIUS TOOLS Inspect.

gti_din_compliant**Options** 0 or 1**Default** 1

Defines whether the numbering should be similar to DIN 6770 (1), or whether there should be a new number for each symbol according to the defined rules (0). If the option is activated, gti_number_sort_at_height, gti_number_sort_at_type and gti_numbering_all_sheets get ignored.

gti_excel_export_file**Options** Filename**Default**

Defines the default name of an exported report file. The file extension (.xlsx or .xlsm) must also be specified. You can use the GENIUS TOOLS for Creo variables. (For example: "=%PART_NO%_%CAD_REVISION%_%CAD_CREATED_ON%")

gti_excel_export_path**Options** Path**Default**

Defines the default path for saving reports.

gti_excel_template**Options** Filename**Default** gti_template_de.xlsx

Defines the name of the basic Excel template.

gti_fillup_places**Options** 0 to 5**Default** 3

Defines the maximum number of leading zeros that numbers are filled up with.

gti_folder**Options** Path or foldername**Default** %gt_resource_folder%inspect\

Defines the folder containing the symbols, tables and the definitions.

gti_lang**Options** Language code (en, de, es, ...)**Default** de

Defines additional languages for displaying UI elements using two-character language codes.

gti_number_range_per_sheet**Options** 0 or 1**Default** 0

Defines for multi-sheet drawings whether symbols have a number range for the whole drawing (0) or per sheet (1).

gti_number_sort_at_height**Options** -1, 0 or 1**Default** 0

Defines whether symbols are numbered by height in the drawing asc (1), desc (-1) or creation order (0).

gti_number_sort_at_type**Options** -1, 0 or 1**Default** 0

Defines whether symbols are numbered by type asc (1), desc (-1) or creation order (0).

gti_numbering_all_sheets**Options** 0 or 1**Default** 0

Defines whether symbols should be numbered across all sheets.

gti_revision_excel_coloring**Options** 0 or 1**Default** 1

Defines whether the symbols should be colored in the Excelexport.

gti_revision_excel_export_file**Options** Filename**Default**

Defines the default name of an exported report file. The file extension (.xlsx or .xlsm) must also be specified. You can use the GENIUS TOOLS for Creo variables. (For example: "=%PART_NO%_%CAD_REVISION%_%CAD_CREATED_ON%")

gti_revision_excel_export_path**Options** Path**Default**

Defines the default path for saving reports.

gti_revision_excel_template**Options** Filename**Default** gti_revision_template_de.xlsx

Defines the name of the basic Excel template.

gti_revision_folder

Options Path or foldername

Default %gt_resource_folder%inspect\

Defines the folder for the search for the base templates for Excel.

gti_revision_parameter

Options Parametername

Default %CAD_REVISION%

Defines the parameter which is read out for the revision display.

gti_shape_and_position_as_note

Options 0 or 1

Default 1

Defines whether GENIUS TOOLS Inspect interprets notes with contained shape and position tolerance symbols as main type "shape and position". (0 - note, 1 - shape- and position)

gti_show_old_symbols

Options 0 or 1

Default 1

Defines whether currently existing symbols from old revisions are also displayed.

gti_start_file

Options Filename

Default gti_definition.xml

Defines the name of the basic definition file.

gti_start_number

Options 0 or 1

Default 1

Defines the starting number for incrementing inspection symbols.

GENIUS TOOLS Library

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Library are administered.

gtl_action_copy_set_file_as_common_name

Options 0 or 1

Default 1

If active (1), a copied file is renamed with the Common Name.

gtl_action_insert_with_mapkey

Options 0 or 1

Default 1

Defines whether inserting a part from GENIUS TOOLS Library is done by mapkey or a programatic function (different user experience).

gtl_action_mfg_insert_as_fixture

Options 0 or 1

Default 1

Defines whether insert as fixture / copy insert as fixture is available (1) or not (0).

gtl_action_open_defined_simp_rep

Options Name of the simplified representation

Default DEFAULT ENVELOPE REP

Defines the simplified representation that get called by "Open / Insert simplified graphics"

gtl_blue_color

Options Hexadecimal 000000 - FFFFFFFF

Default 0000FF

Specifies the color value for blue in GENIUS TOOLS Library.

gtl_close_detail_panel_after_copy**Options** 0 or 1**Default** 1

Defines whether the detail dialog is closed (1) or remains open (0) after a file is copied.

gtl_close_detail_panel_after_insert**Options** 0 or 1**Default** 0

Defines whether the detail dialog is closed (1) or remains open (0) after a file is inserted.

gtl_close_detail_panel_after_open**Options** 0 or 1**Default** 1

Defines whether the detail dialog is closed after a file is opened or inserted (1) or remains open (0).

gtl_copy_check_existence**Options** 0 or 1**Default** 1

Defines whether the new names are checked for existence before the start of a copy process (could take a long time).

gtl_copy_drawings_with_same_name**Options** none, part, assem or both**Default** %rename_drawings_with_object%

Defines whether drawings of the same name are copied as well. Default is the value of the Creo option "rename_drawings_with_object". Possible options: none - drawings are not copied. assem - Only for assemblies. part - Only for parts. both - both.

gtl_copy_remove_wt_templates_from_ws**Options** 0 or 1**Default** 0

Defines whether the template for copying should be removed from the workspace (only Windchill) after the copy process (1) or not (0).

gtl_db_path**Options** Folderpath**Default** %gt_resource_folder%library\

Defines the path to the library databases.

gtl_def_lang**Options** Language abbreviation (en, de, es, ...)**Default** en

Defines an alternative language code for the case that the translation (defined in gtl_lang) is not found in the library database.

gtl_detail_image_folder**Options** Path or foldername**Default** %gtl_current_db_path%

Defines the folder for images displayed in the detail window: %gtl_current_db_path% - uses /library//img_detail/Otherwise, the specified path is used.

gtl_detail_window_autoincrease_size**Options** 0 or 1**Default** 1

Defines whether the configuration options gtl_detail_window_select_height, gtl_detail_window_select_width, gtl_detail_window_udf_forms_height and gtl_detail_window_udf_forms_width are used (1) or not (0).

gtl_detail_window_detail_image_height**Options** Any number**Default** 0

Defines the height of a detail image.

gtl_detail_window_detail_image_show_title**Options** 0 or 1**Default** 1

If a detail image for an object is found, its title will be displayed next to it (1) or not (0).

gtl_detail_window_detail_image_width**Options** Any number**Default** 0

Defines the width of the detail image.

gtl_detail_window_height**Options** Any number**Default** 450

Defines the initial height of the detail dialog in pixel.

gtl_detail_window_hide_details**Options** 0 or 1**Default** 0

Defines whether object details are always shown (0) or hidden initially (1).

gtl_detail_window_move_by_tree_width**Options** 0 or 1**Default** 1

Defines whether the detail dialog is displayed next to the model tree. (0 - No, 1 - Yes)

gtl_detail_window_preselected_tab**Options** 0 to 2**Default** 2

Defines the default tab of the Detail window in GENIUS TOOLS Library (0: Details, 1: Selection 2: Form).

gtl_detail_window_preselected_tab_gph**Options** -1 to 2**Default** -1

Defines the default tab of the Detail window in GENIUS TOOLS Library (-1: Inherit from gtl_detail_window_preselected_tab, 0: Details, 1: Selection 2: Form).

gtl_detail_window_select_height**Options** Any number**Default** 600

Defines the minimal height of the Detail window in pixel after changing to the "Selection" tab.

gtl_detail_window_select_width**Options** Any number**Default** 300

Defines the minimal width of the Detail dialog in pixel after changing to the "Selection" tab.

gtl_detail_window_selection_information_height**Options** Any number**Default** 4

Defines the height of selection information in rows.

gtl_detail_window_show_both_languages**Options** 0 or 1**Default** 0

Defines whether both languages (gtl_lang and gtl_def_lang) are displayed in the Detail window (1) or gtl_lang only (0).

gtl_detail_window_show_status_in_head_area**Options** 0 or 1**Default** 0

Defines whether the status should be displayed inside the head area of the detail dialog (0 - No, 1 - Yes).

gtl_detail_window_show_variant_attribute_type**Options** 0 or 1**Default** 0

Shows attribute types (D:, P:) in the Selection tab of the Detail window of GENIUS TOOLS Library (1).

gtl_detail_window_udf_forms_height**Options** Any number**Default** 700

Defines the minimal height of the Detail dialog in pixel after changing to the Forms tab (UDF Forms).

gtl_detail_window_udf_forms_width**Options** Any number**Default** 350

Defines the minimal width of the Detail dialog in pixel after changing the Forms tab (UDF Forms).

gtl_detail_window_width**Options** Any number**Default** 300

Defines the initial width of the Detail dialog in pixel.

gtl_dnd_enabled**Options** 0 or 1**Default** 1

Activates (1) or deactivates (0) Drag and Drop for GENIUS TOOLS Library in Creo 3.0.

gtl_downsync**Options** 0 or 1**Default** 1

Defines whether libraries are cached locally on initial call (1) or if they are always read from GT_RESOURCE_FOLDER (0) This configuration option does not affect the editor.

gtl_downsync_path**Options** Folderpath**Default** %appdata%\INNEO\GENIUS_TOOLS\for_Creo\library\

Defines the path for locally caching libraries. Depends on gtl_downsync.

gtl_editor_create_db_security_copy_once_a_day**Options** 0 or 1**Default** 1

Defines whether a backup of a database is created once a day after opening it with the Library Editor (1) or not (0).

gtl_editor_file_import_action_fallback**Options** 0 to 262143**Default** 16383

Defines one or more actions which are added to library objects during import or object creation via addition (31=1+2+4+8+16). 1 - Open, 2 - Insert, 4 - Copy, 8 - Copy insert, 16 - Insert (with reference), 16384 - Merge, 32768 - Die, 65536 - Punch, 131072 - Insert as Copy Geom

gtl_editor_file_import_check_selection_existance**Options** 0 or 1**Default** 0

Defines whether the existence check during the import is also applied to instances (1). PLEASE NOTE: This affects import times significantly.

gtl_editor_find_double_objects_by_path_and_name**Options** 0 or 1**Default** 0

Defines how duplicate objects are checked (0 - by object name, 1 - by object path and object name).

gtl_editor_link_double_objects**Options** 0 or 1**Default** 1

Defines how doubled objects are treated by GENIUS TOOLS Library (0 - every doubled object is created during the import, 1 - Doubled objects are linked and only created once during the import).

gtl_editor_mnu_creator_automatic_selection**Options** 0 or 1**Default** 1

Defines whether tree view selections are inherited between subnodes of the tree in the MNU export dialog. (Yes - 1, No - 0)

gtl_editor_selections_inherit_instances**Options** 0 or 1**Default** 1

Defines whether images are handed down from a generic to its instances. (0 - No, 1 - Yes)

gtl_editor_sqlite_allow_unsecure_write_operations**Options** 0 or 1**Default** 1

Defines whether unsecure but faster write operations are used for database access. (0 - No, 1 - Yes)

gtl_editor_use_black_on_white_for_screenshots**Options** 0 or 1**Default** 1

Defines, if images inside of a batch rework get created with the system colors "Black on White" (1) or the current settings (0).

gtl_favorite_button1_liblink**Options** Any text**Default** company

Defines the file name of the first favorite library (this is sepearted of the others).

gtl_favorite_button2_liblink**Options** Any text**Default** designtools

Defines the file name of the second favorite library.

gtl_favorite_button3_liblink**Options** Any text**Default** planttools

Defines the file name of the third favorite library.

gtl_favorite_button4_liblink**Options** Any text**Default** user

Defines the file name of the fourth favorite library.

gtl_favorite_button_show**Options** 0 or 1**Default** 1

Defines whether the buttons for favorites are displayed in GENIUS TOOLS Library. (0 - No, 1 - Yes)

gtl_favorite_path**Options** Path or foldername**Default** %appdata%\\INNEO\\GENIUS_TOOLS\\for_Creo\\library\\

Defines the path to save GT Library favorites.

gtl_filter_blue**Options** 0 or 1**Default** 1

Default filter setting for the filter "Blue".

gtl_filter_blue_text**Options** Any text**Default**

Defines the description of the status "Blue", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_green**Options** 0 or 1**Default** 1

Default filter setting for the filter "Green".

gtl_filter_green_text**Options** Any text**Default**

Defines the description of the status "Green", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_lilac**Options** 0 or 1**Default** 1

Default filter setting for the filter "Lilac".

gtl_filter_lilac_text**Options** Any text**Default**

Defines the description of the status "Lilac", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_normal**Options** 0 or 1**Default** 1

Default filter setting for the filter "Normal".

gtl_filter_normal_text**Options** Any text**Default**

Defines the description of the status "Normal", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_red**Options** 0 or 1**Default** 1

Default filter setting for the filter "Red".

gtl_filter_red_text**Options** Any text**Default**

Defines the description of the status "Red", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_teal**Options** 0 or 1**Default** 1

Default filter setting for the filter "Teal".

gtl_filter_teal_text**Options** Any text**Default**

Defines the description of the status "Teal", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_filter_yellow**Options** 0 or 1**Default** 1

Default filter setting for the filter "Yellow".

gtl_filter_yellow_text**Options** Any text**Default**

Defines the description of the status "Yellow", displayed in the Detail dialog. If the option is empty, the language-dependend default is used.

gtl_green_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** 008800

Specifies the color value for green in GENIUS TOOLS Library.

gtl_gtf_save_forms_in_model**Options** 0 or 1**Default** 1

Defines whether Forms are copied into a model (1) or not (0) during the creation of a new copy of a model.

gtl_gtng_gtf_show_name_dialog_everytime**Options** 0 or 1**Default** 0

Defines whether the name dialog is displayed always (1), or only if it has not been opened manually before (0).

gtl_gtng_new_name_rule**Options** Any Text**Default** @number@@oldname@

Defines the rule for generating names for objects that should be copied. @number@ = generated name @oldname@ = name of the object

gtl_gtng_overwrite_std_number_definition**Options** 0 or 1**Default** 1

Defines whether the Name Generator definition @number@@oldname@ is replaced by @number:%gtl_gtng_standard_db_filter_for_file_copy%@oldname@ also if the definition has been set explicitly.

gtl_gtng_standard_db_filter_for_file_copy**Options** Any text**Default**

Limits the displayed name configurations from GENIUS TOOLS Name Generator. If only one result remains, this name configuration is automatically used.

gtl_gtp_start_gtp_after_model_creation**Options** Any Text**Default**

Defines the model types (prt,asm,dw), seperated by comma, that should start GENIUS TOOLS Parameter after model creation.

gtl_home_db**Options** Filename**Default**

When containing a path to a database the path bar of GTL will display an icon showing a house. Clicking this icon will open this database.

gtl_img_create_detail_size**Options** Any Number > 0**Default** 200

Defines the size for image creation for details.

gtl_img_create_tooltip_size**Options** Any Number > 0**Default** 200

Defines the size for image creation for tooltips.

gtl_img_detail_size**Options** Number >= 20**Default** 100

Defines the rendering size for detail images in pixel.

gtl_img_size**Options** 100 or 40**Default** 100

Defines the displayed icon size in pixel.

gtl_img_switch_size**Options** Number >= 20**Default** 40

Defines the rendering size for switch images in pixel.

gtl_info_folder**Options** Folderpath**Default** %gtl_current_db_path%

Defines a folder that will be searched for information documents for library objects during a batch processing run.

gtl_lang**Options** Language abbreviation (en, de, es, ...)**Default** de

Defines a language which is used to display library objects as a language code.

gtl_lilac_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** FF00FF

Specifies the color value for lilac in GENIUS TOOLS Library.

gtl_list_use_with_double_click**Options** 0 or 1**Default** 0

Defines whether the walk through the catalogue structure goes by single click (0) or double click (1).

gtl_mark_category_images**Options** 0 or 1**Default** 1

Defines whether images of library categories get a triangle symbol for better differentiation.

gtl_mdI3d_search_by_bounding_box**Options** 0 or 1**Default** 1

Defines the 3D search behavior together with gtl_mdI3d_search_by_voxel.

gtl_mdI3d_search_by_voxel**Options** 0 or 1**Default** 1

Defines the 3D search behavior together with gtl_mdI3d_search_by_bounding_box.

gtl_mdI3d_search_factor_bb_bounding_box**Options** 0 to 1**Default** 0.5

Defines the billing factor. The values entered for the three configuration options gtl_mdI3d_search_factor_bb_bounding_box, gtl_mdI3d_search_factor_bb_bounding_box_unnorm und gtl_mdI3d_search_factor_bb_mass_center must add up to 1 (= 100%). They determine the weighting of the individual search options.

gtl_mdI3d_search_factor_bb_bounding_box_unnorm**Options** 0 to 1**Default** 0

Defines the billing factor. The values entered for the three configuration options gtl_mdI3d_search_factor_bb_bounding_box, gtl_mdI3d_search_factor_bb_bounding_box_unnorm und gtl_mdI3d_search_factor_bb_mass_center must add up to 1 (= 100%). They determine the weighting of the individual search options.

gtl_mdI3d_search_factor_bb_mass_center**Options** 0 to 1**Default** 0.5

Defines the billing factor. The values entered for the three configuration options `gtl_mdI3d_search_factor_bb_bounding_box`, `gtl_mdI3d_search_factor_bb_bounding_box_unnorm` und `gtl_mdI3d_search_factor_bb_mass_center` must add up to 1 (= 100%). They determine the weighting of the individual search options.

gtl_mdI3d_search_factor_bounding_box**Options** 0 to 1**Default** 0.5

Defines the billing factor. The values entered for the two configuration options `gtl_mdI3d_search_factor_bounding_box` and `gtl_mdI3d_search_factor_voxel` must add up to 1 (= 100%). They determine the weighting of the individual search options.

gtl_mdI3d_search_factor_voxel**Options** 0 to 1**Default** 0.5

Defines the billing factor. The values entered for the two configuration options `gtl_mdI3d_search_factor_bounding_box` and `gtl_mdI3d_search_factor_voxel` must add up to 1 (= 100%). They determine the weighting of the individual search options.

gtl_mdI3d_search_voxel_per_axe**Options** Any number**Default** 10

Defines the number of voxels for 3D resolution. Please note: This value must be consistent in the database and in the program.

gtl_parameter_multiple_value_separator**Options** One letter**Default**

Specifies the character that separates multi-value parameter values.

gtl_red_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** 880000

Specifies the color value for red in GENIUS TOOLS Library.

gtl_retrieve_run_batch**Options** Any text**Default**

Defines a batch file that is called in preparation for a file load to session.@name@ will be replaced by object name.@path@ will be replaced by object path.

gtl_run_mode**Options** 0 or 1**Default** 1

Defines the display mode of GENIUS TOOLS Library. (0 - as external dialog, 1 - inside the Creo navigator pane)

gtl_search_btn_search**Options** 0 or 1**Default** 0

Defines whether a search is performed after a click on the extended search button. (0 - No, 1 - Yes)

gtl_search_no_type_no_folder**Options** 0 or 1**Default** 0

Defines whether folders will be searched (1) or not (0) when the type flag is not set in the search options.

gtl_search_tree_dependent**Options** 0 or 1**Default** 1

Defines whether the complete database or only categories/subcategories get searched. The search in categories doesn't show objects from selections. (0 - No, 1 - Yes).

gtl_search_values_per_page**Options** Any number**Default** 25

Number of rows per page. Values <1 and >999 are read as 25.

gtl_show_both_languages**Options** 0 or 1**Default** 0

Defines whether both languages (gtl_lang and gtl_def_lang) are displayed in the library browser (1) or only gtl_lang with the fallback gtl_def_lang (0).

gtl_show_category_names**Options** 0 or 1**Default** 0

Defines whether category names are displayed.

gtl_show_font_size**Options** Any number**Default** 5

Defines the font size in the library browser.

gtl_show_object_names**Options** 0 or 1**Default** 1

Defines whether object names are displayed in GENIUS TOOLS Library. (0 - No, 1 - Yes)

gtl_show_path**Options** 0 or 1**Default** 1

Defines whether the path of underlying categories will be shown (1) or not (0).

gtl_show_tooltip_image**Options** 0 or 1**Default** 1

Defines whether tooltip images are displayed.

gtl_start_db**Options** Filename**Default**

Defines the library loaded on startup. If empty, the alpha-numerical first is used.

gtl_teal_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** 008888

Specifies the color value for teal in GENIUS TOOLS Library.

gtl_tooltip_image_folder**Options** Path or foldername**Default** %gtl_current_db_path%

Defines the folder for tooltip images: %gtl_current_db_path% -
uses /library//img_tooltip/Otherwise it uses the path given in this value

gtl_yellow_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** FFA500

Specifies the color value for yellow in GENIUS TOOLS Library.

GENIUS TOOLS Material

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Material are administered.

gtm_allow_edit_favorites

Options 0 or 1

Default 1

Defines whether the user is allowed to edit the favorites.

gtm_can_set_to_system_default

Options 0 or 1

Default 1

Defines whether system default is settable or not.

gtm_close_at_set_material

Options 0 or 1

Default 1

Defines whether GT Material should be restarted after material declaration.

gtm_command_file

Options Filepath

Default %gt_resource_folder%\material\material.db

Defines the database used by GENIUS TOOLS Material and GENIUS TOOLS Material Editor.

gtm_db_def_lang

Options Language abbreviation (en, de, es, ...)

Default en

Defines a language code for a language to be used if the translation is not found in gtm_db_lang.

gtm_db_lang

Options Language abbreviation (en, de, es, ...)

Default de

Defines the displayed language of material attributes as a language code (standard is the Creo language).

gtm_delete_not_current_materials

Options 0 or 1

Default 0

This configuration deletes all materials that do not match the current material when a new material is set. This could cause troubles while using:- Relations- Pro/Program- Family tables- Multi body- ...Setting is explicitly not recommended.

gtm_editor_material_browser_path

Options Path

Default %GTS_SERVERONLY_DIR%
\\tools\\freeware_gt_material_browser\\GT_Material_Browser.exe

Defines the path where GT Material Browser get searched. If it is found it could be started from GT Material Editor. At Version 1.0.4.0 also path transfer ist supported (Temp path if WT connection is set).

gtm_exclude_material_from_update_all

Options Any Text

Default

Defines a material that is excluded from the update of all materials (without extension).

gtm_favorite_file

Options Filepath

Default %appdata%
\\INNEO\\GENIUS_TOOLS\\for_Creo\\material\\favorites.txt

Defines the file whether the favorite materials get saved.

gtm_infoDoc_folder

Options Path or foldername

Default %gt_resource_folder%\material\info\

Defines the directory containing information documents for GENIUS TOOLS Material, if no path is specified.

gtm_remember_selected_filter

Options Any number

Default 1

The number of layers that should be kept.

gtm_replace_materials_with_same_name

Options 0 or 1

Default 1

Defines the behavior of GENIUS TOOLS Material if an assigned material has the same name as the new assigned material (0 - the old material will not be replaced, 1 - the old material will be replaced).

gtm_show_body_selection

Options 0 or 1

Default 1

Defines whether the body selector would be shown or not.

gtm_show_infoDoc_in_list

Options 0 or 1

Default 1

Defines whether the info symbol should already be shown in the shortlist or only after picking the material.

gtm_show_properties_button**Options** 0 or 1**Default** 1

Defines whether the "Open material properties" control is shown in GT Material (1) or not (0).

gtm_visible_details_rows_set**Options** 8-21**Default** 9

Defines the height of the material list in rows.

GENIUS TOOLS Multibody To Assembly

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Multibody To Assembly are administered.

gtmba_body_parameter_export_rule**Options** Any Text**Default** ^.*\$

Defines the rule for determining which body parameters are to be exported. In addition, all parameters beginning with "PTC_" are ignored.

gtmba_gtng_common_name_rule**Options** Any Text**Default** @filename@

Defines the rule for generating common names for bodies. @number@ = generated name @oldname@ = name of the body or the current part @filename@ = generated file name

gtmba_gtng_filter**Options** Any text**Default**

Limits the displayed name configurations from GENIUS TOOLS Name Generator. If only one result remains, this name configuration is automatically used.

gtmba_gtng_name_rule**Options** Any Text**Default** @number@@oldname@

Defines the rule for generating names for bodies. @number@ = generated name@oldname@ = name of the body or the current part

gtmba_material_checked**Options** 0 or 1**Default** 1

Defines whether the check mark for the material export is set when it is called up for the first time.

gtmba_parameter_checked**Options** 0 or 1**Default** 1

Defines whether the check mark for the parameter export is set when it is called up for the first time.

gtmba_part_parameter_export_rule**Options** Any Text**Default** ^.*\$

Defines the rule for determining which part parameters are to be exported. In addition, all parameters beginning with "PTC_" are ignored.

gtmba_select_last_export_at_start**Options** 0 or 1**Default** 1

Defines whether the last export is started in update mode.

gtmba_start_model_dir**Options** Path**Default** %start_model_dir%

Defines the full path to the directory containing start parts and assemblies.

gtmba_template_designasm**Options** Path**Default** %template_designasm%

Defines the model used as the default assembly template.

gtmba_template_solidpart**Options** Path**Default** %template_solidpart%

Defines the model used as the default part template.

GENIUS TOOLS Name Generator

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Name Generator are administered.

gtng_copy_template_if_filter_db_not_found**Options** Any Text**Default**

Defines whether and which template would be used for the creation of undefined filter databases. If the definition is empty no db would be created.

gtng_db_name_filter**Options** Any text**Default**

All file names of configuration definitions are searched for the value defined here. Only definitions containing this value are displayed in Name Generator.

gtng_folder**Options** Path or foldername**Default** %gt_resource_folder%\name_generator\

Defines a directory containing the global name configurations for Name Generator.

gtng_local_folder**Options** Path or foldername**Default** %appdata%\INNEO\GENIUS_TOOLS\for_Creo\name_generator\

Defines a directory containing the local name configurations for Name Generator.

gtng_use_windchill_credentials_for_server_request**Options** 0 or 1**Default** 1

If a Windchill server is recognized, this option defines whether the login data is requested (1) or an URL without login data is retrieved (0).

GENIUS TOOLS Parameter

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Parameter are administered.

gtp_alternative_background_color**Options** Any text**Default**

Defines the color of every even row in the parameter form. Enter the colour as RGB values (e.g.: 122,45,89). The default color (grey) is used if the config option is empty. If the value is "n" or "no", no background color will be used.

gtp_ask_for_save**Options** 0 or 1**Default** 1

A window will pop up in the GTP after changing the model or closing the dialog, if there are parameters with changed values. This behaviour will be deactivated if the option is "0".

gtp_bold_parameter_values**Options** 0 or 1**Default** 0

Defines whether the parameter values are displayed in bold typeface. (0 - No, 1 - Yes)

gtp_check_connections**Options** 0 or 1**Default** 1

Defines whether database connections and filters should be checked after importing the parameter values and missing values be set automatically. (0 - No, 1 - Yes)

gtp_db_folder**Options** Path or foldername**Default** %gt_resource_folder%parameter\database\

Defines the folder for the databases which are used for database nodes.

gtp_default_value**Options** Any text**Default** -

Defines the default value for parameter values. This value is added to value lists in GTP regardless of whether it is valid for a parameter. If an empty value is set as the default value, it is not added to value lists. The default value is considered as an empty value when checking for empty values, e.g. in mandatory fields.

gtp_designate**Options** 0 or 1**Default** 1

Defines whether the checkbox „Remove all designations [in Windchill mode]“ in the tools menu of GT Parameter is displayed. (0: no, 1: yes)

gtp_do_not_save_conflicts**Options** 0 or 1**Default** 1

Defines whether you can save when filter conflicts occur. (0 - Yes, 1 - No)

gtp_do_not_save_empty_mandatory**Options** 0 or 1**Default** 1

Defines whether you can save when mandatory fields are left blank. (0 - Yes, 1 - No)

gtp_do_not_save_format_conflicts**Options** 0 or 1**Default** 1

Defines whether you can save when format conflicts occur. (0 - Yes, 1 - No)

gtp_dock_dialog_to_mdl**Options** 0 or 1**Default** 1

Defines the behavior of the GTP dialog. If the value is "1" the dialog belongs to the specific model window. If the value is "0" the dialog can be open and the user can interact with the model window.

gtp_dropdown_height_max**Options** Any number**Default** 10

Defines the maximum number of elements in a dropdown list displayed when lists from databases, CSV or text files are used in GT Parameter.

gtp_editor_new_def_without_sys_params**Options** 0 or 1**Default** 0

Defines whether existing system parameters of Creo/Windchill should be included when a new parameter definition is created from a model. (0 - System parameters are not included, 1 - System parameters are included into the definition)

gtp_editor_open_csv_program**Options** Filename**Default** notepad.exe

Defines the standard program to edit CSV and text files.

gtp_file**Options** Filepath**Default** %GT_RESOURCE_FOLDER%parameter\gtp_int_de\gtp_int_de.xml

Defines the path to the parameter definition file for GT Parameter (fallback).

gtp_file_param**Options** Any text**Default** MC_CHECKTYPE

Defines a parameter of the type string, which is used to automatically select a parameter definition file in conjunction with "gt_lst".

gtp_fill_empty_description**Options** 0 or 1**Default** 0

Defines whether an empty description of a Creo parameter should be overwritten with the description from the current parameter definition. (0 - No, 1 - Yes)

gtp_filter_auto_fill_back**Options** 0 or 1**Default** 1

Defines whether the filtering is set automatically if only one selection is possible. (0 - No, 1 - Yes)

gtp_filter_auto_single_fill**Options** 0 or 1**Default** 1

Defines whether the last available hit of a filtering is set automatically into a parameter value field (1) or not (0).

gtp_gtr_rules**Options** Path**Default** %gt_resource_folder%parameter\\

Defines the GT Value Transfer config load path, except it is not explicit defined.

gtp_lang**Options** Language abbreviation (en, de, es, ...)**Default** de

Defines language codes for additional languages to support language-dependent parameter titles in the parameter definition of GT Parameter.

gtp_lock_change_generic**Options** 0 or 1**Default** 1

Defines whether parameters of instances that are not in the family table may be changed in GTP. (1 - Yes, 0 - No) For the setting 1, if there are no other restrictions for the parameter of an instance, a symbol is shown to mark that the parameter is not in the family table.

gtp_lock_rel_locked_params**Options** 0 or 1**Default** 1

Defines whether parameters locked by relations may be changed in GTP.

gtp_lst**Options** Filepath**Default** %GT_RESOURCE_FOLDER%parameter\gtp.lst

Defines the path to the list file containing the overview of all available parameter definitions.

gtp_model_tree_column_width**Options** Any Number > 0**Default** 10

Defines the width of additionally displayed parameter columns, measured in the standard width for characters.

gtp_model_tree_columns**Options** Any text**Default**

Defines the parameters which are displayed as an extra column in the model list.

gtp_overwrite_description**Options** 0 or 1**Default** 0

Defines whether a description (not empty) of a Creo parameter can be overwritten by GENIUS TOOLS Parameter. (0 - No, 1 - Yes)

gtp_regen**Options** 1, 0, 2, or -1**Default** 1

Defines whether models will be automatically regenerated after saving and whether the checkbox „Regenerate after saving“ in the tools menu of GT Parameter is displayed. (0: display unchecked box, 1: display checked box, 2: no automatic regeneration and hide checkbox, -1: automatic regeneration and hide checkbox)

gtp_save_hidden**Options** 0 or 1**Default** 0

Defines whether hidden parameters are transferred into the model on Save. (0 - No, 1 - Yes)

gtp_save_model_at_save_press**Options** 0 or 1**Default** 0

Defines whether the model should be saved at GT Parameter save 1 or not 0*.

gtp_show_duplicate_warning**Options** 0 or 1**Default** 1

Defines whether a warning message box is shown if duplicated parameters occur (1) or only an information message in the message area of Creo (0).

gtp_show_hidden_params**Options** 0 or 1**Default** 0

Defines whether the hidden parameter tab is displayed in GT Parameter (1) or not (0).

gtp_show_mdllist**Options** 0 to 2**Default** 2

Defines whether the model list is displayed: 0 - list is hidden, 1 - list is always displayed, 2 - list is model-dependent.

gtp_show_server_conflict_dlg**Options** 0 or 1**Default** 0

Uses the server conflict dialog to check the status of a model in Windchill (0 - The dialog has to be opened manually (lock symbol) 1 - The dialog is shown after every model change if Creo cannot determine the status without the dialog).

gtp_start_drw**Options** 0 or 1**Default** 1

Defines whether the parameters of the active model (0) or the drawing parameters (1) are displayed first by GT Parameter in drawing mode.

gtp_start_gtp_after_model_creation**Options** Any Text**Default**

Defines the model types (prt,asm,drw), separated by comma, that should start GENIUS TOOLS Parameter after model creation.

gtp_use_type_insensitive_dbs**Options** 0 or 1**Default** 1

Defines how to handle integer and double values in databases.1: The values are saved as strings in the database and will be handled as strings.0: The double values are stored as rounded values in the database and it should be queried for ranges, not for values.

gtp_web_server_url**Options** Path or foldername**Default**

Defines the path to the web server for database queries.

GENIUS TOOLS Quick Access

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Quick Access are administered.

gtqa_always_option**Options** 0 or 1**Default** 0

Defines whether the modus option "Always" is displayed and usable in the Quick Access Editor.

gtqa_background_circle**Options** 0 or 1**Default** 1

Defines whether the inner circle of Quick Access is visible.

gtqa_background_picture**Options** Path and/or file name with extension .png**Default** gtqa_quick_access.png

Defines the path or file name of the image that is used as logo in the ring menu of GENIUS TOOLS Quick Access. The image has to be saved as a PNG file in one of these folders: \quick_access\img_background\\text\resource\

gtqa_change_closed_groups**Options** 0 or 1**Default** 0

Defines whether the locked groups (8 and 9) are shown and editable in the QuickAccess editor.

gtqa_command_file**Options** Filepath**Default** %gt_resource_folder%\quick_access\quick_access_%
GT_CREO_VERSION%.db

Defines the path to the database used by Quick Access and Quick Access Editor.

gtqa_command_icon_folder**Options** Path or foldername**Default** %gt_resource_folder%\quick_access\img_w20\

Defines the directory for the GENIUS TOOLS Quick Access command icons.

gtqa_db_def_lang

Options Language abbreviation (en, de, es, ...)

Default en

Defines a language code for a language to be used for GENIUS TOOLS Quick Access and its Editor if the translation is not found in gtqa_db_lang.

gtqa_db_lang

Options Language abbreviation (en, de, es, ...)

Default de

Defines a language code for a language to be used for the tooltips (standard is the Creo language).

gtqa_editor_create_db_security_copy_once_a_day

Options 0 or 1

Default 1

Defines whether a backup of the database is created once a day after opening it with the Library Editor (1) or not (0).

gtqa_group_bow

Options 0 or 1

Default 0

Defines the group display style. (0 = straight, 1 = curved)

gtqa_local_command_file

Options Filename

Default

Defines a second, local database for Quick Access and Quick Access Editor. e.g. %appdata%/inneo/genius_tools/quick_access/quick_access.db

gtqa_local_command_group_split**Options** 0 to 9 or "all"**Default** all

Defines which groups of the Quick Access are read from the local database (specify the highest group) and can be configured by user with the Quick Access Editor (0-9) or all (all).

gtqa_local_command_icon_folder**Options** Path or foldername**Default**

Defines the directory for the icons used by Quick Access.e.g. %appdata
%/inneo/genius_tools/quick_access/img_w20/

gtqa_no_tooltips**Options** 0 or 1**Default** 0

Defines whether tooltips are displayed in Quick Access. (0 - tooltips are displayed, 1 - tooltips are not displayed)

gtqa_show_admin_switch**Options** 0 or 1**Default** 1

Defines whether a switch to change between the global and local database is shown.

gtqa_start_mapkey**Options** Any text**Default** <

Defines the mapkey for starting Quick Access.

gtqa_tooltip_background_alpha_channel**Options** 0 to 254**Default** 180

Defines the opacity of the background from 0 - transparent to 254 - opaque

gtqa_tooltip_time**Options** Any number**Default** 1.0

Defines the time until tooltips in GENIUS TOOLS Quick Access are displayed in seconds (Default: 1.0).

gtqa_undock_groups_with_middle_mouse_button**Options** 0 or 1**Default** 1

Undock the open group as a separate window by use of the middle mouse button.

gtqa_undock_groups_with_right_mouse_button**Options** 0 or 1**Default** 0

Undock the open group as a separate window by use of the right mouse button.

gtqa_zoom**Options** 1.0 to 2.0**Default** 1.6

Defines the zoom factor of GENIUS TOOLS Quick Access. Zoom $\geq 1.0 < 1.4 \rightarrow 20 \times 20$ px
IconsZoom $\geq 1.4 < 1.9 \rightarrow 30 \times 30$ px IconsZoom $\geq 1.9 \rightarrow 40 \times 40$ px Icons

GENIUS TOOLS UDF Forms

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS UDF Forms are administered.

gtuf_create_layer**Options** 0 or 1**Default** 1

Defines whether a layer should be created with the name of the UDF and whether the help features should be placed on it.

gtuf_data_folder**Options** Path or foldername**Default** %gt_resource_folder%udf_forms\gt_udf

Defines the path where the UDF models are located.

gtuf_dec_places**Options** 0 to 6**Default** 2

Defines the number of decimal places displayed in tables in GENIUS TOOLS UDF Forms.

gtuf_definition_folder**Options** Path or foldername**Default** %gt_resource_folder%udf_forms\gt_definition

Defines the path where the UDF definitions are located.

gtuf_external_data_folder**Options** Path**Default** %gt_resource_folder%udf_forms\\gt_data

Defines which path should be used for external data that are defined without a specific path.

gtuf_lang**Options** Language abbreviation (en, de, es, ...)**Default** de

Defines language codes for additional languages in which UI elements should be displayed.

gtuf_save_xml_in_mdl**Options** 0 or 1**Default** 1

Defines whether the XML definitions of GENIUS TOOLS UDF Forms are written to the model. (0 - XML definitions are saved in external files, 1 - XML definitions are written to the model)

gtuf_separator**Options** Any character**Default** ;

Defines the separator for reading CSV files.

gtuf_show_create_by_creo**Options** 0 or 1**Default** 0

Defines whether the function "Create new UDF group by Creo std. Dialog" in GENIUS TOOLS UDF Forms is disposal.

gtuf_show_create_without_references**Options** 0 or 1**Default** 1

Defines whether the button to create a UDF without predefined references is shown (1) or not (0).

gtuf_show_status**Options** 0 or 1**Default** 1

Defines whether the status area below the input fields should be displayed or not.

GENIUS TOOLS Utilities

In the configuration file *gt_modules.cfg* the settings for GENIUS TOOLS Utilities are administered.

General Utilities Configuration Options

gtu_autostart_mapkey_inline**Options** 0 or 1**Default** 0

Defines whether the mapkey should be startet synchronously or asynchronously.

gtu_autostart_mapkey_mapkey**Options** %%Any text**Default**

Defines the Mapkey. WARNING: The mapkey must be written in one line. Mapkey from the config.pro file can be used by: %%MapkeyName;

gtu_comp_file_name**Options** Path or foldername**Default** gtu_component_parameters.xml

File name of configuration xml.

gtu_lang**Options** Language abbreviation (en, de, es, ...)**Default** en

Defines the display language of GENIUS TOOLS Utilities.

gtu_start_autostart_mapkey**Options** 0 or 1**Default** 1

Defines whether a defined mapkey "gtu_autostart_mapkey_mapkey" should be started. (0 - No, 1 - Yes)

gtu_start_close_all_windows**Options** 0 or 1**Default** 1

Defines whether the button for "Close all other windows" is displayed. (0 - No, 1 - Yes)

gtu_start_copyCParamToSubsCParam**Options** 0 or 1**Default** 1

Defines whether the utility "GENIUS TOOLS Copy Component Parameter To Substitution Component Parameter" is available. (0 - No, 1 - Yes)

gtu_start_extendedDimensionFunctions**Options** 0 or 1**Default** 0

Defines whether the module for changing dimensions is displayed (0 - No, 1 - Yes).

gtu_start_loadSaveConverter**Options** 0 or 1**Default** 1

Defines whether the utility "GENIUS TOOLS Load Save Converter" is available. (0 - No, 1 - Yes)

gtu_start_openGeomOrigin**Options** 0 or 1**Default** 1

Defines whether the button for "GT Open Base Model" is displayed. (0 - No, 1 - Yes)

gtu_start_pdm_commands**Options** 0 or 1**Default** 1

Defines whether the buttons for "Open workspace" and "Open commonspace" are displayed. (0 - No, 1 - Yes)

gtu_start_selectSurfacesByColor**Options** 0 or 1**Default** 1

Defines whether the utility "GENIUS TOOLS Select Surfaces by Color" is available. (0 - No, 1 - Yes)

gtu_start_toggleSymbolGroups**Options** 0 or 1**Default** 1

Defines whether the "GtuToggleSymbolGroups" button is provided (0 - No, 1 - Yes).

gtu_start_ui_change**Options** 0 or 1**Default** 1

Defines whether the the module UI Change gets started. (0 - No, 1 - Yes)

gtu_start_usage_logger**Options** 0 or 1**Default** 0

Defines whether the the usage logger is active. (0 - No, 1 - Yes)

gtu_ui_change_check_material_check_material_has_no_param**Options** 0 or 1**Default** 1

Defines whether a material without a revision parameter generates a warning in the material check (1) or not (0).

gtu_ui_change_check_material_check_material_not_in_db**Options** 0 or 1**Default** 1

Defines whether a material which is not in the database generates a warning in the material check (1) or not (0).

gtu_ui_change_check_material_check_multiple_materials**Options** 0 or 1**Default** 1

Defines whether multiple material files with the same name generate a warning in the material check (1) or not (0).

gtu_ui_change_check_material_check_no_material**Options** 0 or 1**Default** 1

Defines whether no material generates a warning in the material check (1) or not (0). This error message appears up to Creo version 6.

gtu_ui_change_check_material_check_old_material**Options** 0 or 1**Default** 1

Defines whether a different revision parameter of a material generates a warning in the material check (1) or not (0).

gtu_ui_change_check_material_check_only_current_material**Options** 0 or 1**Default** 0

Defines whether only the active material is checked.

gtu_ui_change_check_material_system_material_is_wrong**Options** 0 or 1**Default** 1

Defines whether the Creo std. material ptc_system_mtrl_props as current would be displayed as an failure case (1) or not (0).

gtu_ui_change_check_material_version**Options** 0 or 1**Default** 0

Defines whether the material version check should be done. (0 = No, 1 = Yes)

gtu_ui_change_check_material_version_parameter**Options** Any text**Default** REVISION

Defines the parameter that is filled with the material version.

gtu_ui_change_hole_optionmenu**Options** 0 or 1**Default** 0

Defines whether the HoleMenu should be expanded or if it keeps its standard height. This should not be used with Creo 5.0 or higher, because PTC changed the custom settings in newer versions.

gtu_ui_change_hole_optionmenu_length**Options** Any number >0**Default** 20

Defines the number of rows shown in the HoleMenu.

gtu_ui_change_show_forms_icon**Options** 0 or 1**Default** 1

Defines whether the Forms icon is shown if a model is displayed with Forms (1) or not (0).

gtu_ui_change_show_info**Options** 0 or 1**Default** 1

Defines whether the text from gtu_ui_change_show_info_text will be displayed in the Creo Window if there is no model loaded (1) or not (0).

gtu_ui_change_show_info_background**Options** Hexadecimal 000000 - FFFFFFFF**Default**

Specifies the color value for the background in GENIUS TOOLS Utilities Show Info.

gtu_ui_change_show_info_color**Options** Hexadecimal 000000 - FFFFFFFF**Default** 000000

Specifies the color value for the text in GENIUS TOOLS Utilities Show Info.

gtu_ui_change_show_info_text

Options	Any text
Default	%GT_VERSION_STRING%
	%PROE_START%
	INNEO Solution GmbH

Text output that will be displayed if gtu_ui_change_show_info = 1.

gtu_usage_logger_file

Options	Filename with extension .db
Default	usage.db

Defines the file name for the usage logger database.

gtu_usage_logger_path

Options	Path or foldername
Default	%gt_resource_folder%\utilities\usage_logger\

Defines the path in which the usage logger database (gtu_usage_logger_file) is written.

3D Note Form

gtu_3d_note_form_filter_hole_notes

Options	0 or 1
Default	1

Defines whether notes on holes are hidden when the dialog box is opened (1, filter is enabled) or not (0).

gtu_3d_note_form_filter_with_input_panels

Options	0 or 1
Default	1

Defines whether notes without input panels are hidden when the dialog box is opened (1, filter is enabled) or not (0).

gtu_3d_note_form_pagesize**Options** Any number >0**Default** 15

Define the number of shown lines per page

gtu_start_3d_note_form**Options** 0 or 1**Default** 1

Defines whether the button "GENIUS TOOLS 3D Note Form" is displayed. (0 - No, 1 - Yes)

Open Base Model

gtu_start_openGeomOrigin**Options** 0 or 1**Default** 1

Defines whether the button for "GT Open Base Model" is displayed. (0 - No, 1 - Yes)

Extend Relations

gtu_relationextension_calculateBoundingBox**Options** 0 or 1**Default** 1

Defines whether the commands GT_CalculateBoundingBox and GT_CalculateBodyBoundingBox are available in the relations.

gtu_relationextension_doubleToString**Options** 0 or 1**Default** 1

Defines whether the command GT_DoubleToString is available in the relations.

gtu_relationextension_round**Options** 0 or 1**Default** 1

Defines whether the command GT_Round is available in the relations.

gtu_relationextension_unitToString**Options** 0 or 1**Default** 1

Defines whether the command GT_UnitToString is available in the relations.

gtu_relationextension_updateBodyParamMass**Options** 0 or 1**Default** 1

Defines whether the command GT_UpdateBodyParamMass is available in the relations.

gtu_relationextension_updateBodyParamMaterial**Options** 0 or 1**Default** 1

Defines whether the command GT_UpdateBodyParamMaterial is available in the relations.

gtu_relationextension_updateBodyParamVolume**Options** 0 or 1**Default** 1

Defines whether the command GT_UpdateBodyParamVolume is available in the relations.

gtu_start_relationExtension**Options** 0 or 1**Default** 1

Defines whether the utility with relation extensions is available. (0 - No, 1 - Yes)

CS Assembler

gtu_csassembler_component_cs_name

Options Component name

Default PLACEMENT_CS

The name of the inser coordinatesystem from the component that should be assembled.

gtu_csassembler_maximal_cs_count

Options Any number ≥ 0

Default 5000

Defines the maximal used number of coordinate systems.

gtu_csassembler_multi_level

Options 0 or 1

Default 1

Defines whether coordinate systems from assembled components should be used as target coordinate systems inside the same assemble task.

gtu_csassembler_xml_path

Options Path or foldername

Default

Start path to choose a XML-file.

gtu_start_csassembler

Options 0 or 1

Default 1

Defines whether the the module CS Assembler gets started. (0 - No, 1 - Yes)

Select Surfaces by Color

gtu_start_selectSurfacesByColor

Options 0 or 1

Default 1

Defines whether the utility "GENIUS TOOLS Select Surfaces by Color" is available. (0 - No, 1 - Yes)

Component Parameters

gtu_comp_file_name

Options Path or foldername

Default gtu_component_parameters.xml

File name of configuration xml.

gtu_start_component_params

Options 0 or 1

Default 1

Defines whether the button "Component params" is displayed. (0 - No, 1 - Yes)

Load Save Converter

gtu_start_loadSaveConverter

Options 0 or 1

Default 1

Defines whether the utility "GENIUS TOOLS Load Save Converter" is available. (0 - No, 1 - Yes)

Points

gtu_points_autoselect_points_with_rule

Options Any regular expression

Default

Defines a rule to preselect Points at start up.

gtu_points_curve_chain_precision

Options Any real number between 0 and 1

Default 0.1

Defines the standard value for curve displays and results with `gtu_points_curve_output_type` in a definition for displaying curves.

gtu_points_curve_output_type

Options 0 to 3

Default 0

Defines the preselection for displaying curve points
0 - References
1 - Catenary Curve
2 - Absolute Length
3 - Relative Length

gtu_points_precision

Options C++ Float printf definition (e.g. .6 for 6 decimals)

Default 15.6

Defines the number of decimals.

gtu_points_remove_double_selected_points

Options 0 or 1

Default 1

Defines whether points that are defined twice should be removed from the saved file.

gtu_points_write_filename

Options Any text

Default

Defines the export file name. You can use variables.

gtu_points_write_ijk

Options 0 or 1

Default 0

Defines whether the normal to next surface should be written to the file.

gtu_points_write_template_extension**Options** Any text**Default**

Defines the file extension at the usage of templated export.

gtu_points_write_template_filenames_fill_up_length**Options** Any Number > 0**Default** 20

Defines the fill up length at usage from modelnames in templates.

gtu_points_write_template_footer**Options** Any Text**Default**

Defines a footer for the .dat file.

gtu_points_write_template_header**Options** Any Text**Default**

Defines a header for the .dat file.

gtu_points_write_template_line_left**Options** Any Text**Default**

Defines a line start for the .dat file.

gtu_points_write_template_line_right**Options** Any Text**Default**

Defines a line end for the .dat file.

gtu_points_write_template_modelnames**Options** 0 or 1**Default** 0

Defines if model names should be written into the template.

gtu_points_write_template_names**Options** 0 or 1**Default** 0

Defines whether names should be written to the output file.

gtu_points_write_template_names_fill_up_length**Options** Any number ≥ 0 **Default** 20

Fill up the name up to # characters.

gtu_points_write_template_names_split**Options** Any Text**Default** –

Replace the defined string through a space character.

gtu_points_write_template_seperator**Options** Any Letter**Default**

Defines the seperator for outputs with template.

gtu_points_write_use_template**Options** 0 or 1**Default** 0

Defines whether the gtu_points_write_template_* definitions should be used (can also be selected in the options tab at runtime).

gtu_start_points**Options** 0 or 1**Default** 1

Defines whether the button "GTU Points" is displayed. (0 - No, 1 - Yes)

Create search.pro***gtu_create_search_pro_exclude_current_path*****Options** 0 or 1**Default** 1

Defines whether the current working directory is included in the search.pro file (0) or not (1).

gtu_create_search_pro_exclude_file**Options** Filename**Default**

Defines a file containing search paths which are not included in the new search.pro file.

gtu_create_search_pro_line_start**Options** Any character**Default**

Defines characters which are inserted at the beginning of each line.

gtu_create_search_pro_path_end_with_slash**Options** 0 or 1**Default** 1

Defines whether a slash (/) (or backslash \) is added to the end of a file path (1) or not (0).

gtu_create_search_pro_standard_save_folder**Options** 0 to 2**Default** 0

Defines the standard save location. 0 - Creo standard directory, 1 - directory of the current object, 2 - current working directory.

gtu_create_search_pro_standard_save_name**Options** Filename**Default** search.pro

Defines the suggested file name of the search.pro file in the save dialog.

gtu_start_create_search_pro**Options** 0 or 1**Default** 1

Defines whether the button for "Create Search.pro" is displayed. (0 - No, 1 - Yes)

Edit Standard Texts

gtu_start_edit_default_texts**Options** 0 or 1**Default** 1

Defines whether the module to edit the default values is enabled. (0 - No, 1 - Yes)

Show Pitch

gtu_show_pitch_check_param**Options** Any text**Default** STEIGUNG

Defines a language-dependend feature parameter. If defined, it is checked if the parameter is existing. If the parameter does not exist, the function is unavailable.

gtu_show_pitch_check_param_fallback**Options** Any text**Default** PITCH

Defines a language-dependend feature parameter. If defined, it is checked if the parameter is existing. If the parameter does not exist, the function is unavailable.

gtu_show_pitch_text_definition**Options** Any text**Default** M @D x &STEIGUNG:FID_@feat_thread_id@[.2]

This value has to be customized for "Show Pitch" to work with the set Creo language.

gtu_show_pitch_text_definition_fallback**Options** Any text**Default** M @D x &PITCH:FID_@feat_thread_id@[.2]

This value has to be customized for "Show Pitch" to work with the set Creo language. The fallback is used if gtu_show_pitch_check_param can not be found.

gtu_start_show_pitch**Options** 0 or 1**Default** 1

Defines whether the button "Show Pitch" is displayed. (0 - No, 1 - Yes)

Table to CSV

gtu_start_table_to_csv**Options** 0 or 1**Default** 1

Defines whether the the module Table to CSV gets started. (0 - No, 1 - Yes)

gtu_table_to_csv_export_file**Options** Filename**Default**

Defines the file name when exported. May contain variables. If this value is empty, the regular file name will be used.

gtu_table_to_csv_export_folder**Options** Path or foldername**Default**

Defines the destination folder of the CSV file

gtu_table_to_csv_export_sep**Options** Any character**Default** ;

Defines the separator for writing CSV files.

gtu_table_to_csv_table_template**Options** Filename**Default**

Template for Table to CSV.

gtu_table_to_csv_write_file_as_utf8**Options** 0 or 1**Default** 1

Defines whether a CSV is written as ASCII (0) or UTF8 (1).

gtu_table_to_csv_write_file_as_utf8_with_bom**Options** 0 or 1**Default** 0

Defines whether the UTF8 file is written with (1) or without (0) Byte Order Mark (BOM).

Table to Excel and Copy Table 1:1 to Excel

gtu_start_table_to_excel**Options** 0 or 1**Default** 1

Defines whether the button "Copy Table to Excel" is displayed. (0 - No, 1 - Yes)

gtu_table_to_excel_autosave**Options** 0 or 1**Default** 0

Defines whether the Excel table to be created is written to the defined report folder without a save dialog in the one-to-one export. See configuration "gtu_table_to_excel_report_folder".

gtu_table_to_excel_copy_borders**Options** 0 or 1**Default** 1

Defines whether the table boundaries will be copied to the Excel table. (0 - No, 1 - Yes)

gtu_table_to_excel_erase_replaced_comments**Options** 0 or 1**Default** 0

Defines whether the comments in the Excel sheets will be deleted after filling (1) or not (0).

gtu_table_to_excel_name_by_template**Options** Any text**Default**

With the parameters and variables defined here, a name for the output file of "Table to Excel" is generated automatically.

gtu_table_to_excel_open_export**Options** 0 or 1**Default** 1

Defines whether a report should be opened after export.

gtu_table_to_excel_report_folder**Options** Path or foldername**Default**

Defines the standard directory for saving Excel reports.

gtu_table_to_excel_run_check**Options** 0 or 1**Default** 1

Specifies whether to check whether an Excel session is open.

gtu_table_to_excel_std_template_name**Options** Any text**Default**

Defines a name of an Excel template, selected at the start of the dialog.

gtu_table_to_excel_template_folder**Options** Path or foldername**Default** %gt_resource_folder%\utilities\table_to_excel\

Defines the path to the template directory for "Table to Excel" (.xlsx-files).

gtu_table_to_excel_use_model_instead_of_drawing_name**Options** 0 or 1**Default** 0

Defines whether the drawing name (0) or the model name (1) are used for naming the Excel table.

Create Tolerance Table

gtu_start_tolerance_table**Options** 0 or 1**Default** 1

Defines whether the button "Create Tolerance Table" is displayed. (0 - No, 1 - Yes)

gtu_tol_table_creo_insert**Options** 0 or 1**Default** 1

Defines the insertion of the tolerance table (0 - insertion without preview, 1 - insertion with preview on the mouse cursor).

gtu_tol_table_decimal_marker_follow_dtl**Options** 0 or 1**Default** 1

Reads the DTL file (1*) or not (0). If the DTL option decimal_marker = COMMA is set, the representation of numerical values will be changed from period to comma as decimal separator.

gtu_tol_table_fitsize_color**Options** 0,1,2,3,4,5,6,8,9,10 oder 12**Default** 9

Defines the color of the fitsizes in the tolerance table. The following colors can be used: 0 Red (PRO_COLOR_LETTER) 1 Green (PRO_COLOR_HIGHLITE) 2 White (PRO_COLOR_DRAWING) 3 Background (PRO_COLOR_BACKGROUND) 4 Grey (PRO_COLOR_HALF_TONE) 5 Blue (PRO_COLOR_EDGE_HIGHLIGHT) 6 Grey (PRO_COLOR_DIMMED) 8 Magenta (PRO_COLOR_ERROR) 9 Cyan (PRO_COLOR_WARNING) 10 Green (PRO_COLOR_SHEETMETAL) 12 Brown (PRO_COLOR_CURVE)

gtu_tol_table_fitsize_height**Options** Positive real number**Default** 2.5

Defines the text height of the tolerance table column "Fitsize".

gtu_tol_table_fitsize_text**Options** Any text**Default**

Defines the column name "Fitsize". If nothing is specified, the standard of GENIUS TOOLS is used, based on the current Creo language. You can split multi-line headings by using /n.

gtu_tol_table_fitsize_width**Options** 1-999**Default** 19

Defines the width of the fitsize column.

gtu_tol_table_font**Options** Any text**Default**

Defines the font of the tolerance table. The Creo standard font is used, if no value is set.

gtu_tol_table_form**Options** A or B**Default** A

Defines the type of the tolerance table. The available table configurations can be found in the online help.

gtu_tol_table_hidden_font**Options** Any text**Default** isofont

Defines the font for the hidden table header. If no font is defined the header would get the drawing font.

gtu_tol_table_lower_text**Options** Any text**Default**

Defines the column name "Minimum". If nothing is specified, the standard of GENIUS TOOLS is used, based on the current Creo language. You can split multi-line headings by using /n.

gtu_tol_table_minimal_pending_zeros**Options** 0 to 999**Default** 3

Defines the minimum number of characters of a tolerance value which are displayed.

gtu_tol_table_only_dims_from_current_sheet**Options** 0 or 1**Default** 1

Defines whether all dimensions (0) or only the dimensions of the current sheet (1) are displayed in the tolerance table.

gtu_tol_table_origin_at_bottom_right**Options** 0 or 1**Default** 0

Defines the origin of the table and its growth direction: 0 - The origin of the table is at the top left and its growth direction to the bottom right. 1 - The origin of the table is at the bottom right and its growth direction to the top left. These configuration did affect the table placement only with gtu_tol_table_creo_insert = 0.

gtu_tol_table_show_diameter**Options** 0 or 1**Default** 1

Defines whether a diameter sign is displayed on diameter tolerances. (0 - No, 1 - Yes)

gtu_tol_table_show_plus_at_positive**Options** 0 or 1**Default** 0

Defines whether a plus sign is displayed on positive numbers (1) or not (0).

gtu_tol_table_show_prefix**Options** 0 or 1**Default** 1

If the prefix sign should be shown or not

gtu_tol_table_show_sign_before_zero**Options** Any text**Default**

If a table cell contains only a zero, the text defined here is displayed before this zero.

gtu_tol_table_sort_order**Options** ASC or DESC**Default** ASC

Defines whether the table is sorted in ascending or descending order.

gtu_tol_table_std_height_plus**Options** Positive real number**Default** 2.5

Defines the cell height with the inner text height (text height + standard height = cell height).

gtu_tol_table_text_color**Options** 0,1,2,3,4,5,6,8,9,10 or 12**Default** 0

Defines the text color of the tolerance table. The following colors can be used: 0 Red (PRO_COLOR_LETTER) 1 Green (PRO_COLOR_HIGHLIGHT) 2 White (PRO_COLOR_DRAWING) 3 Background (PRO_COLOR_BACKGROUND) 4 Grey (PRO_COLOR_HALF_TONE) 5 Blue (PRO_COLOR_EDGE_HIGHLIGHT) 6 Grey (PRO_COLOR_DIMMED) 8 Magenta (PRO_COLOR_ERROR) 9 Cyan (PRO_COLOR_WARNING) 10 Green (PRO_COLOR_SHEETMETAL) 12 Brown (PRO_COLOR_CURVE)

gtu_tol_table_text_height**Options** Positive real number**Default** 2.5

Defines the text height of the tolerance table column tolerance.

gtu_tol_table_tolerance_color**Options** 0,1,2,3,4,5,6,8,9,10 or 12**Default** 0

Defines the color of the tolerances in the tolerance table. The following colors can be used:

- 0 Red (PRO_COLOR_LETTER)
- 1 Green (PRO_COLOR_HIGHLIGHT)
- 2 White (PRO_COLOR_DRAWING)
- 3 Background (PRO_COLOR_BACKGROUND)
- 4 Grey (PRO_COLOR_HALF_TONE)
- 5 Blue (PRO_COLOR_EDGE_HIGHLIGHT)
- 6 Grey (PRO_COLOR_DIMMED)
- 8 Magenta (PRO_COLOR_ERROR)
- 9 Cyan (PRO_COLOR_WARNING)
- 10 Green (PRO_COLOR_SHEETMETAL)
- 12 Brown (PRO_COLOR_CURVE)

gtu_tol_table_tolerance_height**Options** Positive real number**Default** 2.5

Defines the text height of the tolerance table column header.

gtu_tol_table_tolerance_minimal_pending_zeros**Options** -1 till 9**Default** 2

Defines the number of decimal places for the values in the column Tolerance. 0: No decimal places are displayed. 1...9: Number of decimal places. The last digit will be rounded if necessary. -1: Decimal places are shown without alteration, i. e. no zeros are added.

gtu_tol_table_tolerance_text**Options** Any text**Default**

Defines the column name "Tolerance". If nothing is specified, the standard of GENIUS TOOLS is used, based on the current Creo language. You can split multi-line headings by using /n.

gtu_tol_table_tolerance_width**Options** 1-999**Default** 19

Defines the width of the tolerance column.

gtu_tol_table_upper_text**Options** Any text**Default**

Defines the column name "Maximum" is defined. If nothing is specified, the standard of GENIUS TOOLS based on the current Creo language is used. You can split multi-line headings by using /n.

Open/Create Drawing***gtu_ord_copy_common_name_on_drw_create*****Options** 0 or 1**Default** 1

Defines whether the PTC_COMMON_NAME remains the standard name of the drawing file (0) or the PTC_COMMON_NAME is copied from the model (1).

gtu_ord_createdrw**Options** Any text**Default**

Defines a mapkey which is started after the create drawing dialog is opened. If a submapkey should be used please write it like %%mapkey_name;.

gtu_ord_drw_name**Options** Any Text or Empty**Default** @mdlname@

Defines the name of the drawing that should be created or opened. The use of variables is possible. If empty, the WT number generator is used. Definition e.g. :

\$\$repl\$MO-\$DW-\$@mdlname@\$

gtu_ord_ignore_name_for_common_name**Options** 0 or 1**Default** 1

0 - The common name is set as expected, 1 - If the common name is the same as the model name, it will not be set.

gtu_ord_pdm_auto_open_one_drawing**Options** 0 or 1**Default** 0

PDM - If there is only one drawing it should be opened

gtu_ord_pdm_close_dialog_after_show_url**Options** 0 or 1**Default** 1

PDM - Dialog get closed after display of the drawing details page

gtu_ord_pdm_file_name_attribute**Options** Any text**Default** objCadModelName

Defines the Windchill attribute in which the file name will be found. It may be differ depending on the specific Windchill installation and version. (e.g. name, number, objCadModelName)

gtu_ord_pdm_look_for_used_parts**Options** 0 or 1**Default** 0

Defines whether drawings should also be searched that reference the current part as a subpart (REST API support required).

gtu_ord_pdm_show_Thumbnails**Options** 0 or 1**Default** 1

Defines whether thumbnails from Windchill (when using REST API) should be downloaded (1) or not (0).

gtu_ord_pdm_shown_attributes**Options** Any text**Default**

Type in additional attributes to be shown. These must be deposited in Windchill. This may be parameters of other object attributes (e.g. status, version,...) Type in the ids of the attributes as comma separated values (e.g. "REVISION,VERSION,STATUS" or "IBA|REVISION,IBA|VERSION,IBA|STATUS")

gtu_ord_pdm_sort_out_doubles**Options** 0 or 1**Default** 1

PDM - Drawings that are inside the resource and drawing model list get filtered

gtu_ord_post_drw_name**Options** Any text**Default**

Defines a suffix for the file name of drawings. = PREFIX + + SUFFIX + .DRW

gtu_ord_pre_drw_name**Options** Any text**Default**

Defines a prefix for the file name of drawings. = PREFIX + + SUFFIX + .DRW

gtu_ord_try_to_use_selected_part_if_inside_asm**Options** 0 or 1**Default** 1

Defines whether the select model request is shown every time (0) or only if no model is already selected (1).

gtu_ord_use_pdm_server**Options** 0 or 1**Default** 1

Try to use Windchill plugin while connected with Windchill

gtu_start_open_create_drawing**Options** 0 or 1**Default** 1

Defines whether the button for "Open/Create Drawing" is displayed. (0 - No, 1 - Yes)

gtu_tol_table_sort_order**Options** ASC or DESC**Default** ASC

Defines whether the table is sorted in ascending or descending order.

Full Backup

gtu_start_fullbackup**Options** 0 or 1**Default** 1

Defines whether GENIUS TOOLS Full Backup can be started by users. (0 - No, 1 - Yes)

Command Control

gtu_command_control_configuration

Options NAME_1:MODE_1|NAME_2:MODE_2|NAME_3:MODE_3

Default ProCmdMdlTreeWfChkInExp:UNAVAILABLE

Defines the commands and the defined modes:NAME_1:MODE_1|NAME_2:MODE_2|NAME_3:MODE_3Mode:UNAVAILABLEINVISIBLEDISALLOW

gtu_start_command_control

Options 0 or 1

Default 0

Defines whether GENIUS TOOLS Command Control would be startet. (0 - No, 1 - Yes)

Work Dir Manager

gtu_start_work_dir_manager

Options 0 or 1

Default 1

Defines whether GENIUS TOOLS Work Dir Manager can be started by users. (0 - No, 1 - Yes)

gtu_work_dir_manager_always_at_front

Options 0 or 1

Default 0

Definies whether the dialog stays on top, or falls behind others.

gtu_work_dir_manager_autodelete_after_days

Options Any Number > 0

Default 1

Defines the number of days until paths get deleted from list.

gtu_work_dir_manager_first_chars**Options** Any Number**Default** 10

Defines the number of characters from left side to shorten the path for display.

gtu_work_dir_manager_last_chars**Options** Any Number**Default** 25

Defines the number of characters from right side to shorten the path for display.

gtu_work_dir_manager_save_path**Options** Path**Default** %appdata%\INNEO\GENIUS_TOOLS\for_Creo\work_dir_manager\

Defines the path for system files.

19 Variables

The following variables can be used in various GENIUS TOOLS for Creo applications and can be combined with regular text.

Example: The specification `@date@_Project5_@mdl@` resolves to `2017-16-02_Project5_prt0001`.

If a variable cannot return a value, the `gt_replace_character_if_not_found` configuration option determines the returned value. If the configuration option is not set, the variable name will be returned as entered.

User input

Instead of using a variable which is resolved automatically, you can also use text input by a user.

To cause a prompt (text box) to be displayed to the users, use the format `==LabelText==`. The text enclosed by double equal signs is displayed in the input prompt. This text, including the equal signs, is then replaced by the text entered by the user.

Example for generating a file name:

`@date@_==Please enter descriptive name==_@mdl@`

Date and time information

Text operation	Description	Example
@date@	Returns the current date in yyyy-mm-dd format to facilitate sorting.	2017-13-02
@datede@	Outputs the current date in German dd.mm.yyyy format to facilitate sorting.	13.02.2017
@dateen@	Output the current date in English dd-mm-yyyy format.	13-02-2017
@dateus@	Output the current date in American mm-dd-yyyy format.	02-13-2017
@dateusshort@	Output the current US-American format, short notation mm-dd-yy.	02-13-17

Text operation	Description	Example
@time@	Output the current time in a notation that can also be used for filenames, short notation HH-MM (0-23)-(0-59).	14-15
@timede@	Output the current time in German format, short notation HH:MM (0-23):(0-59).	14:15
@timeen@	Output the current time in US-American format, short notation hh:MM a (1-12):(0-59) AM/PM.	02:15 AM

Object information

Text operation	Description	Example
@curworkdir@	Returns the current working directory.	c:\temp
@clipboard@	Returns the current buffer memory.	
@common@	Returns the common name.	
@count_form@	Returns the number of internal forms of a model.	
@mdlIn@	Returns the current model name.	
@mdlpath@	Returns the current directory of the current model.	
@mdlpathr@	Returns the current directory of the current model with a slash instead of a backslash ("/" instead of "\").	
@genname@	Returns the generic name of the current instance.	
@feat_id@	Returns the feature ID of the currently selected feature	7400
@filename@	Returns the file name.	

Text operation	Description	Example
@filepath@	Returns the current directory of the model.	
@fileversion@	Returns the current version of the file with prefixed "." (number following the file suffix e.g.: test.prt.2).	.1
@mdlno@	Returns the current version of the file without prefixed "." (number following the file suffix e.g.: test.prt.2).	1
@mdltype@	Returns the type of a model. (part, assembly, drawing)	PRT
@selmdl@	Returns the name of the currently selected model.	
@selmdlpath@	Returns the path of the currently selected model.	

Variables with limited usage

Some variables can only be used in certain GENIUS TOOLS components.

Text operation	Description	GENIUS TOOLS Modul	Beispiel
@counter@	Generates a continuous number.	Quick Access, Library, Name Generator	
@number@		Quick Access, Library, Name Generator, Multibody to Assembly	
@is_embedded@	Returns "1" for embedded models, otherwise "0".	Value Transfer	

Text operation	Description	GENIUS TOOLS Modul	Beispiel
@oldname@		Quick Access, Library, Name Generator, Multibody to Assembly	

Drawing information

Text operation	Description	Example
@pageno@ or @sheetno@	Output the current page on the drawing.	
@sheetimagestyleclass="Default" scale@	Output the current page base imagestyleclass="Default" scale of the drawing.	2:1
@sheetformat@	returns the (paper) format of the current drawing (A0-A4 in compliance with DIN 476 or A-F in compliance with ANSI A).	A3 or B
@maxpage@	Output the total number of pages in the current drawing.	

Parameter information

Text operation	Description	Example
%PARAM%	Returns the value of a parameter. The input between the percent signs may vary.	%DESIGNATION%
%curmod:PARAM%	Returns a parameter value of the current model of a drawing into a drawing. The replacement of PARAM may vary.	% curmod:DRAWINGNUMBER%

Text operation	Description	Example
	Outside of drawings this notation does not make sense.	
%dubase:PARAM%	Returns a double Value in e^ notation instead of the regular output rounded to 6 decimal places. This can also be used in combination with <i>curmod</i> and <i>curmat</i> . The order % curmat:dubase:PARAMETER% must be observed.	% dubase:DOUBLE_PARAMETER%
\$env-var\$	Outputs an environment variable of Model Processor/from Windows.	\$USERNAME\$ \$COMPUTERNAME\$ \$HOMEDRIVE\$ \$LOGONSERVER\$ \$USERDOMAIN\$

Text operations for variables

A regular expression cannot contain string operations.

The first character of the count always has position 0.

Text operation	Description	Example
\$\$sub\$x\$y\$TEXT\$\$	Cuts an input out of a text (TEXT) starting with x and ending with y; x and y must be numeric values. Make sure the text is of appropriate length.	\$\$sub\$2\$5\$@para@\$
\$\$pre\$x\$TEXT\$\$	Sub-string from start (character at position 0) till character x.	
\$\$pree\$x\$TEXT\$\$	Sub-string from start (character at position 0) till character x before the end of the string.	
\$\$poste\$x\$TEXT\$\$	Sub-string with length x, measured from the end of the string. Is	

Text operation	Description	Example
	equivalent to the last x characters of the string.	
\$\$posts\$x\$TEXT\$\$	Sub-string till the end, starting at character x. Thus is equivalent to variable number of output characters.	
\$\$repl\$x\$y\$TEXT\$ \$	Replace x with y in the text. TEXT can be any text or another variable.	

20 Regular Expressions

Use Regular Expressions in GENIUS TOOLS for Creo to check value inputs or allow only rule-compliant inputs to be saved.

Character	Description
\	Indicates the following character as a special or verbatim character. For example "n" is corresponds to the character "n". "\n" corresponds to a line-break character. The sequence "\\" corresponds to "\", "(" corresponds to "(".
^	Corresponds to the beginning of the input.
\$	Corresponds to the end of the input.
*	Corresponds to the proceeding character zero or multiple times. For example "zo*" matches either "z" or "zoo".
+	Corresponds to the proceeding character one or multiple times. "zo+" for example matches "zoo", but does not match "z".
?	Corresponds to the proceeding character zero or one time. For example "a?ve?" matches the "ve" in "never".
.	Corresponds to all single characters except for a line-break character.
(Pattern)	Matches Pattern and saves the equivalent. The compared substring can be retrieved from the resulting matches listing using the elements [0]...[n]. For comparing of characters put in parentheses () use "(" or "\".
x y	Corresponds to either x or y. For example matches "l red" either "l" or "red". "(l r)ed" matches "led" or "red".
{n}	n is a positive integer. Corresponds to exactly n times. "o{2}" for example does not match the "o" in "Robert" but the first two "o"s in "Boooooat".

Character	Description
<code>{n,}</code>	<code>n</code> is a positive integer. Corresponds to at least <code>n</code> times. " <code>o{2}</code> " for example does not match the " <code>o</code> " in " <code>Robert</code> " but all " <code>o</code> "s in " <code>Boooooat</code> ". " <code>o{1,}</code> " is equivalent to " <code>o+</code> ". " <code>o{0,}</code> " is equivalent to " <code>o*</code> ".
<code>{n,m}</code>	<code>m</code> and <code>n</code> are positive integers. Corresponds to at least <code>n</code> and maximum <code>m</code> times. For example " <code>o{1,3}</code> " matches the first three " <code>o</code> "s in " <code>Boooooat</code> ". " <code>o{0,1,}</code> " is equivalent to " <code>o?</code> ".
<code>[xyz]</code>	A group of characters. Corresponds to any of the included characters. " <code>[abc]</code> " for example matches the " <code>a</code> " in " <code>falling</code> ".
<code>[^xyz]</code>	A group of excluded characters. Corresponds to any character not included. " <code>[^abc]</code> " for example matches the " <code>f</code> " in " <code>falling</code> ".
<code>[a-z]</code>	A character range. Corresponds to any character in the specified range. For example, " <code>[a-z]</code> " matches any lowercase alphabetic character in the range from " <code>a</code> " to " <code>z</code> ".
<code>[^m-z]</code>	An excluded range of characters. Corresponds to any character not included in the specified range. " <code>[m-z]</code> " for example matches all characters not included in the range from " <code>m</code> " to " <code>z</code> ".

Examples

Regular expression	Description	Example
<code>[a-zA-Z]*_[a-zA-Z]*</code>	Any alphabetic string with an underscore	<code>user_tbx</code>
<code>[0-9]{5}</code>	Five random numbers	<code>12345</code>
<code>^. {7}\$</code>	7 random characters	<code>t_p.prt</code>
<code>^[A-Z]{1}[a-z]{2,10}</code>	A capital letter at the beginning followed by two to ten lower case letters	<code>Tuser</code>

Regular expression	Description	Example
[0-9]{2}\.[0-9]{2}\.[0-9]{4}	Date format	01.08.1975

21 Available Arithmetic Operations

You can use arithmetic operations and mathematical functions in different modules of GENIUS TOOLS for Creo.

These operations differ in notation from the default Creo calculations. To use an arithmetic operation in an input field, start with the equal sign, then enter the mathematical function.

Examples

=12-d4

=d42/23

Please note: Units like degree (in angular dimensions) or the length information are ignored. Only the values are used for calculations.

Mathematical Function	Description	Example
+	Addition	=d27+5
-	Subtraction	=12-d4
/	Division	=d42/23
*	Multiplication	=d31*3
Math.sqrt(x)	Square root	=Math.sqrt(9)
Math.pow(x,y)	Exponentiation (x to the power of y)	=Math.pow(d2,3)
Math.abs(x)	Absolute value	=Math.abs(-5)
Math.round(x)	Round	=Math.round(2.565)
Math.ceil(x)	Round up to next integer	=Math.ceil(3.6)
Math.floor(x)	Round down to previous integer	=Math.floor(3.4)

22 Frequently Asked Questions

On this page you find a summary of frequently asked questions on GENIUS TOOLS for Creo and possible solutions.

General questions

What happens if the license for GENIUS TOOLS for Creo gets lost?

If the floating license (e.g. due to Windows hibernation mode) is lost, all GENIUS TOOLS applications and their functions are deactivated. To avoid loss of data all windows stay opened. As soon as the license is regained you can continue working as usual. Regaining a license from the network may require some time.

Why are not all GENIUS TOOLS for Creo apps and editors displayed in the GENIUS TOOLS ribbon?

GENIUS TOOLS for Creo is a product group divided into different products. Depending on the product purchased, only the included modules are displayed.

The GENIUS TOOLS modules are mode-dependent. Check if you are in the proper Creo mode!

Make sure that all required Creo licenses have been loaded. Even if you can open parts and assemblies, the GENIUS TOOLS for Creo ribbon will only be displayed with the corresponding Creo license available.

The GENIUS TOOLS modules and editors can also be deactivated individually with configuration settings.

Excel connection does not work. What can I do?

Make sure Microsoft Office 2010 or newer is installed. In many cases the Excel connection also works with older Office versions (minimum 2007) but some GENIUS TOOLS functions require Office 2010; therefore proper operation of GENIUS TOOLS for Creo with older Office versions cannot be guaranteed.

If the error message *Failed to initialize Microsoft Excel* is displayed in the Creo log, this can be due to a corrupted Office installation. Try to execute the office repair option.

Specific questions about GTfC modules

Library: Umlauts in my library object status are not displayed. Is there a solution?

Check the *gt_library.cfg* and the *gt_headerless_files_are_utf8* configuration options in *gt_main.cfg*. As the status names are specified via configuration options it is possible that the CFG file was not saved correctly and that the umlauts are misinterpreted.

Name Generator: My Name Generator database is not used automatically in my Quick Access mapkey. I always have to select the correct name configuration in the selection dialog. What can I do?

Do you have multiple name configurations with similar file name? The filter rules do not search for unique equivalents. For instance, if you have two name configurations (*FileName.db* and *FileName_User.db*), and a mapkey with `number:FileName`, the selection dialog will always appear. Take care of uniqueness at name configuration designation. Rename the name configurations (e.g. *FileName_Global.db* and *FileName_User.db*) and specify the uniquely identifiable name configuration in the mapkey.

This behavior also occurs in environments with local and global name configurations.

Parameter: I receive a save/regenerate error; what solutions are available?

Have you activated the units relation in Creo? Check the Creo configuration option `relations_num_const_units`. Try to specify values with units (10[kg] instead of 10).

Check the info icons in the parameter form and in the tabs and check the status signal light. Pay attention to mandatory parameters!

Open/create drawing: I receive an error message that a drawing already exists even though it does not open via Open/create drawing.

This error may occur if drawing models or frames for a drawing are missing. Unfortunately, the repair dialog cannot be opened due to an API error. You have to open the drawing manually.

23 Copyright

Copyright 2023 by:

INNEO Solutions GmbH

IT-Campus 1

73479 Ellwangen

Germany

This documentation is protected by copyright. All rights reserved.

Without prior written consent of an authorized representative of INNEO Solutions GmbH it must not be copied, photocopied, reproduced, translated, communicated or converted to electronic or machine readable form in whole or in part.

The unauthorized use of the documentation can lead to a claim for liquidated damages or legal prosecution. INNEO Solutions GmbH does not accept liability for possible faulty information in this documentation and the consequences resulting from such.

Note on registered trademarks:

Most of the software, hardware and trade names mentioned in this documentation are also registered trademarks of the respective software manufacturers.

Registered trademarks and trade names of INNEO Solutions GmbH:

GENIUS TOOLS, Startup TOOLS, INNEO

- 3 -

3D Model compare 193

- A -

Action 181
 ASCII file 498
 Assembly 184, 493
 complete assembly 185
 create 294
 export 294
 from bodies 294
 standard part with selection table 207
 Assembly Report
 configure position attributes 57
 configure table view 56
 define head parameters 55
 define number ranges 58
 define table attributes 57

- B -

Batch mode 240
 Bodies to disband 294

- C -

Check revision status 396
 Close
 all other windows 477
 Color selection
 select surfaces by color 537
 Color selector 590
 Commands
 ribbon menu commands hide/disable 478
 Common name 26, 300, 328
 Component
 assemble automatically 493
 Component parameter 479
 create 483
 Components
 replace 95
 suppress 95
 Configuration concept 551
 Configuration layers 551
 overwrite 568
 Configuration Options 593
 environment variables 574

paths 573
 save 569
 setting values 566

Configuration overview 561
 Creo Advanced Assembly Extension (AAX)
 GENIUS TOOLS Multibody to Assembly 294
 Creo object types
 search 190

- D -

Databases
 default texts 495
 material database 268
 multilingual 213, 495
 update 576
 Dimension 81, 524, 542
 DIN 6770 129
 Directory 546
 Drawing
 create 536
 open 536

- E -

Embedded data 592
 Environment variables 555, 568, 574
 ERP system 212
 Excel 25, 111, 129, 461, 506, 509, 695

- F -

Family table 437
 add dimensions 81
 import 254
 Family table variants
 name assignment 306
 Form 86, 93
 Forms
 use 204

- G -

General Configuration Options
 Assembly Report 602
 Dimension 606
 Forms 607
 Inspect 610
 Library 615
 Material 635

General Configuration Options

- Multibody To Assembly 638
- Name Generator 640
- Parameter 641
- Quick Access 648
- UDF Forms 652
- Utilities 654
- GENIUS TOOLS Inspect Revision 169
- GENIUS TOOLS Library 11
- GENIUS TOOLS Model Processor 321
- GENIUS TOOLS Parameter 11
- GENIUS TOOLS Startup TOOLS 573
 - environment variables 555
 - working directory 555
- Global name configuration 313
- Glossary
 - GENIUS TOOLS Assembly Report 26
 - GENIUS TOOLS Forms 86
 - GENIUS TOOLS Inspect 129
 - GENIUS TOOLS Inspect Revision 169
 - GENIUS TOOLS Library 183
 - GENIUS TOOLS Parameter 328
 - GENIUS TOOLS UDF Forms 437

- I -

- Information overview 538
- Inspection symbols 129
 - hide/unhide 136
 - number 136
 - position 133, 136

- J -

- JavaScript 48, 116, 396, 455, 577

- L -

- Library 181
 - create 251
 - optimize search 258
 - read in additional information 258
- Library category
 - create 252
- Library objects
 - create 252
 - update 256
- License 22, 695
- Local name configuration 313

- M -

- Mapkey 85, 222, 416
 - intelligent 306, 320, 321, 405
- Mass
 - assign 345
 - calculate 345
- Material
 - Material version check 288
- Materials
 - assign 274
 - check 288
 - check in a model 279
 - delete 277
 - disable Update materials function 278
 - filter material properties 272
 - material directory 283
 - material properties 272
 - search 273
 - update 278
 - value 272
 - version 288
- MNU export 243
- Model
 - Variants 93
- Model properties
 - change 93
- MTL files 291
- MUI files
 - import 265
- Multibody 294

- N -

- Name configuration
 - global 313
 - local 313
- Name Generator
 - Fallback 316
 - Integration in Quick Access 320
 - Mapkey 320
 - Windchill 318
- Naming 306
- Notes
 - edit multiple notes 471
 - with calculations 473
- Number ranges 25, 50

- O -

Objects
 back up 527
 Object collector 244
 Overwrite
 configuration layers 568

- P -

Parameter 25, 396
 add 359
 create table parameters 42
 parameter definitions 355
 parameter management concept 329
 reading parameter values from text files 338
 use of variable parameters 437
 Parameter values
 copy in symbols 163
 link in symbols 163
 Parts list
 bulk bill of materials 25
 multi-level bill of materials 25
 removing single lines 65
 single-level bill of materials 25
 PDM system 212
 Pitch
 show 542
 PLM system 212
 Points
 export 498
 Position number
 position number assignment 25
 Position PreSave rule 49
 programm editors
 access 575
 PTS file 498

- Q -

QAEX file 422
 Quick Access
 mode dependency 407
 selection dependency 408
 Quick Access Editor
 mapkey definition 416

- R -

Regeneration time 526
 Regular Expressions 349, 691
 Relations
 extend 519
 Report definition 26
 automatic update 62
 create 54
 external 28
 internal 28
 Reports 26
 create 510
 export 25
 REST API 368
 Revision 169
 Revision parameter 176
 Rounding according to IEEE 754 519

- S -

Save
 configuration options 569
 objects from older Creo versions in currently
 used version 533
 Search.pro 487
 Shortcut 405
 Snapshot
 create 173
 Start mechanism 553
 start switches 575
 Startup TOOLS 11
 environment variables 555
 working directory 555
 Status dialog 338
 Status indicator 338
 Subscription license 22
 GENIUS TOOLS Multibody to Assembly 294
 SYM file 133, 545
 Symbol definition attributes 163
 Symbol variants
 toggle 545
 Symbols
 switch between all variants of a grouped
 symbol 545

- T -

Tables

Tables

- export 136
- export a placed drawing table 507
- export a report table 509
- export One-to-One 506
- export without template 506
- fill and export via template 507
- hide/unhide 136
- position 136
- tolerance table 491

Template 25, 169, 184, 186, 503, 517

- create template for export 512

Tolerance table

- create 491

Tooltip 406

- U -

UDF

- as layer 453
- place dependently 443

UDF Forms

- use 204

UDF groups

- create 437
- edit subsequently 437

- V -

Value Transfer 430

Value-dependent unit specifications on drawings 396

Variables 685

- date and time information 685
- drawing information 688
- object information 686
- parameter information 688
- text operations 689
- with limited usage 687

- W -

Windchill 45, 176, 181, 212, 291, 314, 328, 368, 536

- using position values 63

Working directory 546, 555

- Z -

Zoom 425