GENIUS TOOLS®

INNEO®
That's IT.

GENIUS TOOLS Library Data Importer

11.0.0.0

Documentation

© 2024 INNEO Solutions GmbH



www.inneo.com

Table of Contents

1 GENIUS TOOLS Library Data Importer 2

1.1 General requirements ... 2
1.2 Preparations in GTL ... 4
1.3 Encrypting login data ... 6

2 Configuration 8

2.1 Configuration file ... 9
2.2 Alternative configuration files ... 15
2.3 Overwriting individual configuration options ... 15

3 Import from Windchill via SavedSearch and Rest-API 17

3.1 Creating a SavedSearch ... 17
3.2 Defining configuration options ... 19
3.3 Adapting the Rest XSLT file ... 21

4 Import from Windchill via the Rest API 24

4.1 Defining configuration options ... 24
4.2 Defining the Rest query ... 25
4.3 Testing the Rest query with Swagger UI ... 27

5 Import from other systems 32

5.1 Import via CSV ... 32
5.1.1 Rule files ... 33
5.1.2 CSV configuration options ... 34
5.1.3 CSV with and without header ... 34
5.2 Import via XML ... 36
5.2.1 XML configuration options ... 36
5.2.2 XML files ... 36
5.2.3 XSLT files ... 37

6 Copyright 39

1 GENIUS TOOLS Library Data Importer

GENIUS TOOLS Library Data Importer is a tool for importing Creo model data from an existing library or directory sstructure into a library for GENIUS TOOLS Library.

Typically, the Creo model data is imported from PTC Windchill. However, you can also provide data from other systems as import files (XML or CSV) for Library Data Importer.

Library Data Importer fills a library database for GENIUS TOOLS Library with library objects. Library objects refer to Creo models and provide structured data management, but they do not contain model data themselves. The corresponding Creo model has to be located in the path specified on the library object (the object source).

Library objects can contain metadata (additional information). Additional information can include descriptive parameters of a Creo model, the object type (prt, asm, sym, etc.), or the current status of the library object. This metadata makes is possible to quickly find the library objects in the Library Browser in GENIUS TOOLS Library.

1.1 General requirements

You need a number of helper and configuration files to use Library Data Importer.

Konfiguration settings are made in the file *conf\main.cfg*, see Configuration 8 .

Other helper files are required depending on the type of import you want to use. Examples or templates for all types of rule files can be found under *GT_Library_DataImporter\software\conf*.

There are two main types of import

– data import from Windchill, see Import from Windchill via SavedSearch and Rest-API 17 and Import from Windchill via the Rest API 24

– data import from other systems using CSV or XML import files, see Import from other systems 32

The configuration required for each type of import is described in the respective section of this documentation.

Example import files can be found under *GT_Library_DataImporter\software\PollingExamples*.

Imported data: Attributes and parameters for library objects

For each imported library object, the attribute *name* has to be specified. The library object to be created or updated is identified by its name.

© 2024 INNEO Solutions GmbH

If the object name is not contained in the data to be imported, but you want to add the data to existing library objects, you can use the configuration option `gtl_dataimporter_reverse_lookup_parameter` to specify a parameter name. The library object to import to is determined via the value of this parameter, and the data to be imported is added to the existing object. The parameter value has to be the same in the import data and on the library object.

The following other data items can be specified directly on import:

– path: object source for the library object (path to the model)

– objType: object type for the library object

– info: path / URL to the information document of the library object

– status: status of the library object

– title_de, title_en etc. for names (titles) of the library objects in different languages

– tree: name of the parent object to assign the library object to a category. During import, the ID of the parent object is determined using the name, and the imported object is linked to the parent object as a child. The parent object, that is, the target category, has to exists already.

Images / thumbnails can be imported in the Base64 format and have to specified as *thumbnail*. Please take care to use the standard Base64 format. Base64 data from Windchill is in the standard format. When you use a different data source, characters that are not allowed in URLs may have been replaced. The format with replacements cannot be read by Library Data Importer.

For data from PTC Windchill the object ID is used, e.g.:

```
OR:wt.epm.EPMDocument:29758012
```

Additional parameters can be imported as `P:PARAMETERNAME`, dimensions as `D:DIMENSIONNAME`.

Starting the program

When you have prepared all necessary helper files, start Library Data Importer by running *DataImporter.exe*. You can use the file *task.cmd* to pass start parameters.

Library Data Importer runs without displaying a graphical user interface. After the program has terminated, you can check the results in the database of the target library in GENIUS TOOLS Library and in the log files.

Error and success directories

On data import, Library Data Importer processes the data in a polling directory. Import files that have been processed successfully are moved to the *Success* directory. Import files that could not be processed fully are moved to the *Error* directory.

© 2024 INNEO Solutions GmbH

If multiple objects are imported from the same source (e.g., from Windchill, or from a CSV files with multiple lines), and an error occurs during data base import preparation, the entire file is moved to the *Error* directory, and no changes are made in the database.

If an import file contains multiple objects with the same name, an error occurs and the entire file is moved to the *Error* directory. However, all objects except the ones with names that are not unique are written to the database. An entry is created in the log file. The affected names are explicitly listed at the end of the log file.

Files that are still in the polling directory when the program terminates are moved to the *Error* directory.

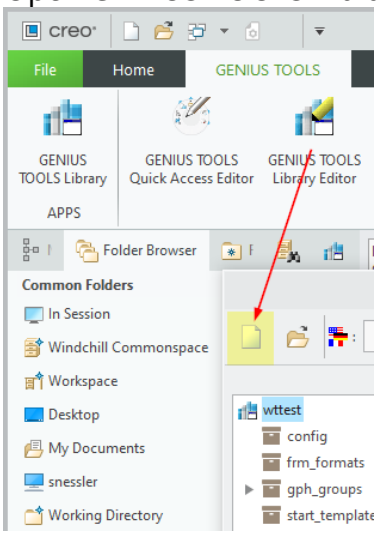
1.2 Preparations in GTL

This section describes preparations you have to make in the GENIUS TOOLS Library data structure before importing data with Library Data Importer.

Creating a new library

To create a new library for the data to be imported, proceed as follows:

1. Open GENIUS TOOLS Library Editor and click *New library*.



2. Enter a name for the new library, verify the language settings and confirm by clicking *OK*. The new library is created and opened.

Importing categories

If you want to repeatedly import data from Windchill, for example to keep the data in GENIUS TOOLS Library current, take care to create the required categories before the first import, or import the categories before the first import.

To assign objects to a category on import, the category already has to be present. If you want to assign objects to a category on import, the import data has to specify the

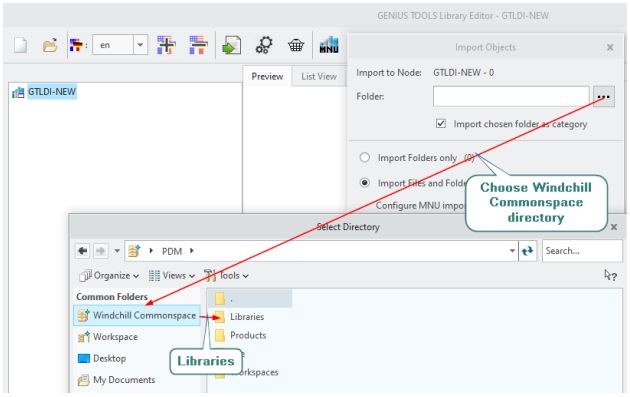
© 2024 INNEO Solutions GmbH

attribute *tree*, which contains the name of the parent object. Also consult the list of data items imported for library objects under GENIUS TOOLS Library Data Importer[2] .

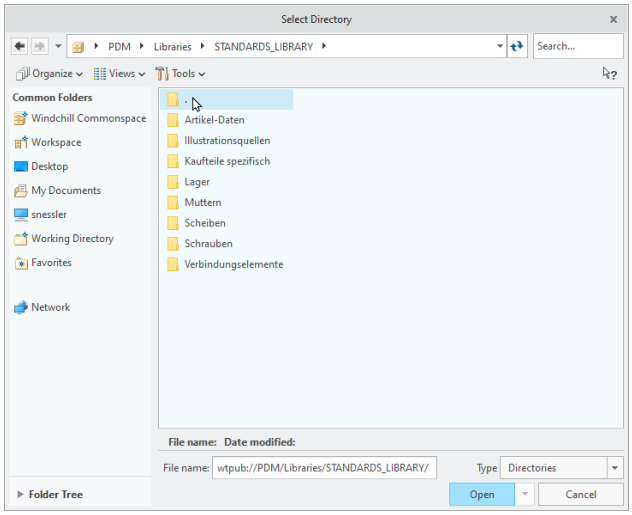
During a repeated import, objects will only be updated if they are found in the same category as during the first import. Otherwise, objects will be created again in the category where Library Data Importer expects them due to the first import.

To import categories from Windchill, please proceed as follows:

1. In GENIUS TOOLS Library Editor, open the import wizard.
2. As the folder to be imported, select the *Libraries* folder in the Windchill Commonspace.

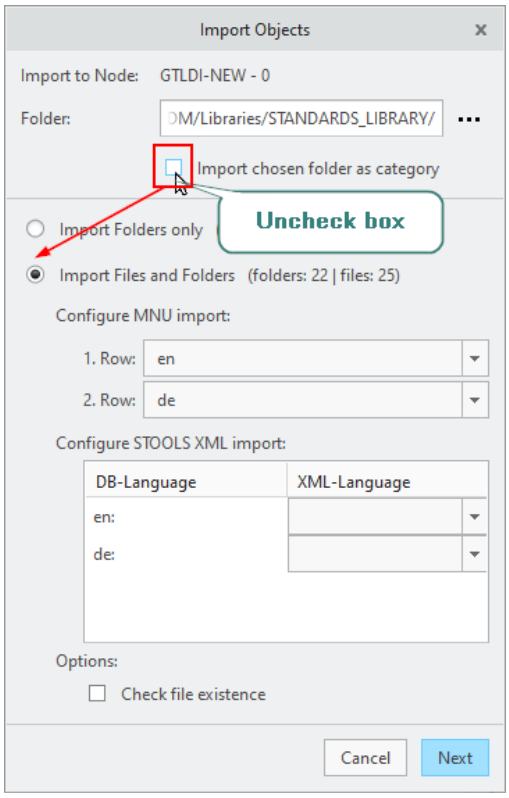


3. Select the required library and click *Open*.



4. Set the import options. Remove the setting *Import selected folder as category*.

© 2024 INNEO Solutions GmbH



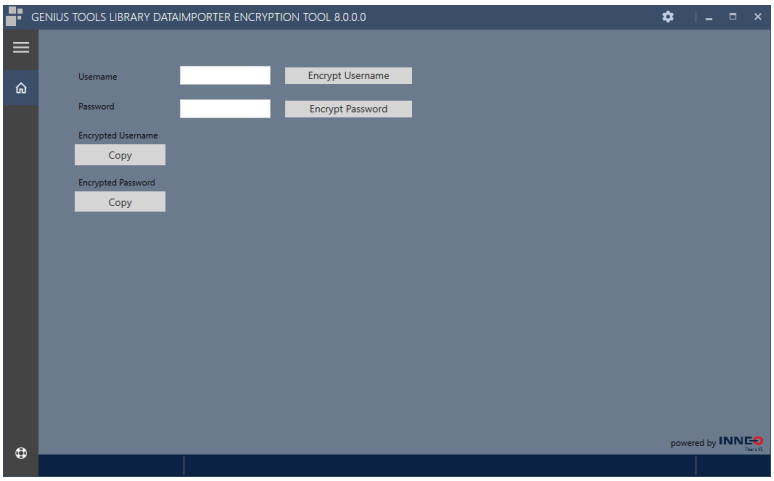
5. Select the required categories and click *Import*. The categories are imported and can be assigned to the library objects to be imported by Library Data Importer.

1.3 Encrypting login data

When importing data from Windchill, Library Data Importer requires a Windchill login. In order to be able to securely transmit the login data, Library Data Importer comes with an encryption tool. EncryptionTool encrypts the login data in a way that Library Data Importer can work with.

EncryptionTool is started independently of Library Data Importer by running the executable file *EncryptionTool.exe*.

© 2024 INNEO Solutions GmbH



EncryptionTool user interface

To encrypt the user name or password, enter the text into the corresponding input field and click *Encrypt*. The encrypted value is displayed and copied to the clipboard. In this way, you can immediately insert the value in the CFG file.

If you encrypt both values, please note that only the value encrypted last is on the clipboard. The current clipboard value is highlighted in yellow in EncryptionTool.

If you click *Encrypt* again, the input value is newly encrypted, even if it has not changed.

If you change the value in one of the input fields, the corresponding encrypted value is deleted and the clipboard is emptied.

You can use the *Copy* buttons to add one of the encrypted values to the clipboard again. Alternatively, you can select the encrypted text as usual and copy it to the clipboard using Ctrl+C.

© 2024 INNEO Solutions GmbH

2 Configuration

The behavior of Library Data Importer is determined by the configuration options in the file *conf\main.cfg*.

The configuration options define the data source. Which type of file do you want to process? Are you importing data from PTC Windchill?

The configuration options also determine the location of all directories, rule files, and the log file. Further configuration options contain additional parameters for data processing.

The configuration file is located in the installation directory under *<Drive>\<InstallationDirectory>\GT_Library_DataImporter\Software\conf\main.cfg* and can be changed using any text editor.

Required configuration options

Typically, not all configuration options are needed.

Which set of configuration options has to be defined depends on the source for the import data, or the method with which Windchill data is to be imported.

If multiple data sources should be processed in one run, the configuration options for all data sources have to be defined.

Default values for some options are defined directly in the program source code, but can be overwritten by setting values in a CFG file. The values can also be changed by overwriting them in the file *GT_Library_dataImporter\task.cmd*.

The following configuration options are always required and have a hard-coded default value.

- gtl_dataimporter_log_dir (It is recommended not to change this.)
- gtl_dataimporter_polling_dir
- gtl_dataimporter_success_dir
- gtl_dataimporter_error_dir
- gtl_dataimporter_sleeptime_int

When you change directory paths, make sure that the specified directory exists and that Library Data Importer has read and write access.

The following configuration option is also required, but it does not have a hard-coded default value. It has to be passed for each program start.

- gtl_dataimporter_db_path

© 2024 INNEO Solutions GmbH

This configuration option defines the path to the GENIUS TOOLS Library database. This path is created automatically when you create a library. You can copy it from GENIUS TOOLS Library Editor and paste it into the CFG file.

2.1 Configuration file

gtl_dataimporter_db_path	All types of import
Path to the GENIUS TOOLS Library database, including the file name. Has to be defined in the CFG file, the *task.cmd* or via the console start command. Default path pattern: *Drive\DirectoryStructure\gt_resource_folder\library\DatabaseName.db*	

gtl_dataimporter_reverse_lookup_parameter	All types of import
Specifies a parameter to assign import data items to an existing object in GENIUS TOOLS Library. Typically, the attribute *name=* is passed for each object during data import. If the object name is not contained in the import data, the specified parameter is checked. Using the parameter value, which has to be the same in the import data and on the library object, the correct library object is determined, and the data items from the import are added to the existing object. Example: *P:PartNumber*	

gtl_dataimporter_multiple_value_separator	All types of import
Character separating multiple values for the same parameter. If no character is specified (default), but the import data contains multiple values for the same parameter, an error occurs and the affected object is not imported. If a character is specified, multiple values for the same parameter will be imported and separated by the defined character.	

© 2024 INNEO Solutions GmbH

gtl_dataimporter_log_dir	All types of import
Last and next-to-last log file with success and error messages are saved to the specified directory. It is recommended to use the default path in order to avoid write access conflicts. Name of the log file: *log.txt* The next-to-last log file is renamed to *oldLog.txt*. Do not include the file name in the specified path. Enter an absolute path or a path relative to *DataImporter.exe*. Default value: ..\..*data**log* relative to *DataImporter.exe* Even if you do change the path, please do not delete the original log folder.	

gtl_dataimporter_polling_dir	All types of import
All files containing data on objects to be imported are placed in the specified directory for checking. Enter an absolute path or a path relative to *DataImporter.exe*. Default value: ..\..*data**polling* relative to *DataImporter.exe* Even if you do change the path, please do not delete the original folder.	

gtl_dataimporter_success_dir	All types of import
All successfully processed files are moved to the specified directory by Library Data Importer. Enter an absolute path or a path relative to *DataImporter.exe*. Default value: ..\..*data**success* relative to *DataImporter.exe* Even if you do change the path, please do not delete the original log folder.	

gtl_dataimporter_error_dir	All types of import
All files containing objects that cannot be imported are moved to the specified directory by Library Data Importer. Enter an absolute path or a path relative to *DataImporter.exe*. Default value: ..\..*data**error* relative to *DataImporter.exe* Even if you do change the path, please do not delete the original log folder.	

© 2024 INNEO Solutions GmbH

gtl_dataimporter_sleeptime_int	All types of import
Time lapse in milliseconds between saving the files to be checked in *polling* and processing them, to buffer delays when the data to be checked is saved automatically. Default value: *2000*	

gtl_dataimporter_unsecure_db_connection	All types of import
Defines whether the data base connection will be maintained throughout a Library Data Importer run (1/true). The database will be blocked for any other access, but the processing time for Library Data Importer will be shorter. If the setting is at 0/false, a new database connection will be opened and closed for each object to be saved.	

gtl_dataimporter_change_parameter	All types of import; optional; recommended but not required
Name of the parameter to be used for checking for object changes. The parameter has to be present for objects to be imported in order to prevent overwriting unchanged or newer values in the database. The parameter typically contains a date. Different date formats can lead to newer values being overwritten. Please use a clearly-defined company-wide standard. Default value: *P:CHANGEDATE*	

gtl_dataimporter_db_thumbnail_path	All types of import; optional; recommended but not required
Absolute or relative path to the database's image file directory. Default path pattern: *Drive\DirectoryStructure\gt_resource_folder\library\DatabaseDirectory* Based on *img, img_w40* and *img_w100* will be created if they do not exist. Preview images for objects should be passed to GENIUS TOOLS Library.	

gtl_dataimporter_wc_server	All Windchill imports
Name of the Windchill server in a part URL ending in */Windchill*, e.g. *http://ServerAddress:Port/Windchill*	

© 2024 INNEO Solutions GmbH

gtl_dataimporter_wc_login_name	All Windchill imports
Windchill user name (unencrypted), has to be present for Windchill communication either in the unencrypted or in the encrypted form. Library Data Importer prefers the encrypted form. For information on encryption, please refer to Encrypting login data[6] .	

gtl_dataimporter_wc_login_password	All Windchill imports
Windchill password (unencrypted), has to be present for Windchill communication either in the unencrypted or in the encrypted form. Library Data Importer prefers the encrypted form. For information on encryption, please refer to Encrypting login data[6] .	

gtl_dataimporter_wc_login_name_encrypted	All Windchill imports
Windchill user name (encrypted), has to be present for Windchill communication either in the unencrypted or in the encrypted form. Library Data Importer prefers the encrypted form. For information on encryption, please refer to Encrypting login data[6] .	

gtl_dataimporter_wc_login_password_encrypted	All Windchill imports
Windchill password (encrypted), has to be present for Windchill communication either in the unencrypted or in the encrypted form. Library Data Importer prefers the encrypted form. For information on encryption, please refer to Encrypting login data[6] .	

gtl_dataimporter_wc_rest_xslt	Windchill Rest API with or without SavedSearch
Absolute or relative path to the XSLT file for processing data obtained from Windchill via the Rest API. A commented example can be found under *Drive\DirectoryStructure\GT_Library_DataImporter\software\conf\rest.xslt*	

gtl_dataimporter_odata_maxpagesize	Windchill Rest API with or without SavedSearch
Provides the possibility to change the number of results per search process and thus the processing speed of the software, e. g. if the server has a low performance. The value must range from 1 to 20,000. Default: 500 results per query.	

© 2024 INNEO Solutions GmbH

gtl_dataimporter_rest_select	Windchill Rest API with SavedSearch
List of all attributes to be evaluated for the objects to be imported. The attribute names are separated by commas, and the list contains no spaces. The recommended example which works with the example XSLT *rest.xslt* is the following: *CADName,container,folder,state,version,MATERIAL,DESCRIPTION_1_DE,DESCRIPTION_2_DE,DESCRIPTION_1_EN,DESCRIPTION_2_EN,thePersistInfo.modifyStamp*	

gtl_dataimporter_wcSearch_name	Windchill Rest API with SavedSearch
Name of the SavedSearch in Windchill to be used for finding the objects.	

gtl_dataimporter_wc_rest_query	Windchill Rest API without SavedSearch

Rest URL for queries that are not based on a SavedSearch. The content consists of the SELECT elements of the query.

Parts:

– http://

– Address:Port

– /Windchill/servlet/rest/structure/objects

– ?%24filter=

– (endswith(name%2C%20'prt')%20or%20endswith(name%2C%20'asm')%20%20or%20endswith(name%2C%20'drw')%20or%20endswith(name%2C%20'gph')%20or%20endswith(name%2C%20'mfg')%20or%20endswith(name%2C%20'lay')%20or%20endswith(name%2C%20'frm')%20or%20endswith(name%2C%20'tbl')%20or%20endswith(name%2C%20'sym')%20or%20endswith(name%2C%20'txt')%20or%20endswith(name%2C%20'sec'))%20and%20state%20eq%20'RELEASED'

– &%24select=CADName%2Ccontainer%2Cfolder%2Cversion%2CMATERIAL%2CDESCRIPTION_1_DE%2CDESCRIPTION_2_DE%2CDESCRIPTION_1_EN%2CDESCRIPTION_2_EN%2CthePersistInfo.modifyStamp

– &typeId=wt.epm.EPMDocument

– &queryLimit=

© 2024 INNEO Solutions GmbH

gtl_dataimporter_rest_query_int	Windchill Rest API without SavedSearch
Integer value to define the number of objects to be fetched with each Rest request. If few objects are expected, set a low number. If many objects are expected, set a higher number. The value *0* selects the largest possible interval, that is, all objects from the list selected in Windchill.	

gtl_dataimporter_rest_query_days	Windchill Rest API without SavedSearch
Integer value to define the number of past days to use in the search for objects. The value *0* selects the longest possible period.	

gtl_dataimporter_wc_webapi_xslt	Windchill query with web API
XSLT file for processing objects that were found via Web API/JSP Action Call. Use is not recommended.	

gtl_dataimporter_xml_rule_file	XML (other systems)
XSLT file for processing XML files from other systems than Windchill, e.g., *Drive\DirectoryStructure\GT_Library_DataImporter\software\conf\xml_rule.xslt*	

gtl_dataimporter_csv_rule_file	CSV (other systems)
Rule file in the TXT format to define the attributes and their position in CSV import files. Depending on the CSV structure, attributes are defined by the column name in the header row, or by the position in each row. Examples for CSV files with or without header can be found under: – *Drive\DirectoryStructure\GT_Library_DataImporter\software\conf\csv_rule_header.txt* – *Drive\DirectoryStructure\GT_Library_DataImporter\software\conf\csv_rule_headerless.txt*	

gtl_dataimporter_csv_header_bool	CSV (other systems)
Boolean value (0/1 or false/true) to define whether the CSV import file has a header row (1/true) or not (0/false).	

gtl_dataimporter_csv_separator_char	CSV (other systems)
Separator character in the CSV import files, typically either comma (,) or semicolon (;).	

© 2024 INNEO Solutions GmbH

Format hints

The commented default CFG file *main.cfg* is located under *Drive\DirectoryStructure\GT_Library_DataImporter\software\conf*.

A CFG file has to exist under *software\conf*.

You can fill the default CFG file with the required values for your import, or use it as a template for alternative configurations, see also Alternative configuration files[15].

A configuration file has to have the extension *.cfg*.

Each row in a CFG file follows this pattern:

```
gtl_dataimporter_key = Value
```

There may be no line break between the key and the value.

Comments or unused lines can be marked with a leading semicolon:

```
;Comment
```

Empty lines are ignored by Library Data Importer.

2.2 Alternative configuration files

If you want to switch between different configuration settings for Library Data Importer, you can store different CFG files.

To use a defined configuration file in the software, you can call it either directly via the start command or via the file *task.cmd*.

A path to a defined configuration file is passed immediately after the program call, either as an absolute path or as a path relative to *DataImporter.exe*. The start parameter has to be prefixed with `-p`.

```
-p <Drive>:\<Path>\CFGFile.cfg
```

All values contained in the CFG file are set as specified. Values not contained in the file keep using the default value or are read from the *main.cfg* as before.

Example from the file *task.cmd*:

```
@echo off
cd software/bin/
DataImporter.exe -p C:\GT_Library_DataImporter\software\conf\alternative.cfg
```

2.3 Overwriting individual configuration options

You can change the value of individual configuration options directly via the start command or via the file *task.cmd*.

© 2024 INNEO Solutions GmbH

To pass configuration options, use the prefix `-c` followed by the name and value of the option: `-c gtl_dataimporter_config_<Name> <Value>`

Example from the file *task.cmd*:

```
@echo off

cd software/bin/

DataImporter.exe -p ..\GT_Library_DataImporter\software\conf\alternative.cfg
-c gtl_dataimporter_config_csv_header_bool 0
```

If you make multiple modifications to the same configuration option, the last value passed is used.

Values that are not modified keep using the default value or are read from the *main.cfg* as before.

© 2024 INNEO Solutions GmbH

3 Import from Windchill via SavedSearch and Rest-API

This section describes data import from Windchill where the data to be transferred is defined by a SavedSearch in Windchill. This is the recommended type of import for data from Windchill, as well as the main application of Library Data Importer.

To import data from Windchill using a SavedSearch, you have to make the following preparations:

– Creating a SavedSearch 17

– Defining configuration options 19

– Adapting the Rest XSLT file 21

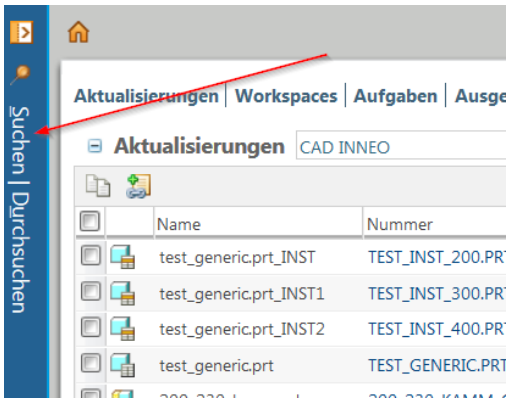
Starting the program

When you have prepared all necessary helper files, start Library Data Importer by running *DataImporter.exe*. You can use the file *task.cmd* to pass start parameters.

Library Data Importer runs without displaying a graphical user interface. After the program has terminated, you can check the results in the database of the target library in GENIUS TOOLS Library and in the log files.

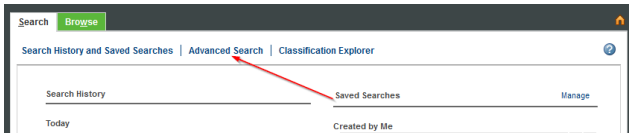
3.1 Creating a SavedSearch

The recommended method for defining the data to be imported from Windchill is to define and save a search query in Windchill. This search query can then be addressed by its name and used for data transfer.

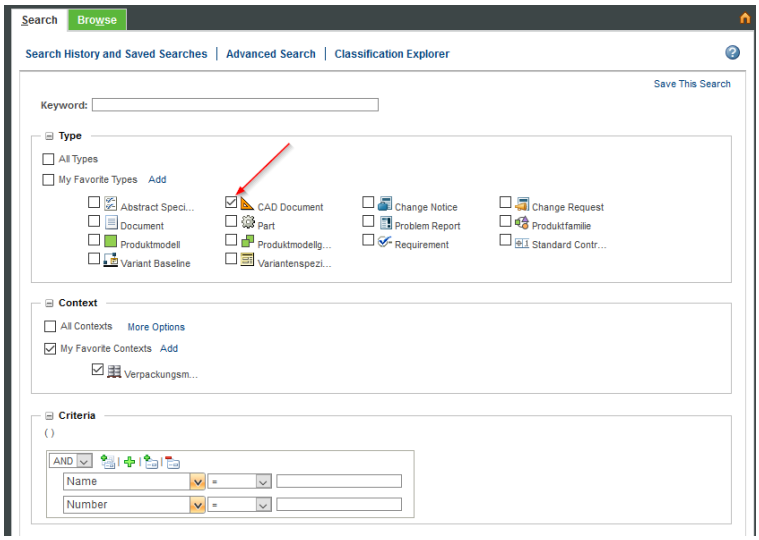


© 2024 INNEO Solutions GmbH

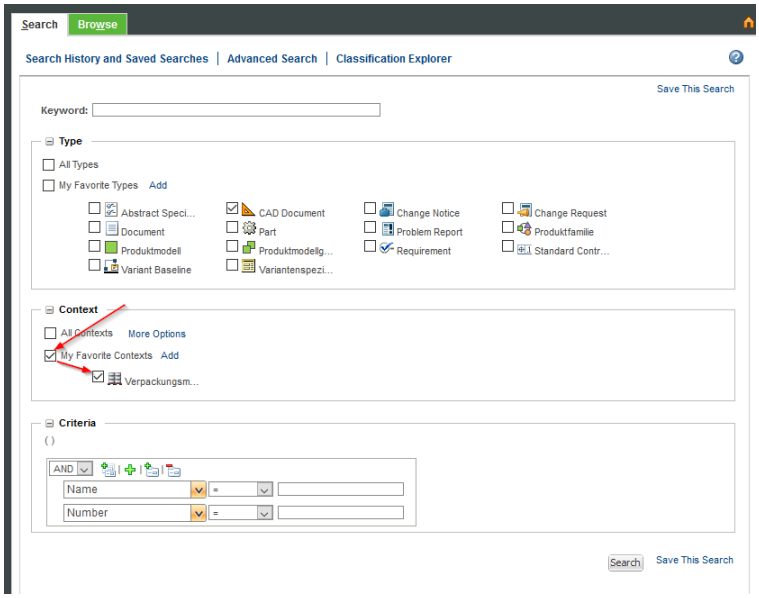
Extended search:



Select favorites: first types, then contexts

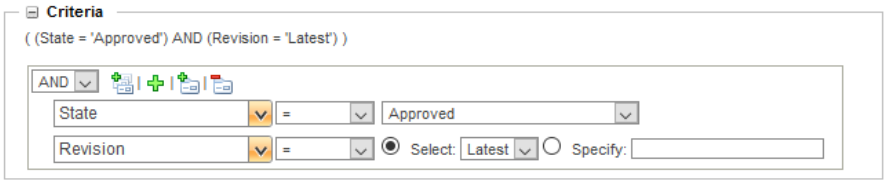


In the *Type* segment: When checking *My Favorite Types*, all types will be selected automatically. Alternatively, check the box *CAD Document* directly. Be careful not to select the box *Document*.



© 2024 INNEO Solutions GmbH

Define criteria:



The criteria displayed at first are linked with AND by default:

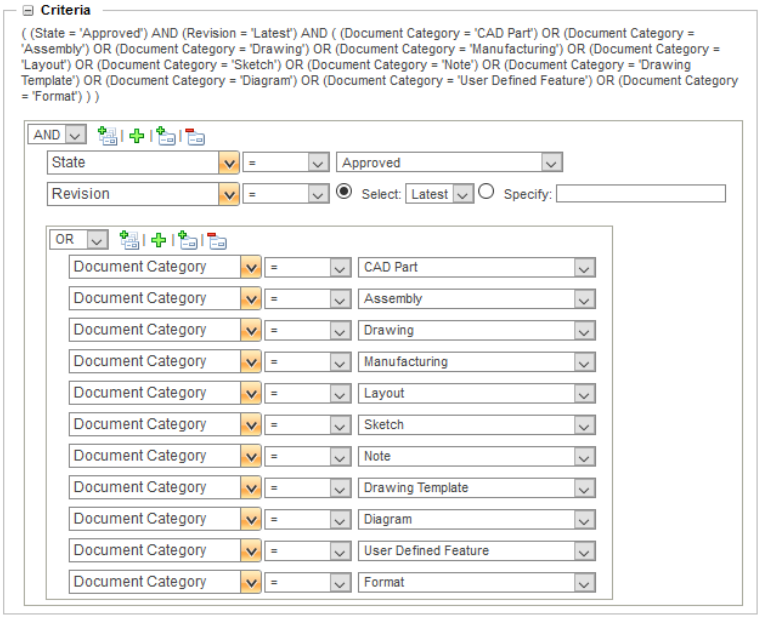
– Life cycle status (for example, if you only want to import only data that has been cleared)

– Latest change (period to look into the past; for regular automated imports, select the period so that there are no overlaps in the automated process)

You can create further selection criteria.

In the example, a filter by document revision has been created:

1. Add a new criterion via *Add attribute*: select latest revision
2. Add a group, and link attributes with OR within that group to select the required document types



Save the search and define a name for it.

3.2 Defining configuration options

This section describes configuration options that have to be set for importing data from Windchill via a SavedSearch.

For more information on configuration options, please also refer to Configuration file[9] .

© 2024 INNEO Solutions GmbH

configOption	Example
gtl_dataimporter_db_path =	Drive: \path\gt_resource_folder\library\GTL_db.db
gtl_dataimporter_db_thumbnail_path =	Drive: \path\gt_resource_folder\library\GTL_db\
gtl_dataimporter_wc_server =	http://WT-Server:PortNr/Windchill
gtl_dataimporter_wc_login_name =	wcadmin
gtl_dataimporter_wc_login_password =	wcadmin
gtl_dataimporter_wc_login_name_encrypted =	Iq3V,2T@E9?{]mHtBf9]24.]&lt\z2cN)}Tp&? U9zq/H{@5O(#0yRE&+IQr"3G? nU<Tud9"8TiIOmti?0c6PXT%TQo7} Nu(jJ.4r?-,W(t0}NOQ951\L)mlE| WnIQ>UTZ\xy*Q($P&WS&Cdfy_cB7,:wc) (@X%t_Wap&>x\s;JE@E2Viajs3">dP|}v) e>e\>r"a\8fm,Fd%:<u]m<D:L5:H0#r2% >'*N[RGZ2mAD\Z#X[i<oW{_I:]Lm;m<]?IX
gtl_dataimporter_wc_login_password_encrypted =	+4Tv[QOZK_S9qY/VY5NW;M8V4yft9ps[{A3 qFrS>[crmfyiO07Eq3wIA2J8FotM? t8J0lrM/*mR? HqIXS3A2R9<GW9huMGi&(oH?9AUt[UjRo %ALQRIZOW3iv5mUJ/7IV9pN&M4FnO8} ZodU6vZuF7HV/l} yi/GPtXO\rXheNXMNfQ&Bb&uotNgXzj[9q oY{syJ7| oL[Xf*Xu4gU@qk20"w{CKdC*Psq/8uy86R} 1r20q"1;a*x/R!g@x&Cs4$/y
gtl_dataimporter_wc_rest_xslt =	..\conf\rest.xslt
gtl_dataimporter_change_parameter =	CHANGEDATE
gtl_dataimporter_wcSearch_name =	GTLI2

The login information under `gtl_dataimporter_wc_login_name` and `gtl_dataimporter_wc_login_password` should correspond to a user account that can access the data to be imported.

© 2024 INNEO Solutions GmbH

The login information under `gtl_dataimporter_wc_login_name_encrypted` and `gtl_dataimporter_wc_login_password_encrypted` work in the same way, but are preferred by Library Data Importer over their unencrypted counterparts. If there is both an encrypted and an unencrypted value, the encrypted value will overwrite the unencrypted value. If an encrypted value cannot be decrypted, Library Data Importer tries to use the unencrypted value instead.

If both types of login information are invalid or not present, Library Data Importer aborts. The same happens if the password and user name are both valid, but do not belong together. Please take care to either configure only one type of login information, or to use the same user account for the encrypted and unencrypted settings. It is possible to pass an encrypted password and unencrypted user name or the other way around.

Under `gtl_dataimporter_wcSearch_name`, specify the name of the search.

Specify the attributes to read for each object under `gtl_dataimporter_rest_select`. Take care not to enter space characters in the attribute list. Separate the attribute names with commas. To determine possible attributes, it is helpful to use lists (for example at *Tools* > *Parameters* in Creo if an object is loaded), and you can also use *Swagger UI* in Windchill for constructing queries, or the application *Postman*, which can test Rest queries.

To find the latest change to an object, you can use *thePersistInfo.modifyStamp*. This attribute is assigned to the required parameter in the XSLT file (for example, `P:CHANGEDATE`).

3.3 Adapting the Rest XSLT file

There is a commented example file at *GT_Library_DataImporter\software\conf\rest.xslt*.

Format specifications for XSLT files:

– All nodes have to have a corresponding `AttributeName=` assignment in the XSLT file that uses the value of one or more adequate XML nodes.

– The assignment instructions for each attribute, each parameter and each dimension have to end in `<xsl:text>
</xsl:text>` This instruction adds a minus sign (-) and a line break (Enter) at the end of each object so that Library Data Importer can recognize the end of an object during processing.

– If more than one object is contained in a file, the XSLT file should wrap the nodes belonging to one object in a for-each command, e.g.;
`<xsl:for-each select="root/element/items"> </xsl:for-each>`

– The XML files to be processed are automatically created from the JSON files which are in turn obtained from the Odata API. You don't have to generate the XML files nor edit them. The files consist of 25 data sets each. Files will be requested by Windchill until all objects that were searched for and their data have been processed. All can be found in

© 2024 INNEO Solutions GmbH

the *success* directory after processing (or in the *error* directory if there are records that cannot be processed).

– For each object, the `name` has to be defined.

– If you want to import thumbnail images from Windchill, the label `thumbnail` needs to have the `ID` value assigned. The processing logic uses the fixed ID pattern to access the required graphic file.

– To pass parameters, use `P:ParameterName=`.

– To pass dimensions, use `D:DimensionName=`.

– Language-specific names can be passed using `title_<LanguageCode>=`

– Besides `name`, the labels `path`, `info`, `objType` and `status` are available by default in GENIUS TOOLS Library and do not have to be declared separately as parameters or dimensions.

– You can use the label `tree` to pass the name of a parent object (category).

– To find out the numerical values used in `objType`, please refer to the example file. The value depends on the file type of the corresponding object.

– The check value for object changes has to be passed as a parameter (e.g., `P:CHANGEDATE`).

– In the CFG file, under `gtl_dataimporter_change_parameter`, specify the name of the parameter only (e.g., *CHANGEDATE*). As soon as this parameter is specified for an object, it can always be used during updates to check the changed status, as long you do not define a new check parameter in the CFG file. If you do not use a check parameter, updates will always be run, even if an older version with the same name is used or the object has not changed.

– Specify the path to the XSLT file under `gtl_dataimporter_wc_rest_xslt` as an absolute path or relative to *DataImporter.exe*.

To make it possible to read only defined folders within a library, the entire set of processing instructions in the XSLT file is bracketed by a separate instruction, e.g.:

```
<xsl:if test="normalize-space(attributes/container) = 'Genius Tools Library' and normalize
…(all other instructions)
</xsl:if>
```

In the example, only objects from the library *GENIUS TOOLS Library* (*attributes/container*) that are located in the folder *test_gle* (*attributes/folder*) will be read.

Alternatively, the example file contains a commented-out variant to exclude individual Windchill folders. This variant can be used if you want to read multiple folders and exclude only a few, or if there are individual objects on the same level as the folders in the directory structure, and you want to read the individual objects, too:

```
<xsl:if test="normalize-space(attributes/container) = 'Genius Tools Library' and normalize
…(all other instructions)
```


© 2024 INNEO Solutions GmbH

```
</xsl:if>
```

In the example, only objects from the library *GENIUS TOOLS Library* (*attributes/container*) will be read, and the folders (*attributes/folder*) *config, frm_formats, gph_groups* and *start_templates* are excluded.

You could also combine both types of instruction if you follow the logic rules of the XSLT format.

© 2024 INNEO Solutions GmbH

4 Import from Windchill via the Rest API

This section describes data import from Windchill where it is not possible to create a SavedSearch to define the data to be imported.

To import data from Windchill using the Rest API, but no SavedSearch, you have to make the following preparations:

– Defining configuration options[24]

– Defining the Rest query[25]

– Testing the Rest query with Swagger UI[27]

– Adapting the Rest XSLT file. The required XSLT file is the same one that is used for Windchill import with a Saved Search, see Adapting the Rest XSLT file[21].

Starting the program

When you have prepared all necessary helper files, start Library Data Importer by running *DataImporter.exe*. You can use the file *task.cmd* to pass start parameters.

Library Data Importer runs without displaying a graphical user interface. After the program has terminated, you can check the results in the database of the target library in GENIUS TOOLS Library and in the log files.

4.1 Defining configuration options

This section describes configuration options that have to be set for importing data from Windchill using the Rest API but no SavedSearch. Mostly the same configuration options are required as when you import Windchill data with a SavedSearch.

For more information on configuration options, please also refer to Configuration file[9] .

For information on the structure of the Rest query, please refer to Defining the Rest query[25].

configOption	Example
gtl_dataimporter_db_path =	Drive:\Folder_Structure\gt_resource_folder\library\GTL_db.db
gtl_dataimporter_db_thumbnail_path =	Drive:\Folder_Structure\gt_resource_folder\library\GTL_db\

© 2024 INNEO Solutions GmbH

configOption	Example
gtl_dataimporter_wc_server =	http://WT-Server:PortNr/Windchill
gtl_dataimporter_wc_login_name =	wcadmin
gtl_dataimporter_wc_login_password =	wcadmin
gtl_dataimporter_wc_rest_xslt =	..\conf\rest.xslt
gtl_dataimporter_change_parameter =	CHANGEDATE
gtl_dataimporter_wc_rest_query =	http://<Windchillserver:Port>/Windchill/servlet/rest/structure/objects?%24filter=(endswith(name%2C%20'prt')%20or%20endswith(name%2C%20'asm')%20%20or%20endswith(name%2C%20'drw')%20or%20endswith(name%2C%20'gph')%20or%20endswith(name%2C%20'mfg')%20or%20endswith(name%2C%20'lay')%20or%20endswith(name%2C%20'frm')%20or%20endswith(name%2C%20'tbl')%20or%20endswith(name%2C%20'sym')%20or%20endswith(name%2C%20'txt')%20or%20endswith(name%2C%20'sec'))%20and%20state%20eq%20'RELEASED'&%24select=CADName%2Ccontainer%2Cfolder%2Cversion%2CMATERIAL%2CDESCRIPTION_1_DE%2CDESCRIPTION_2_DE%2CDESCRIPTION_1_EN%2CDESCRIPTION_2_EN%2CthePersistInfo.modifyStamp&typeId=wt.epm.EPMDocument&queryLimit=
gtl_dataimporter_rest_query_int =	10
gtl_dataimporter_rest_query_days =	1

4.2 Defining the Rest query

The Rest query, which has to be defined as a configuration option, consist of the following elements:

© 2024 INNEO Solutions GmbH

– URL to the Windchill Rest API

– filter for the objects to be imported

– SELECT: Which attributes of the filtered objects should be read?

You can construct your query based on the example in the *main.cfg* file, or use the *Swagger UI* tool in Windchill to define it, see Testing the Rest query with Swagger UI 27.

URL
Enter an URL under `gtl_dataimporter_wc_rest_query =` that defines the path to the Windchill Rest API. The SELECT data is appended to the URL.

The static part is *http://<ServerAddress>:<Port>/Windchill/servlet/rest/structure/objects*, where you have to enter the server name and port.

Filter
After the URL, define a filter, for example to restrict the data import to certain file types and lifecycle status values: *?%24filter=<For filter definition see example value>*.

Please note that characters not allowed in a URL have to be replaced, for example:

– %2C = ,

– %20 = space

– %24 = $

Example filter criterion: `(endswith(name%2C%20'prt')`.

SELECT
After the filter, define the SELECT data, specifying which attributes of the filtered objects to display, for example: *&%24select=CADName%2Ccontainer%2Cfolder%2Cversion%2CMATERIAL%2CDESCRIPTION_1_DE%2CDESCRIPTION_2_DE%2CDESCRIPTION_1_EN%2CDESCRIPTION_2_EN%2CthePersistInfo.modifyStamp*

The SELECT always ends with the AND-linked type ID and the number of records that should be fetched with the query, for example:
&typeId=wt.epm.EPMDocument&queryLimit=

Please note that you must NOT set a value for the trailing *queryLimit=* Please also refer to the description of the configuration option `gtl_dataimporter_rest_query_int` under Configuration file 9 .

Further information for query definition

If you do not define a query period or maximum number of objects or set the values to *0*, all data will be read and processed. This takes significantly longer, but should be done for the first data import to transfer all data to GENIUS TOOLS Library.

© 2024 INNEO Solutions GmbH

Query period for object changes
Depending on how often you want to import data, use the configuration option `gtl_dataimporter_rest_query_days =` to specify the period in days which you want to look back from the current date to find changed objects in Windchill.

If you have a lot of changes, use a shorter period, maybe 1-2 days, if you have only a few changes, select a longer period, such as 7 days.

If you run Library Data Importer at regular intervals, it is recommended to synchronize the period with the program start frequency, that is, select a period that is sure to cover the time since the last run in order not to miss any changes.

Use the *Changedate* parameter to make sure that no objects that have already been changed are overwritten in the case of overlapping query periods.

Maximum number of objects
The value of the *queryLimit* is added automatically by Library Data Importer as specified under `gtl_dataimporter_rest_query_int =` This makes sure that all new records get processed, which might not be the case for a query limit entered statically into the URL.

For the maximum number of objects, refer to the average number of transferred objects. For example, if you run Library Data Importer daily and have an average of 10 objects with changes per day, 10 is a reasonable value for the maximum number. If you expect larger amounts of data to be transferred, for example because you start fewer runs or have a lot of small changes, you should set a larger maximum number.

The maximum number setting is a way to optimize data transfer time.

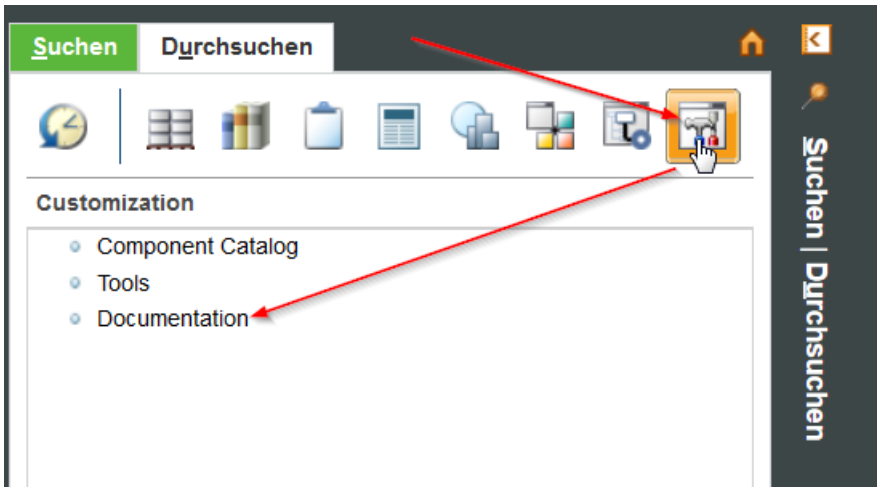
4.3 Testing the Rest query with Swagger UI

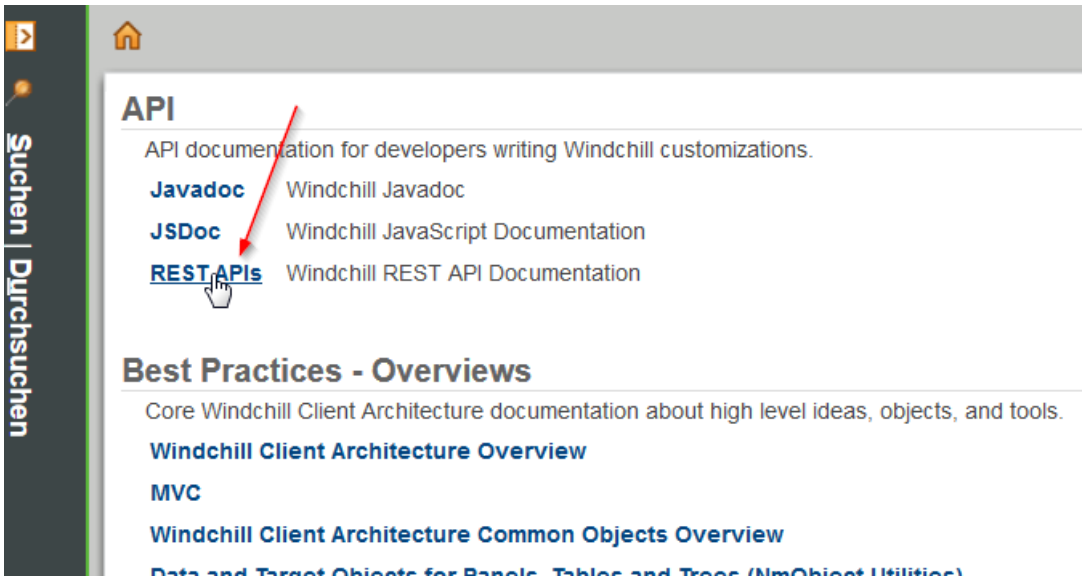
Swagger UI is a tool that comes with Windchill and lets you test or create Rest queries.

For information on the structure of the Rest query, please refer to Defining the Rest query 25.

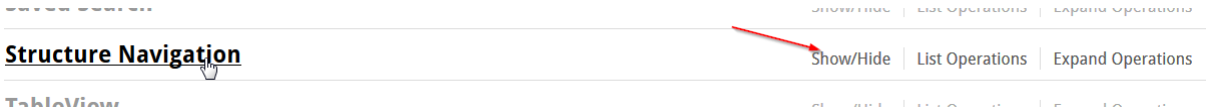
You find Swagger UI here:

© 2024 INNEO Solutions GmbH





Under *Structure Navigation,* you can create Rest queries.



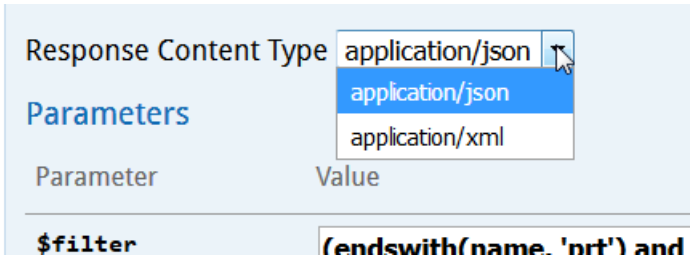
Wählen Sie */structure/objects* aus.



Scrollen down to see the input fields under *Parameters*.

© 2024 INNEO Solutions GmbH

Under *Response Content Type,* select *application/json*.



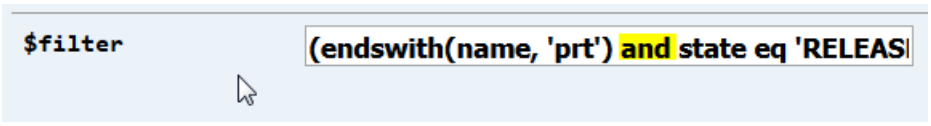
Parameters

Parameter	Value	Description	Parameter Type	Data Type
$filter	(required)	**An OData filter criteria in the form of "<propertyName> eq '<value>' [and ...]" where <property-name> is either 'number' or 'name'.For example:** `$filter=number eq '0000222341' or name eq 'PTC' or startswith(name, 'PTC') or view eq 'wt.vc.views.View:21281' and state eq 'INWORK' and iterationIdentifier eq '1' and versionIdentifier eq 'A'`	query	string
navigationCriteria		Windchill OID of the navigation criteria or the navigation criteria name. This is used to choose the version of the identified object. If not specified, then the latest version is chosen.	query	string
$select		An OData select expression that is a comma-separated list of property names.	query	string
typeId	(required)	**The type on which the query should happen, must be instance of Iterated class e.g. WCTYPE|wt.doc.WTDocument or wt.doc.WTDocument.Supports only one type at a time.**	query	string
queryLimit		The number of results which the query should return	query	integer

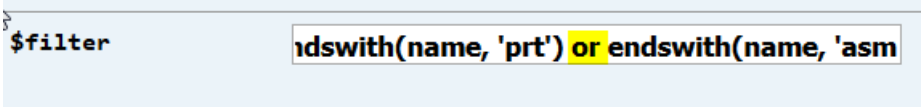
Here, you can create Rest query with the elements *$filter*, *$select*, *typeId* and *queryLimit*.

You can make the required input without having to replace characters unsuitable for an URL.

Filter example:

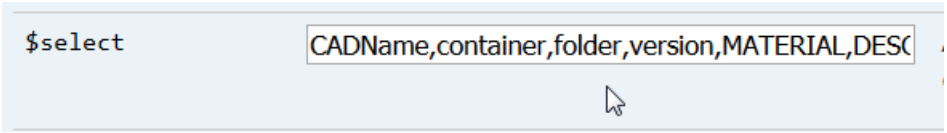


File types are linked with OR, use AND for other data items such as the lifecycle status.

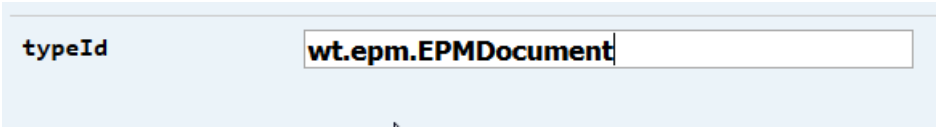


© 2024 INNEO Solutions GmbH

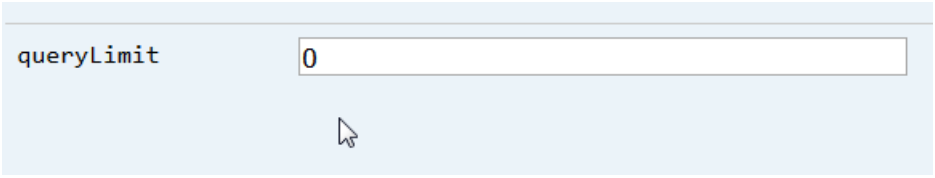
When you have defined the filter, select the required attributes. The attributes are listed separated by commas. You can use the example from the file *main.cfg*.



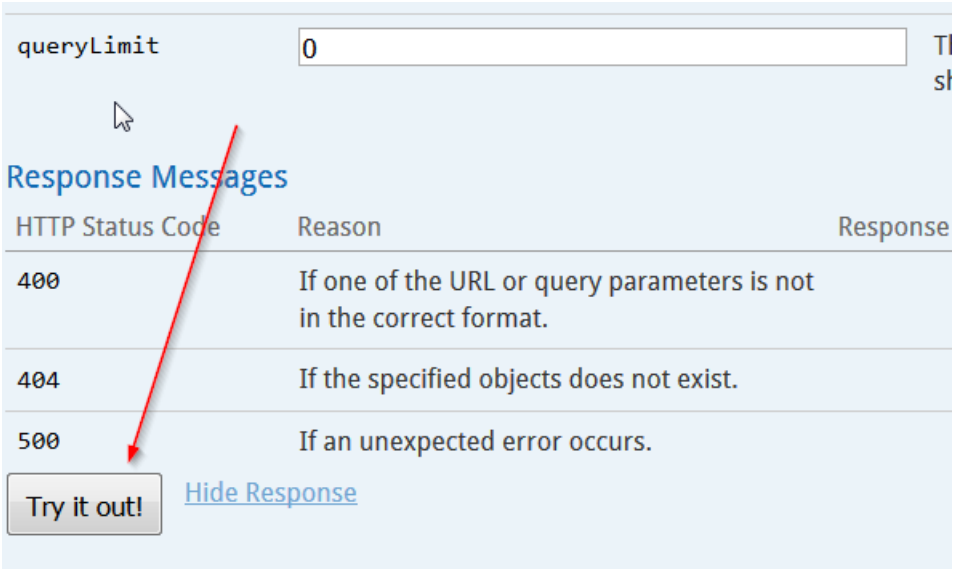
Finally, go to *typeID* and specify an iterateable class that the objects belong to, typically *wt.epm.EPMDocument*.



Enter any value for the *queryLimit* to make sure that the item *&queryLimit=* is added to the URL. You then have to remove the numerical value from the URL.



Click *Try it out*.



The system processed the defined query, which can take some time depending on the amount of data.

The output is displayed under *Response Body*.

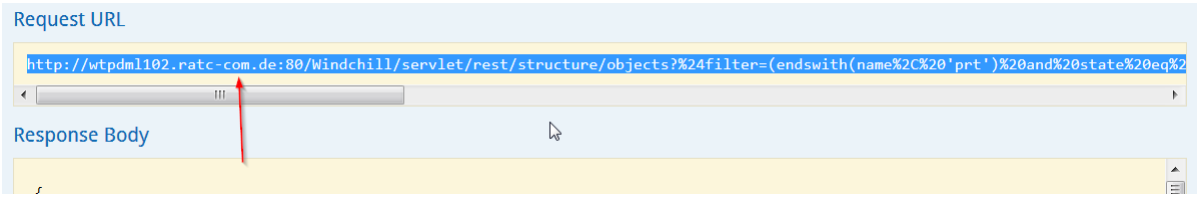
© 2024 INNEO Solutions GmbH

Response Body

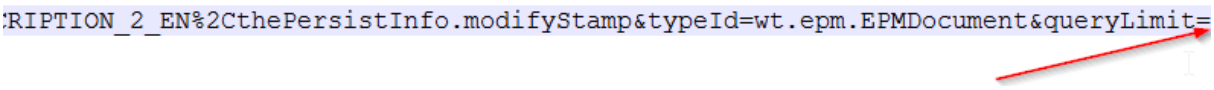

```
{
  "items": [
    {
      "id": "OR:wt.epm.EPMDocument:30758899",
      "typeId": "WCTYPE|wt.epm.EPMDocument|de.ratc_com.DefaultEPMDocument",
      "attributes": {
        "container": "Maschinenbau",
        "folder": "/Default/CAD-Daten",
        "version": "A.5",
        "CADName": "200_203_traegerplatte.prt",
        "MATERIAL": "S235JR",
        "thePersistInfo.modifyStamp": "2018-10-18T10:53:17Z"
      }
    },
    {
      "id": "OR:wt.epm.EPMDocument:29758012",
      "typeId": "WCTYPE|wt.epm.EPMDocument|de.ratc_com.DefaultEPMDocument",
      "attributes": {
        "container": "Maschinenbau",
```


Verify whether the response contains all the data that you want to transfer to GENIUS TOOLS Library.

Copy the generated query URL from the field *Request URL* to the configuration file.



Remember to remove the value for *queryLimit*.



© 2024 INNEO Solutions GmbH

5 Import from other systems

Besides transferring data from Windchill, you can also transfer data from other systems to GENIUS TOOLS Library.

The procedure for creating the import files depends on the source system. Please refer to the documentation for your data source.

GENIUS TOOLS Library Data Importer can handle two types of import files from other systems:

– CSV (with or without a header row), see Import via CSV 32

– XML, see Import via XML 36

Examples or templates for all required rule files can be found under *GT_Library_DataImporter\software\conf*.

Examples for import files to be processed can be found under *GT_Library_DataImporter\software\PollingExamples*.

Other file formats

If you need to process other file formats, such as JSON, that do not originate from Windchill, it is recommended to transform the files into XML files that can then be read by Library Data Importer with the help of a corresponding XSLT file.

Starting the program

When you have prepared all necessary helper files, start Library Data Importer by running *DataImporter.exe*. You can use the file *task.cmd* to pass start parameters.

Library Data Importer runs without displaying a graphical user interface. After the program has terminated, you can check the results in the database of the target library in GENIUS TOOLS Library and in the log files.

5.1 Import via CSV

Many systems can export data records into CSV files.

Depending on the system, the different objects are listed line-by-line in the CSV file, with the attributes separated by commas or semicolons.

Depending on the system, each line corresponds to an object or a separate file is created for each object. Library Data Importer can handle both types of import data.

Also, two different CSV formats are accepted:

© 2024 INNEO Solutions GmbH

– with header

– without header

The header row lists the names of the attributes. The attribute values follow in the other rows, keeping to the same sequence as the header row.

If there is no header row, only the attribute values are listed with the same sequence in each row.

In both cases, each attribute name or value is delimited by the separator character, that is, a comma, semicolon or other character.

In the simplest case, the CSV file contains only the values to be imported. However, in some cases full records are exported, including attributes that are not meant to be transferred to GENIUS TOOLS Library.

In any case, the rule file ensures that the relevant data is imported.

If individual columns in the CSV file are not required, they can be left out in the rule file.

Empty values are marked in the CSV file by multiple separator characters directly following each other.

If the separator character occurs within the value for a column, the entire value has to be escaped by enclosing it in straight quotation marks.

The CSV files to be processed have to be saved to the *polling* directory. They have to have the file name extension **.csv*.

The rule file has to be placed in the *conf* directory, or as specified in the configuration option `gt_dataimporter_csv_rule_file`.

5.1.1 Rule files

Rule files are TXT files that tell Library Data Importer where in the CSV files to look for which attribute values.

Example rule files can be found under *conf* as

– *csv_rule_header* (for CSV with header)

– *csv_rule_headerless* (for CSV without header)

The rule files contain the names of the attributes, followed by = and the position in the CSV file.

Here is an example line for CSV with header:

```
name = Description
```

This means that the value for *name* can be found in each row at the position in which the label *Description* is placed in the header row.

© 2024 INNEO Solutions GmbH

Without header, the position is given as a numerical value:

```
name = 0
```

This means that the value for *name* can be found in each row at the first position. The first position is addressed by the value *0*.

Refer to the example rule files to find the attribute labels for GENIUS TOOLS Library, such as *name*, *path*, *objType*, *title_en*, *title_de* and *info*. Additional parameters or dimensions can be passed with the prefix *P:* or *D:*

5.1.2 CSV configuration options

This section describes configuration options that have to be set for importing data from CSV files.

For more information on configuration options, please also refer to Configuration file[9] .

Configuration option	Example
gtl_dataimporter_csv_rule_file =	..\conf\csv_rule_header.txt
gtl_dataimporter_csv_header_bool =	1
gtl_dataimporter_csv_separator_char =	,
gtl_dataimporter_db_path =	Drive: \Folder_Structure\gt_resource_folder\library \GTL_db.db
gtl_dataimporter_db_thumbnail_path =*	Drive: \Folder_Structure\gt_resource_folder\library \GTL_db\
gtl_dataimporter_change_parameter =*	CHANGEDATE

*optional

5.1.3 CSV with and without header

For an example CSV import file with a header, refer to *PollingExamples/csvHeaderKommaBild.csv*. The corresponding rule file in the *conf* directory is *csv_rule_header.txt*.

© 2024 INNEO Solutions GmbH

```
name = bezeichnung
path = pfade
objType = objektTypen
thumbnail = bild
status = status
title_en = titel_en
title_de = titel_de
P:MASSE = P1
P:MATERIAL = P2
P:REVISION = P3
D:LAENGE = D1
D:BREITE = D2
D:HOEHE = D3
info = info
P:CHANGEDATE = P4
```

Rule file for CSV file with header row

```
bezeichnung,pfade,objektTypen,status,titel_en,titel_de,P1,P2,P3,D1,D2,D3,info,bild,P4
06screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,06screwU,06SchraubeU,5g,Stahl allg.,V.06,50mm,5mm,8mm,"http://www.lmgtfy.com",,03.10.2018 06:34:03
07screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,07screwU,07SchraubeU,5g,Stahl allg.,V.07,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
08screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,08screwU,08SchraubeU,5g,Stahl allg.,V.08,50mm,5mm,8mm,"http://www.lmgtfy.com",,31.09.2018 06:35:03
09screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,09screwU,09SchraubeU,5g,Stahl allg.,V.09,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
10screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,10screwU,10SchraubeU,5g,Stahl allg.,V.10,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
11screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,10screwU,10SchraubeU,5g,Stahl allg.,V.10,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
13screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,10screwU,10SchraubeU,5g,Stahl allg.,V.10,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
14screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,10screwU,10SchraubeU,5g,Stahl allg.,V.10,50mm,5mm,8mm,"http://www.lmgtfy.com",/9j/4AAQSkZJRgABAQEASABIAAD/
```

CSV with header row

For an example CSV import file without a header, refer to *PollingExamples/csvOhneHeaderKommaBild.csv*. The corresponding rule file in the *conf* directory is *csv_rule_headerless.txt*.

```
name = 0
path = 1
objType = 2
status = 3
title_en = 4
title_de = 5
P:MASSE = 6
P:MATERIAL = 7
P:REVISION = 8
D:LAENGE = 9
D:BREITE = 10
D:HOEHE = 11
info = 12
thumbnail = 13
P:CHANGEDATE = 14
```

Rule file for CSV file without header row

```
ohne01screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,01screwU,01SchraubeU,5g,Stahl allg.,V.01,50mm,5mm,8mm,"http://www.l
ohne02screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,02screwU,02SchraubeU,5g,Stahl allg.,V.02,50mm,5mm,8mm,"http://www.l
ohne03screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,03screwU,02SchraubeU,5g,Stahl allg.,V.03,50mm,5mm,8mm,"http://www.l
ohne04screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,04screwU,04SchraubeU,5g,Stahl allg.,V.04,50mm,5mm,8mm,"http://www.l
ohne05screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,05screwU,02SchraubeU,5g,Stahl allg.,V.05,50mm,5mm,8mm,"http://www.l
ohne06screwU.prt,"%GT_RESOURCE_FOLDER%data\GTL_Test_DB\",10,2,05screwU,02SchraubeU,5g,Stahl allg.,V.05,50mm,5mm,8mm,"http://www.l
```

CSV without header row

© 2024 INNEO Solutions GmbH

5.2 Import via XML

Many systems can export data records into XML files.

The XML files can then be prepared for Library Data Importer using an appropriate XSLT file.

The exact XML structure depends on the source system.

The *PollingExamples* directory contains an example XML file (*xmlBsp.xml*), and the *conf* directory contains an example XSLT file (*xml-rule.xslt*) that matches it.

5.2.1 XML configuration options

This section describes configuration options that have to be set for importing data from XML files.

For more information on configuration options, please also refer to Configuration file[9] .

Configuration option	Example
gtl_dataimporter_xml_rule_file =	..\conf\xml_rule.xslt
gtl_dataimporter_db_path =	Drive: \Folder_Structure\gt_resource_folder\library \GTL_db.db
gtl_dataimporter_db_thumbnail_path =*	Drive: \Folder_Structure\gt_resource_folder\library \GTL_db\
gtl_dataimporter_change_parameter =*	CHANGEDATE

*optional

5.2.2 XML files

XML files are very freely configurable in their structure. For general information on the XML format, you can refer to *https://wiki.selfhtml.org*.

Basically, the object data contained in the XML file wraps nodes or subnodes for each attribute, which in turn contain the values to be set.

Ideally, XML nodes are available for the object name, path, type, status and other general information. Further data items can be passed as parameters and dimensions. To create an

© 2024 INNEO Solutions GmbH

object title in different languages, use nodes that pass *title* attributes with attached language codes.

The exact node structure is defined in the source application.

Thumbnail images can be passed as text values in the Base64 format.

5.2.3 XSLT files

For general information on XSLT rule files for Library Data Importer, please refer to Adapting the Rest XSLT file[21].

Format specifications for XSLT files:

– All nodes have to have a corresponding `AttributeName=` assignment in the XSLT file that uses the value of one or more adequate XML nodes.

– The assignment instructions for each attribute, each parameter and each dimension have to end in `<xsl:text>
</xsl:text>` This instruction adds a minus sign (-) and a line break (Enter) at the end of each object so that Library Data Importer can recognize the end of an object during processing.

– If more than one object is contained in a file, the XSLT file should wrap the nodes belonging to one object in a for-each command, e.g.;

```
<xsl:for-each select="root/element/items"> </xsl:for-each>
```

– Language-specific titles can be passed using `title_<LanguageCode>=`, e.g.:

```
<xsl:if test="title_en and normalize-space(title_en)!=''">
        <xsl:text>title_en=</xsl:text>
        <xsl:value-of select="normalize-space(title_en)"/>
      <xsl:text>&#xa;</xsl:text>
</xsl:if>
```

Examples

However the nodes in the XML file are named, the XSLT file is used to process them for Library Data Importer.

Check the example file *xml-rule.xslt* to see how nodes are selected and how their content is assigned to a label that GENIUS TOOLS Library can understand.

Example: Name

```
<xsl:if test="name and normalize-space(name)!=''">
        <xsl:text>name=</xsl:text>
        <xsl:value-of select="normalize-space(name)"/>
      <xsl:text>&#xa;</xsl:text>
      </xsl:if>
```

This example looks for nodes called *name* and assigns their value to the label *name=* An enter instruction is then added.

© 2024 INNEO Solutions GmbH

The equal sign = after the label and the enter instruction are both mandatory for Library Data Importer to assign the values correctly and to recognize the end of an object and the beginning of the next.

Example: Parameter

```
<xsl:if test="P[@key='MASSE'] and normalize-space(P[@key='MASSE'])!=''">
        <xsl:text>P:MASS=</xsl:text>
        <xsl:value-of select="normalize-space(P[@key='MASSE'])"/>
      <xsl:text>&#xa;</xsl:text>
      </xsl:if>
```

The example reads a parameter and checks which part of the XML file is defined as the key for the parameter within the node name. The corresponding value from the XML file is assigned to the label *P:MASS*= An enter instruction is then added.

The colon in the *P:* prefix is mandatory for Library Data Importer to process the value as a parameter for GENIUS TOOLS Library. The prefix for dimensions is *D:*

The XML node name given as the key does not have to match the name of the GENIUS TOOLS Library parameter. In the example, the key for the parameter *P:MASS*= could also be defined as `P[@key='WEIGHT']` as long as the node is called like this in the XML file to be processed.

© 2024 INNEO Solutions GmbH

6 Copyright

Copyright 2024 by:

INNEO Solutions GmbH

IT-Campus 1

73479 Ellwangen

Germany

This documentation is protected by copyright. All rights reserved.

Without prior written consent of an authorized representative of INNEO Solutions GmbH it must not be copied, photocopied, reproduced, translated, communicated or converted to electronic or machine readable form in whole or in part.

The unauthorized use of the documentation can lead to a claim for liquidated damages or legal prosecution. INNEO Solutions GmbH does not accept liability for possible faulty information in this documentation and the consequences resulting from such.

Note on registered trademarks:
Most of the software, hardware and trade names mentioned in this documentation are also registered trademarks of the respective software manufacturers.

Registered trademarks and trade names of INNEO Solutions GmbH:
GENIUS TOOLS, Startup TOOLS, INNEO