INNEO®
That's IT.

GENIUS TOOLS®

Model Processor

Version 10.0.0.0

Help and documentation

© 2023 INNEO Solutions GmbH

www.inneo.com

1 About Model Processor 8

2 News 10

2.1 News in Release 10.0.1.0 10
2.2 News in Release 10.0.0.0 11
2.3 News in Release 9.0.1.0 12
2.3.1 News in Release 9.0.1.1 14
2.4 News in Release 9.0.0.0 14
2.5 News in Release 8.0.1.0 15
2.6 News in Release 8.0.0.0 16
2.7 News in Release 7.0.1.0 17
2.8 News in Release 7.0.0.0 20

3 Program window 23

4 Interface - Model Processor UI 26

5 Action list 29

5.1 Actions 30
5.1.1 Environment 32
5.1.1.1 Clear all windows 32
5.1.1.2 Erase undisplayed models 33
5.1.1.3 Grouping 34
5.1.1.4 Request user input 36
5.1.1.5 Save configuration options 37
5.1.1.6 Set configuration options 38
5.1.1.7 Set environment variable 39
5.1.1.8 Set user variable 40
5.1.1.9 Set user variable from URL 41
5.1.1.10 Set working directory 43
5.1.1.11 Start JavaScript 44
5.1.1.12 Start MSDOS command 46
5.1.1.13 Start mapkey 48
5.1.1.14 Start sync app function 50
5.1.1.15 Start trail file 51
5.1.1.16 Write file 52
5.1.1.17 Write to Creo Log 54
5.1.2 PTC Windchill PDM 54
5.1.2.1 Add file to workspace 54
5.1.2.2 Check in model 55
5.1.2.3 Check out model 56
5.1.2.4 Clear workspace 57
5.1.2.5 Remove file from workspace 58
5.1.2.6 Set server active 58

5.1.3 Model ... 59
5.1.3.1 Add 3D annotation ... 59
5.1.3.2 Backup action ... 61
5.1.3.3 Cancel feature position ... 62
5.1.3.4 Compute mass ... 62
5.1.3.5 Copy model ... 63
5.1.3.6 Create DM code as dot matrix ... 64
5.1.3.7 Create QR code as dot matrix ... 66
5.1.3.8 Edit PTC_COMMON_NAME (common name) ... 68
5.1.3.9 Edit model dimension from file ... 70
5.1.3.10 Embed Models ... 72
5.1.3.11 Insert first features ... 72
5.1.3.12 Load tolerance table ... 74
5.1.3.13 Regenerate model ... 75
5.1.3.14 Remove feature ... 76
5.1.3.15 Rename feature ... 77
5.1.3.16 Rename model ... 79
5.1.3.17 Replace a model ... 79
5.1.3.18 Rescale feature positions in assemblies ... 81
5.1.3.19 Save model ... 82
5.1.3.20 Set accuracy ... 83
5.1.3.21 Set all dimension bounds ... 84
5.1.3.22 Set feature insert position ... 85
5.1.3.23 Set model options (DTL) ... 86
5.1.3.24 Set style ... 87
5.1.3.25 Set tolerance ... 88
5.1.3.26 Set units ... 89
5.1.3.27 Set user variable from bounding box ... 91
5.1.3.28 Start ModelCHECK ... 93
5.1.4 Model - Views ... 94
5.1.4.1 Add views ... 94
5.1.4.2 Rename view ... 96
5.1.4.3 Set view ... 96
5.1.5 Model - Relations ... 97
5.1.5.1 Add relations ... 97
5.1.5.2 Add feature relations ... 99
5.1.5.3 Edit relations ... 100
5.1.5.4 Edit feature relations ... 103
5.1.5.5 Delete relations ... 105
5.1.5.6 Delete feature relations ... 107
5.1.5.7 Set relations unit sensitive ... 109
5.1.6 Model - Export ... 110
5.1.6.1 Export PARASOLID file ... 111
5.1.6.2 Export Creo View file ... 112

5.1.6.3 Export IGES file ... 113
5.1.6.4 Export JPG file ... 115
5.1.6.5 Export PDF/U3D file ... 116
5.1.6.6 Export SAT file ... 117
5.1.6.7 Export STEP file ... 119
5.1.6.8 Export STL file ... 120
5.1.7 Model - Family table ... 121
5.1.7.1 Delete the family table instances ... 121
5.1.7.2 Split up family table ... 122
5.1.7.3 Unlock family table instances (all) ... 123
5.1.7.4 Verify family table ... 124
5.1.8 Model - Appearance ... 125
5.1.8.1 Set appearance ... 125
5.1.8.2 Set appearance from RGB ... 126
5.1.9 Model - Layer ... 129
5.1.9.1 Add layer ... 129
5.1.9.2 Add layer state ... 131
5.1.9.3 Delete layers / Move features ... 132
5.1.9.4 Delete layer state ... 134
5.1.9.5 Group layers ... 135
5.1.9.6 Rename layers ... 135
5.1.9.7 Set layer state ... 137
5.1.9.8 Set layer status ... 138
5.1.10 Model - Combined state ... 139
5.1.10.1 Add combined state ... 139
5.1.10.2 Delete combined state ... 141
5.1.10.3 Set combined state ... 142
5.1.11 Model - Explode state ... 143
5.1.11.1 Add explode state ... 143
5.1.11.2 Delete explode state ... 143
5.1.11.3 Set explode state ... 144
5.1.12 Model - Cross Section ... 145
5.1.12.1 Activate cross section ... 145
5.1.12.2 Add cross section ... 146
5.1.12.3 Delete cross section ... 147
5.1.12.4 Show cross section ... 148
5.1.12.5 Update cross sections in the model ... 149
5.1.13 Model - Material ... 150
5.1.13.1 Add material ... 150
5.1.13.2 Add material parameter ... 151
5.1.13.3 Delete material ... 153
5.1.13.4 Edit material parameter ... 154
5.1.13.5 Edit material (from file) ... 155
5.1.13.6 Replace material ... 157

5.1.14 Model - Parameter 159
5.1.14.1 Add alternative model parameter to family table 159
5.1.14.2 Add feature parameter 161
5.1.14.3 Add parameter 162
5.1.14.4 Add associative parameter to flexible dimension 164
5.1.14.5 Change all designated flags 165
5.1.14.6 Delete feature parameter 166
5.1.14.7 Delete model parameter 167
5.1.14.8 Edit drawing parameter from model 168
5.1.14.9 Edit model parameter 170
5.1.14.10 Edit model parameter from file 174
5.1.14.11 Edit parameter from dictionary 175
5.1.14.12 Edit parameter from table cell 178
5.1.14.13 Edit parameter from symbol text 180
5.1.14.14 Fill component parameter by reference 181
5.1.14.15 Rename parameter 182
5.1.15 Model - Pro/PROGRAM 183
5.1.15.1 Delete Pro/PROGRAM input lines 183
5.1.15.2 Delete Pro/PROGRAM 185
5.1.16 Model - Simplified Representation 186
5.1.16.1 Create a simplified representation 186
5.1.16.2 Delete simplified representation 188
5.1.16.3 Edit simplified representation 188
5.1.16.4 Rename simplified representation 189
5.1.16.5 Set simplified representation 190
5.1.17 Drawing 191
5.1.17.1 Edit model notes from file 191
5.1.17.2 Remove all annotations 193
5.1.17.3 Remove unused drawing models 194
5.1.17.4 Replace format 194
5.1.17.5 Replace view model display 197
5.1.17.6 Set current drawing model 198
5.1.17.7 Set current sheet scale 199
5.1.17.8 Show unbend features on views 200
5.1.18 Drawing - Export 200
5.1.18.1 Export 2D PDF file 200
5.1.18.2 Export DWG file 202
5.1.18.3 Export DXF file 203
5.1.18.4 Export TIFF file 204
5.1.19 Drawing - Note 205
5.1.19.1 Add a note 206
5.1.19.2 Delete a note 207
5.1.19.3 Move a note 209
5.1.20 Drawing - Symbol 211

5.1.20.1 Add symbol ... 211
5.1.20.2 Switch symbol group variant ... 213
5.1.20.3 Create bar code (PDF 417) ... 215
5.1.20.4 Create DM code symbol ... 216
5.1.20.5 Create QR code symbol ... 218
5.1.20.6 Delete a symbol ... 219
5.1.20.7 Remove symbol paths ... 221
5.1.20.8 Move a symbol ... 222
5.1.20.9 Update symbol definitions ... 223
5.1.21 Drawing - Table ... 224
5.1.21.1 Add a table ... 224
5.1.21.2 Delete a table ... 226
5.1.21.3 Move a table ... 227
5.2 Model filter ... 229
5.2.1 Assembly - Explode state exists ... 231
5.2.2 Assembly - Style state exists ... 232
5.2.3 Drawing - Current model ... 233
5.2.4 Drawing - Dimension text ... 234
5.2.5 Drawing - Format name ... 234
5.2.6 Drawing - Formats on sheets ... 235
5.2.7 Drawing - Is multi model drawing ... 236
5.2.8 Drawing - Sheet size ... 237
5.2.9 Drawing - Symbol on drawing ... 238
5.2.10 Drawing - Symbol in drawing ... 238
5.2.11 Drawing - Table on drawing ... 239
5.2.12 Environment - Creo version ... 240
5.2.13 Environment - Environment variable ... 241
5.2.14 Environment - File exists ... 241
5.2.15 Environment - Model exists ... 242
5.2.16 Environment - User Parameter ... 243
5.2.17 Model - Dimension tolerance ... 244
5.2.18 Model - Double defined parameters ... 244
5.2.19 Model - File name ... 245
5.2.20 Model - Is regenerateable ... 246
5.2.21 Model - Is modified ... 246
5.2.22 Model - Layer ... 247
5.2.23 Model - Model subtype ... 248
5.2.24 Model - PDM object ... 249
5.2.25 Model - Parameter ... 250
5.2.26 Model - Parameter type ... 251
5.2.27 Model - Path ... 252
5.2.28 Part - Current material name ... 252
5.2.29 Part - Is Skeleton ... 253
5.2.30 Part - Material parameter ... 254

5.2.31 Solid - Accuracy ... 255
5.2.32 Solid - Combined State ... 256
5.2.33 Solid - Contains circular dependencies ... 256
5.2.34 Solid - Cross Section exists ... 257
5.2.35 Solid - Feature exists ... 258
5.2.36 Solid - Feature from type ... 259
5.2.37 Solid - Feature is flexible ... 260
5.2.38 Solid - Feature suppressed ... 260
5.2.39 Solid - Geometry check exists ... 261
5.2.40 Solid - Instance with name ... 262
5.2.41 Solid - Is generic ... 263
5.2.42 Solid - Is insert mode active ... 263
5.2.43 Solid - Is instance ... 264
5.2.44 Solid - Layer state exists ... 265
5.2.45 Solid - Relation ... 266
5.2.46 Solid - Simplified representation exists ... 267
5.2.47 Solid - Tolerance ... 267
5.2.48 Solid - Unit system ... 268
5.2.49 Solid - Unregenerated instances ... 269
5.2.50 Solid - View exists ... 270
5.3 Reports ... 270
5.3.1 Open CSV report ... 272
5.3.2 Report dependencies ... 274
5.3.3 Report drawing data ... 276
5.3.4 Report table cell from drawing ... 279
5.3.5 Report family table ... 282
5.3.6 Report feature positions ... 283
5.3.7 Report feature parameters ... 284
5.3.8 Report geometry properties ... 285
5.3.9 Report layers ... 286
5.3.10 Report material ... 287
5.3.11 Report model data ... 288
5.3.12 Report parameters ... 291
5.3.13 Report Pro/PROGRAM ... 292
5.3.14 Report relations ... 294
5.3.15 Report universal ... 295
5.3.16 Write a list report ... 297
5.3.17 Write informative PDF ... 301
5.4 Variables ... 306
6 Mode configuration 314
6.1 Batch processing configuration ... 316
6.1.1 Extended options ... 319
6.1.2 Check for duplicated files ... 320

6.1.3 Searching in the file list ... 321
6.1.4 Modifications of the file base ... 321
7 RegExp - Regular Expressions 323
8 Mapkeys 325
9 Project Explorer 327
9.1 Model Processor Zip - MPZ ... 329
10 Project properties 332
11 Best practice 336
12 FAQ 338

1 About Model Processor

What is Model Processor?

Model Processor supports you in processing large amounts of data or complex tasks

– time-saving

– repeatable

– traceable

– transparent

The key areas of usage are

– Model analyses

– Model overhauls

Applications

Powerful tool to analyze Creo Parametric data

– Qualitative statements about model settings in master data

– Requirement for expenditure assessment for PDM introductions

– Requirement for a realistic assessment of suggested CAD work regulations

Powerful tool to overhaul Creo Parametric data

© 2023 INNEO Solutions GmbH

– Support in the PDM introduction process (master data, library adaptation, subsequent parameter update, ...)

– Support for the reinforcement of CAD work regulations

– Improvement of the model data quality

Use in interactive and automated batch processing mode possible.

Program philosophy

1. Models are overhauled by a steadily increasing number of actions. The tool should relieve the operator of as many tasks as possible.
2. Operating the individual actions should be as easy and efficient as possible (beginners and advanced users, enhanced filters, ...).
3. Large data volumes should be overhauled without user intervention over night.
4. Logs provide information on Model Processor activities at any time.
5. Each setting and privilege should be available at any time all over the world (provided that internet is available).

Typical sequence of a model overhaul

Why are these individual steps necessary?

– Many models are modified in a short time.

– Frequently, the effect of a modification is not immediately obvious.

  – Example: Changing the material can lead to invalid relations (MATERIAL=MATERIAL_PARAM("CONDITION)).

– Individual actions offer complex definitions.

  – Example:"NO" to be inserted at the 7th place of the "CLASSIFICATION" parameter of all models that contain a "PARTTYPE" parameter with the value "*NORM".

© 2023 INNEO Solutions GmbH

2 News

You can see the changes in context of the previous version.

2.1 News in Release 10.0.1.0

Model Processor

- **New actions:**
  - Remove feature
  - Run sync app function
  - Embed models
  - Run JavaScript
  - Write to file
  - Remove file from Workspace

move Set model options (DTL) from group "Environment" to "Model"

- **Functional extensions of actions:**
  - Group action - all model components (recursive)
  - Group action - all simplified representations
  - Start ModelCHECK - Show Report added
  - Set View - Refit after set view added
  - Group action - added "also on current model" for instances, components and dependencies
  - Request user input - Now 20 description lines are possible
  - Split up family table - The family table head can now be preserved
  - Add relations - check for line length now give only a warning if a comment line would be longer than 80 letters
  - Set user input from URL - XSLT support added
  - Set user input from URL - maximum feedback Variable length extended

© 2023 INNEO Solutions GmbH

- **Functional extensions of filter:**
  - Accuracy - Variable support added

- **Fixed failures in actions:**
  - Delete ProProgram input parameters - Crash on deeper temp directories
  - Set tolerance - Lost extended filters at load in MP
  - Start ModelCHECK - Crash on drawings

- **New variables**:
  - @creoname:prt@
  - @creoname:asm@
  - @creoname:drw@
  - @wtservername@
  - @wtserverurl@
  - @wtserverws@

Model Processor User

Now deletes locks on intermediate generics.

- **New configurations:**
  - mpu_dll_id
  - mpu_dll_id_file

2.2 News in Release 10.0.0.0

Model Processor

Creo 10.0 Support

- **New actions:**
  - Suppress feature

- **Fixed failures in actions:**

© 2023 INNEO Solutions GmbH

- Edit parameter from dictionary - Solve failure at different memory constellations (Crash)

- **New variables**:
  - time_with_seconds - Time with seconds (H-MM-SS)

Model Processor User

- **New configurations:**
  - mpu_write_reports_in_time - Write the report at every model change, not at the end of MPU run

2.3 News in Release 9.0.1.0

Model Processor

Added support for Creo +

Support of WT SolidWorks-Import improved

Count problems at batch mode solved

Preload models for batch mode added - WARNING! this configuration also affects other run modes.

Control characters \u0001 and \u0002 in relations get now replaced by (CS) to avoid the destruction of the xml save file.

Change / Unify of terms that are used in the tasks: Export STEP file, Export PDF file and Export 2D PDF file

- **New actions:**
  - Export PARASOLID file
  - Export STL file
  - Edit parameter from symbol text
  - Remove symbol paths
  - Add feature relations
  - Edit feature relations
  - Delete feature relations
  - Create DM code smybol

© 2023 INNEO Solutions GmbH

- Create DM dot matrix

- **New variables:**
  - %conf:CONFIGURATIONNAME% - returns the value of a Creo config option

- **Functional extensions of actions:**
  - Add model relations - Error message by lines that are longer than 80 letters are added
  - Add feature Parameter - Add to family table support
  - Delete Material - Need graphical representation to update the material list inside the model
  - Set user variable from URL - Proxy Support
  - Report drawing data - added symbol paths
  - Report drawing data - added detail note fonts
  - Report drawing data - implemented detail notes
  - Update symbol definitions - New option for better scaling and controlling the update
  - Export PDF file - added output mode (U3D, U3D PDF, Named Views)

- **Functional extensions of filters:**
  - Model - Creo Version - Supports now Creo 9.0 and Creo 10.0
  - Model - File name - supports now variables

- **Fixed failures in actions:**
  - Edit model dimension from file - Solve . problems by the use of file names
  - Delete model relations - IF / ENDIF detection improved

- **Fixed failures in filters:**
  - Model - PDM Object - Failure at PDM Objekt Checked out solved
  - Model - File name - Support for regular expressions fixed

Model Processor User

Change of the model after finish a run with a huge log tree fixed.

© 2023 INNEO Solutions GmbH

2.3.1 News in Release 9.0.1.1

Model Processor

- **Funktionale Erweiterungen an Funktionen:**
  - Gruppierung - all Model components (recusive / in session) - nur nicht unterdrückte
  - Farbeffekt setzen von RGB - Unterstützung für Oberflächenparameter hinzugefügt

- **Fehlerbehebungen in Funktionen:**
  - CSV Report - doppelte Spalte "File name"
  - CSV Report - fehlende Reportausgaben bei doppelungen von Reportaktionen in Gruppierungen
  - Start MSDos Kommando - Fehlerausgabe wenn Kommando > 255 Zeichen
  - Loop durch Umgebungsvariablennutzung behoben

Model Processor User

2.4 News in Release 9.0.0.0

Model Processor

Official Support for Creo 9.0 added.

- **Fixed failures:**
  - Add ASM / PRTs if file names / file types written in upper case.
  - Failure while read the log files if notes have line breaks solved.
  - Bug at groups with B number higher than 100 solved.

- **Functional extensions of actions:**

© 2023 INNEO Solutions GmbH

- Edit model parameter from file - Old Param ### can now replace the file name as selector for the line
- Add feature parameter - Support of variables for values

- **Fixed failures in actions:**
  - Add a note - Bug at read of multiline files solved
  - Add a note - Bug at selection of notes to read the position solved
  - Add feature parameter - Change the Headlines of the Table
  - Edit model parameter from file - Bug at the read from csv files (crash if file is not present)

2.5 News in Release 8.0.1.0

Model Processor

- **Functional extensions of actions:**
  - Grouping - Filetype can now be set for the Import of STEP/STP
  - Grouping - New Support for IGES as other model (PRT, ASM, DRW).
  - Set appearance from RGB - Filter features by parameters has not a consistence.
  - Grouping - Remove loaded / created model from session
  - Checkout Model - Add instance level

- **Fixed failures in actions:**
  - Export STEP file - Replace Options by config-file.
  - Write a list report - Failure at the report of instance parameters solved.
  - Add layers - Failure at the creation of layers with a template without a Creo defined default_layer_model solved (only first model).

© 2023 INNEO Solutions GmbH

2.6 News in Release 8.0.0.0

Model Processor

Added official support for Creo 8.0.

- **New actions:**
  - Write Creo Log
  - Add assoziative parameter to a flexible dimension

- **Neue filter:**
  - Model - Is modified
  - Model - Dimension tolerance

- **New variables:**
  - @regen_status@ - return the model regeneration status

- **Functional extensions of actions:**
  - Create backup - Reset model path through second save
  - Create a simplified representation - added Default Envelope
  - Edit simplified representation - added Default Envelope
  - Group - RMB-menu has now the same actions as the body action list
  - Set user variable from URL - Support user credentials with variables
  - Set user variable - File / Folder path - start directory variable support

- **Functional extensions of filters:**
  - Model - Parameter - Parametername now supports variables
  - Model - Is regenerateable - force regeneration

- **Fixed failures in actions:**
  - Set user variable - Bug at the file ending support at load / save option solved
  - Set material - Set material without path definition

© 2023 INNEO Solutions GmbH

- Set dimension from CSV - File name column changed
- Check in model - Drawing filter wasn't saved
- Check out model - Drawing filter wasn't saved
- Delete relations - If there was a \ as last character on the last line Creo crashed
- Report - New line creation at model change in MP Batchmode works now also if the model is renamed inside the action list.

- **Fixed failures in filters:**
  - Environment - Environment variable - Fixed the comparison with the use of variables

- **Fixed failures in variables:**
  - @mdltype@ - Can now also be used for drawings
  - §ENVIRONMENT VARIABLE§ - can now be used inside the head area

Model Processor User

- Fasten up the subsequent works:
  - Assemble the log tree
  - Clean up the allocated memory
- Crash at multiple model selection (current model is an assembly) but not the current assembly solved

Model Processor Worker Extension

- Change of the trigger behavior (after_open).
- Change of the trigger behavior (occupy_post).

2.7 News in Release 7.0.1.0

Hotfix 01

- Fixes failure inside variable replacement
  - Environment variables
  - double %% to single %

© 2023 INNEO Solutions GmbH

- String replacement

Model Processor

- Support for DXF and Inventor imports
- Failure for the count of done actions fixed
- Changelog removed and added to the help files
- Windchill version flag fixed
-
- **Removed functionality:**
  - Set simplified representation for parts, because the functionality isn't supported by Creo API

- **New Actions:**
  - export 2D PDF file

- **New Variables:**

@mdltype@ - Output of the current model type (File extension)

@workdir:<PATH>@ - If a trigger is used in PTC Creo View Adapter visualization this variable can returns the right subfolder inside the pubtemp / transfer folder. This variable should used favoured with GENIUS TOOLS Model Processor Worker Extension.

- **Functional enhancements on Actions:**
  - Set user variable from URL - CSRF Token can be transfered
  - Request user input - File / Folder - Dialog
  - Export pdf file - Use of an export profile
  - Set drawing format - Variable support added
  - Set layer status- Variable support added
  - Add 3d annotation - Variable support added
  - Group action - Group act on simplified representations
  - Insert first features - can now add axis

- **Functional enhancements on Filters:**

© 2023 INNEO Solutions GmbH

- Model - Parameter - can now work on explicit parameters (PTC_ ...) that are not inside the parameter list. (e.g. PTC_COMMON_NAME)
- Environment - User variable - Variable support added
- Environment - Environment variable - Variable support added

- **Fixed failures in Actions:**
  - Set appearance from rgb - Filter at feature parameter is fixed
  - Group action - Dependencies - Failure at second call is fixed
  - Report Model data - simplified representation at parts
  - Edit parameters from file - fixed

- **Fixed failures in filters:**
  - Model - Parameter - Can now check boolean parameters
  - Drawing - Sheet size --> < and > are switched, old definition would be updated automatically

Model Processor User

- New Product Model Processor Worker Extension added for the enhancement of PTC Creo View Adapter in the visualization step of PTC Windchill

- Instances filter and current model fixed filter in combination
- Delete of the UNDO / REDO stack at runtime fixed
- Change of the function at the after_open trigger. It now triggers after open, but before show.
- main.cfg - Cleaned up and new configurations added
- Minimize Button added
- Dialog order fixed, so the info dialog is now before the main dialog
- Automatically unlock of PTC Windchill locked models

© 2023 INNEO Solutions GmbH

2.8 News in Release 7.0.0.0

Model Processor

- Creo 7.0 Support
- Reports are now core functionality and can be used in MP and MPU
- Mapkeys and Report-Actions can be used inside of Groups

- **Removed functionalities:**
  - Remove of the online licensing
  - Remove of the change variable from the UI
  - Remove of the support for asynchronous mapkeys (only MP) from the run mapkey action
  - Remove of the action Report to xls
  - Remove of the action Edit parameter from file history

- **New actions:**
  - Solid - Rename view
  - Solid - Fill component parameters by reference
  - Drawing - Remove all annotations
  - Drawing - Remove unused drawing models
  - Drawing - Show unbend features on views
  - Report - Report feature parameters
  - Report - Report feature positions

- **Functional extensions of actions:**
  - Add views - SMT Flat view
  - Write informative PDF - Show errors in relations
  - Report layers - Can now report layers from drawings
  - Report drawing data - Amount of sheets based on sheet size
  - Report drawing data - Amount of shown dimensions
  - Report drawing data - Amount of views

© 2023 INNEO Solutions GmbH

- Report drawing data - Amount of symbols
- Report model data - Amount of features
- Report model data - Flexible features
- Edit simplified representations - Include / Exclude of features by name or type
- Add simplified representations - Add to assemblies
- Group - Over all sheets - Option to set the model if the current model has no defined sheet
- Request user input - File / Folder - Dialog
- Add combined state - Set the simplified representation MASTER
- Add combined state - define note and geometry status
- Add material / Material filter - Creo 7.0 support

- **New variables:**
  - nozero: - Cut pending zeroes from floating-point numbers
  - @curdrwmdltype@ - type from the current drawing model
  - @cursheetsize@ - sheet size from the current sheet
  - @cursheetcurmdlscale@ - sheet scale from current sheet
  - @cursheetcurmdlscaleISO@ - sheet scale from current sheet (in ISO notation)
  - @cursheetname@ - current sheet name
  - @cursheetformat@ - current sheet frame name
  - @oid@ - Windchill OID

- **New filters:**
  - Model - Double defined parameters

- Reports can now grow in 2 dimensions
- Change of the used Java-Version
- Update the help to a new system

Model Processor User

- Reports are now core functionality and supported in MPU

© 2023 INNEO Solutions GmbH

- Support special charakters in paths (to MPZ files)
- Support for borrowing together with GT Starter

© 2023 INNEO Solutions GmbH

3 Program window

Once Model Processor has been installed, it can be started via its icon. Model Processor starts only when Creo (Pro/ENGINEER) is running. If this is not the case, a window appears that tells you that Model Processor is waiting.

An active connection to Creo is not established if the waiting screen is skipped. Model Processor must be restarted if the connection to Creo is to be established later.

Model Processor automatically connects with the running Creo. Creo may only be started in a single instance.

Waiting window

Main program window

The Model Processor program window is independent of Creo. This means that the window is - like any other Windows program window - in the background when it is not active.

© 2023 INNEO Solutions GmbH

The current state of the connection to Creo is shown in the bottom right corner of Model Processor. Connection states:

	A connection to Creo could not be established, or the connection was disconnected.
	A connection to Creo is established.
	A connection to Creo is established. Creo is connected to Intralink.
	A connection to Creo is established. Creo is connected to Product Point.
	A connection to Creo is established. Creo is connected to Windchill.

It is possible that a submodule can not cooperate with some PDM solutions, although Model Processor has identified states.

Model Processor menu

© 2023 INNEO Solutions GmbH

Select the logo in the upper left corner to navigate to the Model Processor menu. In the Model Processor menu, you can (as you can do with F1) open Model Processor Help, modify various program parameters, select the language, and update Model Processor.

© 2023 INNEO Solutions GmbH

4 Interface - Model Processor UI

1. Project functions
2. Model selection and mode configuration
3. Action lists
4. Action details
5. Log pane

Project functions

The project functions contain the control elements that control the overall workflow in the Model Processor. The project functions also contain the MPZ management components.

– Starts a run over all selected models

– Aborts a started run

– Checks all actions for the existence of specified folders and files

– Creates a new Model Processor project

– Loads a new Model Processor project (MPZ)

– Saves the current Model Processor project

– Saves the current Model Processor project as a backup copy

© 2023 INNEO Solutions GmbH

– Starts a management UI to process and manage the MPZ contents

– Permits PDM information and project information to be viewed and modified

– Starts a test environment for RegExp 323

– Opens the help on the help page that corresponds to the current circumstances

Saving and creating a backup copy

Saving automatically saves into the specified file. A file name need only be entered if the project does not yet have a file name assigned. After file name and path have been entered, the file is saved under the specified name. The file can be renamed as required. The file name is always retained during loading so that the file is always saved under the file name that was used for invoking the MPZ file.

Create a backup copy creates a file with an additional date key in the file name (e.g. *filename_20130370-093051.mpz*). This file name convention is hidden when a Model Processor project file is opened. You can find it when you switch the file type over to *backup copy* in the open dialog. When you select this file type, ONLY backup copies are shown here. After you have opened a backup copy, note that saving it will overwrite the current project file. The backup copy never overwrites itself. It is thus always retained as a secured version.

Mode configuration

A run always includes a data base and an action list. A run is not possible if one of the two is missing. In mode configuration, you can use 4 different possibilities to select the data base that is to be used for the run.

– Current model - the actions of the action list are only applied to the current model in Creo

– All in - all models that are currently in the session / workspace are listed and overhauled one by one in succession. A restriction of the processed models on the basis of the file suffix is possible.

– Batch processing - here, you can load and overhaul the data base from a PDM system or from the hard disk. If you want to save the overhauled data, you must include a corresponding action.

– Create new model - creates a new model to which the actions are applied.

More details about this topic can be found under Mode 314.

Action lists

Use the button in the action list area to create a new action list. The new action list requires a name. It has 3 different areas.

– Header area - is folded in at the beginning of a project. It contains all actions that must be processed ONCE at the beginning of a run.

© 2023 INNEO Solutions GmbH

– Body area - is performed repeatedly for each model. Each action in the list is applied to each file that is determined by the data base in the run configuration.

– Footer area - like the header area, the footer is executed ONCE. In contrast to the header area, it is executed at the end of the run after all files of the data base have been processed.

More details about this topic can be found under Action list 29.

Action details

The part that shows the action details is a part of the action lists. The action configuration in this area opens whenever an action is selected in an action list.

Logs

Four different logs are written for the standard configuration.

– Overview - has one entry per processed file. It shows the number of actions that were processed properly, and the number of errors/warnings that occurred in the related files.

– Log - is the largest log. It contains a lot of information in addition to the complete information in the other three logs. Writing this log takes a long time, however. It is expedient to use it during tests and during runs on the current model. For runs using a large data base, it should be deactivated for performance reasons. It is the only log that is updated in parallel. It can therefore be written even after the run proper has already been completed.

– Warnings - all warnings that occurred during the run are listed here.

– Errors - all errors that occurred during the run are listed here.

© 2023 INNEO Solutions GmbH

5 Action list

User Interface

There is a drop-down box on the UI that permits the action lists in the MPZ 329 to be opened. Merely select an action list from the list. The currently open action list is saved automatically when you open a new action list.

Further actions:

– Creates a new action list

– Opens the current XML viewer or Internet Explorer, and shows the action list in a print-friendly view

– Displays the action list information and offers options for renaming the action list

– Deletes the selected action list

Keyboard options

Inside the lists (Head, Model, Foot) as well as inside groups the following keys can be used:

Arrow up	Selects the element above in the action list

© 2023 INNEO Solutions GmbH

Arrow down	Selects the element below in the action list
Page up	Selects the element one scrollbar length above
Page down	Selects the element one scrollbar length below
Home	Selects the first element in the list
End	Selects the last element in the list
DEL	Deletes the selected element
Insert	Adds a new element below the selected element
*(Numblock)	Enables / Disables the selected element

Action lists and action groups

Actions are joined in action lists. An action list can be applied once (header and footer area) or repeatedly (body area - once to each model). Actions are subdivided into groups (topics) in order to permit the correct action to be found rapidly when an action is added.

All 3 action lists, that is,

– header area - is processed once prior to model overhaul

– body area - is processed repeatedly; once for each model

– footer area - is processed once after model overhaul

can be saved in an XML file together, and thus be reused.

5.1 Actions

Action types

Model Processor actions are small blocks that perform certain actions with models. Actions can have very different tasks. The type of an action is shown by color.

© 2023 INNEO Solutions GmbH

Action properties

Each action has properties. The description exists with all actions. All the other properties depend on the concrete action. For example, the properties of the *Add parameter* action are names, values, types, ... of the parameters that are to be added.

The help on the pages of the individual actions contains more information about the individual properties of the actions.

Most actions can additionally be restricted by filters. If, for example, a function should only be executed in a part with a parameter *OTTO*, you can use a filter to define this restriction.

More about filter possibilities can be found in the section Filters 229.

Action messages

Each action has outputs. An output can be of the following types:

– Log information

– Warnings

– Error messages

Each output is of the structure: Date; time; action number, description

The action number consists of a letter and a number. The letter identifies the list (H...header, B...body, F...footer). The number identifies the position in the list.

© 2023 INNEO Solutions GmbH

Guest mode

In guest mode, you can create only one action list per MPZ.

5.1.1 Environment

This group combines actions. Please choose an action.

5.1.1.1 Clear all windows

The action closes all windows and purges the session if desired.

© 2023 INNEO Solutions GmbH

General description

Clear all windows

Description:

1 ☑ Erase undisplayed models

This task is ignored in mode "current model"

1. Models that are not displayed are removed from the memory after all windows have been closed.

This action closes all open Creo windows. The action should not be used in the modes *Current model* and *Create new model*. When the checkbox *Erase undisplaye models* is active, models that are no longer displayed after the windows have been closed are removed from the session.

MP User: This action is not executed here.

5.1.1.2 Erase undisplayed models

The action erases all models that are not displayed from the session.

General description

Erase undisplayed models

Description:

This task is ignored in mode "all in ...".

The *Erase undisplayed models* action removes all models that are not displayed from the memory. It thus corresponds to the Creo function.

In most lists, this function only makes sense when it is used in the header or footer area.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.1.3 Grouping

The action groups multiple actions under one action. Applied filters have effect on the operating performance of sub-actions.

General description

1. Expand or collaps the group action in the parent list
2. Load another action list into the group
3. Automatically update a loaded action list if the original list is changed
4. Grouped sub-actions
5. Definition of model to be overhauled or current model

The *Group* action groups multiple actions (sub-actions) into a single one. If execution of the *Group* action is suppressed by general or advanced filters, execution of all sub-actions will be suppressed as well.

Groupings can be nested within each other. You can modify how a grouping is displayed using the ▼ button (1):

Expanded

© 2023 INNEO Solutions GmbH

Collapsed

Actions can be moved between the individual groupings and the main windows using drag and drop. At this, it should be noted that the visible area of an expanded action cannot be dragged and dropped or selected.

If another action list has been loaded (2) into the group (take care to avoid recursive calls), use the auto-update function to update that list ON LOADING the action.

If the action list is to be used with MPU, please note that the auto-update is only done in MP. If you change an action list loaded into a group, open the parent action list again, then save the MPZ to get the changes.

Mode setting (5)

– *currently reworking model*
  Continue work on the model currently being overhauled.

– *all instances (at drawing current model instances)*
  All instances of a generic will be processed with the action in the list (4). For drawings, this refers to the instances of the current drawing model.
  recursive defines whether sub-instances should also be processed, or only instances from the first level.

– *a new drawing created*
  A new drawing is created in memory. The drawing can be defined via a template and is then used as the current model for the actions from the list (4).
  If you do not explicitly save the drawing (Save model) 82, it will be discarded at the end of the action list.

– *all model dependencies (in session)*
  The actions in the list (4) are executed on all models loaded into the session as dependencies.

– *for loop*
  A loop repeats the grouping. A counter starts at 0 and runs for the specified number of times. If you set the counter to 1, the loop runs once. You can also use a user variable in the counter that can be changed at loop runtime (see Set user variable) 40.

– *other model*
  Specify a file name that defines the other model for which the actions from the list (4) should be executed. Variables can be used in specifying the file name (see Variables 306).
  You can define the Import type for STEP and IGES. It make sense to set the hidden config option *intf3d_in_as_part* = *yes* (read also https://community.ptc.com/t5/Creo-Parametric-Ideas/Fix-Import-STEP-Assembly-as-Part-Components-in-Wrong-Position/idi-p/660685).

– *all sheets (Model selected through frames)*
  The program iterates through all sheets of a drawing and executes the actions from the list

© 2023 INNEO Solutions GmbH

(4) for each sheet.
Single-model drawings: The current drawing model remains set.
Multi-model drawings: Via the drawing frame, the program tries to identify the model that was active when the drawing frame was set. This model is then set as the active model in the drawing ant the action list (4) executed for the sheet. When all sheets have been processed, the active model for the drawing is reset.

– custom simplified representation
It would called a defined simplified representation. All group action(4) would be used on this.

5.1.1.4 Request user input

The action opens a window to query MP/MPU variable which afterwards can be utilized with %user:VARIABLE_NAME%.

General description

1. Name of the user input
2. Description of the user input
3. Type of the user input
4. Specification of the selected values / default value of the variable

© 2023 INNEO Solutions GmbH

The *Request user input* function opens a window and requests an input from the user. This input can be specified in more details via the action.

The input over all elements can always be used wherever variables can be used (except reports). The following syntax must be observed:

`%user:VARIABLE_NAME%`, z. B. `%user:VISUALLY_CHECKED%`.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.5 Save configuration options

The action saves specified configuration options under a name. The saved values can be reset with Set configuration options 38.

General description

1. Name for the list of keys and values
2. Specification of the keys that are to be saved

© 2023 INNEO Solutions GmbH

The *Save configuration options* action saves configuration options in a table structure in the memory, that can later be reset with the action Set configuration options 38. The action stores keys and values as pairs. Just select the name of a value list in the setting action. To create a value list, specify a name in the text box (1). Only the keys are specified under *Values* (2), as the values for the keys are read from Creo.

Information about description and model filters can be found under Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.6 Set configuration options

The action sets a specified Creo configuration option to a specified value or sets values from a previously saved list.

General description

1. Select the name of a saved value list if you want to write back stored data
2. Specify key-value pairs

The *Set configuration options* action can carry out two different activities. Using lists from the Save configuration options action, it can reset key-value pairs to the previously saved

© 2023 INNEO Solutions GmbH

values. This requires the corresponding value list to be selected in the selection box (1). Furthermore, you can specify new key-value pairs in the table (2) that are to be set.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.7 Set environment variable

The action sets the value of an environment variable.

General description

1. Name of the environment variable
2. Value of the environment variable

The action *Set environment variable* sets an environment variable (1) to a defined value (2) in the Creo memory. The variable can then be used by other synchronous applications or from Creo.

For information on using variables in text fields (1+2), pleaser refer to Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.8 Set user variable

The action sets a user variable to a value without asking for user input.

General description

1. Name of the variable that can later be accessed as `%user:NAME%`
2. Value of the variable
3. Increment the current variable value by 1

The action *Set user variable* sets a value that is NOT DEFINED IN CREO in Model Processor. The value can later be accessed as `%user:NAME%` in all fields that support variables.

`%user:NAME%` will resolve to the value of the user variable. If you activate the increment setting (3), the value will be transformed in to a number that can be incremented.

Here are example results for an increment of 1:

1 => 2

2inch => 3

inch => 1

As text strings cannot be transformed into number, an error will be output as 0, then incremented to 1.

See also:

Request user input 36

© 2023 INNEO Solutions GmbH

Set user variable from URL 41

Set user variable from bounding box 91

Please note that user variables will be deleted after each Model Processor or Model Processor User run. All values not explicitly assigned and saved will be lost.

For information on using variables in text fields (1), pleaser refer to Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.9 Set user variable from URL

The action sets a user variable to a value without asking for user input.

© 2023 INNEO Solutions GmbH

General description

1. Name of the variable that can later be accessed as `%user:NAME%`
2. The URL that returns the value
3. User name and password if they are required to access the URL
4. PTC Windchill CSRF Token for ODATA REST API would be requested and send for the defined Request
5. Define a Proxy Server if needed for the connection
6. Post Variable that would be send to the server. You can use b64 and b64uc inside of it (look also at Variables 306)

The action *Set user variable from URL* sets a value that is NOT DEFINED IN CREO in Model Processor. The value can later be accessed as `%user:NAME%` in all fields that support variables.

`%user:NAME%` will resolve to the value of the user variable.

The URL (2) specified for requesting the value may contain variables and thus model-specific information. An XSLT parser for accessing the *WC 11.0 REST API* is not supported at the moment.

If the URL (2) requires user authentication, you can store user name and password under (3).

© 2023 INNEO Solutions GmbH

The value returned by the URL is shortened to 200 characters, or at the first Enter (\n) or CarriageReturn (\r) character, then saved in the user variable.

If you want to assign the value to a parameter later on, please note that parameters only support a value length of 80 characters.

See also:

Request user input 36

Set user variable 36

Set user variable from bounding box 91

Please note that user variables will be deleted after each Model Processor or Model Processor User run. All values not explicitly assigned and saved will be lost.

Should a Request be send to PTC Windchill ODATA REST API, which need a CSRF Token, so this setting (4) could request the Token and use it while sending the file to the REST API.

If you want to send a file or informations inside the post variable (e.g. JSON) to the server, you can do this by the variable declaration(6). If you want to embed files you can use the Base 64 / Base 64 uc support. The maximum size of post variables are 5 MB.

For information on using variables in text fields (1,2,6), pleaser refer to Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.10 Set working directory

The action sets the current Creo working directory to this folder.

© 2023 INNEO Solutions GmbH

General description

1. The folder to which should be switched.

The action sets the current Creo working directory to the specified folder (1). Variables may be used in the specification.

Information about variable entries for the folder (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.1.11 Start JavaScript

Die Aktion führt ein JavaScript in einem ECMA Interpreter aus um Nutzervariablen zu verarbeiten.

© 2023 INNEO Solutions GmbH

Allgemeine Beschreibung

1. Der Name der Variablen, welche später mit `%user:NAME%` angesprochen wird in die die Rückgabe "return value" geschrieben wird.
2. Der JavaScript Quelltext zur Verarbeitung von Variablen.

Die Aktion führt ein JavaScript aus. Hierbei wird für das JavaScript der Befehlssatz aus dem ECMAScript 5.1 (2015+) eingesetzt.

Folgende Befehle stehen zusätzlich zur Verfügung

getUserVar("VARIABLE") - holt den Wert einer Nutzervariablen

setUserVar("VARIABLE", "WERT") - setzt den Wert einer Nutzervariablen

log_error("MESSAGE") - schreibt in das Log als Fehler

log_warning("MESSAGE") - schreibt in das Log als Warnung

log_message("MESSAGE") - schreibt in das Log als Loginformation

alert("MESSAGE") - schreibt in das Log als Loginformation

© 2023 INNEO Solutions GmbH

Alle Eingaben sollten per getUserVar geholt werden und per setUserVar geschrieben werden.

Siehe auch:

Nutzereingabe abfragen 36

Nutzereingabe von URL setzen 41

Nutzereingabe von Bounding Box befüllen 91

Achtung: Nutzereingaben werden am Ende des Laufs von Model Processor / User gelöscht. Somit sind alle nicht explizit zugewiesenen Werte dann verloren.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	X	
Teil - Instance	X	
Baugruppe	X	
Baugruppe - Instance	X	
Zeichnung	X	

5.1.1.12 Start MSDOS command

The action executes a system command.

© 2023 INNEO Solutions GmbH

General description

1. Text box to specify an MSDOS command that is to be started, including parameters.
2. Text box to insert arguments. If you use Option (3) you have to define all arguments in these.
3. Start the Process as a parallel process without waiting on its end.
4. Use this selection to specify that the command is executed before the file is opened. Works only if the command is at location B0.
5. Waiting time in seconds after the asynchronous start of the command from (1).

A command that is to be executed in an MSDOS shell can be entered in the text box (1) of the *Start MSDOS command* action. This can open a graphical window or not. The end of the command is not waited for, parallel execution is therefore possible.

Direct shell commands, such as `copy` etc. cannot be executed. To use such a command, you must save it in a Windows batch or command file. This file can then be executed.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.13 Start mapkey

The action executes a specified mapkey (the entire mapkey can be copied into the action).

General description

1. Input of a mapkey in default length, or abbreviated.
2. Repeat range for the mapkey. Here you can select the objects the specified mapkey is to be applied to one by one in succession.

The *Start mapkey* action executes any mapkey in Creo. Enter the mapkey in the text box (1).

To address a mapkey with its name, you must use the following syntax (note the `%%`).

```
%%MAPKEYNAME;
```

In addition, all variables from the variable list 306 can be used in mapkeys. Here is a useful selection:

– @date@

– @datede@

– @dateen@

– @dateus@

– @dateusshort@

– @mdln@

– @number@

– %PARAM%

© 2023 INNEO Solutions GmbH

An exact discussion of the possibilities of the variables 306 can also be found on the related help page.

Note: When using paths, it should be noted that mapkeys require four-fold backslashes. Therefore individual backslashes have to be replaced with 4 backslashes. This happens particularly often when using environment variables.

`$$repl$\$\\\\$$TEMP$$$` - Replaces the backslashes compatibly for mapkeys when using the environment variable *Temp*.

Repeat range

Execute for

- *only once* - Default, the mapkey will be executed only once.
- *sheet* - Here, you can execute a mapkey for each sheet of a drawing. The sheets are iterated automatically. The mapkey is executed again for each sheet. @*sheetno*@ contains the current sheet number.
- *instance* - Here a mapkey will be executed for each instance. For drawings, each instance of the current model. @*mdln*@ contains the instance name here, not the name of the drawing model as usual. The repeat range can be used only in connection with MP User mode.

Synchronous mode

All mapkeys would be run in synchronous mode. A mapkey is always fully blocking when it is executed.

Advantages:

– Feedback and execution of the mapkey is much faster.

– The mapkey cannot be aborted by a timeout as long as it is still running.

– Windows that are open after mapkey completion are closed automatically to permit an orderly sequence.

Disadvantages:

– Errors in the mapkey frequently lead to a Creo crash or to undefined states in Creo.

– Any `mapkey continue` with leading backslash must be removed from the mapkey. Failure to do so will lead to execution errors.

– In some cases, pressing an OK button causes Creo to crash. Prior to handing over the MPZ you should therefore test the operability of your mapkey.

Some mapkeys can also be found here 325.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.14 Start sync app function

Die Aktion startet eine Funktion von einer bereits in Creo gestarteten Applikation anhand des Funktionsnamens und der Applikations ID.

Allgemeine Beschreibung

1. ID der Applikation (z.B. aus der Umgebungsvariable DLL_ID)
2. Funktionsname der exportierten Funktion
3. Liste an Eingabeargumenten

© 2023 INNEO Solutions GmbH

4. Angabe ob die Ausgabeargumente folgend als Nutzervariablen zur Verfügung gestellt werden sollen

Die ID und der Funktionsname müssen beim Start der Aktion definiert sein. Eingabeargumente(3) können angegeben werden. Diese sind im Format:

ARGUMENTNAME=WERT

ARGUMENTNAME2=WERT2

anzugeben. Hierbei können Variablen 306 verwendet werden. Eingabeargumente werden immer als STRING übergeben.

Die Ausgabeargumente werden immer mit dem Namen und Wert befüllt. Ausgabeargumente sollten immer als STRING übergeben werden.

Informationen zu Variablen finden Sie unter Variablen 306.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	X	
Teil - Instance	X	
Baugruppe	X	
Baugruppe - Instance	X	
Zeichnung	X	

5.1.1.15 Start trail file

The action executes a specified trail file.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the Creo trail file that is to be started.

The *Start trail file* action starts a Creo trail file that can be specified in the text box (1).

@project_dir@ is located at: *Project directory/models*

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.1.16 Write file

Die Aktion erzeugt eine Textdatei mit gegeben Inhalt.

© 2023 INNEO Solutions GmbH

Allgemeine Beschreibung

Die Aktion "Datei schreiben" erzeugt eine Text-Datei im UTF8 Format in gegebenem Pfad und füllt sie mit definiertem Text. Im Text können Variablen 306 enthalten sein.

Siehe auch:

Nutzereingabe abfragen 36

Nutzereingabe von URL setzen 41

Nutzereingabe von Bounding Box befüllen 91

Achtung: Nutzereingaben werden am Ende des Laufs von Model Processor / User gelöscht. Somit sind alle nicht explizit zugewiesenen Werte dann verloren.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	X	
Teil - Instance	X	

© 2023 INNEO Solutions GmbH

Modelltyp	Anwendbar	Bemerkungen
Baugruppe	X	
Baugruppe - Instance	X	
Zeichnung	X	

5.1.1.17 Write to Creo Log

The action writes the message to the message area.

General description

The defined message would be written to the Creo message area. You can use variables inside the message test.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.2 PTC Windchill PDM

This group combines actions. Please choose an action.

5.1.2.1 Add file to workspace

The action adds a file to the workspace.

© 2023 INNEO Solutions GmbH

General description

1. Path / name of the file to be added to the workspace
2. The model to which the file should be attached as secondary content. If no model is specified here, the file is added to the workspace as primary content (if possible)

The action adds a defined model (1) to the workspace. Use it, for example, to attach external file formats created by exports, such as STEP, to another model (2) as secondary content.

The model specified as the target for secondary content (2) does not have to be the current model.

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.2.2 Check in model

The action checks in a model.

© 2023 INNEO Solutions GmbH

General description

1. Specify the path under which to check in the model if it has not been checked in before. Otherwise, the model will keep its previous path.

Information about variable entries for the text box (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.2.3 Check out model

The action checks out a model.

General description

The action checks out the current model from Windchill.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.2.4 Clear workspace

The action removes models from the workspace.

General description

The action removes all models from the workspace that can be removed. For a model to be removed, it must not be used in the current session any longer.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.2.5 Remove file from workspace

Die Aktion entfernt eine Datei aus dem Workspace.

Allgemeine Beschreibung

Die Aktion entfernt eine definierte Datei aus dem Workspace. Instanzen müssen hierbei mit Generic geschrieben werden (Instanz<Generic>.Typ). Wird ein Generic entfernt werden auch alle Instanzen entfernt.

VORSICHT: Es können auch Dateien entfernt werden welche sich aktuell in der Sitzung befinden.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	X	
Teil - Instance	X	
Baugruppe	X	
Baugruppe - Instance	X	
Zeichnung	X	

5.1.2.6 Set server active

The action sets a server active that has already been defined in Creo.

© 2023 INNEO Solutions GmbH

General description

1. Name of the server. The server already has to be present in Creo.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.3 Model

This group combines actions. Please choose an action.

5.1.3.1 Add 3D annotation

The action adds a 3D annotation to the model or changes an existing one. It can be anchored to the basic coordinate system.

© 2023 INNEO Solutions GmbH

General description

1. Specify the name of the 3D note you want to create. The 3D note is created with the default name if a name is not specified.
2. Specification of the content of the 3D annotation in textual form
3. Selects whether the 3D annotation shall be placed on the first located coordinate system
4. Specifies whether the content of a 3D annotation that exists under the selected name shall be overwritten with the new content

The action *Add 3D annotation* creates a new 3D annotation that is added to the model. Name and content of the 3D annotation are set with the texts from the text boxes (1,2). The 3D note is only placed if selection (3) defines that it is to be placed on the first coordinate system found. Selection (4) controls whether the content of a 3D annotation that exists under the selected name should be overwritten with the new content, or whether a new annotation should still be created. If a new annotation is created, the specified name is not set, as only one element of a given name may exist.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.3.2 Backup action

The action saves the model to a specified path.

General description

1. The path under which the backup copy is to be saved.

The *Backup action* creates a backup copy of the edited model. In the text box (1), you can specify the path under which the backup copy should be created. The open model, NOT the backup copy, is used in all the further actions of the action list.

You cannot specify a name for the backup copy. The name of the original model is always used for the backup copy, too.

CAUTION: The action first creates the backup copy, then the model is saved again at its original location (relocated). Therefore, a new version is also created at the original location. This behavior can be deactivated with the option(2).

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.3.3 Cancel feature position

The action cancels the feature position offset.

General description

The action *Cancel feature position* aborts the feature position offset. All features that were suppressed due to a position offset will be resumed.

For additional information, please refer to the help page Set feature insert position 85.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.3.4 Compute mass

The action regenerates the mass of the model by means of a defined coordinate system.

General description

1. Text box to specify a coordinate system. The default coordinate system is used if no entry is made here.

© 2023 INNEO Solutions GmbH

The *Compute mass* action creates the *pro_mp_mass* parameter and calculates the mass of the part or the assembly. In the text box (1) you can specify the name of a coordinate system that is used for the mass calculation. Model Processor runs with default settings require the *Compute mass* action to be used before the mass parameter is used in a relation.

The action does no longer have to be used before regeneration. Currently, the regeneration function automatically sets the density to 1 if it was not defined before. See also the Changelog Model Processor 1.2 to Model Processor 2.0.

CAUTION: If the part does not yet have a density and material assigned, density 1.0 will be assigned when the mass is regenerated.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.5 Copy model

The action creates a copy of a model defined by name under a different name (see also Backup action 61).

General description

1. Name of the model that is to be copied to the left (with extension) and the new name of the model (without extension) to the right

© 2023 INNEO Solutions GmbH

The *Copy model* action copies a model under the same path so that an identical model with a new name is created. The extension is to be considered when specifying the names:

– left column: with extension

– right column: without extension

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.3.6 Create DM code as dot matrix

The action creates a DM code as dot matrix table pattern.

To create a DM code in a drawing, please use the action Create DM code symbol 216.

© 2023 INNEO Solutions GmbH

General description

1. Coordinate system in whose X and Y axes (positive) the QR code is to be created
2. Name for the generated table pattern
3. Input of the text encoded within the DM code
4. Specification of the pixel size, depending on the unit system
5. Specification whether the following 3 mapkeys (9) should be executed or not
6. 3 mapkeys to create geometry on the QR code dots

The action *Create DM code as dot matrix* creates a dot matrix, which includes geometry if option (5) is selected. To do so, a coordinate system (1) must be specified, a name for the pattern (2) to be created and a text (3) to be encrypted. Variables may be used in the text (3).

© 2023 INNEO Solutions GmbH

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.7 Create QR code as dot matrix

The action creates a QR code as dot matrix table pattern.

To create a QR code in a drawing, please use the action Create QR code symbol 218.

© 2023 INNEO Solutions GmbH

General description

1. Coordinate system in whose X and Y axes (positive) the QR code is to be created
2. Name for the generated table pattern
3. Input of the text encoded within the QR code
4. Specification of bit length per character. This also affects which characters may be written.
5. Specification of the symbol color by specification of system colors
6. Case-sensitivity

© 2023 INNEO Solutions GmbH

7. Specification of the pixel size, depending on the unit system
8. Specification whether the following 3 mapkeys (9) should be executed or not
9. 3 mapkeys to create geometry on the QR code dots

The action *Create QR code as dot matrix* creates a dot matrix, which includes geometry if option (8) is selected. To do so, a coordinate system (1) must be specified, a name for the pattern (2) to be created and a text (3) to be encrypted. Variables may be used in the text (3).

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.8 Edit PTC_COMMON_NAME (common name)

The action sets the common name (PTC_COMMON_NAME) to a specified value.

© 2023 INNEO Solutions GmbH

General description

1. Input box for the new PTC_COMMON_NAME
2. Specifies whether the PTC_COMMON_NAME should be changed for PDM systems, too
3. Specifies whether all instances of a generic should be included in the rework

The *Edit PTC_COMMON_NAME* action changes the internal parameter PTC_COMMON_NAME to the input (1). You can use the variables specified underneath the text box. Bear in mind that modifying the parameter can lead to problems with PDM systems. If you are using a PDM system, only modify the common name if you are sure that there can be no negative effects on your system.

If the option for reworking all instances (3) is enabled, the PTC_COMMON_NAME is set in accordance with the rule for each instance (including in multi-level family tables). The rule is applied again for each instance.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

Messages

Log information

Edit PTC_COMMON_NAME	Start of the action
mdl not found	The model could not be accessed
replace parameters	The system looks for parameter dependencies (by variables), and replaces them.
replace others	Other variable dependencies are looked for and replaced.
change config options	The creo/pro configuration is modified to allow renaming in PDM systems.
set common name	The new "common name" is set

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.9 Edit model dimension from file

The action changes dimensions based on the data in a CSV or XLS file.

© 2023 INNEO Solutions GmbH

General description

1. Specify the CSV or XLS file
2. Specify the encoding of the CSV file under (1)
3. Specify the separator character in the CSV file under (1)

The action *Edit model dimension from file* sets the dimensions specified in a file to the values specified in the same file.

Dimension that should be filled need to already existing inside the model.

Following cells inside the CSV file would be evaluated:

– *File name* - defines the file name

– *File path* - defines the file path

– *### VALUE* - the value that should be set on a dimension with the name ###

Example of a CSV file:

```
Filename;LENGTH VALUE;
test.prt;355;
peter.prt;400;
```

Information about variable entries for the text field (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.10 Embed Models

Die Aktion bettet alle Modelle und Submodelle in die Baugruppe ein.

Allgemeine Beschreibung

Alle Modelle und Submodelle innerhalb der Baugruppe werden eingebettet in die Baugruppe.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	-	
Teil - Instance	-	
Baugruppe	X	
Baugruppe - Instance	-	
Zeichnung	-	

5.1.3.11 Insert first features

The action adds the first features (one coordinate system and 3 planes).

© 2023 INNEO Solutions GmbH

General description

1. Specification of the name for the first coordinate system
2. Specification whether 3 planes should be created as well.
3. Specification of the names for the three default planes to be created on the axes of the default coordinate system
4. Sequence of the four new features, the coordinate system is NOT made dependent of the datum planes

This action creates the first coordinate system on the model origin if it does not yet exist. The origin exists in each model, even if not indicated by a visible element. In the text box (1), you can specify the name of the coordinate system that is to be created.

Additionally, you can create 3 datum planes by selecting option (2). These are located on the default coordinate system on the X, Y and Z axis. The names of the planes to be created can be specified via the text boxes (3).

Using the option (4), you can change the sequence of the newly created features.

WARNING! The action will create no new features if the model does not have any features yet; i.e., no start objects will be created in a completely empty model.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	-	
Drawing	-	

Messages

Log information

insert first features	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.12 Load tolerance table

The action sets or replaces a tolerance table in the model to be processed.

General description

1. Path to the tolerance table to be loaded

The action loads the specified tolerance table (1) into the model. Existing tolerance tables of the same name will be replaced.

The tolerance table name is specified in the file to be loaded. This means that the general tolerance table can also be replaced with this action if the file header matches it.

Example file header for a general tolerance table:

```
...
"Table_Type" "General_Dims"
"Table_Name" "Defaults"
"Table_Unit" "millimeter"
"Range_Unit" "millimeter"
...
```


© 2023 INNEO Solutions GmbH

Information about variable entries for the text field (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.13 Regenerate model

The action regenerates the model.

General description

1. Specification of the maximum number of regenerations of a model
2. Igrnore the *previously generated* status used by Creo
3. Update model tree after regeneration

For the *Regenerate model* action, you can enter (1) to specify the maximum number of regenerations. The action regenerates the current model. If another regeneration is required afterwards, regeneration will be repeated up to the maximum number of regenerations.

If option (2) is selected, each feature is regenerated and the system disregards whether or not Creo states that is has already been regenerated. Consequently, regeneration takes longer and each part is always regenerated as often as specified under (1). In this case, the number set at (1) is not understood as the maximum, but as the required number of regenerations.

© 2023 INNEO Solutions GmbH

If option (3) is selected, the model tree of the regenerated model is updated after regeneration. This is of interest only at the last regeneration of a model. Within a batch processing this is not needed at all. If a mapkey is run after regeneration that accesses the model tree, the tree has to be updated before.

CAUTION: The way this action works has changed with version Model Processor 2.0. Regenerating no longer requires a preceding *Compute mass* action.

CAUTION: If the part has not yet been assigned a density and material, density 1.0 will be assigned during regeneration.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

Messages

Log information

regenerate model	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.14 Remove feature

Die Aktion löscht definierte KEs.

© 2023 INNEO Solutions GmbH

Allgemeine Beschreibung

1. Filter um die KEs zu bestimmen welche gelöscht werden sollen
2. Angabe ob nur KEs gelöscht werden sollen die unterdrückt sind und dem Filter(1) entsprechen

Durch die Aktion *KE löschen* können einzelne Konstruktionselemente (KEs) gelöscht werden. Hierzu muss der Filter definiert werden, welcher bestimmt welche KEs gelöscht werden sollen. Im Namenfilter kann hierfür RegExp 323 verwendet werden.

Mit der Option(2) kann angegeben werden ob die zu löschenden Features zusätzlich unterdrückt sein müssen. Alle Features werden in einem Schritt gelöscht. Sollten nicht löschtbare Features definiert sein so schlägt die Aktion fehl.

Informationen zur Beschreibung und den Modellfiltern finden Sie unter Modellfilter 229.

Voraussetzungen

Modelltyp	Anwendbar	Bemerkungen
Teil	X	
Teil - Instance	-	
Baugruppe	X	
Baugruppe - Instance	-	
Zeichnung	-	

5.1.3.15 Rename feature

The action changes the name of a feature to the new, specified name.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the features that are to be renamed (left) and of the new names (right)

The *Rename feature* action permits individual design elements (features) to be renamed. Specify the current name of the feature in the left column of the table (1), and the new name of the feature in the right column.

The action supports RegExp 323 in the left column of the table. Variables 306 are supported in the right column. All names must be entered in a Creo-compatible way.

If you want to see the modification, you must regenerate the model in order to update the model tree in Creo.

CAUTION: Only named features can be renamed. Unnamed features (Example: *Extrude 1*) can not be renamed. *Extrude 1* etc. are designations Creo assigns to features without a name.

Information about variable entries for the right side of the table can be found at Variables 306 .

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.3.16 Rename model

Renames a model. (CAUTION: Models not in memory will not be able to find the renamed model afterwards.)

General description

1. Name of the model that is to be renamed to the left (with extension) and the new name of the model (without extension) to the right

The *Rename model* action renames a model:

– left column: old name with extension, you can use RegExp 323 and Variables 306

– right column: new name without extension, you can use Variables 306

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.3.17 Replace a model

Replaces a first-level component with another model (like Creo *relationless replace*).

© 2023 INNEO Solutions GmbH

General description

1. Name of the model that is to be replaced to the left (without extension) and the new model (without extension) to the right
2. Models with the same name will be loaded into the session

The *Replace model* action replaces a model with another model (1) or loads a model into the session (2).

Reloading a model can be used, for example to loadsevered family table instances 122. The assembly is loaded without the family table instances (>=WF5). Afterwards, the family table instances of the broken-up family table are reloaded with the replace model action (only set (2)). The family table instances are automatically replaced with the now separate parts. Note that for graphic representation the Set simplified representation 190 action should be executed next. With one regeneration (force regeneration) the tree in the representation can then be cleaned up as well.

Alternatively, it is also possible after reloading the models to save, close and reload the assembly . After refreshing the tree, you can already see if the parts were replaced properly, but Creo doesn't recognize it for the graphical representation. But after loading the saved part again Creo will load the newly assembled part (family table variant independent from family table).

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.3.18 Rescale feature positions in assemblies

The action recalculates assembly positions that are referenced in a coordinate system if the size of the parts has changed by a given factor.

General description

1. Scale factor

The action adjusts component positions that can be wrong due to scaled parts. This happens only when import geometry (STEP) is scaled, or with assemblies in which parts are directly referenced to the base coordinate system.

Scaling can, for example, be performed with the following mapkey across all parts.

```
~ Command `ProCmdAOScaleModel` ;0.8;~ Activate `0_std_confirm` `OK`;
```

Information about description and model filters can be found in Model filter 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

Messages

Log information

rescale ASM positions	Start of the action

© 2023 INNEO Solutions GmbH

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.19 Save model

The action saves the model.

General description

1. Selects the criterion for saving the model

The *Save model* action saves the model after the selected event (1) has occurred.

Events:

– Only if no errors occured - The model is saved if an error did not occur in the Model Processor during the previous actions.

– Always - The model is always saved. Any errors are ignored.

Creo automatically causes the related generic to be saved when an instance is saved.

MP User

This action is not performed in MP User. The entire model, together with all dependent revised models, can be saved in one operation by saving the topmost assembly.

Any other implementation of saving is not recommended, as this, depending on the trigger configuration, may result in loops which cause the running Creo system to crash.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	X	
Drawing	X	

Messages

Log information

save model	Start of the action
saved	The model was saved.

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.20 Set accuracy

The action changes the accuracy of the model to the specified type and value.

General description

1. Selects the type of accuracy that is to be set
2. Enter the accuracy that is to be set

The *Set accuracy* action sets the absolute or relative (1) accuracy of a model. Taking the usual Creo limits into account, you can enter the value in the text box (2). The value can not always be adjusted via the program. There are constellations where manual changes work, while program-controlled changes fail.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

Messages

Log information

set accuracy	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.3.21 Set all dimension bounds

The action sets all dimension bounds for all dimensions of a model.

General description

1. Type of dimension bound to be set

© 2023 INNEO Solutions GmbH

The action *Set all dimension bounds* sets the dimension bounds to the specified mode (1) for all dimensions.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.22 Set feature insert position

With this action the insertion point is moved to underneath the specified feature.

General description

1. The feature underneath which the insertion point should be located is specified here

The *Set feature insert position* action moves the insert point in the tree to the position underneath the feature which is specified by ID, type (the first one to be found) or name. RegExp 323 may be used with selection by name.

Before setting a new insert position offset, the old offset MUST be aborted using the Cancel feature position 62 action. Otherwise, suppressed elements will not be resumed on changing the insert position.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.3.23 Set model options (DTL)

The action modifies drawing or model options using a specified DTL file.

General description

1. Specify the DTL file that is to be loaded

To modify the drawing options you must create a DTL file. Using this action, the DTL file is added to the current drawing.

The DTL file is a standard PTC file. The first line of the DTL file must contain a comment (line that begins with !). Any settings of the drawing options can be modified in the following lines.

A DTL file must always be saved in the UTF-8 character set. Typical contents of a DTL file:

```
! This is the dtl for the expansion of our standardized parts
default_font    inneo_de
text_height 4.50000
```

More detailed information can also be found in the PTC manual at *Set drawing options*. Here, you can also find a list of all possible options.

@project_dir@ is located at: *Project directory/models*

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.3.24 Set style

The action changes the component display style.

General description

1. Name of the style to be set

The action *Set style* sets a defined display style (1) in an assembly.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.3.25 Set tolerance

The action modifies the set tolerance. The tolerance limits must be ensured here.

© 2023 INNEO Solutions GmbH

General description

1. Select the tolerance you want to set

The action sets a specific tolerance that can be selected from the option box (1). The model is opened visibly, and the required tolerance is set.

CAUTION! In a conversion from ANSI to DIN/ISO, or from DIN/ISO to ANSI, manually entered tolerances and selected tolerance tables are lost.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.3.26 Set units

The action changes the used unit system to the selected one.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the unit system that is to be set
2. Specification of a newly created unit system
3. Specification of the conversion options into the selected unit system

The *Set units* action sets the unit model selected at *Type* (1) for the model that is to be processed. The selected conversion (3) is performed when the units are set.

The options under (2) permit a new unit system to be created. It is specified via the name and the units. An existing unit system of the same name is selected without changes, NOT deleted. Consequently, the selected unit system can sometimes not correspond to the units you entered.

Conversion (3):

– Change value - adjusts the existing values to the new unit system

– Keep value - retains the numeric values, and changes the unit. In conjunction with other models, this can lead to an increased number of errors.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.3.27 Set user variable from bounding box

The action sets a user variable to a value without asking for user input.

General description

1. Name of the variable that can later be accessed as `%user:NAME%`
2. Defines whether the model should be rotated to find the minimal bounding box
3. Defines whether the values should be sorted by size to facilitate value comparison
4. Defines the number format (default 6 decimal places, floating)

The action *Set user variable* sets a value that is NOT DEFINED IN CREO in Model Processor. The value can later be accessed as `%user:NAME%` in all fields that support variables.

`%user:NAME%` will resolve to the value of the user variable.

© 2023 INNEO Solutions GmbH

The option *rotate Model* (2) defines whether the model should be rotated along its edges to find the minimal bounding box. When rotating, diameter values are also recognized and the *Diameter* value set. Otherwise, the diameter value is set as *-1*.

The option *sort values* (3) defines that the return values for height, width and lengths are set sorted by size. This means that height, widths and lengths are not defined relative to a coordinate system (or a rotated coordinate system if you set (2)), but assigned by size. This makes it easier to compare the values, for example to box sizes.

The number format (4) is defined as follows:

– % -> start symbol

– . -> symbol for number of places

– f -> specifier

Supported specifiers:

– d -> integer

– f -> floating-point number

– e -> exponential number

– g -> shortest representation (f or e)

– x -> hexadecimal number

Examples:

– %.6f -> 6 decimal places

– %4.6f -> 4 integer places, 6 decimal places

– %.6d -> 6-digit integer with leading zeroes

– %6d -> 6-digit integer with leading space characters

See also:

Request user input 36

Set user variable from URL 41

Set user variable from bounding box 91

Please note that user variables will be deleted after each Model Processor or Model Processor User run. All values not explicitly assigned and saved will be lost.

For information on using variables in text fields (1), pleaser refer to Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.3.28 Start ModelCHECK

The action starts ModelCHECK in Update mode with specified or default configuration.

General description

The *Start ModelCHECK* action executes ModelCHECK, and creates the related parameters at the models that are to be overhauled.

CAUTION: ModelCHECK is executed in Update mode in this action.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.4 Model - Views

This group combines actions. Please choose an action.

5.1.4.1 Add views

The action adds views to the model. There are 7 default views and one custom view.

General description

1. Presetting. All list boxes can be filled with pre-defined settings.
2. Names of the views, including the visible axis names
3. Specification of a name for the Creo standard view
4. Creating a new view with rotation about the corresponding axes X,Y,Z
5. Import the current values for the rotation specifications under 4 from Creo, rounded to precision 6
6. For flattened sheetmetal parts only: If you specify a name for a view here, a view of the green side of the flattened sheetmetal is created.

© 2023 INNEO Solutions GmbH

The *Add views* action permits views of any name to be added. The orientation of the views is specified via the axes (2). The standard view (3) in Creo is a 3D view that is created as a function of the selected configuration options.

Available presettings (1) are:

– English

– German

– International

See the presettings (1) on the UI as an example. Creating views on the basis of named planes is not possible.

If the *Name* field of the user-defined view (4) contains a value, a view is created with the specified rotation about the X, Y, and Z axis. The values can be taken from existing views in the view dialog, or be selected as required. They must be in the range Creo expects them to be.

As a basis, you can read the current values from Creo.

Example values for ISO views:

Top Left Front

X: 45 Y: 35.2644 Z: 30

Top Left Back

X: 45 Y: -35.2644 Z: -30

Top Rear Back

X: -45 Y: -144.7356 Z: 30

Top Rear Front

X: -45 Y: -144.7356 Z:-30

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	Regenerate indirectly via instance
Assembly	X	
Assembly - instance	-	Regenerate indirectly via instance

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.4.2 Rename view

The action renames a given view to a new name.

General description

1. The name of the view that should be renamed
2. The new view name

This action renames a given view (1) to a new name (2).

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.4.3 Set view

The action rotates the model to the specified view.

© 2023 INNEO Solutions GmbH

General description

1. Sets the default view. Inputs in the text box (2) are ignored.
2. Name of the view that should be set

This action sets a specific view that is defined by name as the current view, and adjusts it to fit in the window. This action is frequently used before images or PDFs are created. The action requires a visibly opened model.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.5 Model - Relations

This group combines actions. Please choose an action.

5.1.5.1 Add relations

The action adds relations / post regeneration relations to the model.

© 2023 INNEO Solutions GmbH

General description

1. Lines to be added to the relations
2. Lines to be added to the post-regeneration relations.
3. Specifies whether the lines are added at the beginning or at the end of the relations
4. Deactivating error checking (can cause errors)

In the *Add model relation line* action, the relations entered in the text boxes (1,2) are appended to the related relation. Use the selection (3) to specify whether the relation is added at the beginning or at the end of the relations. Use the Edit relations 170 action to add the relation at a different point (not the beginning or the end).

You may use `%PARAMETERNAME%` to write a parameter value into a relation. Use `%%` to write a percent sign.

Deactivate the error checking function (4) only if you are sure that the errors in the relations do not influence the execution of the newly added lines. The result of the relations would not be reliable and the data quality after rework would not be reliable.

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	-	Indirectly via regenerate instance
Assembly	X	
Assembly - instance	-	Indirectly via regenerate instance
Drawing	-	

5.1.5.2 Add feature relations

The action adds relations / post regeneration relations to the model.

General description

1. Name of the feature which relations should be changed
2. Lines to be added to the relations
3. Specifies whether the lines are added at the beginning or at the end of the relations
4. Deactivating error checking (can cause errors)

In the *Add feature relations* action, the relations entered in the text box (2) are appended to the related relation. Use the selection (3) to specify whether the relation is added at the beginning or at the end of the relations. Use the Edit feature relations 103 action to add the relation at a different point (not the beginning or the end).

© 2023 INNEO Solutions GmbH

You may use `%PARAMETERNAME%` to write a parameter value into a relation. Use `%%` to write a percent sign.

Deactivate the error checking function (4) only if you are sure that the errors in the relations do not influence the execution of the newly added lines. The result of the relations would not be reliable and the data quality after rework would not be reliable.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	Indirectly via regenerate instance
Assembly	X	
Assembly - instance	-	Indirectly via regenerate instance
Drawing	-	

5.1.5.3 Edit relations

The action modifies the words or phrases in relations / post regeneration relations in a model line by line or adds a block of lines.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the relation types where changes are to be made
2. Deactivating error checking (can cause errors)
3. Specification of the lines that are to be modified, separated by Enter. Each line stands for a line that is to be located and modified; RegExp 323.
4. Specification of the word that is to be replaced. * replaces the entire line; RegExp 323.
5. Input box for the letters, words that are to be inserted.
6. Lines to be added. The lines are added as a block.
7. Specify whether the block shall be added above or below the line specified at (7).
8. Specify the line above or below which you want to insert the lines; RegExp 323.

© 2023 INNEO Solutions GmbH

The action *Edit model relations* can carry out two tasks: Editing existing lines (upper box) and adding new lines (bottom box).

The selection (1) enables you to choose the type of relations you want to modify or extend.

Modifying existing lines

To modify existing lines you must enter the lines you want to edit - separated by Enter - in the text box (2). In the text box (3) you can define the text area you want to replace. The wildcard character * on its own stands for the entire line. Line breaks are captured. The relation is edited beyond line breaks. The text to be inserted can be entered in text box (4).

When using RegExp 323 to modify lines, please note that RegExp 323 work in *greedy* mode in MP. Also, the text that is to be replaced must be marked with start (^) and end ($) (e.g., `/^NAME$/`).

Adding lines

In contrast to adding lines with the *Add relations* action, adding lines via *Edit model relations* is not able to append lines specifically at the end of the relations. Consequently, there are advantages and disadvantages in using the actions.

When you add lines, you can enter a block of lines in the text box (5). This block will then be positioned in existing relations via the selection (6) and the entered line (7).

CAUTION: When the line occurs repeatedly (matching definition from input (7)), the block (text box (5)) will also be added repeatedly.

Use `%PARAMETERNAME%` in the boxes (2,4,5) to write a parameter value into a relation. Use `%%` to write a percent sign in these boxes.

Information about variable entries for the text boxes (2,4,5) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Example

Appending values to a line:

To append values to a line, the line first needs to be found using the line specification, and then it will be replaced completely by the new line. @*line*@ is used to re-use the line content. Lines: `/DIM_X_ACS_GBA_X_MI=\d*/\d*/`

Replace by: `@line@+HUB`

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.5.4 Edit feature relations

The action modifies the words or phrases in relations / post regeneration relations in a model line by line or adds a block of lines.

General description

1. Name of the feature which relations should be changed
2. Deactivating error checking (can cause errors)

© 2023 INNEO Solutions GmbH

3. Specification of the lines that are to be modified, separated by Enter. Each line stands for a line that is to be located and modified; RegExp 323.
4. Specification of the word that is to be replaced. * replaces the entire line; RegExp 323.
5. Input box for the letters, words that are to be inserted.
6. Lines to be added. The lines are added as a block.
7. Specify whether the block shall be added above or below the line specified at (7).
8. Specify the line above or below which you want to insert the lines; RegExp 323.

The action *Edit feature relations* can carry out two tasks: Editing existing lines (upper box) and adding new lines (bottom box).

Modifying existing lines

To modify existing lines you must enter the lines you want to edit - separated by Enter - in the text box (2). In the text box (3) you can define the text area you want to replace. The wildcard character * on its own stands for the entire line. Line breaks are captured. The relation is edited beyond line breaks. The text to be inserted can be entered in text box (4).

When using RegExp 323 to modify lines, please note that RegExp 323 work in *greedy* mode in MP. Also, the text that is to be replaced must be marked with start (^) and end ($) (e.g., `/^NAME$/`).

Adding lines

In contrast to adding lines with the *Add relations* action, adding lines via *Edit model relations* is not able to append lines specifically at the end of the relations. Consequently, there are advantages and disadvantages in using the actions.

When you add lines, you can enter a block of lines in the text box (5). This block will then be positioned in existing relations via the selection (6) and the entered line (7).

CAUTION: When the line occurs repeatedly (matching definition from input (7)), the block (text box (5)) will also be added repeatedly.

Use `%PARAMETERNAME%` in the boxes (2,4,5) to write a parameter value into a relation. Use `%%` to write a percent sign in these boxes.

Information about variable entries for the text boxes (2,4,5) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Example

Appending values to a line:

To append values to a line, the line first needs to be found using the line specification, and then it will be replaced completely by the new line. @*line*@ is used to re-use the line content. Lines: `/DIM_X_ACS_GBA_X_MI=\d*/\d*/`

© 2023 INNEO Solutions GmbH

Replace by: `@line@+HUB`

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.5.5 Delete relations

The action deletes lines from the relations / post regeneration relations which can be specified by individual lines or as blocks.

© 2023 INNEO Solutions GmbH

General description

1. Specifies the type of relations in which you want to search and delete
2. The lines that are found for deletion are turned into comments, and not deleted.
3. Deactivating error checking (can cause errors)
4. Lines that are separated by a line break are handled, searched and deleted individually.
5. A block that is to be deleted is identified by a start line (5) and an end line (6).
6. End line of a block that is to be deleted
7. The end of an if block is recognized automatically. Box (6) is blocked for entries, and ignored during processing.

The *Delete model relations* action deletes specified lines from relations. Checkboxes (1) enable you to select the relation types in which you want to search and delete. Use the option (2) to choose whether the selected lines are to be turned into comments rather than deleted.

The lines that are to be deleted can be specified in the text box (4). Each line corresponds to a new delete command. If a line in this text box contains `endif`, for example, each line that

© 2023 INNEO Solutions GmbH

contains `endif` will be deleted. Each line is considered independently, irrespective of the previous line.

To delete a block of lines, you can define a start line and an end line.

Example:

```
if (TEST=14)
    PARAMETER2 = "HELLO"
endif
```

Start at line: `*if*(TEST=*`

End at line: `*endif*`

* is used here as a wildcard character. It says that all blocks that contain `if` with the parameter `TEST` are deleted up to the `endif`. The wildcard is used here to ignore any blanks as well as the value the `TEST` parameter is compared to.

To search explicitly for a line that contains an asterisk, you must use **. A single asterisk is ALWAYS interpreted as a wil card. Otherwise, you may use enhanced RegExp 323.

Option (7) can also be selected to recognize the end of an if block . The associated `endif` is recognized and used automatically here. This is useful when several if blocks are nested.

Deactivate the error checking function (3) only if you are sure that the errors in the relations do not influence the execution of the newly added lines. The result of the relations would not be reliable and the data quality after rework would not be reliable.

Information about variable entries for the text box (4) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.5.6 Delete feature relations

The action deletes lines from the relations / post regeneration relations which can be specified by individual lines or as blocks.

© 2023 INNEO Solutions GmbH

General description

1. Name of the feature which relations should be changed
2. The lines that are found for deletion are turned into comments, and not deleted.
3. Deactivating error checking (can cause errors)
4. Lines that are separated by a line break are handled, searched and deleted individually.
5. A block that is to be deleted is identified by a start line (5) and an end line (6).
6. End line of a block that is to be deleted
7. The end of an if block is recognized automatically. Box (6) is blocked for entries, and ignored during processing.

The *Delete feature relations* action deletes specified lines from relations. Use the option (2) to choose whether the selected lines are to be turned into comments rather than deleted.

The lines that are to be deleted can be specified in the text box (4). Each line corresponds to a new delete command. If a line in this text box contains `endif`, for example, each line that

© 2023 INNEO Solutions GmbH

contains `endif` will be deleted. Each line is considered independently, irrespective of the previous line.

To delete a block of lines, you can define a start line and an end line.

Example:

```
if (TEST=14)
    PARAMETER2 = "HELLO"
endif
```

Start at line: `*if*(TEST=*`

End at line: `*endif*`

* is used here as a wildcard character. It says that all blocks that contain `if` with the parameter `TEST` are deleted up to the `endif`. The wildcard is used here to ignore any blanks as well as the value the `TEST` parameter is compared to.

To search explicitly for a line that contains an asterisk, you must use **. A single asterisk is ALWAYS interpreted as a wil card. Otherwise, you may use enhanced RegExp 323.

Option (7) can also be selected to recognize the end of an if block . The associated `endif` is recognized and used automatically here. This is useful when several if blocks are nested.

Deactivate the error checking function (3) only if you are sure that the errors in the relations do not influence the execution of the newly added lines. The result of the relations would not be reliable and the data quality after rework would not be reliable.

Information about variable entries for the text box (4) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.5.7 Set relations unit sensitive

The action sets relations unit sensitive.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the relation types where changes are to be made
2. Specifies whether unit sensitivity should be set or removed

The action *Set relation unit sensitive* sets or removes (2) the unit sensitivity of relations (1).

This can be done manually as shown here:

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.6 Model - Export

This group combines actions. Please choose an action.

© 2023 INNEO Solutions GmbH

5.1.6.1 Export PARASOLID file

The action generates a PARASOLID file of the model. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. When this checkbox is selected, box (1) is deactivated, and the folder of the model is used (not possible in conjunction with PDM).
3. Name used for saving the model
4. Definition of a PARASOLID export profile file. If the profile is not defined, the standard configuration of Creo would be used.

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.6.2 Export Creo View file

The action generates a Creo View file of the model. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Specifies whether or not the model folder should be used
3. File name to save to
4. Specification of the export format

The *Export as Creo View* action exports assemblies and parts as Creo View files. In the text box (1) you must specify the folder to which you want to save the resulting Creo View files (variables possible). You can also use the selection (2) to use the folder of the model as the output folder.

Enter the name for the export in text box (3). You may use variables here. If the file name extension is not specified, it will automatically be added. Make sure that you do not specify a wrong file name extension for the selected file type. This would create a double suffix (e.g. *pvs.pvz*).

In the export format setting (4) for Creo View, you can select four different formats:

1. .ed
   prt : one .ed file, one .ol file
   asm : one .ed file, many .ol files (one for each part in the assembly)

© 2023 INNEO Solutions GmbH

2. .edz
   one file with suffix .edz
3. .pvs
   prt : one .pvs file, one .ol file
   asm : one .pvs file, many .ol files (one for each part in the assembly)
4. .pvz
   one file with suffix .pvz

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.6.3 Export IGES file

The action generates an IGES file of the model. This requires the output format, the storage location and name to be specified.

© 2023 INNEO Solutions GmbH

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Name used for saving the model.
3. Specify the elements you want to export.
4. Specify the STEP file structure to which you want to export the assembly.

The action *Export as IGES* exports assemblies and parts as IGS/IGES files. In text box (1) you must specify the folder to which you want to save the resulting IGES files.

Enter the name for the export in text box (2). You may use variables here.

In options (3+4), you can enter various export configurations.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Assembly - instance	X	
Drawing	-	

5.1.6.4 Export JPG file

The action generates an image (JPG) of the model with defined height and width. This requires the storage location and the name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. When this checkbox is selected, box (1) is deactivated, and the folder of the model is employed (not possible in conjunction with PDM).
3. Name used for saving the image
4. Specifies whether or not the image should be overwritten if an image of the specified name exists
5. Input of the width in px (without end or blanks)
6. Input of the height in px (without end or blanks)

The *Export JPG file* action saves assemblies and parts as JPG files. It is recommended to deactivate the representation of axes and planes with a mapkey before taking a picture.

To save, you can specify an existing folder in the text box (1), or activate the option (2). The option (2) cannot be used in conjunction with PDM systems. In the text box (3), you can specify the name of the JPG file to be created. When you edit more than one file, you must

© 2023 INNEO Solutions GmbH

ensure that the name contains a variable, so that the images do not overwrite each other. Option (4) specifies whether an image that exists at the specified path and name will be overwritten.

Only figures can be entered in the text boxes (4,5). Here you can define height and width of the image. If you want to use the images for publications, creating the images at a resolution that is higher than required is expedient. They can then be downscaled with an image processing program, which increases the quality.

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.6.5 Export PDF/U3D file

The action exports the model to a 3D PDF/U3D file.

General description

© 2023 INNEO Solutions GmbH

1. Path to save the PDF in. You can also use the model directory.
2. Name of the PDF to be created
3. Profile which defines export settings (optional)
4. Defines the export mode:
   - U3D
   - U3D PDF
   - PDF with named views

The action *Export PDF file* creates a 3D PDF file with the defined name (2) in the defined location (1). The PDF export can be defined using Creo configuration options. For an exact definition of the export settings a profile (3) can be used.

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.6.6 Export SAT file

The action generates a SAT file of the model. This requires the output format, the storage location and name to be specified.

© 2023 INNEO Solutions GmbH

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Name used for saving the model
3. Specify the elements you want to export
4. Specify the STEP file structure to which you want to export the assembly

The action *Export as SAT* exports assemblies and parts as SAT files. In text box (1) you must specify the folder to which you want to save the resulting SAT files.

Enter the name for the export in text box (2). You may use variables here.

In options (3+4), you can enter various export configurations.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

Messages

Log information

Export STEP file	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.6.7 Export STEP file

The action generates a STEP file of the model. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. When this checkbox is selected, box (1) is deactivated, and the folder of the model is used (not possible in conjunction with PDM).
3. Name used for saving the model
4. Definition of a STEP export profile file. If the profile is not defined, the standard configuration of Creo would be used.

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.6.8 Export STL file

The action generates a STL file of the model. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. When this checkbox is selected, box (1) is deactivated, and the folder of the model is used (not possible in conjunction with PDM).
3. Name used for saving the model
4. Definition of a STL export profile file. If the profile is not defined, the standard configuration of Creo would be used.

© 2023 INNEO Solutions GmbH

Information about variable entries for the text box (3) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.7 Model - Family table

This group combines actions. Please choose an action.

5.1.7.1 Delete the family table instances

The action deletes an instance from a family table using the specified instance name.

General description

1. The family table instances to be deleted

For the *Delete family table instances* action, you can specify the instances in the list (1) either with the instance name (example: *PRT0001_INST2*), or with the instance name at the generic (example: *PRT0001<PRT0001_INST2>.prt*). The list (1) may additionally contain any specifications of wildcards or RegExp 323.

© 2023 INNEO Solutions GmbH

Reading from a CSV file adds a maximum of 8000 lines. A list MUST NEVER have more than 9940 lines. Even if this is possible by adding several CSV files. Exceeding this limit corrupts the action list. An expedient way of using more than 8000 lines is to insert several consecutive *Delete family table instances* actions and to import a maximum of 8000 lines with instance names per action.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.7.2 Split up family table

The action saves all instances of a generic as individual files to the source directory.

General description

The *Split up family table* action is used for saving a family table from a model to individual files. Each instance of a family table is saved as a separate file. At the end, the generic contains only an empty family table. However, the generic is not saved in this way. If you want to retain an empty generic, you must carry out the *Save model* action after the *Split up family table* action.

The action creates a copy of the instance with the same name as the instance in the folder of the generic, which will then be deleted from the family table. The generic will be saved in the PDM system automatically and thus will be changed in the workspace. A *Save model* action afterwards is not needed.

The generic is only saved if you use an explicit Save model 82 action.

© 2023 INNEO Solutions GmbH

Within a PDM system, saving the generic is required so that the instance names are free, and the instances can be checked in individually without causing name conflicts.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.7.3 Unlock family table instances (all)

The action unlocks all instances (removes the *locked* symbol) .

General description

The action unlocks all locked family table instances. In multi-level family tables, the instances are set to unlocked on all levels of the family table.

Unlocking instances is important when changes have been made that are to have an effect on instances by means of a verification.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Assembly - instance	X	
Drawing	-	

5.1.7.4 Verify family table

The action verifies the family table of a generic.

General description

1. Should each instance of the family table be regenerated once or twice?
2. Fasten up the verification speed, but only works on single level family tables

The *Verify family table* action creates the verified state of a family table of a generic. Activate the checkbox (1) to specify that each instance is regenerated twice (instead of a single regeneration of the instances). Some instances must be regenerated twice to be verified. This process takes time, however. If you only have single level family tables you can also use the option to only verify unverified instances (2). This do not work with multilevel family tables.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.8 Model - Appearance

This group combines actions. Please choose an action.

5.1.8.1 Set appearance

The action changes the color representation of the model to a palette color.

General description

1. Specification of the color name from the default color table *My color effects* that is loaded in Creo
2. Defines that the complete model is colored in the selected color
3. Defines that all features with the specified name are colored (RegExp 323 possible)
4. Defines that all features of the selected type are colored

The *Set appearance* action sets selected surfaces to the color (1) that is selected by its name from the *My color effects* section.

Selection is possible for all surfaces of the model (2), only surfaces that belong to features with the specified name (3), or surfaces that belong to features of the selected type (4).

Remember that you can use RegExp 323 in the selection by name.

This action can not be used with Wildfire 4.

Information about description and model filters can be found at Model filter 229.

MP User

In MPU the action is applicable only for the current model. The Set appearance from RGB 126 action must be used if you want to set the color for multiple models.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	-	

5.1.8.2 Set appearance from RGB

The action changes the color representation of the model to a declared color. Please use the Set appearance 125 action to modify gloss effects or similar.

© 2023 INNEO Solutions GmbH

General description

1. Color selection
2. Specify whether all set colors should be removed,
3. or whether only the selected color should be removed.

© 2023 INNEO Solutions GmbH

4. Specify the surfaces to be colored
5. Specify whether the surfaces that are to be colored are allowed to be colored beforehand
6. Options to restrict features to be colored

The *Set appearance from RGB* action sets a selected color (1) on selected surfaces.

Selection is possible for all surfaces of the model, only surfaces that belong to features with the specified name, or surfaces that belong to features of the selected type.

Remember that you can use RegExp 323 in the selection by name.

You can also specify whether the surface may be colored beforehand (5), must be colored beforehand, or should always be colored.

The following exceptions can not be colored (with feature selection):

– The first feature can never be colored. If this is a solid, its surfaces can not be colored with this function.
– Subtypes of extrusions can not be colored, such as sweep features.
– When you color the entire model, only the areas are colored that do not have an explicit color assignment. The same applies to removing colors from the entire model.
– With assemblies, colors set for components in the assembly are retained.

If features are colored, for example bores or threads, they can be filtered according to additional criteria.

– has dimension with tolerance: Specify a tolerance. All dimensions of the feature are checked for a matching tolerance, and the feature if only colored if a match is found.
  Example: *H**
– has feature parameter with value: Specify a feature parameter (e.g., thread pitch) for which to color. Currently, only direct comparisons are supported.
  Examples:
  THREAD_SERIES = UNC
  THREAD_SERIES = UNF
  THREAD_SERIES = ISO
– smallest dimension of type diameter: All dimensions of type diameter are collected, and the smallest diameter is compared to the given range.
  Example: Diameter: 6 <= value <= 6, feature of type : FEATURE-HOLE --> Bore with diamter = 6

CAUTION: The color is only displayed after the model has been redrawn. It can therefore be possible that the changed color of the current model is not immediately visible after rework.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	-	

5.1.9 Model - Layer

This group combines actions. Please choose an action.

5.1.9.1 Add layer

The action creates layers with or without rules. Associativity and independence can be set for layers with rules.

General description

© 2023 INNEO Solutions GmbH

1. The path to the template file for layers can be selected with the browse button. A template model is not used if the selection is blank.
2. Name of the newly created layers. If a template model (1) has been selected, the layers are created in accordance with the rules of the template model. The *Sort* column is responsible for the order within the table only. For adding, the current order of the table is always used.
3. Layer options
   – independent: If the newly created layer is a rule layer, the rule is copied in case of independence. Without independence, the model gets a dependency to the template model.
   – associative: If a layer is associative, it includes already existing elements in accordance with the rules. Otherwise, a layer will only be applied to newly created elements.
   – regenerate and remove independence: Regenerates the layer and removes the independence afterwards. This makes the layer dependent on the loaded template model.

The *Add layers* action adds new layers to parts and assemblies.

The text box (1) must remain blank when newly created layers are to be created without rules. It should also be ensured in the Creo options that the `default_layer_model` option is not set or does not refer to a valid model.

To create layers with the rules from a template model, you must enter the path to the template model into the text box (1). When a layer is created from the template model, the layer definitions that exist in the template model for the layer of the same name are then copied into the edited model. Amongst others, you can use the following variables for this purpose:

– *@project_dir@* - describes the path to the MPZ file for Model Processor

– *@default_layer_model@* - describes the path from the Creo configuration `default_layer_model`. No other entries are permitted here

Specifications from *project_dir* do not require a / to follow the specification.

Make sure to assign the version to the template model; or save it without version assignment in *project_dir*.

To use the action, you MUST be able to create *C:\TEMP* or you must have write privileges there.

```
Example: @project_dir@template_layer.prt.5
```

In the table (2) you can specify the names of the layers you want to create. Layers of the same name in the template layer are automatically linked with it. If the template model contains a rule layer, a rule layer is also created in the processed model.

The selection of the options (3) refers to all layers specified in the table (2).

© 2023 INNEO Solutions GmbH

An independently created layer copies all rules of the template model into the current model. If the layer is not created independently, it gets a reference to the template model and functions only when the template model is in the session. Alternatively, the layer tries to reload it.

A layer that is identified as associative uses the specified rules also on existing elements, and adds them in accordance with the rules of the layer. Otherwise, the rules are only applied to newly added elements.

If the model is loaded in Creo as the current model, the icon *Import from Creo* can be used for entering all layers of the model into the table (2).

CAUTION! Processing is always performed in tabular sequence, from top to bottom. The *Sort* box in the table is not taken into account here.

@project_dir@ is located at: *Project directory/models*

Information about variable entries for the text box (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	X	

5.1.9.2 Add layer state

The action creates a layer state with specified layers in their defined status.

© 2023 INNEO Solutions GmbH

General description

1. Name of the layer state
2. Layer name and state of the added layers

The action *Add layer state* creates a layer state with name (1) that is filled with the specified layers (2). In case not all layers are specified, the layer state will be displayed as not fully constrained in Creo, and therefore provides the *Save* option. It is therefore recommended to hide all undefined layers using `* - hide`.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	X	

5.1.9.3 Delete layers / Move features

The action deletes a layer using the specified layer name, or all empty layers. Before doing so, the content can be copied to another layer.

© 2023 INNEO Solutions GmbH

General description

1. Layers to be deleted, RegExp may be used in the table
2. You can specify that you want to copy the contents of the layers that are to be deleted to another layer, before deletion.
3. All empty layers in the model are deleted if this checkbox is set.

The *Delete layers / Transfer features* action deletes all layers that are specified in the table (1). RegExp 323 can be used here.

To copy the contents of a layer to another layer prior to deletion, you can specify the name of the receiving layer in the text box (2). The layer content is not copied anywhere if this text box is blank. RegExp 323 can also be used in this text box.

The checkbox (3) specifies whether all empty layers should be deleted. Empty layers are searched for after the contents have been copied and layers deleted.

CAUTION! With Creo Elements 5 M070, the *Deleted empty layers* function (3) causes a crash.

CAUTION! When testing, layers that are to be deleted must not be selected. This would cause Creo to crash.

Hidden elements is not a layer. It can therefore not be influenced with filters and/or functions.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.9.4 Delete layer state

The action deletes defined layer states using the specified names.

General description

1. Layer states to be deleted. RegExp 323 may be used in the table

The action *Delete layer states* deletes all layer states specified in table (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.9.5 Group layers

The action puts a layer on another existing layer.

General description

1. Specification of all layers that are to be grouped
2. Text box to specify the layer on which the layers that are specified in the table are grouped

The *Group layers* action links all layers that are specified in the table (1) to the layer that is specified in text box (2). Here, you can use RegExp 323 in both entries.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	X	

5.1.9.6 Rename layers

The action changes the name of a layer to the new, specified name.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the layers that are to be renamed
2. Specifies whether the layer content should be copied if a target layer of a corresponding name exists

In the *Rename layer* action, the names of the layers you want to rename must be entered in the left column of table (1).The new names of the renamed layers are entered in the right column.

RegExp can be used in the table column *from layer name*. Make sure that the value of *to layer name* does not match the RegExp under *from layer name* (see example).

With the checkbox (2), you can specify (checkbox not selected) that a layer is only renamed. This will cause an error if a layer with the new name exists.

If you select checkbox (2) and if a layer with the new name exists, the content of the source layer is copied to the target layer, and the source layer is deleted afterwards. The layer is renamed if a layer with the corresponding target name does not exist.

Hidden elements is not a layer. It can therefore not be influenced with filters and/or functions.

Example that causes an error message to be generated:

```
from layer name; to layer name
T*             ; TEST
```

Solution of the problem:

```
from layer name; to layer name
T*             ; TEMP_VARIABLE
TEMP_VARIABLE ; TEST
```

WARNING: This function does not work for assemblies in Wildfire 5.0.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	X	

5.1.9.7 Set layer state

The action sets a defined layer state using the specified name.

General description

1. Name of the layer state to be set, RegExp 323 may be used here

The action *Set layer state* sets the layer state which is defined in the text field (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.9.8 Set layer status

The action sets the layer status temporarily or in saved in the model.

General description

Control elements (1)

– Plus - adds an entry to the table (2)

– Pen - edits a selected entry in the table (2)

– Minus - deletes a selected entry in the table (2)

Table (2) that contains an overview of all layers and the status that is to be changed.

Checkbox (3) specifies whether the status change is only temporary, or is to be saved in the model.

The *Set layer status* action permits the status of layers that are specified in table (2) to be set to a defined value. The status of a layer is not set when the layer cannot be found. Using RegExp 323 permits a large number of layers to be modified via one line in the table.

The table (2) can not be edited directly. Use the control elements above the table for editing.

If the checkbox (3) is selected, the status that is set after the execution of the action is saved in the model. If the checkbox is not selected, the previous layer status is restored when the model is opened the next time (even after the model has been saved).

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	-	Implicitly when the generic is processed
Assembly	X	
Assembly - instance	-	Implicitly when the generic is processed
Drawing	-	

5.1.10 Model - Combined state

This group combines actions. Please choose an action.

5.1.10.1 Add combined state

The action creates a combined state.

© 2023 INNEO Solutions GmbH

General description

1. Name of the combined state
2. The Layer state 131 which will be linked
3. The view 94 which will be linked
4. The simplified representation 186 which will be linked (empty, Name, MASTER or CURRENT)
5. The cross section 146 which will be linked - the view option can be set here (*none* - not active, *back* view or *front* view)
6. The Annotations and Extra Geometry Flags (Creo 7 only) can be set or unset
7. The exploded view that will be linked - exploded or assembled (for assemblies only)
8. The style that will be linked (for assemblies only)

The action *Add combined state* creates a combined state with name (1) which is created using the specified definition.

© 2023 INNEO Solutions GmbH

If an option (2-7) is not defined, it will not be used.

RegExp 323 may be used in the inputs (1-7).

The Creo configuration option 38 `display_combined_views` should be active in order to fully utilize the advantages of combined sates.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.10.2 Delete combined state

The action deletes defined combined states using the specified names.

General description

1. Combined states to be deleted, RegExp 323 may be used in the table

The action *Delete combined states* deletes all combined states specified in table (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.10.3 Set combined state

The action sets a defined combined state using the specified name.

General description

1. Name of the combined state to be set, RegExp 323 may be used here

The action *Set combined state* sets the combined state which is defined under (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.11 Model - Explode state

This group combines actions. Please choose an action.

5.1.11.1 Add explode state

The action adds a defined exploded view with specified name.

General description

1. Name of the new exploded view

The action *Add explode state* adds the exploded view which is defined in the text field (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.11.2 Delete explode state

The action deletes defined exploded views using the specified names.

© 2023 INNEO Solutions GmbH

General description

1. Exploded views to be deleted, RegExp 323 may be used in the table

The action *Delete explode state* deletes all exploded views specified in table (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.11.3 Set explode state

The action sets a defined exploded view using the specified name.

General description

1. Name of the exploded view to be set, RegExp 323 may be used here

© 2023 INNEO Solutions GmbH

The action *Set explode state* sets the exploded view which is defined in the text field (1). RegExp 323 can be used here.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.12 Model - Cross Section

This group combines actions. Please choose an action.

5.1.12.1 Activate cross section

The action sets a defined cross section active using the specified name.

General description

1. Name of the cross section to be activated, RegExp 323 may be used here
2. All cross sections will be deactivated when selected

The action *Activate cross section* activates the cross section which is defined in the text field (1). RegExp 323 can be used here.

If the selection (2) is activated, no cross section will be activated. If a cross section is activated, it will be set inactive.

© 2023 INNEO Solutions GmbH

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.12.2 Add cross section

The action creates a cross section with defined name and properties.

General description

1. Name of the cross section
2. The type of the cross section:
   – Parallel - parallel to a layer
   – Planar - located on a layer
3. PLane name - the name of the layer
4. Distance - the distance to the layer (only for type *parallel*)

The action *Add cross section* creates a cross section with name (1), of type (2).

© 2023 INNEO Solutions GmbH

Depending on the selected typ,e different additional properties have to be specified.

Parallel

– Plane name (3)

– Distance to the layer (4)

Planar

– Plane name (3)

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	X	

5.1.12.3 Delete cross section

The action deletes defined cross sections using the specified names.

General description

1. cross sections states to be deleted, RegExp 323 may be used in the table

The action *Delete cross section* deletes all cross sections specified in table (1). RegExp 323 can be used here.

© 2023 INNEO Solutions GmbH

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.12.4 Show cross section

The action modifies the display status (displayed / hidden) of a cross section specified by name.

General description

1. Name of the cross section to be displayed / hidden, RegExp 323 may be used here
2. Defines whether the cross section(s) (when using RegExp / wildcards) are to be displayed or the display should be deactivated

The action *Show cross section* displays the cross section(s) defined in the entry box (1), or removes the display. RegExp 323 can be used here.

The selection (2) allows to define whether the cross section shall be displayed (selected) or hidden.

Variable input information can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.12.5 Update cross sections in the model

The action updates the data of all cross section to the current data format.

General description

The action *Update all cross sections in model* updates the data of all cross sections to the current data format.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.13 Model - Material

This group combines actions. Please choose an action.

5.1.13.1 Add material

The action adds a material to a model and can assign it.

General description

1. Path to the material file, without file suffix (use the browse icon to select a path)
2. Specifies whether the new material shall be assigned to the current model
3. Specifies whether skeleton models should also be processed

The *Add material* action adds material to a part, and can set the material as the assigned material.

To add material, you must write the path and the name of the material into the text box (1). Ensure that the material is written without a suffix. Creo automatically recognizes the suffix when the material is added.

Variables are available that help to make the directory path specification more generic.

- *@project_dir@* - describes the path to the Model Processor MPZ. In the directory structure, this corresponds to the *materials* directory.
- *@pro_material_dir@* - describes the path in the Creo configuration option `pro_material_dir`.

Neither specification requires a / to follow the specification.

```
Example: @project_dir@leermaterial
```

The selection *Set as current material* (2) permits Model Processor to set the material in the model as the assigned material after it has been added.

Adding material files to assemblies is not supported.

© 2023 INNEO Solutions GmbH

Look at the `material_upd_smt_bend_table` configuration option if you want to replace materials in sheet-metal parts. Here, you can define how to proceed with the *SMT_Y_FACTOR* (bending factor) and the bending tables. You can use the Set configuration option 38 action to edit configuration options during the Model Processor execution time. This option influences only automatic replacement, not manual material processing.

@project_dir@ is located at: *Project directory/materials*

Caution: Option (3) also lets you assign a material to skeleton models. By default, the action does not assign material to skeleton models, as this is not supported in Creo. However, there are some exception where such an action does make sense.

Information about variable entries for the text box (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	-	

5.1.13.2 Add material parameter

The action adds parameters to the current material if they do not yet exist. Furthermore it can change the description for existing parameters.

© 2023 INNEO Solutions GmbH

General description

Control elements (1)

– Plus - adds a parameter to the table (2)

– Pen - edits a selected entry in the table (2)

– Minus - deletes a selected entry in the table (2)

Table (2) that contains an overview of all parameters. You can sort by a column by clicking the column title.

Checkbox (3) specifies whether the description should be changed anyway if a parameter already exists in the model.

The action *Add material parameter* adds a specified number of parameters to the current material. Here, you can make all definitions like in Creo.

Option (3) allows you to modify the description of existing parameters in the material as specified in the table The parameter is not newly created for this purpose. The relations linked with the parameter thus remain linked.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	
Assembly - instance	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.1.13.3 Delete material

The action deletes material using the specified material name, or unused materials.

General description

1. Specification of the material name without file suffix to specify the material that is to be deleted
2. When the option is selected, the material currently assigned to the model is not deleted.
3. Specifies whether skeleton models should also be processed

The *Delete material* action deletes materials from a model. Assemblies are not supported. Only the name (without the file name suffix) of the materials that are to be deleted is entered into the table (1). RegExp 323 can be used here.

Caution: Option (3) also lets you assign a material to skeleton models. By default, the action does not assign material to skeleton models, as this is not supported in Creo. However, there are some exception where such an action does make sense.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	

© 2023 INNEO Solutions GmbH

Assembly - instance	-	
Drawing	-	

5.1.13.4 Edit material parameter

The action modifies material parameters which are specified by their name or value. The value and the type of the parameter can be modified.

General description

Edit current material parameter

Description:

Model filter:

Search in: 1 ◯ Value ◉ Name

Search by: 2

To use Reg exp start and end line with "/"

New type: 3 String

New value: 4

@date@ - current date

@mdln@ - model name, until "."

@mdlpath@ - model path

@number@ - auto increment number

@para@ - parameter value (current parameter)

%PARAM% - parameter value (other parameter)

%curmod:PARAM% - parameter value from active model (only in drw)

§env-var§ - environment variable

$$sub$x$y$TEXT$$ - substring from TEXT, x-start, y-end

$$pre$x$TEXT$$ - substring from start, x-length

$$poste$x$TEXT$$ - substring from end, x-length

$$posts$x$TEXT$$ - substring until end, x-startposition

$$repl$A$B$TEXT$$ - replace in TEXT, A (RegExp possible) through B

%% - is the % sign

1. Specifies whether the entries in the text box (2) are to be compared with the parameter name or the parameter value
2. Input of the comparison value

© 2023 INNEO Solutions GmbH

3. Specification of the parameter type. The parameter is converted into a corresponding type.
4. Input of the new value. Enter *@para@* if you want to retain the old value.

To edit a parameter, you must first identify it. This can be done on the basis of its name or its value (selection (1)). In the following description, this is always referred to as S1. The value of entry (2) is compared with the value of S1. RegExp can be used here.

The parameter type must be chosen in the selection (3). If the parameter found at the material does not correspond to the selected parameter type, the parameter is deleted from the material and a new parameter of the same name is created. To use the old value (if possible) after the type change, you can use *@para@* as the input for the new value (4).

To assign a Boolean value to a parameter, the value in the input mask is defined as *true/false* or *yes/no*. Other language-specific entries are not permitted here. Entries that can not be recognized (e.g. *tre*, *si*, ...) are recognized as *false*.

Parameters that are to be filled must exist in the material.

Information about variable entries for the text box (4) can be found at Variables 306.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	-	
Assembly instance	-	
Drawing	-	

5.1.13.5 Edit material (from file)

The action modifies the material of a model. The new material is specified by an XLS or CSV file.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the CSV / XLS file that is to be read.
2. Specification of the material directory that contains the materials which are used in the file.
3. Specification of the storage format of the CSV file specified at (1).
4. Specification of the delimiter used in the CSV file specified at (1).
5. Handling generics

Path and file name including suffix must be entered in the input box (1). The file is read and the status of the read file is output underneath the text box (1) when you exit the text box (by selecting another item, such as the action description box). In the text box (2), you can specify the material directory that must contain all materials that are used in the file. If a CSV file is used, you must select the file input format via the selection (3 + 4). The option box (5) specifies how instances are handled.

In the CSV file, the following boxes are evaluated:

– file name - specification of the file name

– path - specification of the file path (data from the hard disk with \, data from Windchill server with /). The specifications must end with a separator character.

© 2023 INNEO Solutions GmbH

– ### value - instead of the *file name* cells, you can specify a parameter name with the suffix *value*

– old material - the currently assigned material (the name without extension)

– current material - the material that is to be assigned (the name without extension)

The materials that are to be assigned can be distinguished by *file name* or parameter content (### *value*). The column with the path is only necessary for an exact selection of the file, not for the function. The following examples show some possibilities.

Typical CSV file:

```
file name;current material;
loko.prt;STEEL;
```

Typical CSV file with path specification:

```
file name;path;current material;
loko.prt;C:\testdata\;STEEL;
```

Typical CSV file with parameter value:

```
MATERIAL value;current material;
Steel;STEEL;
```

@project_dir@ is located at: *Project directory/reports*

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	-	
Assembly - instance	-	
Drawing	-	

5.1.13.6 Replace material

The action replaces the material in a model with a new material. If the original material was assigned to the model, the new material is assigned as well.

© 2023 INNEO Solutions GmbH

General description

1. Text box of the material folder
2. Table to specify the material that is to be replaced (left column: the material that is to be replaced, right column: the material that replaces the material in the left column)
3. Select whether you want to replace materials with the same names (if they exist in the selected folder).
4. Select whether you want to delete the replaced materials from the model, or to retain them

The *Replace material* action replaces a the current material (specified in the left column of table (2)) with a new material (right column of table (2)). The current material may also be the assigned material. Both names can be the same. A material can thus be replaced with a new version of itself. Option (4) permits the replaced materials (with different names) to be deleted from the parts. Selection (3) can be used to specify that in addition to the information given in table (2), all materials with the same name (folder => materials on model) are automatically updated.

If the material is stored in the MPZ, you can press the button to select from the MPZ and select the corresponding material. Pressing the transfer button copies the name into the related box. See also Project Explorer 327.

Materials must be specified without suffix.

@project_dir@ is located at: *Project directory/materials*

Information about using Reg Exp 323 can be found in the related help section.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.14 Model - Parameter

This group combines actions. Please choose an action.

5.1.14.1 Add alternative model parameter to family table

The action adds selected alternative model properties to the family table of the model.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the properties (alternative parameters) of the model

The action *Add alternative mass properties to family table* adds selected (1) alternative parameters to the family table of a generic.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.14.2 Add feature parameter

This action adds feature parameters to the features of the model if they do not yet exist. Furthermore, it can set the description for existing parameters.

General description

1. Selection of the features a parameter should be added to
2. Control elements

– Plus - adds a parameter to the table (3)

– Pen - edits a selected entry in the table (3)

– Minus - deletes a selected entry in the table (3)

3. The table shows all parameters. Use the delete button or the Minus to delete a parameter from the table. Select the title element to sort by columns.

The action *Add feature parameter* adds a specified number of parameters to each feature defined in selection (1) of the model to be overhauled. Here, you can make all definitions like in Creo.

The Creo icon with the green arrow permits all parameters to be copied from the module currently loaded in Creo into the table.

Option (4) allows you to modify descriptions of existing parameters in the feature according to the information in the table. The parameter is not newly created for this purpose. Relations linked with the parameter and specified values thus remain intact.

Parameter data types

– Boolean - Yes/No

© 2023 INNEO Solutions GmbH

– String - character string

– Integer - integer number

– Double - real number

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.14.3 Add parameter

The action adds parameters to the material if they do not yet exist. Furthermore, it can set the description and designation status for existing parameters.

General description

1. Control elements

– Plus - adds a parameter to the table (3)

© 2023 INNEO Solutions GmbH

– Pen - edits a selected entry in the table (3)

– Minus - deletes a selected entry in the table (3)

2. The table shows all parameters. Use the delete button or the Minus to delete a parameter from the table. Select the title element to sort by columns.
3. If the parameter exists in the model that is to be processed, you can select here whether or not the description and/or the designated flag should be set.
4. If an instance can be reached (generic, instance), you can create a table column in the generic's family table for the parameter.

The *Add model parameters* action adds a defined number of parameters to each model that is to be overhauled. Here, you can make all definitions like in Creo.

The Creo icon with the green arrow permits all parameters to be copied from the module currently loaded in Creo into the table.

The option (3) permit descriptions and/or designated flags of existing parameters in the model to be modified according to the information in the table The parameter is not newly created for this purpose. The relations linked with the parameter thus remain linked.

If the model is an instance or a generic, the parameter can also be added to the generic family table. This requires *Add to generic family table* (4) to be selected.

In Model Processor version 3.0 and later, you can specify units for parameters that are created as Double.

Parameter data types

– Boolean - Yes/No

– String - character string

– Integer - integer number

– Double - real number

When you create new data, ensure that the name of the data has not yet been used in the models as a dimension name. Model Processor generates an error if this is the case, and is not able to create this parameter (which is not possible manually either).

To edit instances, you should select *Include instances* in the left-hand Mode menu 314.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.14.4 Add associative parameter to flexible dimension

The actions adds an associative parameter to every defined dimension inside a flexible component.

General description

1. Model name of the flexible component model
2. The flexible dimension of the component model
3. Name of the mapped parameter
4. Option to designate the parameter

The action *Add associative parameter to flexible dimension* search for a defined (1 + 2) placed component and extend the dimension (2) by the associative parameter (3). This only works if there is no current defined associative parameter. The parameter can be designated after wards(4).

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	-	
Part - instance	-	

© 2023 INNEO Solutions GmbH

Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.14.5 Change all designated flags

The action changes the designated property to the specified value in each parameter of the model.

General description

1. Specifies whether all parameters should be designated or not designated
2. Do you want to designate in generic if the model is an instance?

The *Change all designation flags* action changes the status of the designated flag of all parameters of a model. You can use the option (1) to specify whether or not all parameters should be designated. Use the checkbox (2) to specify whether or not the change should be made in the generic if the current model is an instance. The action is ignored for the instance if this is not specified.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.14.6 Delete feature parameter

The action deletes a feature parameter using the specified parameter name or value.

General description

1. Do you want to search for the parameter name or parameter value?
2. Specification of the names or values of the parameters that are to be deleted, according to selection (1)
3. If the model is an instance, do you want the parameter to be deleted in the generic (deletes the entire family table), or in the family table (the parameter falls back to the specification of the generic parameter)?

The action *Delete feature parameter* deletes defined parameters from all features of a model. The parameter can be identified by its name or its parameter value (1). RegExp 323 can be used to specify the name / parameter value of the parameter that is to be deleted in table (2).

If the model is an instance:

– *Delete from generic* deletes the parameter in the generic model. This deletes the parameter in all instances and in the family table.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.14.7 Delete model parameter

The action deletes a model parameter using the specified parameter name or value.

General description

1. Do you want to search for the parameter name or parameter value?
2. Specification of the names or values of the parameters that are to be deleted, according to selection (1)
3. If the model is an instance, do you want the parameter to be deleted in the generic (deletes the entire family table), or in the family table (the parameter falls back to the specification of the generic parameter)?

The "Delete model parameter" action deletes defined parameters. The parameter can be identified by its name or its parameter value (1). RegExp 323 can be used to specify the name / parameter value of the parameter that is to be deleted in table (2).

If the model is an instance:

– *Delete from generic* deletes the parameter in the generic model. This deletes the parameter in all instances and in the family table.

© 2023 INNEO Solutions GmbH

– *Delete only in family table* deletes the parameter in the family table of the generic. This assigns the instance the value of the generic.

To delete Intralink or PDM parameters, you must set the hidden configuration option `allow_create_pdm_param = yes`.

Within a PDM system, you must additionally disconnect the connection to the server, and reestablish it afterwards. Using the action *Start mapkey* is recommended.

Example:

– Mapkey : Disconnect server

– Delete parameters: PROI_*

– Mapkey : Connect server

CAUTION: Deleting PDM parameters can lead to inconsistent data. Utilization with or without PDM system can be impaired.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.14.8 Edit drawing parameter from model

The action fills the value of a parameter from a drawing with a model parameter value. The model for this can be selected in different ways.

© 2023 INNEO Solutions GmbH

General description

1. The parameter name to use the model from the declared parameter (must exist and be declared)
2. The method of selection when the parameter (1) is not found or filled with a model that does not exist
3. The parameters that shall be transferred with specification of source and target parameter (both must exist)
4. The type of output to be generated in case of errors

The action *Fill drawing parameter from model parameter* fills the value of a parameter specified in table (3) with the value of a parameter from a model. Three eventualities may occur:

– no model in the drawing - nothing will be transferred

– one model in the drawing - the model that is found will be used

– multiple models in the drawing - the selection will be performed via the defined selection option (2)

Selection options for multi-model drawings (2):

– Active model

– Model selection through frames - the formats on all sheets will be checked, and if the same model was used everywhere, this model will be used

© 2023 INNEO Solutions GmbH

– throw error message - throws an error message that may prevent saving, for example

– do nothing

After specification of how to select the model (2) and which parameters shall be transferred (3) it may be specified what should happen if the corresponding parameter does not exist at the model (4). This may cause either an error (allows further evaluation) or a warning.

Parameters to be filled must already exist at the drawing.

The parameter (1) must exist at the drawing. In this parameter the name of the selected model will be deposited. Thus the next time the selection (2) will not be made repeatedly but the last chosen model will be used. If this behavior is not desired, you can add the parameter (Add parameter 162) beforehand and delete it (Delete model parameter 167) again afterwards.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly instance	-	
Drawing	X	

5.1.14.9 Edit model parameter

The action modifies model parameters which are specified by their name or value. The value and the type of the parameter can be modified.

© 2023 INNEO Solutions GmbH

General description

1. Specifies whether the entries in the text box (2) are to be compared with the parameter name or the parameter value
2. Input of the comparison value
3. Specification of the parameter type. The parameter is converted into a corresponding type.
4. Input of the new value. Enter *@para@* if you want to retain the old value.
5. If the model is an instance and has a generic, do you want to change it?
6. Do you want to change each instance if the model is a generic?

© 2023 INNEO Solutions GmbH

To edit a parameter, you must first identify it. This can be done on the basis of its name or its value (selection (1)). In the following description, this is always referred to as S1. The value of entry (2) is compared with the value of S1. RegExp can be used here.

The parameter type must be chosen in the selection (3). If the parameter found at the material does not correspond to the selected parameter type, the parameter is deleted from the material and a new parameter of the same name is created. To use the old value (if possible) after the type change, you can use *@para@* as the input for the new value (4).

If the model is an instance, it will only be processed if checkbox (5) is selected. If the checkbox is selected, the result can vary depending on the overhaul, or whether it is a generic or a family table parameter.

To assign a Boolean value to a parameter, the value in the input mask is defined as *true/false* or *yes/no*. Other language-specific entries are not permitted here. Entries that can not be recognized (e.g. *tre*, *si*, ...) are recognized as *false*.

The rule specified in the text box (4) is newly evaluated for each instance when you rework a generic, if all instances are reworked (6 checked). Afterwards, the value is assigned to each instance. You must ensure that the value is in the family table. In a multi-stage family table, this action is only carried out for instances where the parameter is in the family table of the direct parent generic, even if the parent is itself an instance.

Parameters that are to be filled must exist in the model.

Information about variable entries for the text box (4) can be found at Variables 306.

Information about description and model filters can be found at Model filter 229.

Remove restricted

© 2023 INNEO Solutions GmbH

To remove the restricted flag with the Model Processor from one or more parameters, you must use the Creo restriction file. First, you must create a list of the general restrictions of all parts that are to be overhauled. You can use the report actions for this purpose. Next, you must create a file that contains the individual restrictions.

Below is an example that deletes ALL restrictions (except restrictions that were created by relations).

```
restricted.lst
ND_ParamDefArr_K01 = {
}
```

If you want to delete only a part of the restrictions, the remaining restrictions must be redefined in the file.

```
restricted.lst
ND_ParamDefArr_K01 = {
{ Name = test2
  Type = string
  Default = 'blue_new'
  Enum = { 'red_new', 'green_new', 'blue_new' }
}
}
```

The resulting file must now be entered in the Creo *config.pro*.

```
restricted_val_definition C:\restricted.lst
```

Next, you must restart Creo to activate the restriction file. The entered file is not used without a restart. After the restart, you must record a mapkey. This requires a part with restricted parameters to be open. Use the following procedure:

– Open the Creo parameter editor

– Select *Tools -> Update restriction definition* from the menu group

– Select the checkbox in the dialog

– Terminate the dialog with *Assign*

– Terminate the parameter editor with *OK*

After the mapkey has been recorded, you can use the Start mapkey 48 action to use the mapkey also for other models.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Assembly instance	X	
Drawing	X	

5.1.14.10 Edit model parameter from file

The action modifies the value of model parameters which are specified by their name and the model name (in CSV/XSL files).

General description

1. Specification of the CSV or XSL file that is to be read
2. Specification of the storage format of the CSV file specified at (1)
3. Specification of the separator character of the CSV file that is specified at (1)
4. Handling instances
5. Selects whether the parameter shall also be edited in the family table if the part is a generic
6. Selects whether a blocked instance shall automatically be unblocked

Path and file name including suffix must be entered in the input box (1). The file is read and the status of the read file is output underneath the text box (1) when you exit the text box (by selecting another item, such as the action description box). If a CSV file is used, you must select the file input format via the selection (2 + 3). The options (4-6) specify how instances and generics are handled.

© 2023 INNEO Solutions GmbH

Parameters that are to be filled must exist in the model.

The following fields in the CSV file are evaluated:

- *file name* - specification of the file name
- *Old param ###* - Define a parameter ### as alternative to the file name (file name should not be defined) - no instance support
- *file path* - specification of the file path
- *### Value* - the value of parameter ### that is to be set

Typical CSV file:

```
file name;DESIGNATION Value;
test.prt;test file;
peter.prt;Peter's part;
```

When you activate reworking instances in the generic family table (5), they are only specified in the CSV with instance name and suffix (as in *Instance.prt*).

@project_dir@ is located at: *Project directory/reports*

Information about variables for the text box (1) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.14.11 Edit parameter from dictionary

The action writes the translation (created using CSV, MDB or XLS files) of a parameter value into a parameter value.

© 2023 INNEO Solutions GmbH

General description

1. Source of the translation
2. File that contains the translations
3. Source parameter - input of the parameter name
4. Source language - specification of the source parameter (1) language
5. Specify whether parts / individual words of the value shall be translated
6. Specify the contents of the target parameter if a translation / partial translation could not be found

The *Edit model parameter from dictionary* action uses a dictionary to fill a parameter. This requires source (3) and target (4) parameter to be specified after the list. The source parameter must contain meaningful and unambiguous words.

Example:

Power (Macht)	Voltage (Spannung)
Gear (Zahnrad)	Gearbox (Getriebe)

The data base can easily be created in Excel (two-column Excel document) and be saved as a CSV file. The different formats for saving can be selected from the drop-down box (1). Available formats:

– .csv / .txt - UTF8 - tab-separated

– .csv / .txt - UTF8 - semicolon-separated

© 2023 INNEO Solutions GmbH

– .csv / .txt - UTF8 - comma-separated

– .csv / .txt - ISO-8859-2 - tab-separated

– .csv / .txt - ISO-8859-2 - semicolon-separated

– .csv / .txt - ISO-8859-2 - comma-separated

When you translate parts, they are tested in word groups reduced bit by bit starting from the front. A value of *I will be translated*, for example, will be searched in the following sequence for a translation. If possible, parts (5) are translated.

– I will be translated

– I will be

– I will

– I

– will be translated

– will be

– will

– be translated

– translated

If a translation is found for a part, the translation is used. The search for a translation is continued for the remaining text.

The selection (6) is used if a translation could not be found (for parts, if no part could be translated). The following options exist:

– do nothing - the old value remains in the target parameter

– fill up with empty value - the target parameter receives the specified value from the *Empty Value* box (6)

– copy the source value - the target parameter is set to the value of the source parameter

A free database for translation can be downloaded from *dict.cc*. Ensure that you select the correct translation / direction of translation for the database.

The *pmm.mdb* from the Startup TOOLS can also be used as source file for translation. However, this works only if you have Microsoft Access installed on your PC, or if you have added the Access ODBC component to your 32-bit JDBC component management manually.

You can find it in a 64-bit system under *%systemdrive%\Windows\SysWoW64\Odbcad32.exe*

Bear in mind that only the new title *Microsoft Access Driver (*.mdb, *.accdb)* is supported. The title *Microsoft Access Driver (*.mdb)* is not supported. It will generate an error message that appears in the dialog.

© 2023 INNEO Solutions GmbH

You can download the Microsoft Access driver individually from the address *http://www.microsoft.com/en-us/download/details.aspx?id=13255*.

If the driver still cannot be located automatically, create a system DSN link to your database under the name *MSAlink*.

Make sure that the installed Office version and the version of the employed driver (x86, x64) is compatible with your installed Creo. Since a 64-bit Creo can only cooperate with a 64-bit database, you need a 64-bit Office to be installed to be able to use the driver.

From Model Processor version 3.0 onwards, direct server access is no longer supported.

@project_dir@ is located at: *Project directory/reports*

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.14.12 Edit parameter from table cell

The action modifies the value of a model parameter to the value from a defined table cell.

© 2023 INNEO Solutions GmbH

General description

1. It can be selected whether a parameter with the name defined in (4) should be filled in the drawing or in the model.
2. Defines in which drawing sheets to look (selected: only current sheet, not selected: all sheets)
3. Support functions to read the table definition
4. Specification of the table and the parameter into which the data is to be read
5. Support function to read the cells to be read

The action *Edit parameter from table cell* fills a parameter defined in the table (4) with the content of a defined table cell. It can be selected whether the parameter to be filled should be the one in the drawing or the one in the active model of the drawing (1).

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.14.13 Edit parameter from symbol text

The action modifies the value of a model parameter to the value from a variable text in a symbol instance.

General description

1. It can be selected whether a parameter with the name defined in (2) should be filled in the drawing or in the model.
2. Defines the symbol (name of the symbol definition), the variable text and the parameter that should be filled.

© 2023 INNEO Solutions GmbH

The action *Edit parameter from symbol text* fills a parameter defined in the table (2) with the content of a defined variable text. It can be selected whether the parameter to be filled should be the one in the drawing or the one in the active model of the drawing (1).

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.14.14 Fill component parameter by reference

The action sets the value for a component parameter using an assembly reference.

General description

1. Name of the parameter to be set
2. Value for the parameter. The name of the assembly reference is available as *@ref_name@*.

The action *Fill component parameter by reference* sets the value for a component parameter using an assembly reference. It only works for components with a single assembly reference.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.14.15 Rename parameter

The action modifies the name of a model parameter.

General description

1. Specifies whether the entries in the text box (2) are to be compared with the parameter name or the parameter value
2. Input of the comparison value

© 2023 INNEO Solutions GmbH

3. Input of the new name

To edit a parameter, you must first identify it. This can be done on the basis of its name or its value (selection (1)). In the following description, this is always referred to as S1. The value of entry (2) is compared with the value of S1. RegExp can be used here.

The parameter that is to be renamed must exist in the model. The parameter name to which the parameter is to be renamed must not yet be used by another parameter.

CAUTION! With family tables this action may be very time-consuming. Therefore automatic restart after x seconds in the project settings should be re-considered.

Information about variable entries for the text box can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.1.15 Model - Pro/PROGRAM

This group combines actions. Please choose an action.

5.1.15.1 Delete Pro/PROGRAM input lines

The action deletes Pro/PROGRAM lines describing INPUT parameters from the model.

© 2023 INNEO Solutions GmbH

General description

1. Input of the parameters that are to be deleted.

The *Delete model Pro/PROGRAM input lines* action deletes lines that contain the specified parameters from the INPUT block of the Pro/PROGRAM code.

To search explicitly for a line that contains an asterisk, you must use **. A single asterisk is ALWAYS interpreted as a wildcard character. You may also use enhanced RegExp 323.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

Messages

Log information

delete model Pro/PROGRAM input lines	Start of the action
start found at : #	The lines starting at line #

© 2023 INNEO Solutions GmbH

end found at : #	End at line #
deleted	Were deleted

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.15.2 Delete Pro/PROGRAM

The action deletes declared Pro/PROGRAM lines from the model.

General description

1. Lines that are separated by a line break are handled, searched and deleted individually
2. A block that is to be deleted is identified by a start line (2) and an end line (3).
3. The end line of a block that is to be deleted
4. The related ENDIF is recognized automatically. Specifying the end line is not necessary.
5. Start and end line of the block are not deleted.

The *Delete model Pro/PROGRAM lines* action deletes specified lines in the Pro/PROGRAM code of a part or an assembly.

© 2023 INNEO Solutions GmbH

The lines that are to be deleted can be specified in the text box (1). Each line corresponds to a new delete command. If a line in this text box contains ENDIF, for example, each line that contains ENDIF will be deleted. Each line is considered independently, irrespective of the previous line.

To delete a block of lines, you can define a start line and an end line.

Example:

```
INPUT
    entries
END INPUT
```

Start at line: INPUT

End at line: END INPUT

Option (5) must be set since the INPUT and the END INPUT tag are to remain.

To search explicitly for a line that contains an asterisk, you must use **. A single asterisk is ALWAYS interpreted as a wildcard character. You may also use enhanced RegExp 323.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.16 Model - Simplified Representation

This group combines actions. Please choose an action.

5.1.16.1 Create a simplified representation

The action adds a simplified representation to the model.

© 2023 INNEO Solutions GmbH

General description

1. The name of the newly created simplified representation
2. The type of the newly created simplified representation
   - exclude
   - reverse
   - include
   - exclude
   - substitute
   - geom
   - graphics
   - symb
   - none
   - Default Envelope

The *Create simplified representation* action creates a new simplified representation with the given name and type.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.1.16.2 Delete simplified representation

The action deletes one or multiple simplified representations by their name.

General description

1. Specification of the simplified representations that are to be deleted

With the *Delete simplified representation* action, a simplified representation defined by its name (1) is deleted from the model.

To specify the names of the simplified representations, Wildcards and RegExp 323 may be used.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.16.3 Edit simplified representation

The action modifies the default definition of a simplified representation.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the name of the representation that is to be edited
2. Selection of the representation form of all subordinate components that is to be set.
3. Input of the name of the simplified representation, if *user-defined* was selected as representation form
4. Additive to the std. definition you can include features with defined properties inside the simplified representation. The properties can be defined by the control elements (4).

The *Edit simplified representation* action redefines a simplified representation in an assembly. Settings (simplified representations) of the features of the first layer are set to the value from the selection (2). If *user-defined* was chosen in the selection (2), the system looks for simplified representations of the name from text box (3) and uses them.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.16.4 Rename simplified representation

The action changes the name of a simplified representation to the new, specified name.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the simplified representations that are to be renamed (left) and of the new names (right).

The *Rename simplified representation* action permits individual simplified representations to be renamed. Specify the current name of the representations in the left column of the table (1) and the new name of the representations in the right column.

The action does not support RegExp. All names must be entered in a Creo-compatible way (the entry is not case-sensitive).

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	-	
Assembly	X	
Assembly - instance	-	
Drawing	-	

5.1.16.5 Set simplified representation

The action sets a simplified representation defined by name or type.

© 2023 INNEO Solutions GmbH

General description

1. Select the type of the representation that shall be set (1. *User defined*, 2. - 4. default definitions)
2. Input the name of the simplified representation if *User defined* has been selected at (1)
3. Just update the representations in memory by setting and resetting

The *Set simplified representation* action sets a simplified representation that is saved in the model. A representation can be selected either by its name (*User defined*), or a default representation is selected. The text box to enter a name (2) will be displayed only for *User defined*. If the representation should only be updated and not remain changed, option (3) may be selected. This option, after setting the representation, resets it to the previous one.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.1.17 Drawing

This group combines actions. Please choose an action.

5.1.17.1 Edit model notes from file

The action modifies drawing texts in a drawing which is specified by an XLS or CSV file.

© 2023 INNEO Solutions GmbH

General description

1. Specification of the CSV / XLS file that is to be read.
2. Specification of the storage format of the CSV file specified at (1).
3. Specification of the delimiter used in the CSV file specified at (1).
4. Defines whether only drawing texts on the current sheet should be considered.

For the *Replace drawing texts (from file)* action, the CSV or XLS file to be read must be specified in text box (1). The columns that are shown in the *List report* must be given in the file. A more detailed declaration of the file is possible using the options (2,3).

There are two possible types of CSV files:

1. Item value old; Item value
   Replaces a value in any cell that corresponds to the value of `Item value old` with `Item value`.
2. Filename;ID; Item value
   Replaces a value in the cell that has the same ID and is in the same part.

Do not forget to also specify the title row within the file.

Example (the displayed parameter LANG is replaced with LOCAL_LANG, both read from the first part):

```
Item value old;Item value
&lang:1;&local_lang:1
```

The notes that are to be filled must already exist in the model.

@project_dir@ is located at: *Project directory/reports*

Information about variable entries for the text box (1) can be found at Variables 306.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly instance	-	
Drawing	X	

5.1.17.2 Remove all annotations

The action removes all annotations from a drawing.

General description

The action *Remove all annotations* removes all shown annotations from all sheets of a drawing.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.17.3 Remove unused drawing models

The action removes unused models from a drawing.

General description

The action *Remove unused drawing models* removes unused models from a drawing.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.17.4 Replace format

The action replaces a drawing frame (format) with a new drawing frame. Tables that should not be replaced can be specified.

© 2023 INNEO Solutions GmbH

General description

1. Folder that contains the drawing frame files
2. Processing of multi-model drawings
3. Process current sheet only or all sheets of the drawing
4. In the left column of the table you have to specify the drawing frame without suffix. In the second column of the table you have to specify the drawing frame with suffix.

The action *Replace format* replaces a drawing frame with a new drawing frame. Specify the folder that contains the drawing frame in the text box (1). The individual drawing frames that are to be replaced are entered in table (4). The left column contains the current drawing frame without suffix. The new drawing frame with suffix must be entered in the right column.

The format frames can be downloaded from different systems. This requires the related folders to be selected.

– normal file path

– *@project_dir@* - from MPZ file

– *@pro_format_dir@* - from the directory selected in the configuration setting `PRO_FORMAT_DIR`

– empty - similar to *@pro_format_dir@*, loading from PDM system with preceding `PRO_FORMAT_DIR`

© 2023 INNEO Solutions GmbH

If you have downloaded data in the Creo session from a folder for which you set `PRO_FORMAT_DIR` (the *Set drawing frame* action also changes `PRO_FORMAT_DIR`), you should start a new Creo session if a fault occurs, in order to start with the `PRO_FORMAT_DIR` default settings.

Possible drawing frame inputs in the table:

First column

– Format name without extension

– *noFormatFound* - if a format has not yet been assigned

Second column

– Format name with extension

– **.frm* - a format drawing frame

– **.drw* - a drawing from which the format name is used. With *:<Number>* after the file name, you can specify the page from which the format frame shall be used.

Third column

– Empty - all tables of the old drawing frame are removed

– xxx:yyy - only the tables are removed that are not in the frame of the relative specification *xxx:yyy* (e.g. 123:828). The coordinates of a table can be found via the table data report.

@project_dir@ is located at: *Project directory/models*

Use the options (2+3) to define which model should be the active model for the drawing frame (2) and which sheets should be processed (3). If the check for the active model should not be run only once for the entire drawing, but repeated for each sheet, you have to configure a grouping action.

Example:

© 2023 INNEO Solutions GmbH

In the first column of the table (4), you can use RegExp 323.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.17.5 Replace view model display

The action replaces a specified representation style in a drawing view with another representation style.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the representation style to be replaced
2. Selection of the line representation that is to be set

The action *Replace view model display* replaces the line representation of views in drawings. In the selection (1) you can choose the settings for which a replacement should be performed. Use the selection (2) to choose the representation style to be set.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.17.6 Set current drawing model

The action changes the current model of a drawing.

General description

1. Name of the model that is to be set as the current drawing model

© 2023 INNEO Solutions GmbH

The action *Set current drawing model* sets the model specified under (1) (PRT or ASM) as the current model. The model already has to be assigned to the drawing as one of the drawing models.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.17.7 Set current sheet scale

The action set the sheet scale of the current sheet.

General description

The action *Set current sheet scale* set the sheet scale of the current sheet to a specified value. The value need to be defined in the calculated value. E.g. for an ISO value 1:10 its 0,1. The insert of a fraction is not possible.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	

© 2023 INNEO Solutions GmbH

Assembly - instance	-	
Drawing	X	

5.1.17.8 Show unbend features on views

The action show the unbend features in all views inside a drawing.

General description

The action *Show unbend features on views* cycle through all views of a drawing and show all axis that are created by unbend feature inside the view. This is often need for the export of the drawing.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.18 Drawing - Export

This group combines actions. Please choose an action.

5.1.18.1 Export 2D PDF file

The action exports the drawing to a PDF file.

© 2023 INNEO Solutions GmbH

General description

1. Path to save the PDF in. You can also use the model directory.
2. Name of the PDF to be created
3. Profile which defines export settings (optional)
4. Is no profile defined you can use the options to define the settings (recommended)

The action *Export PDF file* creates a 3D PDF file with the defined name (2) in the defined location (1). The export definition can be defined by Creo settings or explicit(4). For an exact definition of the export settings a profile (3) can be used.

Information about variable entries for the text boxes (1,2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	

© 2023 INNEO Solutions GmbH

Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.18.2 Export DWG file

The action creates a DWG file of the drawing. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Name used for saving the model
3. Specify the elements you want to export

The action *Export DWG file* exports a drawing as a DWG file. In text box (1), you must specify the folder to which you want to save the resulting DWG file.

Enter the name for the export in text box (2). You may use variables here.

In option (3), you can enter various export configurations.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.18.3 Export DXF file

The action creates a DXF file of the drawing. This requires the output format, the storage location and name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Name used for saving the model
3. Specify the elements you want to export

The action *Export DXF file* exports a drawing as a DXF file. In text box (1), you must specify the folder to which you want to save the resulting DXF file.

Enter the name for the export in text box (2). You may use variables here.

In option (3), you can enter various export configurations.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.18.4 Export TIFF file

The action generates an image (TIFF) of the model. This requires the storage location and the name to be specified.

General description

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. Name used for saving the model
3. Specifies whether an existing file shall be overwritten
4. For multi-sheet drawings with sheets of different sizes, drawing rotation has to be disabled

© 2023 INNEO Solutions GmbH

5. Specifies how mult-sheet drawings should be output
   - current sheet -> with _sheetno: Only the current sheet, with the sheet number appended after the name (2)
   - current sheet -> without _sheetno: Only the current sheet, without changes to the name (2)
   - all sheets -> multiple files: All sheets in the drawing, each with the sheet number appended after the name (2)

The *Export TIFF file* action exports drawings to TIFF files. Changes to the TIFF export can be made using configuration options.

The following settings are recommended:

– tiff_compression - g4

– tiff_type - grayscale

– raster_plot_dpi - 400

The name of the output file is always determined by (2) in combination with the option selected at (5).

To set configuration options, you can use the Set configuration options 38 action in the header area.

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.19 Drawing - Note

This group combines actions. Please choose an action.

© 2023 INNEO Solutions GmbH

5.1.19.1 Add a note

The action adds a note from a TXT file to a specified position in a drawing.

General description

1. Note file (TXT)
2. Text height
3. Positions:
   - Format name - you CAN specify a drawing frame here to add the note only to this drawing frame
   - Sheet - you CAN specify a sheet number here to add the note only to this sheet
   - Position - you HAVE to specify a position to place the note at

With the *Add note* action, a note is added to a drawing. This note is placed on the sheet according to the specified position.

When a format name is specified, adding of the note will be restricted to sheets with this format.

When a sheet is specified, adding of the note will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

© 2023 INNEO Solutions GmbH

– Top right: -0:-0

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

Messages

Log information

add note	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.19.2 Delete a note

The action deletes a note on basis of its Position, or by specification of text from a drawing.

© 2023 INNEO Solutions GmbH

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to delete the note only in this drawing frame
   - Sheet - you CAN specify a sheet number here to delete the note only on this sheet
   - Position - you CAN to specify a position at which the note is placed
   - Note text - you CAN specify a note text to be deleted

With the *Delete note* action, a note will be deleted from a drawing. To do so, there are different ways to further declare this note.

When a format name is specified, deletion of the note will be restricted to sheets with this format.

When a sheet is specified, deletion of the note will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The note text does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

Note that in each table row, a position and / or a note text must be specified. When position and note text are specified, they are linked with OR, thus one of the conditions being true causes deletion.

```
Format name AND Sheet AND (Position OR Note text)
```


© 2023 INNEO Solutions GmbH

Format names and sheet numbers that are not specified match with all. Positions and note texts that are not specified match with nothing.

Use the auxiliary functions to make selection of the position easier.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

Messages

Log information

delete note	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.19.3 Move a note

The action moves a note on a drawing to a new specified position.

© 2023 INNEO Solutions GmbH

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to move the note only in this drawing frame
   - Sheet - you CAN specify a sheet number here to move the note only on this sheet
   - Old position - you CAN specify the position of the note before moving
   - Note text - you CAN specify the text of the note to be moved
   - New position - you HAVE to specify a position for the note to be moved to

With the *Move note* action, a note on a drawing will be moved to a new position. This note is repositioned on the sheet according to the specified position.

When a format name is specified, moving of the note will be restricted to sheets with this format.

When a sheet is specified, moving of the note will be restricted to the sheet with this page number.

The old and new position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The note text does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

If you specify an *Old position* and a note text, they are linked with OR, thus one of the conditions being true causes the note to be moved.

```
Format name AND Sheet AND (Position OR Note text)
```


© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

Messages

Log information

move note	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.20 Drawing - Symbol

This group combines actions. Please choose an action.

5.1.20.1 Add symbol

The action adds a symbol from a SYM file to a specified position in a drawing.

© 2023 INNEO Solutions GmbH

General description

1. Specify symbol file
1. Positions:
   - Format name - you CAN specify a drawing frame here to add the symbol only to this drawing frame
   - Sheet - you CAN specify a sheet number here to add the symbol only to this sheet
   - Position - you HAVE to specify a position to place the symbol at

With the *Add symbol* action, a symbol is added to a drawing. This symbol is placed on the sheet according to the specified position.

When a format name is specified, adding of the symbol will be restricted to sheets with this format.

When a sheet is specified, adding of the symbol will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

MP User

If this action is used in MPU, it is imperative to use the Creo2 version of the DLL.

© 2023 INNEO Solutions GmbH

Creo2 only: If the action is triggered with a mapkey in MPU, a change of surface elements is necessary after execution of the mapkey to update the drawing tree.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.20.2 Switch symbol group variant

The action changes the displayed variant at a given symbol group.

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to edit the symbol only in this drawing frame
   - Sheet - you CAN specify a sheet number here to edit the symbol only on this sheet
   - Old position - you CAN specify the position of the symbol
   - you HAVE to specify Old position if Symbol name is not specified. Wildcard * can be used.
   - Symbol name - you CAN specify the name of the symbol to be changed

© 2023 INNEO Solutions GmbH

– you HAVE to specify Symbol name if Old position is not specified. Wildcard * can be used.

– New variant - you HAVE to specify the name of the variant to be displayed.

With the *Change symbol group* action, you can switch to a new variant defined in a symbol group on a drawing.

When a format name is specified, changing the symbol will be restricted to sheets with this format.

When a sheet is specified, changing the symbol will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The symbol name does not always have to be specified completely. You may use Wildcards and RegExp 323 in it.

If you specify a position and a symbol name they are linked with OR, thus one of the conditions being true causes the symbol to be changed.

```
Format name AND Sheet AND (Position OR Symbol name)
```

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.1.20.3 Create bar code (PDF 417)

The action creates a bar code (standard 417) as a symbol in a drawing. Afterwards, you need to place the symbol using Add symbol 211.

General description

1. Specify a name for the drawing symbol. The name is essential for placing the symbol.
2. Enter the text to be encoded in the bar code
3. Specify the error correction level (redundancy)
4. Specify the symbol color by entering a system color
5. Specify the pixel size based on the drawing's unit system
6. Specify a factor by which to reduce the bar width. Pixel size - slim factor = bar width

The action *Create bar code (PDF 417)* creates a bar code symbol in a drawing. The symbol then needs to be placed in the drawing using the action Add symbol 211.

To create a symbol, specify the symbol name (1) and bar code text (2). You also need to specify the error correction level (3), a system color (4) and the pixel size (5).

You can use variables in the text field (2).

Note that the slim factor (6) does not reduce the bar code width. It only reduces the width of the individual bars, which improves readability for some scanners.

© 2023 INNEO Solutions GmbH

Example with a slim factor / pixel size ratio of 2/3

Example without slim factor

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.20.4 Create DM code symbol

The action creates a DM(Data Matrix) code as symbol feature in a drawing. Afterwards, you need to place the symbol using Add symbol 211.

To create a QR code in 3D, please use the action Create DM dot matrix 64.

© 2023 INNEO Solutions GmbH

General description

1. Specify a name for the drawing symbol. The name is essential for placing the symbol.
2. Enter the text to be encoded in the DM code
3. Specify the symbol color by entering a system color
4. Specify the pixel size based on the drawing's unit system

The action *Create DM code symbol* creates a DM code symbol in a drawing. After that, the symbol needs to be placed on the drawing using the action Add symbol 211.

The symbol name (1) and text (2) for the QR code to be created can be specified. You also need to specify the system color (3) and the pixel size (4).

Variables may be used in the text (2).

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	

© 2023 INNEO Solutions GmbH

Assembly - instance	-	
Drawing	X	

5.1.20.5 Create QR code symbol

The action creates a QR code as symbol feature in a drawing. Afterwards, you need to place the symbol using Add symbol 211.

To create a QR code in 3D, please use the action Create QR dot matrix.

General description

1. Specify a name for the drawing symbol. The name is essential for placing the symbol.
2. Enter the text to be encoded in the QR code
3. Specify the error correction level (redundancy)

© 2023 INNEO Solutions GmbH

4. Specify the bit-length per character. This also affects which characters can be written.
5. Specify the symbol color by entering a system color
6. Specify case-sensitivity
7. Specify the pixel size based on the drawing's unit system

The action *Create QR code symbol* creates a QR code symbol in a drawing. After that, the symbol needs to be placed on the drawing using the action Add symbol 211.

The symbol name (1) and text (2) for the QR code to be created can be specified. You also need to specify the error correction level (3), character encoding (4), system color (5), case-sensitivity (6) and the pixel size (7).

Variables may be used in the text (2).

Information about variable entries for the text box (2) can be found at Variables 306.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.20.6 Delete a symbol

The action deletes a symbol from a drawing by its position or name.

General description

© 2023 INNEO Solutions GmbH

1. Positions:
   - Format name - you CAN specify a drawing frame here to delete the symbol only in this drawing frame
   - Sheet - you CAN specify a sheet number here to delete the symbol only on this sheet
   - Position - you CAN to specify a position at which the symbol is placed
   - Note text - you CAN specify a symbol name to be deleted

By the *Delete symbol* action, a symbol will be deleted from a drawing. To do so, there are different ways to further declare this symbol.

When a format name is specified, deletion of the symbol will be restricted only to sheets with this format.

When a format name is specified, deletion of the symbol will be restricted to sheets with this format.

When a sheet is specified, deletion of the symbol will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The symbol name does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

Note that in each table row, a position and / or a symbol name must be specified. When position and symbol name are specified, they are linked with OR, thus one of the conditions being true causes deletion.

```
Format name AND Sheet AND (Position OR Symbol name)
```

Format names and sheet numbers that are not specified match with all. Positions and symbol names that are not specified match with nothing.

Use the auxiliary functions to make selection of the position easier.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	

© 2023 INNEO Solutions GmbH

Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

Messages

Log information

delete symbol	Start of the action

Errors

xxx - failure at run command in Creo	An error occurred during access to Creo.

5.1.20.7 Remove symbol paths

The action remove symbol paths removes the symbol paths at symbol definitions inside a drawing.

General description

The action *Remove symbol paths* deletes all paths from symbol filtered by textbox(1). Afterwards you can use Update symbol definitions 223 to update your symbols. All instances of the symbol (placed on the drawing) would be updated too.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	

© 2023 INNEO Solutions GmbH

Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.20.8 Move a symbol

The action moves a symbol on a drawing to a new specified position.

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to move the symbol only in this drawing frame
   - Sheet - you CAN specify a sheet number here to move the symbol only on this sheet
   - Old position - you CAN specify the position of the symbol before moving
   - Note text - you CAN specify the name of the symbol to be moved
   - New position - you HAVE to specify a position for the symbol to be moved to

With the *Move symbol* action, a symbol on a drawing will be moved to a new position. This symbol is repositioned on the sheet according to the specified position.

When a format name is specified, moving the symbol will be restricted to sheets with this format.

When a sheet is specified, moving the symbol will be restricted to the sheet with this page number.

The old and new position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

© 2023 INNEO Solutions GmbH

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The symbol name does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

If you specify an *Old position* and a symbol name, they are linked with OR, thus one of the conditions being true causes the symbol to be moved.

```
Format name AND Sheet AND (Position OR Symbol name)
```

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.20.9 Update symbol definitions

The action updates all symbol definitions used in a drawing.

General description

The action *Update symbol definitions* updates all symbol definitions used in a drawing (and consequently, after reloading, the symbol instances in the drawing). The directory *PRO_SYMBOL_DIR* is scanned for changes. If you do not want to use the directory

© 2023 INNEO Solutions GmbH

PRO_SYMBOL_DIR, you have to redefine the corresponding configuration option for this action, using Set configuration options 38.

You can define the update process with the Option(2) in combination with the textbox (1). So a symbol can be loaded in multiple times with the same name if it comes from different paths. So make sure that the path for the update is the same as it is used in the current symbol. The action Remove symbol paths 221 could help at the cleanup of symbol definitions and solve problems early in the workflow.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.21 Drawing - Table

This group combines actions. Please choose an action.

5.1.21.1 Add a table

The action adds a table from a TBL file to a specified position in a drawing.

© 2023 INNEO Solutions GmbH

General description

1. Table file (TBL)
2. Positions:
   - Format name - you CAN specify a drawing frame here to add the table only to this drawing frame
   - Sheet - you CAN specify a sheet number here to add the table only to this sheet
   - Position - you HAVE to specify a position to place the table at

With the *Add table* action, a table is added to a drawing. This table is placed on the sheet according to the specified position.

When a format name is specified, adding of the table will be restricted to sheets with this format.

When a sheet is specified, adding of the table will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

Table positioning is based on the table point of origin. The method of creation in Creo is decisive for where the table origin is in the table. The table origin is visible when adding a table (point with which the table is attached to the cursor) and is considered when reading the table position. The table origin cannot be modified after creation of the table.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.21.2 Delete a table

The action deletes a table defined by its position or by specification of text from a drawing.

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to delete the table only in this drawing frame
   - Sheet - you CAN specify a sheet number here to delete the table only on this sheet
   - Position - you CAN to specify a position at which the table is placed
   - Table text (Name:Row:Column) - you CAN specify a text from the table to be deleted

With the *Delete table* action, a table will be deleted from a drawing. To do so, there are different ways to further declare this table.

When a format name is specified, deletion of the table will be restricted to sheets with this format.

© 2023 INNEO Solutions GmbH

When a sheet is specified, deletion of the table will be restricted to the sheet with this page number.

The position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The table text (*Name:Row:Column*) does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

You can specify the row and column as * to access all rows or columns of a table.

Note that in each table row, a position and / or a table text must be specified. When position and table text are specified, they are linked with OR, thus one of the conditions being true causes deletion.

```
Format name AND Sheet AND (Position OR Table text)
```

Format names and sheet numbers that are not specified match with all. Positions and table texts that are not specified match with nothing.

Rows in table (1) that do not have either a position or a table text will be removed when the delete action is executed. Text in any other cells is ignored.

Use the auxiliary functions to make selection of the position easier.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.1.21.3 Move a table

The action moves a table on a drawing to a new specified position.

© 2023 INNEO Solutions GmbH

General description

1. Positions:
   - Format name - you CAN specify a drawing frame here to move the table only in this drawing frame
   - Sheet - you CAN specify a sheet number here to move the table only on this sheet
   - Old position - you CAN specify the position of the table before moving
   - Table text (Name:Row:Column) - you CAN specify a text from the table to be moved
   - New position - you HAVE to specify a position for the table to be moved to

With the *Move table* action, a table on a drawing will be moved to a new position. This table is repositioned on the sheet according to the specified position.

When a format name is specified, moving of the table will be restricted to sheets with this format.

When a sheet is specified, moving of the table will be restricted to the sheet with this page number.

The old and new position must be specified in format (*x:y*). Positioning can also be originated from other drawing corners than bottom left using negative numbers:

– Bottom left: 0:0

– Top left: 0:-0

– Bottom right: -0:0

– Top right: -0:-0

The table text (*Name:Row:Column*) does not always have to be a complete text. You may use Wildcards and RegExp 323 in it.

You can specify the row and column as * to access all rows or columns of a table.

If you specify an *Old position* and a table text, they are linked with OR, thus one of the conditions being true causes the table to be moved.

© 2023 INNEO Solutions GmbH

```
Format name AND Sheet AND (Position OR Table text)
```

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

5.2 Model filter

A filter decides whether an action is carried out (true) or not (false). Several filters can be combined to specify the decision more precisely.

Filters can be defined for most actions of the Model Processor. We distinguish between the three standard filters and the enhanced filters.

Each action can have any description assigned in the *Description* (1) box. The description does not have any effect on the behavior of the action. The description is shown in the action list 29 to give a better orientation of the logic contents of the action. It is thus only of use to the user.

Standard filters

The three filters (2,3,4) that are directly related to file types are standard filters. These are:

– Parts (2) - the selected model will be overhauled if it is a part

– Assemblies (3) - the selected model will be overhauled if it is an assembly

– Drawings (4) - the selected model will be overhauled if it is a drawing

Standard filters are not considered in the header and footer and therefore are not shown in the overview either.

© 2023 INNEO Solutions GmbH

Enhanced filters

The enhanced filters are executed when the filter checkbox (5) is set. An enhanced filter can be set globally, or per action that is to be carried out. Clicking on the filter icon opens a menu that permits filters to be added.

Use the *Plus* icon to add filters from the list. Use the *Minus* icon to remove selected actions from the list. All filters in the list can be moved via drag-and-drop within the filter list.

To add a new filter, use the same procedure as for adding an action in an action list 29.

There are two possibilities of interconnecting filters.

In an AND operation with two or more filters (shown by an *&* at the end of the action representation), the combined filter criteria are only met (true) if all part filters are met as well.

&	Filter 1 = true	Filter 1 = false
Filter 2 = true	true	false
Filter 2 = false	false	false

In an OR operation with two or more filters (shown by || at the end of the action representation), the combined filter criteria are met (true) as soon as one of the part filters is met.

© 2023 INNEO Solutions GmbH

||	Filter 1 = true	Filter 1 = false
Filter 2 = true	true	true
Filter 2 = false	true	false

Use the button *Other operation* to toggle between AND and OR operations.

Many filters support RegExp 323. Examples can be found on the related help pages.

Filter combinations can be saved via the floppy disk icon. A filter combination is saved inside the MPZ. From here, it can be loaded via the drop-down list next to the *Save* icon. Filters you load are appended to the list of existing filters. Existing filters are NOT deleted.

Filter combinations can be deleted with the recycle bin icon when they are incorrect, obsolete, or no longer required. You need not select the filter first. A window opens that permits the filters to be selected that you want to delete.

5.2.1 Assembly - Explode state exists

Checks whether the specified explode state exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the explode state exists.

Exclude
The filter returns *true* if the explode state does not exist.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	-	
Assembly	X	
Drawing	-	

5.2.2 Assembly - Style state exists

Checks whether the specified style state exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the style state exists.

Exclude
The filter returns *true* if the style state does not exist.

Prerequisites

Part	-	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.3 Drawing - Current model

Drawing - current Model

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Model name: *

If no current model: ☑ include (If filter is true)

You can use Reg exp in Format name, mind the /.

Reads the current model from a drawing and compares it to the specified name.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the name of the current model of the drawings is the one specified in the *Model name* text box.

You can additionally specify the value returned by the filter when a current model does not exist in the drawing.

Exclude
The filter returns *true* if the name of the current model of the drawings is not the one specified in the *Model name* text box.

The evaluation of *If no current model* is negated.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.2.4 Drawing - Dimension text

Drawing - Dimension text

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Name: *

You can use Reg exp in path, mind the /.

Checks whether the specified dimension text exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the dimension text exists.

Exclude
The filter returns *true* if the dimension text does not exist.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.5 Drawing - Format name

Drawing - Format name

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Format name: *

You can use Reg exp in Format name, mind the /.

Checks whether the name of the format of the current drawing matches the specified value.

© 2023 INNEO Solutions GmbH

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the name of the format of the current drawing matches the specified value.

Exclude
The filter returns *false* if the name of the format of the current drawing matches the specified value.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.6 Drawing - Formats on sheets

Drawing - Formats on sheets

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter: Have

◉ one or more

○ all

sheets a format?

Checks whether formats are assigned to the specified number of drawing sheets.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if formats are assigned to the specified number of drawing sheets.

© 2023 INNEO Solutions GmbH

Exclude
The filter returns *false* if formats are assigned to the specified number of drawing sheets.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.7 Drawing - Is multi model drawing

Drawing - Is multi model drawing

Description:

Filter type: ◉ include

○ exclude

Checks whether the current drawing is a multi-model drawing.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the current drawing is a multi-model drawing.

Exclude
The filter returns *false* if the current drawing is a multi-model drawing.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.2.8 Drawing - Sheet size

Drawing - Sheet size from current sheet

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Height <

Height =

Height >

Width <

Width =

Width >

Empty fields are ignored.

Checks the sheet size of the current drawing sheet.

Boxes without input are not evaluated.

Use the Creo(Pro/E) import icon to read the current sheet sizes.

The filter returns *false* if the model is not a drawing, or a drawing without a sheet.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the current sheet matches all input criteria.

Exclude
The filter returns *false* if the current sheet matches all input criteria.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.2.9 Drawing - Symbol on drawing

Darawing - Symbol on drawing

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter name: /.*/

You can use Reg exp in filter, mind the /.

Checks whether a symbol with the specified name is show in the drawing. A symbol is ignored if it exists in the drawing without being shown on any sheet.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the symbol is displayed in the drawing.

Exclude
The filter returns *true* if the symbol is not displayed in the drawing.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.10 Drawing - Symbol in drawing

Drawing - Symbol in drawing

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter name: /.*/

You can use Reg exp in filter, mind the /.

© 2023 INNEO Solutions GmbH

Checks whether a symbol with the specified name is loaded in the drawing. A symbol is that IS NOT shown but exists in the drawing would be also found.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the symbol is found in the drawing.

Exclude
The filter returns *true* if the symbol is not found in the drawing.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.11 Drawing - Table on drawing

Checks whether a table exists on the specified drawing sheet. If you do not specify a sheet, all sheets will be searched.

The table can be defined by its position or by the content of a cell. If you define both, you can specify whether to link the cell and the position with AND or with OR.

© 2023 INNEO Solutions GmbH

If you want to use Creo variables (such as *&format*), you have to surround the search string with * or use the Creo-specific syntax *{0:&format}*.

For Example, **&format*:4:1* will check *Row 1:, Column: 4, Content: &format*

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the table exists in the drawing.

Exclude
The filter returns *true* if the table does not exist in the drawing.

Prerequisites

Part	-	
Assembly	-	
Drawing	X	

5.2.12 Environment - Creo version

Environment - Creo version

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Version: ☐ Creo 4.0

☐ Creo 5.0

Verifies the current Creo version. This filter does not refer to the current model. It can be used for controlling mapkeys that can be different depending on the Creo version used.

Include
The filter returns *true* if the selected Creo version is in use.

Exclude
The filter returns *false* if the selected Creo version is in use.

© 2023 INNEO Solutions GmbH

5.2.13 Environment - Environment variable

Environment - Environment variable

Description:

Filter type: ◉ include

○ exclude

Variable:

Value: *

You can use Reg exp in name and value, mind the /.

Checks whether the specified environment variable has the specified value.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the environment variable exists with the specified value.

Exclude
The filter returns *false* if the environment variable exists with the specified value.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.14 Environment - File exists

Environment - File exists

Description:

Filter type: ◉ include

○ exclude

Filter:

You can use Reg exp in Path, mind the /.

© 2023 INNEO Solutions GmbH

Checks whether the specified file exists.

To check a directory, you have to make sure that the trailing slash is removed, for example by using `$$pree$1$@mdlpath@$$`

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the file exists.

Exclude
The filter returns *true* if the file does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.15 Environment - Model exists

Checks whether the specified model exists and can be loaded in the current context (*PRO_SEARCH_PATH*, working directory).

Under *with value*, you can specify a parameter whose value should be checked for the model.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

© 2023 INNEO Solutions GmbH

Include
The filter returns *true* if the model can be loaded.

Exclude
The filter returns *true* if the model cannot be loaded.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.16 Environment - User Parameter

Checks whether the specified user variable has the specified value. If you enter * as the value, the value is not evaluated, and only the presence of the user variable is checked.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the user variable exists with the specified value.

Exclude
The filter returns *false* if the user variable exists with the specified value.

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Assembly	X	
Drawing	X	

5.2.17 Model - Dimension tolerance

Model - Dimension tolerance

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Options: ☑ Model contains asymetric tolerance

Checks whether the model contains asymmetric tolerances.

For a description of the basic filter functions, please refer to Model filters 229.

Include
If the model contains asymmetric tolerances, the filter returns *true*.

Exclude
If the model contains no asymmetric tolerances, the filter returns *true*.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.18 Model - Double defined parameters

Model - Double defined Parameters

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Checks whether parameters are defined more than once in a model.

© 2023 INNEO Solutions GmbH

For a description of the basic filter functions, please refer to Model filters 229.

Include
If one or more parameters are defined more than once, the filter returns *true*.

Exclude
If one or more parameters are defined more than once, the filter returns *false*.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.19 Model - File name

Checks whether the name of the current model matches the specified value.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the name of the current model matches the specified value.

Exclude
The filter returns *false* if the name of the current model matches the specified value.

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Assembly	X	
Drawing	X	

5.2.20 Model - Is regenerateable

Model - Is regenerateable

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Regenerates the model and checks whether regeneration was successful.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if regeneration was successful.

Exclude
The filter returns *false* if regeneration was successful.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.21 Model - Is modified

Model - Is modified

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Checks whether the model is changed inside the session.

For a description of the basic filter functions, please refer to Model filters 229.

© 2023 INNEO Solutions GmbH

Include
The filter returns *true* if the model is changed / modified.

Exclude
The filter returns *false* if the model is changed / modified.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.22 Model - Layer

Checks whether the specified layer exists or contains a defined number of elements.

This filter can only be applied to parts and assemblies and always returns *false* for drawings.

Hidden elements is not a layer. It can therefore not be influenced with filters or functions.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if a layer matches the filter criteria.

Exclude
The filter returns *false* if a layer matches the filter criteria.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.23 Model - Model subtype

Checks whether the current model matches the selected subtype.

This filter can only be applied to parts and assemblies and always returns *false* for drawings.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model has the selected subtype.

Exclude
The filter returns *false* if the current model has the selected subtype.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.24 Model - PDM object

Model - PDM object

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Model: ☐ never saved

☐ from HDD

☐ saved in Workspace, but not from Commonspace

☐ from Commonspace : Checked Out

☐ from Commonspace : Not Checked Out

Checks whether the current model is a PDM object and whether it has been checked out.

For a description of the basic filter functions, please refer to Model filters 229.

Include
If the option *is PDM object* has been selected and the current model is a PDM object, the filter returns *true*.

If the option *check if it's checked out* has been selected, the filter only returns *true* if the current model is a PDM object and has the selected check-out state.

Exclude
If the option *is PDM object* has not been selected and the current model is a PDM object, the filter returns *true*.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.25 Model - Parameter

Model - Parameter

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Parameter:

Value: *

Numeral comparison: ○ <

○ <=

◉ =

○ >=

○ >

Epsilon: 0

You can use Reg exp in value if value is string, mind the /.

You can use Reg exp in parameter name if the parameter not start with 'PRO_'.

Checks whether the specified parameters have the specified value.

If the parameter name begins with *PRO_*, you cannot use RegExp under *Parameter*.

If the parameter to be compared is a string parameter, you can use RegExp under *Value*. If the parameter is an integer or double parameter, only the numerical value will be compared. For numerical comparison, you can specify an *Epsilon* (allowable error, +/- range) and a logical operator.

Boolean Parameter-Werte can be defined as positiv/JA with: true, 1, yes, ja, j und y. All other values would be detected as negativ/NEIN.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
If all parameters whose name matches the pattern under *Parameter* have a value that matches the pattern under *Value*, the filter returns *true*.

Exclude
If all parameters whose name matches the pattern under *Parameter* have a value that matches the pattern under *Value*, the filter returns *false*.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.26 Model - Parameter type

Checks whether the specified parameters have the specified type.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
If the parameter whose name matches the pattern under *Name* have a type that matches the chosen *Type*, the filter returns *true*.

Exclude
If the parameter whose name matches the pattern under *Name* have a type that matches the chosen *Type*, the filter returns *false*.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.2.27 Model - Path

Model - Path

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter:

You can use Reg exp in path, mind the /.

Checks whether the current model is in the specified path.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the path of the model matches the specified pattern.

Exclude
The filter returns *false* if the path of the model matches the specified pattern.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.28 Part - Current material name

Part - Current material name

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter: /.*/

You can use Reg exp in filter, mind the /.

If no current material : ☐ include (If filter is true)

© 2023 INNEO Solutions GmbH

Reads the current material from a model and compares it to the specified name.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the name of the current material of the model is the one specified in the *Filter* text box.

You can additionally specify the value returned by the filter when a current material does not exist in the model.

Exclude
The filter returns *true* if the name of the current material of the model is not the one specified in the *Filter* text box.

The evaluation of *If no current material* is negated.

Prerequisites

Part	X	
Assembly	-	
Drawing	-	

5.2.29 Part - Is Skeleton

Checks whether the current model is a skeleton model.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the current model is a skeleton model.

© 2023 INNEO Solutions GmbH

Exclude
The filter returns *false* if the current model is a skeleton model.

Prerequisites

Part	X	
Assembly	-	
Drawing	-	

5.2.30 Part - Material parameter

Material parameter (from current material)

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Parameter:

Value: *

You can use Reg exp in value if value is string, mind the /.

Checks whether the specified material parameter of the current material has the specified value.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the material parameter exists with the specified value.

Exclude
The filter returns *false* if the material parameter exists with the specified value.

Prerequisites

Part	X	
Assembly	-	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.2.31 Solid - Accuracy

Checks whether the accuracy of the current model matches the specified criteria.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the accuracy matches the specified criteria.

Exclude
The filter returns *false* if the accuracy matches the specified criteria.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.32 Solid - Combined State

Solid - Combined State

Description:

Filter type: ◉ include

○ exclude

Filter: /.*/

You can use Reg exp in filter, mind the /.

Checks whether the specified combined state exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the combined state exists.

Exclude
The filter returns *true* if the combined state does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.33 Solid - Contains circular dependencies

Solid - Contains circular dependencies

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Checks whether the current model have circular dependencies.

For a description of the basic filter functions, please refer to Model filters 229.

© 2023 INNEO Solutions GmbH

Include
If the current model have circular references, the filter returns *true*.

Exclude
If the current model have circular references, the filter returns *false*.

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.34 Solid - Cross Section exists

Checks whether the specified cross-section exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the cross-section exists.

Exclude
The filter returns *true* if the cross-section does not exist.

Prerequisites

Part	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.2.35 Solid - Feature exists

Checks whether the specified feature exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the feature exists.

Exclude
The filter returns *true* if the feature does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.36 Solid - Feature from type

Checks whether the current model has a feature with the specified properties.

You can use RegExp 323 in the *Name* field.

CAUTION: Only named features can be filtered by name.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model has a feature with the specified properties.

Exclude
The filter returns *false* if the current model has a feature with the specified properties.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.37 Solid - Feature is flexible

Solid - Feature is flexible

Description:

Filter type: ◉ include

○ exclude

Name: *

You can use Reg exp in path, mind the /.

Checks whether the current model has a flexible feature with the specified name.

You can use RegExp 323 in the *Name* field.

CAUTION: Only named features can be filtered by name.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model has a flexible feature with the specified name.

Exclude
The filter returns *false* if the current model has a flexible feature with the specified name.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.38 Solid - Feature suppressed

Solid - Feature suppressed

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Name: *

You can use Reg exp in path, mind the /.

Checks whether the current model has a suppressed feature with the specified name.

© 2023 INNEO Solutions GmbH

You can use RegExp 323 in the *Name* field.

CAUTION: Only named features can be filtered by name.

You can enter * to search for all suppressed features.

CAUTION: If you are in insert mode, the features suppressed by the insert mode are also recognized as suppressed.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model has a suppressed feature with the specified name.

Exclude
The filter returns *false* if the current model has a suppressed feature with the specified name.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.39 Solid - Geometry check exists

Checks whether the button *Geometry check* is active in the user interface.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the geometry of a feature contains errors which are detected by the geometry check.

© 2023 INNEO Solutions GmbH

Exclude
The filter returns *false* if the geometry of a feature contains errors which are detected by the geometry check.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.40 Solid - Instance with name

Checks whether at least one instance with the specified name exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if at least one instance with the specified name exists.

Exclude
The filter returns *false* if at least one instance with the specified name exists.

Prerequisites

Part	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Drawing	-	

5.2.41 Solid - Is generic

Solid - Is generic

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Checks whether the current model is a generic.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model is a generic.

Exclude
The filter returns *false* if the current model is a generic.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.42 Solid - Is insert mode active

Solid - Is insert mode active

Description:

Filter type: ◉ include

○ exclude

Checks whether the insert mode is active.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the insert mode is active.

© 2023 INNEO Solutions GmbH

Exclude
The filter returns *false* if the insert mode is active.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.43 Solid - Is instance

Checks whether the current model is an instance.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the current model is an instance.

Exclude
The filter returns *false* if the current model is an instance.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.44 Solid - Layer state exists

Solid - Layer state exists

Description:

Filter type: ◉ include

○ exclude

Filter:

You can use Reg exp, mind the /.

Checks whether the specified layer state exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the layer state exists.

Exclude
The filter returns *true* if the layer state does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.45 Solid - Relation

Solid - Relation

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter in: ☑ Relations

☐ Post Regeneration Relations

Search string:

You can use Reg exp in parameter and value, mind the /.

Checks whether the specified search string occurs in the relations or post-regeneration relations of the current model.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the specified search string occurs in the relations or post-regeneration relations of the current model.

Exclude
The filter returns *false* if the specified search string occurs in the relations or post-regeneration relations of the current model.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.46 Solid - Simplified representation exists

Checks whether the specified simplified representation exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the simplified representation exists.

Exclude
The filter returns *true* if the simplified representation does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.47 Solid - Tolerance

Checks whether the tolerance of the current model matches the specified standard.

For a description of the basic filter functions, please refer to Model filters 229.

© 2023 INNEO Solutions GmbH

Include
The filter returns *true* if the tolerance of the current model matches the specified standard.

Exclude
The filter returns *false* if the tolerance of the current model matches the specified standard.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.2.48 Solid - Unit system

Checks whether the specified unit system is used.

For a description of the basic filter functions, please refer to Model filters 229.

Include
The filter returns *true* if the specified unit system is used.

Exclude
The filter returns *false* if the specified unit system is used.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Assembly	X	
Drawing	X	

5.2.49 Solid - Unregenerated instances

Checks whether the current model contains unregenerated instances.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the current model contains unregenerated instances.

Exclude
The filter returns *false* if the current model contains unregenerated instances.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.2.50 Solid - View exists

Solid - View exists

Description:

Filter type: ◉ include (If filter is true)

○ exclude (If filter is false)

Filter:

You can use Reg exp, mind the /.

Checks whether the specified view exists.

For a description of the basic filter functions, please refer to Model filters 229.

For information on how to use regular expressions and variables, please refer to RegExp 323 and Variables 306.

Include
The filter returns *true* if the view exists.

Exclude
The filter returns *true* if the view does not exist.

Prerequisites

Part	X	
Assembly	X	
Drawing	-	

5.3 Reports

Report types

There are three different report types. For each report type, there are own actions and reasons of using exactly this one. The following pages show you the individual types, make you familiar with their advantages and disadvantages, and tell you how you can employ the individual reports to obtain the required result.

Types:

– Informative PDF 301

© 2023 INNEO Solutions GmbH

– CSV report 272

– List report 297

Informative PDF

The simplest report type is the informative PDF. The PDF lists a lot of information about an individual model. The resulting PDF can be configured via the action options. It contains all configurable properties and model configurations.

CSV/XLS report

The action Open CSV report 272 are a processing action for all actions that start with a report. This means that each action that starts with a report requires an open report it can write to.

Each report has an unambiguous name that links the report with the individual actions that write to it. It must be selected in all actions that write to this report. If there is only one report in the action list, it is selected automatically.

Report actions

There are various report actions that can be used with the CSV report.

The Universal report 295 permits many records to be implemented that are not available as finished report actions. For example, a report that says whether geometry checks exist, or whether certain rules were followed to fill a parameter contents. More detailed information can be found on the help pages of the reports.

Before/after report technique

In a before/after report, the report action at the beginning of the action list and the report action at the end of the action list are the same. The same report shows the state of the model before rework and the state after rework. These states are shown on the left and on the right side of the report, respectively.

Since this document contains both values, it may also be used as modification proof for the reworked parts.

List report

Like the *Open CSV report* action, the Create list report 297 action creates a report that can be filled with various data. The difference between the two report actions is in the type of the listing. While a CSV report uses only one line per model, a list report permits as many lines as necessary to be created for each model. This permits lists to be sorted and filtered. A list report can, for example, be used for easily identifying the parameters that are used in a group of models.

The list report should be used in the model area, and use the *Append* setting under *If exists*.

© 2023 INNEO Solutions GmbH

Prior to execution, manually delete the file the list report shall be written to. Failure to do so will cause the list report to be appended to the existing file. In most cases, this would provide data that cannot properly be evaluated.

Summary

There are three different report types.

The *Informative PDF* is a readable and printable report form for changes to the current model, for handing over with a model, or for providing a brief overview of a small number of models.

Due to the larger data volumes, CSV report and list report are better suited for *All in Session*, *All in Workspace* and batch mode. While the CSV/XLS report provides an overview of a list of models, the list report provides an overview of all employed parameters, for example.

5.3.1 Open CSV report

The action generates or extends a CSV report (with specified storage location and name) to which all other report actions can then write.

© 2023 INNEO Solutions GmbH

General description

Open CSV report

Description:

Name: 1 96 CSV-Report

File: 2

@project_dir@ - project directory

@date@ - current date

Line break: 3 (•) Windows ( ) Unix

Charset: 4 ( ) UTF-8 (•) ISO-8859-2 (Excel)

If exists: 5 (•) overwrite ( ) append

Seperator: 6 (•) Semicolon ';' ( ) Comma ','

Options: 7 [x] write filename first

8 [x] replace dot through comma if it's between two numbers

9 [ ] Open CSV after creation

The action ends and close file after end of foot is reached

1. Unique report name, which is used later to assign the report to actions that write into the report
2. Path to the existing or to-be-created CSV file. The variables *@project_dir@* and *@date@* may be used in the name.
3. Specifies the type of line break (*\r\n* or *\n*) that is used for writing the file
4. Character encoding. ISO-8859-2 is recommended for Excel. OpenOffice also supports UTF-8.
5. Specifies how to deal with an existing report file of the same name
6. Specification of the delimiter character that is to be used
7. If this checkbox is selected, the *Filename* is automatically added as the first line in the CSV report.
8. If this checkbox is selected, the decimal point in all numbers is replaced with a comma. This is only done if there is a number before and after the point.

© 2023 INNEO Solutions GmbH

9. After all models have been processed (end of the Model Processor run), the created CSV file is opened.

The *Open CSV report* action is a preparatory action for all report actions. This means that each report action requires an open report it can write to. Each report has a unique name (1) that links the report with the individual actions that write to it. The related file name with the suffix CSV is entered in the text box (2).

The resulting file can later be processed with any program that supports text-based tables. Line breaks (3) and character sets (4) of the processing programs can be different. They can be selected when configuring the report. The default settings (Windows, ISO-8859-2) are made for further processing in Excel under Windows.

Under (5) you can choose whether an existing file is overwritten or whether the captured data should be appended to the existing file. Appending is only expedient when the file content corresponds to the same context, information, and sequence.

Option (6) enables you to select whether the delimiter is a semicolon or a comma. In Germany, a semicolon is the typical delimiter character. Everywhere else, a comma is the typical delimiter character for CSV files.

If (7) is selected, a *Filename* column is inserted into the report as the first column.

Activate option (8) if you want to replace every point that is enclosed by two numbers with a comma.

Activate option (9) if you want to open the CSV file automatically in the default viewer after the entire Model Processor run has been completed.

Since the *Open CSV report* action is in most cases only expedient in the header area, creating this action is only permitted in the header area. Moving it to other areas is possible, however.

The resulting report will always have an English header line. This is the only way to make the reports compatible between all languages. In this way, they may also be used for writing back parameters, for example.

CAUTION! When you work with a checklist, you must bear in mind that the file is always rewritten if you set (5) to *Overwrite*. Old results can easily be lost.

@project_dir@ is located at: *Project directory/reports*

Information about variable entries can be found at Variables 306.

5.3.2 Report dependencies

The report action writes the models on which a model is dependent to an open report.

© 2023 INNEO Solutions GmbH

General description

1. Selection of the report
2. A table column with all files the model is dependent on
3. A table column with all files the model is dependent on (including path)
4. A table column with all paths that contain models the model is dependent on
5. A table column with all layouts the model is dependent on
6. A table column with all dependencies that could not be found
7. A table column with all suppressed features
8. A table column with all inactive features (inactive features are features that are not displayed in the model due to other reasons than being suppressed)
9. A table column with all features that can not be regenerated (they are corrupted / can correspond to missing parts).

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.3 Report drawing data

The report action writes information about a drawing (see also Report model data 288) to an open report.

© 2023 INNEO Solutions GmbH

General description

Report drawing data

Description:

Model filter:

Report: 1

Write: All

2 Model name

3 Common name

4 File name

5 Path

6 Size

7 Date

8 Is modified

Drawings

9 Actual solid

10 Set title column to 'File name'

11 All models

12 All models not displayed as view

13 Number of sheets

14 per sheet size

15 Number of shown dimensions

16 Number of views

17 Number of symbols

18 View - line display style

19 Filter: default

20 View outside the sheet

21 Dimension outside the sheet (dim origin)

22 Used symbols

23 Detail notes

1. Selection of the report
2. A table column with the name of the current model
3. A table column with the contents of the *PTC_COMMON_NAME* parameter

© 2023 INNEO Solutions GmbH

4. A table column with the file name
5. A table column with the file path
6. A table column of the file size. It can not be read from a PDM system, and only after the model has been saved once.
7. A table column of the file date. It can not be read from a PDM system, and only after the model has been saved once.
8. A table column with the *is changed* flag from Creo
9. A table column (if drawing) with the current volume geometry of the drawing
10. Changing the column name of the column created in selection (9) to *File name*. This makes is significantly easier to reuse the report.
11. A table column (if drawing) with a list of all models that are in the drawing
12. A table column (if drawing) with a list of all models that are integrated in the drawing but are not shown in a view. Options of these models can still be shown. Only views are checked here.
13. The count of all sheets in the drawing
14. The count (12) is sorted by sheet sizes. Each size get a single column.
15. The count of shown dimensions in the sheets of the drawing
16. The count of views on the sheets of the drawing
17. The count of symbols on the sheets of the drawing
18. A table column (if drawing) with the number of sheets of each drawing
19. A table column (if drawing) with the line style of each view in the drawing; separated by line break in cell
20. Filter that outputs only views with the selected line style
21. A table column (if drawing) with all dimensions which dimension origin is located outside of the drawing
22. A table column (if drawing) with the names of the symbols shown in the drawing
23. A table column (if drawing) with the contents of the notes used in the drawing

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

All options under *Drawings* are only executed if the model is a DRW file. The actions will not be carried out and the related table columns remain empty if the files are not compatible with the type.

Information about reports in general can be found at Open CSV report 272.

© 2023 INNEO Solutions GmbH

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.4 Report table cell from drawing

The report action writes the content of specified table cells to an open report.

© 2023 INNEO Solutions GmbH

General description

Report drawing table cell

Description:

Model filter:

Report: 1

Cells to report: 2

Table marking	Cell to read	Comment
3		

4

5 Read only from first matching table

6 With sheet segment

7 Write parameters as text

8 replace all " through ' '

9 Remove all returns

You can use as table marking:

row::column::content

x!!y - table position on drawing

x!!y!! +/- range - table position on drawing

Cell to read:

row:column

row:column:line

The count of row and column begins at 1

The count of line begins at 0

You can use Reg exp in cell content

1. Selection of the report
2. Buttons to read table marking data from Creo
3. Table with data on which drawings tables and which cells from these tables should be read

© 2023 INNEO Solutions GmbH

4. Select the cell to be read from the selected table
5. Selects whether only the first matching table or all matching tables should be used
6. Selects whether the page number should precede the output to identify the table
7. Selection whether entered parameters should be read as variable specification or as text
8. Selection, automatic replacement of quotation marks with double apostrophe
9. Selection, delete all line breaks in the selected values

A report action works only with an existing report (1) that the action can write to.

To use the report action to write table cells into a CSV file, you must specify table markings and cells to be read in the table under (3). You can use different ways to identify the table:

– *row::column::content* - enter the row, column, and contents of the cell (RegExp 323)

– *x!!y* - specify the pixels in x and y direction.

– *x!!y!!range* - specify the pixels in x and y direction. You can additionally specify a deviation range within which the table is still recognized.

Cell contents that should be read:

– *row:column* - enter the row and column

– *row:column:line* - enter the row and column. Only the column (line) of the text content is read and output.

To simplify the selection, especially in nested layouts, the buttons at (2,4) read data from a selection you make in Creo. They permit cells to be selected directly in Creo. The selected values are then written to the table (3).

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	-	
Part - instance	-	
Assembly	-	
Assembly - instance	-	
Drawing	X	

© 2023 INNEO Solutions GmbH

5.3.5 Report family table

The report action writes information about a family table and its content to an open report.

General description

1. Selection of the report
2. A table column with the number of instances that exist in the family table of a model
3. A table column with the number of locked instances
4. A table column with the number of instances that have not yet been regenerated / verifie
5. A table column filled with the names of the first x instances. x can be specified in the text box. The individual names are separated with a line break.
6. A table column that specifies whether the family table is multi-level (multi-level: true / single-level: false)

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.6 Report feature positions

The report action writes the position of coodinate systems and datum planes into a report cell.

General description

1. Selection of the report
2. Filter for a feature name
3. Filter for a feature type
4. If only first level, or all component levels should be visited.

All positions would be written in context to the standard coordinate system of the current model.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	

© 2023 INNEO Solutions GmbH

Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.7 Report feature parameters

The report action writes values of feature parameters to an open report.

General description

1. Selection of the report
2. Specification of parameters to report on. You can use RegExp 323 to specify the parameters.
3. Name of a parameter to filter the features by
4. Value of the filter parameter. Only features that have the specified parameter value will be considered.
5. Separator character for outputting the feature parameters

A report action works only with an existing report (1) that the action can write to.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.8 Report geometry properties

The report action writes geometry properties for the selected model to an open report.

General description

1. Selection of the report
2. Decimal places as output specification for the following options
3. Table column with the mass
4. Table column with the volume
5. Table column with the position of the center of gravity
6. Table column with the bounding box dimensions
7. Table column with the geometry surface

© 2023 INNEO Solutions GmbH

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

Enter the number of decimal places to which you want to round under (2). Due to internal errors when calculating floating point numbers, they should not be used without rounding. Unrounded values often have errors in the 10^-15th place.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.9 Report layers

The report action writes information about layers to an open report.

General description

1. Selection of the report
2. Specifies whether the layer status should also be output

© 2023 INNEO Solutions GmbH

3. The layer information's that should be written per model

A report action works only with an existing report (1) that the action can write to.

A table column is created that contains the names of the existing layers. When option (2) is selected, the status of the layer can additionally be output after the name, separated by a colon. The names of the individual layers are separated by line breaks.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.10 Report material

The report action writes information about materials to an open report.

General description

1. Selection of the report
2. A table column with the material that is currently assigned to the model
3. A table column with the names of all materials in the model, separated by line breaks

A report action works only with an existing report (1) that the action can write to.

© 2023 INNEO Solutions GmbH

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	-	
Assembly - instance	-	
Drawing	-	

5.3.11 Report model data

The report action writes information about a model (see also Report drawing data 276) to an open report.

© 2023 INNEO Solutions GmbH

General description

Report model data

Description:

Model filter:

Report: 1

Write: All

2 Model name

3 Model name without generic

4 Common name

5 File version

6 File name

7 Path

8 Size

9 Date

10 Is modified

Solids

11 Accuracy

12 Tolerance

13 Unit system

14 Number of features

15 Flexible features

16 First # feature-types / names 7

17 Cross Sections

18 Pattern

19 Generic name (if model is an instance)

20 simplified representations

21 Regeneration feedback

Parts

22 Used density

1. Selection of the report
2. A table column with the name of the current model (if it is an instance: *instance name<generic name>*)

© 2023 INNEO Solutions GmbH

3. A table column with the name of the current model (if it is an instance, only the instance name)
4. A table column with the contents of the *PTC_COMMON_NAME* parameter
5. A table column with the file version
6. A table column with the file name
7. A table column with the file path
8. A table column with the file size. It can not be read from a PDM system, and only after the model has been saved once.
9. A table column with the file date. It can not be read from a PDM system, and only after the model has been saved once.
10. A table column with the *is changed* flag from Creo.
11. A table column (if solid) with the accuracy
12. A table column (if solid) with the tolerance
13. A table column (if solid) with the unit system
14. A table column (if solid) with the number of features
15. A table column (if solid) with the first feature names (number selectable via text box)
16. A table column (if solid) with a list of flexible features
17. A table column (if solid) with a list of all sections (name / type)
18. A table column (if solid) with a list of all patterns (name / type)
19. A table column (if solid) with the name of the generic if the model is an instance, otherwise blank
20. A table column (if solid) with a list of simplified representations
21. A table column (if solid) with the regeneration status
22. A table column (if part) with the density used

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

All options under *Solids* are only executed if the model is a ASM or PRT file. Options for parts only apply to PRT files. The actions will not be carried out and the related table columns remain empty if the files are not compatible with the type.

CAUTION! File size and file date can only be read if Creo is not connected to a PDM server.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.12 Report parameters

The report action writes information on and values of model parameters to an open report.

General description

1. Selection of the report
2. Specification of parameters to report on. You can use RegExp 323 to specify the parameters. The other options define which information should be output for each of the

© 2023 INNEO Solutions GmbH

parameters selected here. Note that the number of parameters, multiplied by the number of selected options, corresponds to the number of table columns in the report.

3. A table column with the parameter value
4. A table column with the parameter type (string, integer, double, ...)
5. A table column with the *is designated* flag
6. A table column with the parameter description
7. A table column with the specification whether or not the parameter is relation-controlled
8. A table column with the specification whether or not the parameter is type-restricted
9. A table column with the unit of the parameter

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.13 Report Pro/PROGRAM

The report action writes information that is read from Pro/PROGRAM to an open report .

© 2023 INNEO Solutions GmbH

General description

1. Selection of the report
2. Selection of the lines that are to be displayed
3. The block of lines that is to be output, specified by a start line (3) and an end line (4)
4. Specification of the end line (4)

A report action works only with an existing report (1) that the action can write to.

Depending on the input, the report can create different table columns.

If you enter something under (2), a column is created that outputs lines selected by the filter (* outputs all).

If you enter something under (3), a column is created that contains the Pro/PROGRAM code from the defined line to the end.

If you enter something under (3) and under (4), the text block between the specified lines is output.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	

© 2023 INNEO Solutions GmbH

Assembly - instance	X	
Drawing	-	

5.3.14 Report relations

The report action writes information about relations / post-regeneration relations of a model to an open report .

General description

1. Selection of the report
2. A table column with the relations, filtered by text box (6)
3. A table column with the post-regeneration relations, filtered by text box (6)
4. A table column that specifies whether or not errors were found in the relations
5. A table column that specifies whether or not errors were found in the post-regeneration relations
6. A filter which restricts the listed relations

A report action works only with an existing report (1) that the action can write to.

For each of the selected options, a table column is created in the report that is filled with the corresponding information for the individual models.

The line filter (6) filters the output of the options *Relations* (2) and *Post-regeneration relations* (3). * stands for output all lines. RegExp 323 may be used in the filter.

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filter 229.

© 2023 INNEO Solutions GmbH

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.15 Report universal

The report action writes various evaluations of filters to an open report. This way, an individual report line can be declared.

General description

1. Selection of the report
2. Input of the column header
3. Definition of the conditions under which the lines for true or false should be written into the report
4. Output for (3) true
5. Output for (3) false

The *Report universal* action adds a user-defined column to a report. This column has a title (2) and can generate *true* and *false* values via a filter (3). Depending on whether the filter returns *true* or *false*, the corresponding output (4 or 5) is written into the report.

© 2023 INNEO Solutions GmbH

Information about reports in general can be found at Open CSV report 272.

Information about description and model filters can be found at Model filters 229.

Example

Identify all parts that contain *DIN* in their file name, as well as the string *plasma* in the value of the parameter *TEMPERING*.

This is the filter used:

Prerequisites

© 2023 INNEO Solutions GmbH

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	X	

5.3.16 Write a list report

The action generates a report on numerous information from models. Each information item uses one row.

© 2023 INNEO Solutions GmbH

General description

© 2023 INNEO Solutions GmbH

Write list report

Description:

File: 1

@project_dir@ - project directory

@date@ - current date

Line break: 2
- (●) Windows
- ( ) Unix

Charset: 3
- ( ) UTF-8
- (●) ISO-8859-2 (Excel)

If exists: 4
- ( ) overwrite
- (●) append

Separator: 5
- (●) Semicolon ';'
- ( ) Comma ','

Options: 6 [x] replace dot through comma if it's between two numbers

Write : All

7 (●) Parameter

Solid

8 ( ) Instance names

with parameter (comma seperated):

[ ] regenerate Instance to read parameters

Filetypes:

- [x] Generics
- [x] Instances
- [x] Models without familytable

9 ( ) Layers

10 ( ) Dimensions

[x] only named

11 ( ) Dependencies (only first dependency level)

Drawing

12 ( ) Sheet formats / sizes

13 ( ) Annotations / Notes

[x] Text formats

14 ( ) All models

15 ( ) All shown model dimensions

© 2023 INNEO Solutions GmbH

1. Path to the existing or to-be-created CSV file. The variables *@project_dir@* and *@date@* may be used in the name.
2. Specifies the type of line break (*\r\n* or *\n*) that is used for writing the file
3. Character encoding. ISO-8859-2 is recommended for Excel. OpenOffice also supports UTF-8.
4. Specifies how to deal with an existing report file of the same name
5. Specification of the delimiter character that is to be used
6. If this checkbox is selected, the decimal point in all numbers is replaced with a comma. This is only done if there is a number before and after the point.
7. Write all existing parameter names and parameter properties
8. List of all existing instances (with parameters, if required)
9. (Only solid geometry) List all layers including the layer status (if selected)
10. (Only solid geometry) List all dimensions (if selected, only named dimensions)
11. List all dependencies (only first level)
12. (Only drawings) List sheets of drawings and their size
13. (Only drawings) Write all comments, notes, and symbols from drawings
14. (Only drawings) Write all models from drawings in a list
15. (Only drawings) List all displayed model dimensions

The *Write list all report* action, like the Open CSV report 272 action, creates a report that can be filled with different data items. The difference between the two report actions is in the type of listing. While a CSV report uses only one line per model, a list report permits multiple lines to be created for each model. This permits lists to be sorted and filtered.

The list report should be used in the body area of the action list, and use the *Append* setting at selection (4). The *Append* selection is important. Without this selection, each newly edited model overwrites the report.

With many family tables (in particular in the libraries), a report over all instance names can be very large. Processing such a report in Excel can take a lot of time.

Different listing reports should never write to the same file.

The following list reports can be created:

– Parameter

– Instance names

– Sheet drawing frame / sizes

– Annotations / comments

– All models

© 2023 INNEO Solutions GmbH

@project_dir@ is located at: *Project directory/reports*

Information about description and model filters can be found at Model filter 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

5.3.17 Write informative PDF

The action generates a PDF report based on ONE model.

This is the only report that can also be used in the MPU.

© 2023 INNEO Solutions GmbH

General description

© 2023 INNEO Solutions GmbH

Write informative PDF

Description:

Model filter:

Folder:

@project_dir@ - project directory

Folder from model

Export Name: @mdln@

@date@ - current date

@mdln@ - model name, until "."

@number@ - auto incr. number

Overwrite existing

Open PDF after creation

Info box:

Units

Accuracy

Image

Custom text

Materials:

Current

All in model

Parameter:

Parameter

Activate filter

(comma sperated): MC_ERRORS,BEN*

short version

Feature-Parameter

Activate filter

(comma sperated): MC_ERRORS,BEN*

Relations:

Relations

Post Regeneration Relations

Feature-Relations (slow)

© 2023 INNEO Solutions GmbH

Family table: ☑ Number of variants

☑ Names of all variants

More: ☑ Views

☑ Layers

☑ First # features 7

☑ First # 3D-annotations 3

☑ Dependencies

☑ Simplified representations

☑ Interfaces

Use other font(Path):

1. Input box for the path. It must exist on the hard disk, and can be selected via the browse icon.
2. When this checkbox is selected, box (1) is deactivated, and the folder of the model is used (not possible in conjunction with PDM).
3. Name used for saving the PDF
4. Specifies whether or not the PDF shall be overwritten if a PDF of the specified name exists
5. Do you want to open the PDF after it has been created?
6. Output of the units of the unit system
7. Output of the accuracy
8. Output of an image of the model (this requires the graphical representation of the model)
9. Specifies whether the text from the text box (10) shall be written to the PDF
10. Input box for a user-defined text that is written (9) to the PDF
11. Output of the active material
12. Output of all materials in the model
13. Output of all parameters in the model
14. Filtering all parameters, so that only the filtered parameters are output
15. Shorter output format for parameters
16. Output of all parameters that are linked to features, together with feature IDs and feature names. These parameters can be filtered too.
17. Filtering all feature parameters, so that only the filtered parameters are output
18. Output of the relations
19. Output of the post-regeneration relations

© 2023 INNEO Solutions GmbH

20. Output of all relations that are linked to features, together with feature IDs and feature names
21. Output of the number of variants in the family table of the part (provided it has one)
22. Output of the names of all instances of the generic (provided the current model is a generic)
23. Output of all views in the model
24. Output of all layers in the model
25. Output of names and IDs of the first x features
26. Output of names and IDs of the first x 3D comments
27. Output of the dependencies of the model
28. Output of all simplified representations in the model
29. Output of all interfaces of the model
30. Changes the PDF font to the specified one

The *Write informative PDF* action creates a PDF that permits the selected information to be written. Please refer to the list for the possible utilization.

A separate PDF is created for each processed model. When you overhaul several models, deactivating option (5) for opening the report is recommended.

In the text box (28) for a font change, you can specify a font in TTF format. The selected font is used. CAUTION! Writing the PDF does not support UTF-8 characters.

@project_dir@ is located at: *Project folder/reports*

Write access to a folder *c:\temp* is required for creating images.

Information about description and model filters can be found at Model filters 229.

Prerequisites

Part	X	
Part - instance	X	
Assembly	X	
Assembly - instance	X	
Drawing	-	

© 2023 INNEO Solutions GmbH

5.4 Variables

General description

Using variables permits automated value adjustment in many actions of the Model Processor. A value with an entered variable automatically adjusts to the environment.

You can click on the variable entries. The variable values are then automatically entered into the text box provided.

Not every action can support every variable. Please use common sense as to whether a variable can be used. (When analyzing parameters, *@project_dir@* does not make sense, for example.) Unsupported variables remain as they are at the end.

@date@
Output of the current date in formatting / usual for data for easier sorting (yyyy-mm-dd).

Example:

2011-11-12

@datede@
Output of the current date in German formatting / usual for data for easier sorting (dd.mm.yyyy).

Example:

12.11.2011

@dateen@
Output of the current date in English formatting (dd-mm-yyyy)

Example:

12-11-2011

@dateus@
Output of the current date in American formatting (mm-dd-yyyy)

Example:

11-12-2011

@dateusshort@
Output of the current date in American formatting, short notation (mm-dd-yy)

Example:

11-12-11

© 2023 INNEO Solutions GmbH

@time@
Output of the current time in a notation that can also be used for file names. Short notation (HH(0-23)-MM)

Example:

14-15

@time_with_seconds@
Output of the current time in a notation that can also be used for file names. Short notation (HH(0-23)-MM-SS)

Example:

14-15-59

@timede@
Output of the current date in German formatting, short notation (HH(0-23):MM)

Example:

14:15

@timeen@
Output of the current time in America formatting, short notation (hh(1-12):MM a)

Example:

02:15 AM

@curworkdir@
Output of the current working directory

Example:

c:\temp

@mdln@
Output of the current model name

@mdltype@

Output of the current model type (File extension)

@mdlpath@
Output of the current directory of the current model

@selmdln@
Output of the model name of the first selected model in an assembly

@selmdlpath@
Output of the directory of the first selected model in an assembly

© 2023 INNEO Solutions GmbH

@mdlpathr@
Output the current directory of the current model with reverse backslash ("/" instead of "\")

@genname@
Output of the generic name of the current instance. If the currently processed model is not an instance, the output is '-'.

@fileversion@
Output of the current version of the file WITH preceding '.' (number after the file name extension, for example: test.prt.2)

@mdlno@
Output of the current version of the file WITHOUT preceding '.' (number after the file name extension, for example: test.prt.2)

@number@
Output of a consecutively increasing number. The number is incremented upon each call of "@number@". The function does not distinguish between actions. The number is thus also incremented when "@number@" is used in different actions.

@para@
Output of the value of the current parameter (only in "Edit model parameter")

@line@
Output of the value of the current parameter (only in "Edit relations")

@feat@
Output of the name of the current feature (only in "Rename feature" and "Add feature parameter")

@creoname:FILTER@

Retreaves a new file name with Creo from the current source for file names. You can use the different filters prt, asm or drw for it.

Example: @creoname:prt@

@wtservername@

Output of the current server name.

Example: PDM

@wtserverurl@

Output of the current server url.

Example: https://wtpdml120.ratc.de/Windchill

@wtserverws@

Output of the current workspace name.

© 2023 INNEO Solutions GmbH

Only Solids

@regen_status@

Output of the current regeneration status:

- PRO_SOLID_CONNECT_FAILED
- PRO_SOLID_FAILED_REGENERATION
- PRO_SOLID_NEEDS_REGENERATION
- PRO_SOLID_REGENERATED

Only drawings

@pageno@ or @sheetno@
Output of the current page. If the currently processed model is not a drawing, the output is '-'.

@maxpage@
Output of the total number of pages. If the currently processed model is not a drawing, the output is '-'.

@curdrwmdln@
Output of the model name of the current drawing model. If the currently processed model is not a drawing, the output is '-'.

@curdrwmdltype@
Output of the model type of the current drawing model. If the currently processed model is not a drawing, the output is '-'.

@cursheetsize@
Output of the sheet size of the current sheet.

Output: A0, A1, A2, A3, A4, A, B, C, D, E, F, var: x:y, empty

If the currently processed model is not a drawing, the output is '-'.

@cursheetcurmdlscale@
Output of the model scale from the current model based on the current sheet. If the currently processed model is not a drawing, the output is '-'.

Output : 0.5 for 1:2

© 2023 INNEO Solutions GmbH

@cursheetcurmdlscaleISO@
Output of the model scale from the current model based on the current sheet. Try to recalculate it to ISO Norm Values. If the currently processed model is not a drawing, the output is '-'.

Output : 1:2 for 1:2

Denominators from ISO : 2, 3, 5, 10

@cursheetname@
Output of the sheet name based on the current sheet. If the currently processed model is not a drawing, or the current sheet name is not set, the output is '-'.

@cursheetformat@
Output of the sheet format name based on the current sheet. If the currently processed model is not a drawing, or the current sheet has no format set, the output is '-'.

Parameter operations

%PARAM%
Outputs the value of another parameter. Here, the input between the % characters can vary.

Example:

%DESIGNATION%

%length:PARAM%
Output the length of the value of a parameter from the current model. This can also be used in combination with user, curmod and curmat. At this the order %curmat:length:PARAMETER % needs to be observed.

Example:

%length:DESIGNATION%

%curmod:PARAM%
Output of the value of a parameter from the current model of a drawing into a drawing. Here, the replacement of PARAM can vary.

This notation does not make any sense outside of drawings.

Example:

%curmod:DRAWINGNUMBER%

%user:PARAM%
Output of the value of a user variable. Here, the replacement of PARAM can vary.

%conf:PARAM%
Output of the value of a Creo config option. Here, the replacement of PARAM can vary.

© 2023 INNEO Solutions GmbH

%curmat:PARAM%
Output the value of a parameter from the current material of a part. Here, the replacement of PARAM can vary.

The notation will fail at parts without current material or without the material parameter. The extended filter Material parameter 254 can be used to intercept these cases.

Example:

%curmat:SELECT_EN_NAME%

%DubAsE:PARAM%
Output of a double value in e^ notation instead of regular output rounded to 6 decimal places. This can also be used in combination with user, curmod and curmat. At this the order %curmat:dubase:PARAMETER% needs to be observed.

Example:

%dubase:DOUBLE_PARAMETER%

%NoZero:PARAM%
Output of a double value without pending zeros. This can also be used in combination with user, curmod and curmat. At this the order %curmat:dubase:PARAMETER% needs to be observed.

Example:

%nozero:DOUBLE_PARAMETER%

%toupper:PARAM%
Output of the parameter value in upper letters. This can also be used in combination with user.

%toupper:STRING_PARAMETER%

%tolower:PARAM%
Output of the parameter value in lower letters. This can also be used in combination with user.

%tolower:STRING_PARAMETER%

Environment variables

Output of an environment variable of Model Processor/Windows using *§env-var§*.

Example:

§USERNAME§

§COMPUTERNAME§

§HOMEDRIVE§

§LOGONSERVER§

© 2023 INNEO Solutions GmbH

§USERDOMAIN§

String operations

NOTE: An external RegExp can not contain a string operation.

$$sub$x$y$TEXT$$
Cuts from a text (TEXT) an entry that begins at x and ends at y. x and y are numerical values. A text of the specified length must exist. Count starts at 0 for the first character.

Example:

$$sub$2$5$@para@$$

$$pre$x$TEXT$$
Substring from start (character at position 0) to character x.

$$pree$x$TEXT$$
Substring from start (character at position 0) up to x characters before the end of the string.

$$poste$x$TEXT$$
Substring of the length x, measured from the end of the string. Corresponds to the last x characters of the string.

$$posts$x$TEXT$$
Substring up to the end, beginning at character x. Corresponds therefore to a variable number of output characters.

$$repl$A$B$TEXT$$
Replace A with B in TEXT. Using RegExp 323 in input A is possible here.

If a $ character shall be used in A or B, it has to be marked using a subsequent ! Example:

$$repl$.$-$@para@$$
$$repl$/^\w:\\\w+\\\w+\\\w+$!/$$i:\so\soc\traction\active\xxx_t_test\200_tank$$

Folder variables

@project_dir@
Is always replaced with the directory that contains the corresponding data inside the Model Processor project (mpz).

Example:

@project_dir@stahl_c35.mtl corresponds in the Model Processor project to materials\stahl_c35.mtl

@pro_material_dir@
Is replaced with the contents of the configuration option pro_material_dir.

Example:

© 2023 INNEO Solutions GmbH

@pro_material_dir@stahl_c35.mtl

@default_layer_model@
Is replaced with the contents of the configuration option default_layer_model.

Example:

@default_layer_model@

PTC Windchill variables

@workerdir:<PATH>@

If a trigger is used in PTC Creo View Adapter visualization this variable can returns the right subfolder inside the pubtemp / transfer folder. This variable should used favoured with GENIUS TOOLS Model Processor Worker Extension.

Example:

@workerdir:D:\pubtemp\@

@oid@

Returns the current OID of the current reworked model. For this variable a connection to PTC Windchill need to be active and a representation of the object on PTC Windchill site have to exist. This variable should used favoured for ODATA REST API requests.

Example:

@oid@

© 2023 INNEO Solutions GmbH

6 Mode configuration

The next step after selection and setup of the action list 29 is the selection of the data base that shall be used for carrying out the created actions.

4 different data sources / data source groups are available.

1. Current model
2. All in
3. Batch processing
4. Create new model

You can choose any mode as required. With a clear structure (subdivision of the actions into the corresponding areas), each action list can be executed via each database (except *create new model*).

Current model

In the overhaul of the current model, the model is overhauled that is in the Creo window when overhaul is started (pressing the Start button). If the Creo window does not contain a model, the run will be aborted after the execution of the header area of the action list, and completed with the footer area.

© 2023 INNEO Solutions GmbH

All in

In the *All in* group, you must additionally choose between:

– All in session

– All in workspace

Both modes can be restricted via the file type. Thus you can choose that only drawings, assemblies, parts, or any combination thereof is processed.

The selection *All in session* permits all models to be overhauled that are loaded in the Creo session. This includes displayed and undisplayed models. If, in contrast, *All in workspace* is selected (requires a connection to a Windchill PDM system), all data items (after selection) from the PDM system are loaded and processed.

Batch processing

A batch mode run is only possible after it has been configured. The configurator opens when you click on *Configuration*. More details can be found under Batch processing 316.

Additional control elements

The checklist selection is an additional mode configuration element. Using a list, it permits parts that were processed once to be excluded from further processing.

© 2023 INNEO Solutions GmbH

Furthermore, processing instances can also be defined in this area. This setting has only an effect on batch mode and the *All in* modes. The *Current model* mode continues processing instances.

Selecting log report output reduces the speed of processing data records. Activating the log is therefore only recommended for *Current model* and short runs. Overview, warning and error logs are not affected by this setting.

The *License mode* item shows the type of license you are using. If you are using a per-item credit license, the number of files that can still be overhauled is displayed under *Credit*. An exact breakdown of the actions that do or do not require credit items can also be found in the Action overview 30.

6.1 Batch processing configuration

1. Creating or deleting a batch processing file list
2. Importing a list or text file (from PDM system, for example)
3. Exporting the list as an XML rules file

© 2023 INNEO Solutions GmbH

4. Exporting the list as list of file names

5. Searching for duplicated entries as names and / or MD5 hash

6. Settings UI for import data

7. Additional options:

All in session mode: When an assembly is opened, all other data is processed that is subsequently in the session

ASM includes dependencies: The list is enhanced and the data is processed when an assembly is opened that contains data that is not in the list

Process in order: First, process all PRT files, then all ASM files, then all DRW files

Fast open of assemblies: Open assemblies without dependencies

Use defined mapkey for opening files: You can enter the required mapkey here

9. Adding folders from the file system

10. Adding folders from the connected PDM system

11. Adding folders from the current workspace

12. Deleting a selected folder path / a selected file list

13. Selected - recursive inheritance of filters (3,4,5,6)

14. Modifying the filter (parts) for the selected path, including all subpaths if (8) is selected

15. Modifying the filter (assembly) for the selected path, including all subpaths if (8) is selected

16. Modifying the filter (drawing) for the selected path, including all subpaths if (8) is selected

17. Modifying the filter (only the highest version) for the selected path, including all subpaths if (8) is selected

18. Modifying the filter for import data, including all subpaths if (8) is selected

19. Reducing the file list to the entries from X to Y

20. Displaying folders that exist in the file system, but not under the root node

21. Displaying all deactivated files

22. Deactivating all data in the tree whose file name does not comply with the Creo standard

23. Searching all files in the tree (RegExp 323 can be used here)

24. Activating and deactivating entire folders (red strikethrough, are not inherited, can not be used recursively; see also (20))

25. Add to blacklist or remove from blacklist. By default, all selected folders will be added in order to create the batch processing list. Folders or files that you add to the blacklist,

© 2023 INNEO Solutions GmbH

however, will be subtracted. In this way, you can, for example, import a list of files to exclude, then mark them as belonging to the blacklist.

File system

First of all, a new batch processing list (1) is required to start batch processing of some folders from the hard disk. Once this list has been created, you can use the Plus icon (9) to add one or more folders to the list. When you add folders, a UI appears after the selection that adds the selection either

– recursively - including all subfolders of the selected folder, or

– only the selected folder

to the data that is to be processed. The list of data to be processed can now be restricted / stated more precisely with the filters (12 ... 16), or by deactivation of individual files and folders. Deactivation is possible via the right-hand mouse button menu. The inheritance (11) must be observed when the filters are set. When activated, inheritance can pass a filter modification to all subfolders of the selected path.

Import data that is read in Pro/E in conjunction with the overhaul of data must first be admitted in the selected folder via the filter (16). Import data is deactivated in the default setting, and is not processed. If import data is admitted, the type of processing must be selected in the configuration user interface (6).

PDM

If you want to perform batch processing via data in a PDM system, you must first compile a list of the processed data in the PDM system, and export this list into a readable format (Enter-separated TXT file). Remember that you may also use separated files with more information. The file name must then be in the first column. The resulting file can now be imported via the Import icon (2) into batch processing, and then be processed like a normal folder.

You can also use button (10) to import a folder from the PDM system, and button (11) to import the current workspace as file name list.

When you process data in a PDM system, you must frequently check out the data. See also at Project properties 332.

Import data

Creo 1.0 M010 has problems with saving import data in the workspace.

Additional UIs

Check for duplicated files 320

Modifications of the file base 321

© 2023 INNEO Solutions GmbH

Searching in the file list 321

Extended options 319 for batch mode

6.1.1 Extended options

In the extended options, you can define options for process sequence and for opening files.

- "All in session" mode - in Batch mode
- Preload all files
- asm includes dependencies
- process in order (prt, asm, drw)

Please be prepared that a configuration here can interfere with the process / workflow in other modes.

Opening files

The following example demonstrates how to speed up opening files for drawings.

Filter option: DRW

Mapkey:

```
~ Command `ProCmdModelOpen`;
~ Update `file_open` `Inputname` `@filename@.drw`;
~ Command `ProFileSelPushOpen_Rep@context_dlg_open_cmd`;
~ Select `open_rep` `replist` 1 `-2`;
~ Activate `open_rep` `OK`;
```

The following variables can be used within mapkeys:

```
@filename@
@fileext@
@fileversion@
```

The working directory is specified beforehand and does not have to be considered for opening files.

© 2023 INNEO Solutions GmbH

6.1.2 Check for duplicated files

Duplicate files can be found on the UI with their file name (3) and/or hash value (2). The hash values of two files are only identical when the file contents are 100 % identical. The value of a file that was opened and saved in Creo is no longer the same. The file history was changed. Geometrically identical has nothing to do with the hash value.

The batch processing setting is used as the data base for the verification. All located duplicates are shown in the file tree. After selection, you must click on the button (1) to restart evaluation. Once evaluation has been completed, you can click on the export function (4) to write the entire list in a file (that can be transferred or be processed otherwise). Depending on the data volume, exporting can take some time.

When you select a file in the file tree, the files that triggered duplication are shown in the details area. They can be viewed there. Once you have decided on the file you want to keep, you can press the *Disable in list* button (7) to deactivate the other files directly in the list.

The gray, green, and red dots show whether a file name group has already been reduced to a file (green), not yet processed (gray), or processed partly (red).

© 2023 INNEO Solutions GmbH

6.1.3 Searching in the file list

Activate file search to search for files in the data basis. Only folders that were added to batch processing are searched here. Enter the term you are searching for in the text box (1). You may use wild cards and RegExp 323. Press Enter or the search button (2) to start the search.

The search takes file path, file name, and file version into account (unless you search *only latest version*). You can not search for file size or a certain file date.

6.1.4 Modifications of the file base

To find deactivated files in batch processing 316(19), you can use the button that displays the UI shown above. The individual files that are deactivated in the entire data base are shown here. Deactivated folders are not shown here.

© 2023 INNEO Solutions GmbH

To display the UI shown, you must press in batch processing 316(18). All deactivated folders and discrepancies between the selected data basis and the current file situation can be seen on this UI. When you load a project and use an existing data basis, you should check the data base with this UI before you run the project.

© 2023 INNEO Solutions GmbH

7 RegExp - Regular Expressions

General description

All text boxes that are identified with this tag can employ a simplified version or the complete version of the RegExp.

Simplified version

RegExp can be used in many filters and actions.

In the simplified version, only * is used as a wildcard. This sign may be used always and everywhere.

For example:

```
Materials:
Steel 977
Steel_3
St_97
Plastic
```

All steel materials shall be deleted from this material list. You can either specify the materials individually in the *Delete material* action, or delete them with a single line:

```
st*
```

Since the asterisk is only at the end, the material must begin with *st*. The following entry is required to delete only materials that contain *st* and *9*:

```
st*9*
```

As you can see, there is already a lot of freedom here.

Complete RegExp

RegExp can only be used in a line that begins and ends with a / Using RegExp only in parts of an input is not possible.

Automatic case-insensitivity is not active when the complete RegExp is used. It can be restored with (?i).

If we use the wild card example with RegExp:

```
/(?i)st.*/
```

When you start with it, RegExp looks more complicated. However, it offers significantly more options.

The following RegExp is a part of a parameter filter. This filter permits only parameters of the following format:

– 3 letters

© 2023 INNEO Solutions GmbH

– character -

– 4 numbers (digits)

– character -

– 2 numbers

– character -

– 2 numbers

– character -

– a number at the end

```
/\w{3}-\d{4}-\d{2}-\d{2}-\d{1}/
```

Further examples

All except those beginning with 2 digits, an underscore and a letter

```
/^(?!\d{2}_\w).*$/
```

Explanation:

^ - start of input

(?!) - everything except for the following

$ - end of input

.* - all characters in any number

\d - one digit

\d{2} - exactly two digits

\d{1,2} - between 1 and 2 digits (it could also be {1,4} which would mean 1,2,3 or 4)

\w - one letter of the Latin alphabet

All except aba and bab

```
/(?i)^(?!(aba$|bab$)).*$/
```

Explanation:

(?i) - ignores upper / lowercase in the Latin alphabet

^ - end of input

(?!) - everything except for the following

$ - end of input

.* - all characters in any number

With the parentheses, the words aba and bab are negated using (?!). The .* is important because in every RegExp an accepting state always needs to be specified as well, if something is to be accepted.

© 2023 INNEO Solutions GmbH

8 Mapkeys

Create component interface (Creo 2.0)

Filter proposal:

– Feature exists - include - element (e.g. coordinate system)

– Feature exists- exclude - component interface

```
~ Command `ProCmdCompIntrfc` ;
~ Update `Odui_Dlg_00` `t1.set_name_input_panel` `INTF_DEF_CSYS`;
~ FocusOut `Odui_Dlg_00` `t1.set_name_input_panel`;
~ Activate `Odui_Dlg_00` `t1.S.PIM_chk.s0.c0` 1;
~ Trigger `Odui_Dlg_00` `t1.S.C.s0.li.s0.c0` `0`;
~ Trigger `Odui_Dlg_00` `t1.S.C.s0.li.s0.c0` ``;
~ Command `ProCmdMdlTreeSearch` ;
~ Select `selspecdlg0` `SelOptionRadio` 1 `Coord Sys`;
~ Update `selspecdlg0` `ExtRulesLayout.ExtBasicNameLayout.BasicNameList` `DEF_CSYS`;
~ Activate `selspecdlg0` `EvaluateBtn`;
~ Activate `selspecdlg0` `ApplyBtn`;
~ Activate `selspecdlg0` `CancelButton`;
~ Activate `Odui_Dlg_00` `stdbtn_1`;
~ Activate `Odui_Dlg_00` `stdbtn_2`;
~ FocusIn `UI Message Dialog` `yes`;
~ Activate `UI Message Dialog` `yes`;
```

Create axis

Filter proposal:

– Feature exists- include - layer 1

– Feature exists- include - layer 2

– Feature exists - exclude - axis to be created

Actions around this:

– It makes sense to Set feature insert position 85 before, and after the mapkey, to Cancel feature position 62

```
~ Command `ProCmdMdlTreeSearch` ;
~ Open `selspecdlg0` `SelOptionRadio`;
~ Close `selspecdlg0` `SelOptionRadio`;
~ Select `selspecdlg0` `SelOptionRadio` 1 `Feature`;
~ Update `selspecdlg0` `ExtRulesLayout.ExtBasicNameLayout.BasicNameList` `LAYER1`;
~ Activate `selspecdlg0` `EvaluateBtn`;
~ Activate `selspecdlg0` `ApplyBtn`;
~ Update `selspecdlg0` `ExtRulesLayout.ExtBasicNameLayout.BasicNameList` `LAYER2`;
~ Activate `selspecdlg0` `EvaluateBtn`;
~ Activate `selspecdlg0` `ApplyBtn`;
~ Activate `selspecdlg0` `CancelButton`;
~ Activate `main_dlg_cur` `page_Model_control_btn` 1;
```


© 2023 INNEO Solutions GmbH

```
~ Command `ProCmdDatumAxis` ;
~ Activate `main_dlg_cur` `Rename`;
~ Update `main_dlg_cur` `PHTLeft.node_edit` `NEW_AXIS`;
~ Activate `main_dlg_cur` `PHTLeft.node_edit`;
```


© 2023 INNEO Solutions GmbH

9 Project Explorer

The Project Explorer enables you to access all project-specific data, enhance the project with data, or copy data from the project. The UIs of the following pages are similar, they are listed here with the file types you can manage on each page:

1. Checklist - .log
2. Logs - .log
3. Material - .mat / .mtl
4. Models - .prt / .asm / .drw / .txt / .txa / .dtl / .frm
5. Reports - .csv / .pdf / .log
6. Action lists - .xml
7. Model lists - .xml

All files can be edited via drag-and-drop, or via the icons. The Plus icon (10) adds a file from the hard disk to the current display. The Export icon (11) writes the currently selected file into the file system. The Delete icon (12) deletes the currently selected file.

Furthermore, there is a context menu to rename or copy the currently selected file. Double-click on a file to load it with the program that is linked with the file type on your computer. Ensure that saved changes are written to the MPZ.

The pages *Tools* and *Statistics* do not have the same UI.

© 2023 INNEO Solutions GmbH

Tools

The tools offer the possibility of clearing up the MPZ. Deleting logs and checklists (1) can improve size, loading, and saving times of the MPZs. The same can also be done manually with the Explorer. The contents of the MPZ can be shown in the Explorer (2).

Furthermore, you can carry out certain actions in an automated way.

(3) You can create a list that contains all files that are used by functions.

(4) The data that is required by actions can automatically be copied to the MPZ.

(5) All paths in actions can be changed to the data entered by (4) into the MPZ.

When you carry out actions 4 and 5 in succession, the MPZ becomes independent of the file system (CAUTION: This does not apply to the data basis.) and can thus much easier be handed over to colleagues or INNEO support without having to think of dependent data.

© 2023 INNEO Solutions GmbH

Statistics

The *Statistics* page shows all runs (provided that the log files were not deleted) and all related overviews (1), logs (2), warnings (3), and errors (4). You can also show a summary of the files where errors occurred (5).

9.1 Model Processor Zip - MPZ

All relevant data items for a Model Processor run are saved in a project file with the suffix *.mpz (Model Processor Zip). As shown by the name, this is internally a standard ZIP format. However, due to differences in the ZIP standard, you should always use the Model Processor to add and remove data in the MPZ.

The MPZ file contains the following directory structure.

© 2023 INNEO Solutions GmbH

```
<Projektname>.mpz
├── actionlists
├── batchconfigs
├── checklist
├── logs
│   ├── error
│   ├── log
│   ├── summary
│   └── warning
├── materials
├── models
└── system
    └── index.mdlprc
```

And here an example with data.

```
<Projektname>.mpz
├── actionlists
│   ├── normteilecheck.xml
│   ├── braun_layermodell.xml
│   └── fremddatenaufbereitung.xml
├── batchconfigs
│   ├── normteile.xml
│   ├── pdm_produkt_a.txt
│   └── testdaten.xml
├── checklist
├── logs
│   ├── error
│   ├── log
│   ├── summary
│   └── warning
├── materials
│   ├── stahl_allgemein.mtl
│   └── zn_zink.mtl
├── models
│   ├── layermodel_sut.prt
│   └── layermodel_sut.asm
└── system
    ├── index.mdlprc
    └── chosedActionlist.mdlprc
```


© 2023 INNEO Solutions GmbH

When Model Processor opens an MPZ file, the file is unpacked in a temporary directory. This directory can be opened with the Project Explorer in a Windows File Manager. Further information about the Project Explorer can be found when you click here 327.

When you save the MPZ files, the files that are added manually via Project Explorer or Windows File Manager are also packed into the project file.

The free addition of files and the automatic increase of LOG files are the cause that MPZ files are of different sizes even when their contents are similar. Keeping all data items in an MPZ is still expedient. This retains a direct connection between internally kept data and project, and the data versions can not be modified without the knowledge of the project user.

Furthermore, keeping data inside the MPZ makes the transfer of the MPZ to colleagues, customers, or support easier.

© 2023 INNEO Solutions GmbH

10 Project properties

Project properties

All project properties are related to all performed actions inside the project. The UI to define the project properties is shown when a project is created. Alternatively, it can be invoked later from the main UI 26.

On the first page of the project properties, you can assign an a name (1) and a description (2) to the project. The description should give an outline of the project and inform your colleagues briefly of the activity. Creating a version history here can also be expedient. This documents the changes of the project.

A Creo start key (3) can be stored that restarts Creo after each Creo crash or shutdown. The button at the right behind the entry is used for testing the entered command, NOT for the automatic determination of the start command. When it is used, the current Creo session is terminated, and restarted. Next, you must manually reestablish the connection between Model Processor and Creo.

© 2023 INNEO Solutions GmbH

For the restart, enter realistic time values in the boxes (3) and (4). These values control the response during a restart. If the time values are too short, there will never be a connection established. Too long a time gives the restart much more time than is actually required.

Furthermore, you can specify a start cycle (6) to restart Creo automatically every x files.

If Creo does not return from an action, the selection (7) can be used for restarting Creo automatically in such a case. The selected time should be long enough so that restart is not used during the regeneration of a larger module.

The Release licenses (8) action is only required if this does not happen at all, or does not happen sporadically. This is not normally required.

You can specify a time (9) during which Model Processor should not open any files.

PDM information

You do not need the following information if you do not work with a PDM system. Ensure only that *using PDM* is deactivated.

The following input options are important when you work with a PDM system.

© 2023 INNEO Solutions GmbH

The checkbox *Use PDM settings* should be activated.

1. Here you can specify a server registration or copy the current server registration. This is only necessary if you want to pass on the MPZ and expect the server not to be registered.
2. User logon to the PDM system is important and required for each overhaul. This ensures that logon and overhaul is always done with a specific user. Deactivate the checkbox if

© 2023 INNEO Solutions GmbH

you want overhaul is always to be done by the user who is currently logged on. CAUTION: An automatic restart is only possible with active user logon.

3. Checking out the part is only required when the action list modifies the part.
4. Automatic check in of a part is only possible for parts that have never been checked in with a Windchill system. To overhaul data from Windchill, deactivate Check in part and the options (5) and (6). Since the speed of Creo decreases as the number of data in the workspace increases, you should - depending on the computing power - process data packets of approximately 2000 files.
5. Emptying the workspace is only required if you want to analyze a large data volume or if you do not want to empty manually. Note the statements at Check in part (4).
6. Emptying the cache speeds up processing when emptying the workspace (5) is active. Information about malfunctions of the PDM system can be lost here.

MPZ security

This section permits MPZ loading to be secured with a password.

If the checkbox (3) is selected, the system asks for a password to open the MPZ. Opening the MPZ without password is not possible.

If the checkbox (3) is not selected, you can open and use the MPZ, but not modify it. This protects action lists against inadvertent modifications. When you click on the button for the administration UI (which is visible in such cases) and enter the password, you can edit the file at each computer. A person without a password can thus only open and execute MPZ.

© 2023 INNEO Solutions GmbH

11 Best practice

Action structure

Even though actions for the current model can be subdivided as required into header, model, and footer area, the actions should always be subdivided into

– header area - prior to model overhaul

– body area - model overhaul

– footer area - after model overhaul.

This enables the lists to be applied to other data sources, too. When the action lists are used later, failure to do this can lead to an increased effort in understanding and reuse.

Before revising large amounts of data

– Automatic scripts on the PC must be deactivated. All programs that are not required should be exited.

– Windows updates must be deactivated.

– Check the backup and shutdown phases of the PDM server to prevent the PDM server from shutting down/performing backups that would prevent working with it.

– Check the Creo license server for backup scripts and shutdown times.

– Check the Startup TOOLS database and server for shutdown times.

– Users of Windows 7 and Vista should set the following registry keys against execution and waiting of Windows error handling:

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting]
"ForceQueue"=dword:00000001
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\Consent]
"DefaultConsent"=dword:00000001
```

Revising larger data volumes in PDM systems

Although automatic check out of larger data volumes is possible in PDM systems, automatic check in is not possible. Data volumes from PDM systems should therefore only be overhauled in manageable fractions of the overall data volume.

Furthermore, a larger number of files in the workspace slows down Creo noticeably. The number of files (note: an instance is not a file) should therefore not exceed a justifiable volume. Depending on computer speed and hard disk capacity, the recommended limit is 1500 ... 2000 files.

© 2023 INNEO Solutions GmbH

The data volume can be much larger when you write a report. Here, however, you should be aware that Creo will also slow down significantly when a certain number of open and logged data is exceeded. We therefore recommend also a restart here after approximately 1500 ... 2000 files.

The modified model is not saved

There can be an error caused by the `save_objects` configuration option when the log says that a model was saved although it was not saved after it had been modified. This configuration option must be set to `changed_and_specified` to enable Model Processor to save.

Automatic restart of ProE / Creo with Startup TOOLS

To be able to start ProE / Creo automatically with Startup TOOLS, you must extend the start command with `-p:projectname`. Furthermore, under *System => Project selection wizard => Project wizard* in the Startup TOOLS configurator, *Display* must be set to *NO*.

Environment variables

The configuration option `freeze_failed_assy_comp` should be set to *NO*. If this is set to *YES*, errors in assemblies may occur as faulty assemblies can then be loaded, edited, saved and checked into a PDM system.

© 2023 INNEO Solutions GmbH

12 FAQ

The Model Processor tab sometimes disappears when the connection is interrupted while you are working via a remote desktop connection / Netviewer

Up to now, this happens only with PCs that are not equipped with a separate graphics board.

The *Tabs Placement* option can be changed via *Options* -> *Miscellaneous* -> *Windows*. The window is rendered after this change.

Error message: The application could not be initialized properly (0xc0000135)!

Installation of Microsoft .NET Framework Version 1.1 Redistributable (Microsoft Windows download).

Error message: To run this application, you must first install one of the following versions of the .Net Framework : v2.0.50727

Installation of Microsoft .NET Framework Version 2.0 Redistributable

64bit: Microsoft Windows download

32bit: Microsoft Windows download

Error: The application does not connect with Creo (Pro/Engineer) after the start.

– The environment variable PTCNMSPORT is not allowed to exist (it does not exist by default)

– No other asynchronous auxiliary applications are allowed to run

– PTC nmsd service must run (this is the rule)

– Port 1239 must be free, or be used by the PTC nmsd service.

Creo and Model Processor must be restarted after the PTCNMSPORT variable has been deleted.

A restart of both components is recommended if another auxiliary application was connected with Creo, and/or Model Processor was connected with Creo and crashed or caused an error. This process requires terminating *nmsd.exe* via the Windows Task Manager when neither application is running.

If the connection is still not functioning, disconnect peripherals from the PC or Laptop and check with a capture program whether a hardware driver or an other program uses NMSD

© 2023 INNEO Solutions GmbH

port 1239. The assignment can be checked via the Windows command line when *nmsd.exe* has been terminated.

Port 1239 assigned

To find out whether the port is assigned, enter `netstat -ano` via the Windows command line. Look for TCP Port 1239 in the list. Locate the PID number you have found in the list in the Windows Task Manager to find the process.

To display the PID numbers in Windows Task Manager, click on *View -> Select columns -> PID (Process ID)*. Press *OK* to confirm. Next, you can locate the corresponding PID in your Task Manager, reconfigure the related process or stop it.

Error: The application generates a Toolkit error after an update

Please restart Creo and Model Processor. After an update, the old DLLs can still be loaded in Creo. These DLLS are reliably removed when Creo is restarted.

© 2023 INNEO Solutions GmbH