TRANSLATE for CATIA V5 - Creo

USER GUIDE

Release Version: 25.2
Revision: 1.0
Issued: 06/02/2023

© THEOREM SOLUTIONS 2023

Contents

©Theorem Solutions 2023

Overview of CADverter

About Theorem

Theorem Solutions is a world leader in the field of Engineering Data Services and Solutions. This leadership position stems from the quality of our technology and the people in the company. Quality comes not only from the skills and commitment of our staff, but also from the vigorous industrial use of our technology & services by world leading customers.

We are proud that the vast majority of the world's leading Automotive, Aerospace, Defense, Power Generation and Transportation companies and their Supply chains use our products and services daily. Working closely with our customers, to both fully understand their requirements and feed their input into our development processes has significantly contributed to our technology and industry knowledge.

Theorem Solutions is an independent UK headquartered company incorporated in 1990, with sales and support offices in the UK and USA. Theorem has strong relationships with the major CAD and PLM vendors, including; Autodesk, Dassault Systemes, ICEM Technologies (a Dassault company), PTC, SolidWorks, Spatial Technology and Siemens PLM Software. These relationships enable us to deliver best in class services and solutions to engineering companies worldwide.

©Theorem Solutions 2023

Theorem's Product Suite

Theorem have 3 main Product brands. These are:

TRANSLATE

Direct translation of 3D data to or from an alternate CAD, Visualization or Standards Based format.

See our website for more detail.

PUBLISH

The creation of documents enriched with 3D content

See our website for more detail.

VISUALIZE

Visualization for Augmented (AR), Mixed (MR) and Virtual (VR) Reality applications

See our website for more detail.

The Creo Bi-directional CATIA V5 CADverter

The Creo to CATIA V5 CADverter is a direct database converter between Creo and CATIA V5. It enables the user to convert all forms of mechanical design geometry, as well as assembly and attribute information between these two systems.

CADverter can be purchased as a uni-directional, Creo to CATIA V5, or CATIA V5 to Creo product, or as a bi-directional product.

The translator can be invoked in batch mode with the command line interface allowing the conversion process to be integrated into any process oriented operation. Alternatively the conversion process may be operated by using the Theorem Unified Interface.

Primary Product Features

- CADverter converts all geometry
- If assembly data (product structure) is in the file, the assembly structure will be mapped between the two systems as well as colour information
- The user can filter data to optimize the process
- If you wish to visualise and interrogate the CATIA V5 or Creo data this can be done by using the integrated User Interface, which is included with CADverter
- The conversion process can be in Batch Mode or using the Unified Interface
- Command line interface allows process integration

Primary Product benefits?

- Direct conversion between Creo and CATIA V5 reduces processing time, simplifies integration and retains accuracy of the model
- The integrated viewing capability enables visual verification, pre and post translation
- The integrated data filtering options allows selected data ONLY to be processed, enabling optimisation of translations and time savings
- By converting all forms of geometry no data is lost, eliminating the time required to recreate missing data
- With over 20 years industrial use Theorem's product robustness and quality is well proven, reducing your business risk

This document will focus specifically on guidance for the use of the CADverter for CATIA V5 - Creo product. For information regarding any of Theorem's product ranges please contact sales@theorem.com

Getting Started

Documentation

The latest copy of the User Guide documentation can be found on our web site at:

http://www.theorem.com/Documentation

Each product has a specific link that provides user documentation in the form of PDF and Tutorials.

The latest copy of Theorem software can be found via the link above and by searching for the specific product. Each product has a specific link to the Product Release Document, which contains a link to the download location of the installation CD.

Alternatively, you can request a copy of the software to be shipped on a physical CD.

Installation

The installation is run from the .msi file download provided. For full details of the installation process, visit www.theorem.com/documentation and select Creo <> NX from the product selection list.

License Configuration

To run any product a valid license file is required. The Flex License Manager is run from the .msi file download provided. For full details of the installation process, visit www.theorem.com/documentation

Using the Product

To use the product, follow the documented steps found in this document or follow the online video tutorials which can be found from www.theorem.com/documentation

©Theorem Solutions 2023

Using the Product

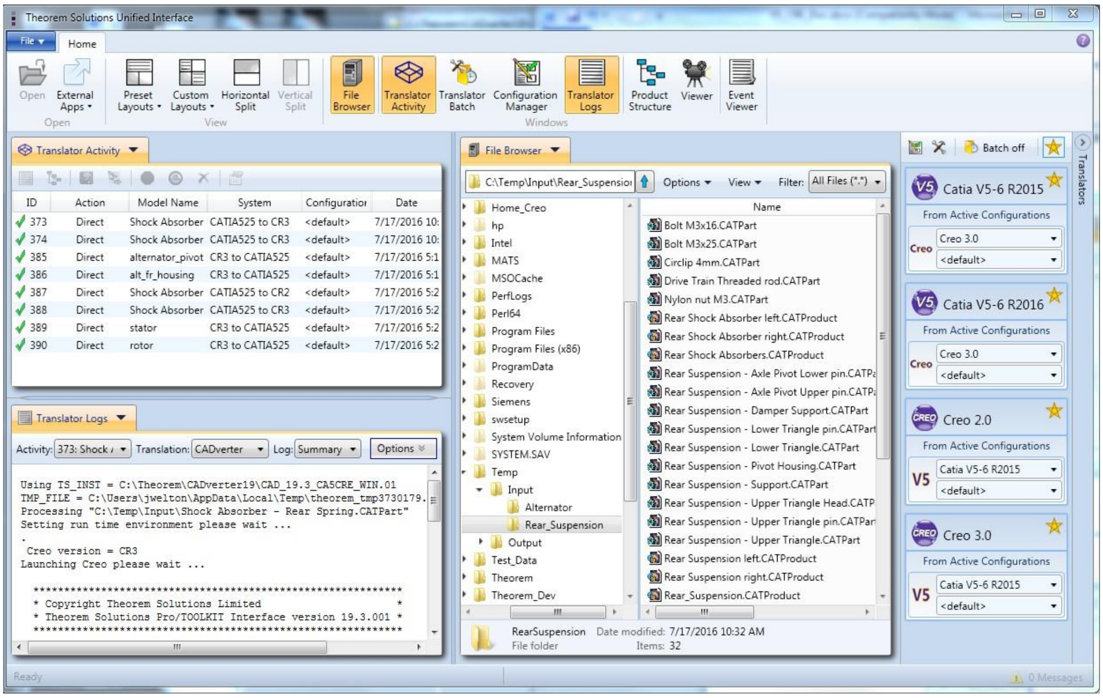
Default Translation – via the Unified Interface

The Unified Interface can be started via the Start Menu – if a shortcut was added during installation.

Alternatively, the Unified Interface can be run via a Windows Explorer selection in:

<UI_installation_directory>\bin\Unified_Interface.cmd

The following interface will be launched:



ID	Action	Model Name	System	Configuratior	Date
✓ 373	Direct	Shock Absorber	CATIA525 to CR3	<default>	7/17/2016 10:
✓ 374	Direct	Shock Absorber	CATIA525 to CR3	<default>	7/17/2016 10:
✓ 385	Direct	alternator_pivot	CR3 to CATIA525	<default>	7/17/2016 5:1
✓ 386	Direct	alt_fr_housing	CR3 to CATIA525	<default>	7/17/2016 5:1
✓ 387	Direct	Shock Absorber	CATIA525 to CR2	<default>	7/17/2016 5:2
✓ 388	Direct	Shock Absorber	CATIA525 to CR3	<default>	7/17/2016 5:2
✓ 389	Direct	stator	CR3 to CATIA525	<default>	7/17/2016 5:2
✓ 390	Direct	rotor	CR3 to CATIA525	<default>	7/17/2016 5:2

Activity: 373: Shock / Translation: CADverter Log: Summary

```
Using TS_INST = C:\Theorem\CADverter19\CAD_19.3_CA5CRE_WIN.01
TMP_FILE = C:\Users\jwelton\AppData\Local\Temp\theorem_tmp3730179.
Processing "C:\Temp\Input\Shock Absorber - Rear Spring.CATPart"
Setting run time environment please wait ...
.
 Creo version = CR3
Launching Creo please wait ...

  ***************************************************************
  * Copyright Theorem Solutions Limited                         *
  * Theorem Solutions Pro/TOOLKIT Interface version 19.3.001 *
  ***************************************************************
```

C:\Temp\Input\Rear_Suspensior Filter: All Files (*.*)

- Home_Creo
- hp
- Intel
- MATS
- MSOCache
- PerfLogs
- Perl64
- Program Files
- Program Files (x86)
- ProgramData
- Recovery
- Siemens
- swsetup
- System Volume Information
- SYSTEM.SAV
- Temp
  - Input
    - Alternator
    - Rear_Suspension
  - Output
- Test_Data
- Theorem
- Theorem_Dev

Name

- Bolt M3x16.CATPart
- Bolt M3x25.CATPart
- Circlip 4mm.CATPart
- Drive Train Threaded rod.CATPart
- Nylon nut M3.CATPart
- Rear Shock Absorber left.CATProduct
- Rear Shock Absorber right.CATProduct
- Rear Shock Absorbers.CATProduct
- Rear Suspension - Axle Pivot Lower pin.CATPa
- Rear Suspension - Axle Pivot Upper pin.CATPa
- Rear Suspension - Damper Support.CATPart
- Rear Suspension - Lower Triangle pin.CATPart
- Rear Suspension - Lower Triangle.CATPart
- Rear Suspension - Pivot Housing.CATPart
- Rear Suspension - Support.CATPart
- Rear Suspension - Upper Triangle Head.CATP
- Rear Suspension - Upper Triangle pin.CATPar
- Rear Suspension - Upper Triangle.CATPart
- Rear Suspension left.CATProduct
- Rear Suspension right.CATProduct
- Rear_Suspension.CATProduct

RearSuspension Date modified: 7/17/2016 10:32 AM
File folder Items: 32

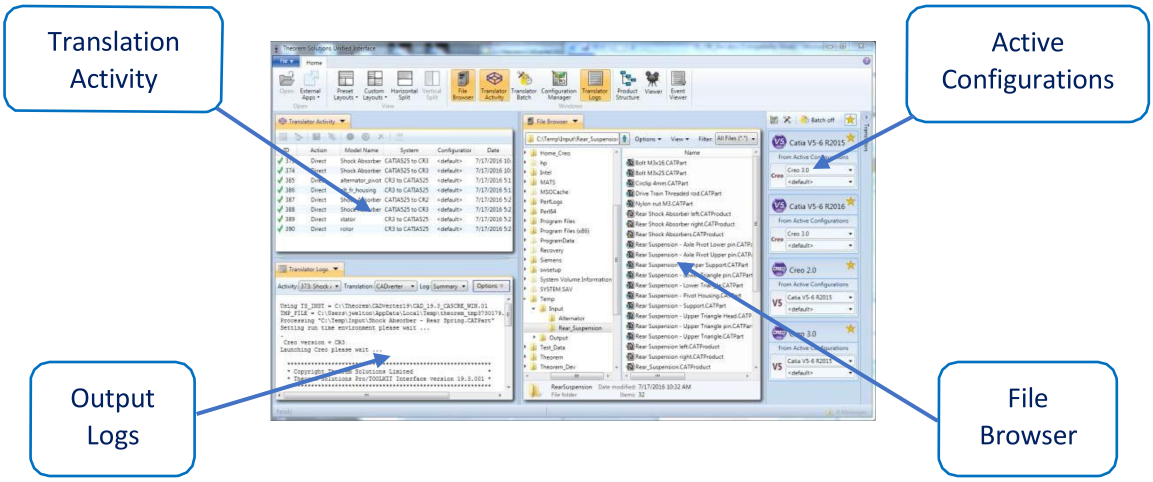
Catia V5-6 R2015
From Active Configurations
Creo Creo 3.0
<default>

Catia V5-6 R2016
From Active Configurations
Creo Creo 3.0
<default>

Creo 2.0
From Active Configurations
V5 Catia V5-6 R2015
<default>

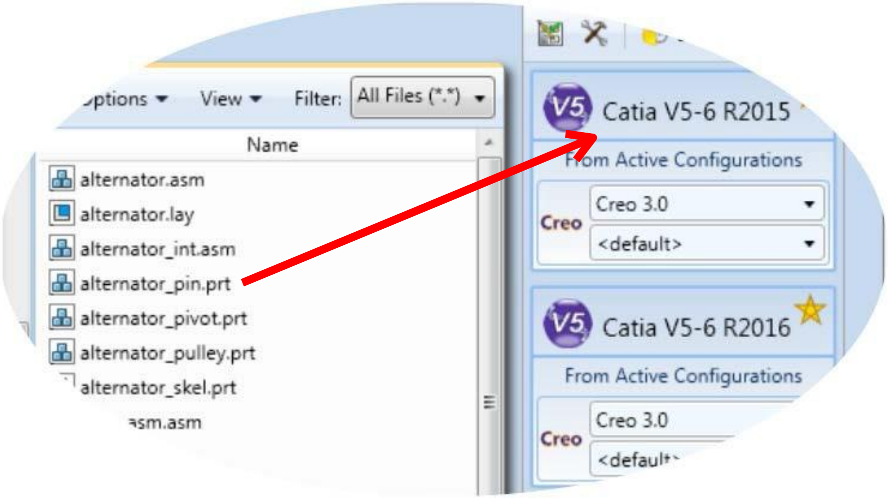
Creo 3.0
From Active Configurations
V5 Catia V5-6 R2015
<default>

The default layout is split into 4 primary areas, which can be altered to the users prefer:



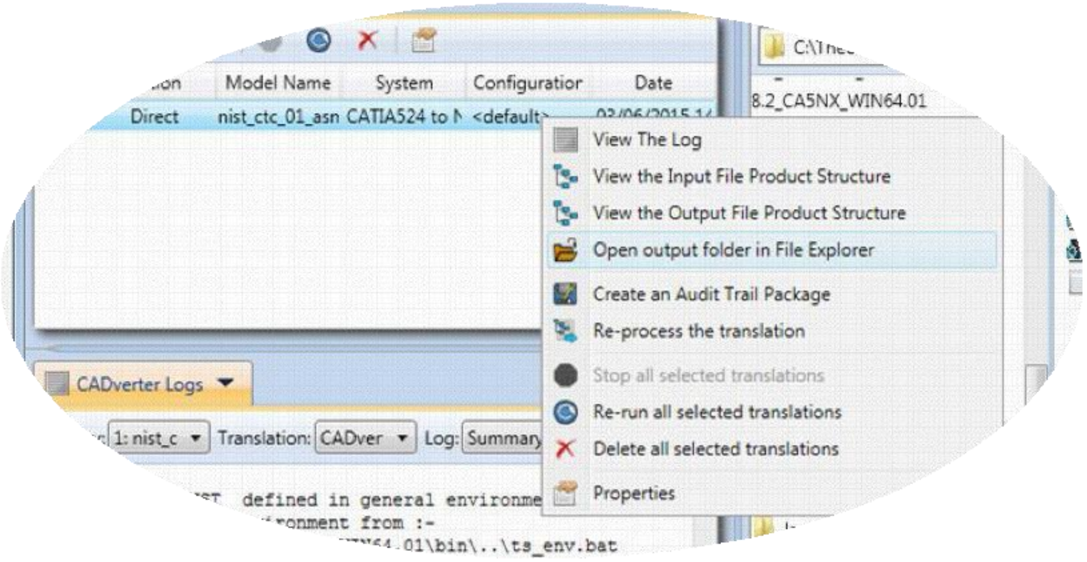
The simplest way to translate from CATIA V5 or CREO is to drag a file from the file Browser Pane on to the Active Configurations for the translation you require.

©Theorem Solutions 2023



On completion, the Unified Interface will display the activity information and details from the log file created during the translation, if requested, in the Translation Activity and Output Log panes, respectively.

The generated output data can be located by selecting the translation from the Activity pane and opening the output folder:


Model Name
System
Configuration
Date
Direct
nist_ctc_01_asm
CATIA524 to N
<default>
02/06/2015 14
C:\The
8.2_CA5NX_WIN64.01
View The Log
View the Input File Product Structure
View the Output File Product Structure
Open output folder in File Explorer
Create an Audit Trail Package
Re-process the translation
Stop all selected translations
Re-run all selected translations
Delete all selected translations
Properties
CADverter Logs
1: nist_c
Translation:
CADver
Log:
Summary

```
ST  defined in general environme
 vironment from :-
 WIN64.01\bin\..\ts_env.bat
```

Default Translation – via the Command Line

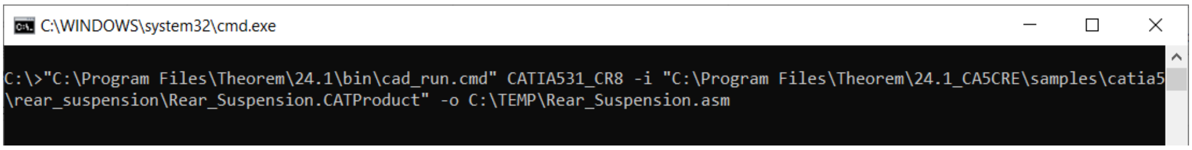
Running a translation via the command line can be carried out via the ***cad_run.cmd*** file located in the ***<installation_directory>\bin*** directory. The format of the command is as follows when translating from CATIA V5 to CREO:

<Translator_installation_directory>\bin\cad_run.cmd CATIA5[XX]_CR[Y] <input_file> <output_file>

The format of the command is as follows when translating from CREO to CATIA V5:

<Translator_installation_directory>\bin\cad_run.cmd CR[Y]_CATIA5[XX] <input_file> <output_file>

Note! Replace the [XX] seen in the example with the version of CATIA V5 you are using and replace the [Y] with the version of Creo you are using. E.g. for CATIA V5-6R2020, change to CATIA530 and for CREO 6 change to CR6:



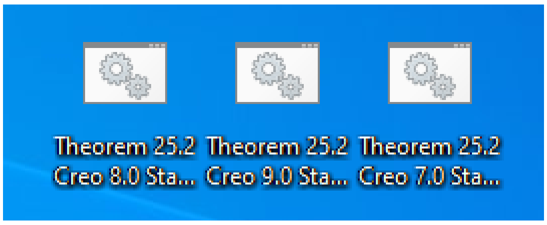
```
C:\>"C:\Program Files\Theorem\24.1\bin\cad_run.cmd" CATIA531_CR8 -i "C:\Program Files\Theorem\24.1_CA5CRE\samples\catia5
\rear_suspension\Rear_Suspension.CATProduct" -o C:\TEMP\Rear_Suspension.asm
```

The example above will translate a CATIA V5 sample file provided within the installation and produce the following output to the target location. In this case:

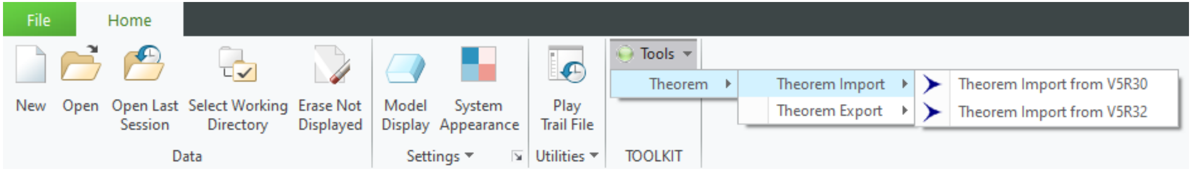
C:\Temp\Rear_Suspension.asm

Default Translation – via the Interactive method

It is possible to run translations via the interactive method for V5<>Creo using the Theorem menu within Creo. To do this launch the start up commands which will be on your desktop or install directory.


Theorem 25.2 Creo 8.0 Sta...
Theorem 25.2 Creo 9.0 Sta...
Theorem 25.2 Creo 7.0 Sta...

The next step would be once this is open within the main ribbon under Tools you will be able to Import or export from and to V5.



CADverter Customization

CADverter allows the information that is read from the source system and written to the target system to be tailored via a set of user specified arguments. Commonly used arguments are supported via the Unified Interface, with Advanced Arguments being described within this document for use in the Unified Interface or via the Command Line invocation.

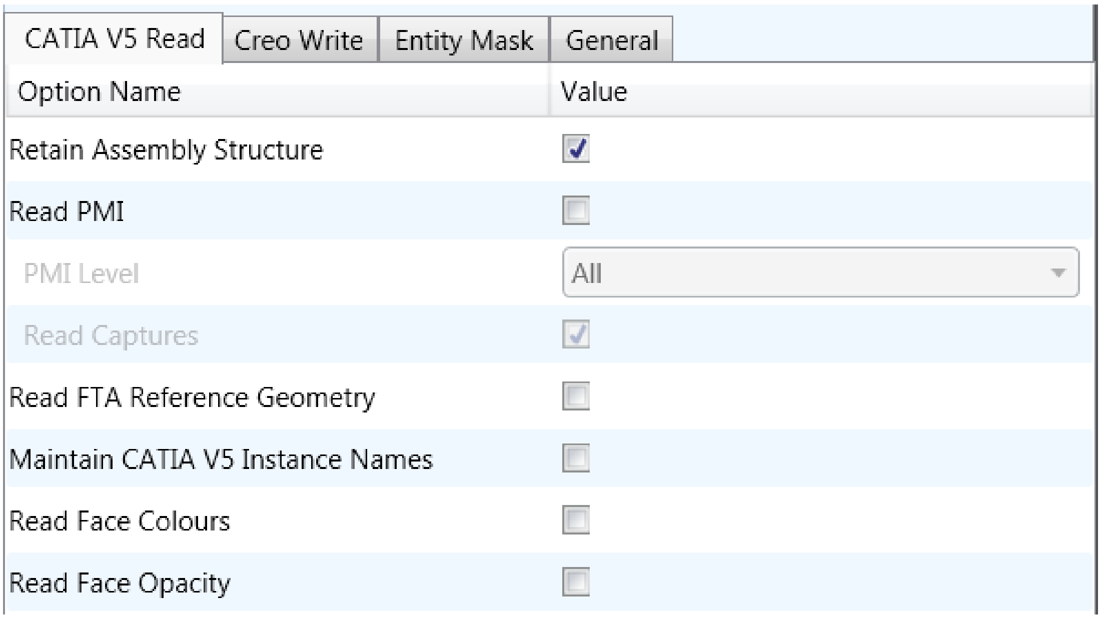
Common Options for CATIA V5 to CREO

Within the Configuration Manager pane of the Unified Interface, arguments that can be specified when publishing CATIA V5 data into CREO are grouped into 4 areas:

- CATIA V5 Read – Those arguments that affect how data is read from CATIA V5
- CREO Write – Those arguments that affect how the data is written to CREO
- Entity Mask – Those arguments that allow specific read entities to be masked
- General – Those arguments that are common to ALL Publishing activities regardless of source data

CATIA V5 Read Arguments

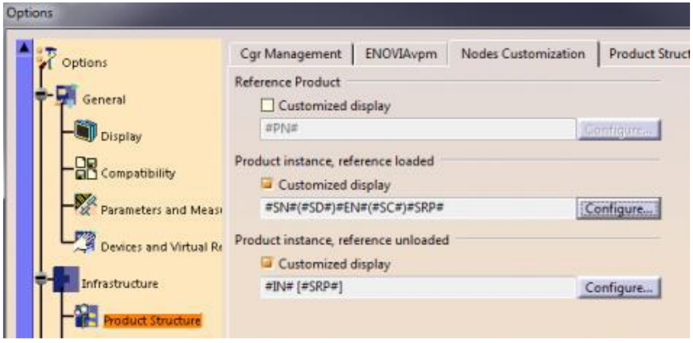
The image below shows the CATIA V5 Read arguments that are available, with their default settings:



Option Name	Value
Retain Assembly Structure	☑
Read PMI	□
PMI Level	All
Read Captures	☑
Read FTA Reference Geometry	□
Maintain CATIA V5 Instance Names	□
Read Face Colours	□
Read Face Opacity	□

Each of these options is described below:

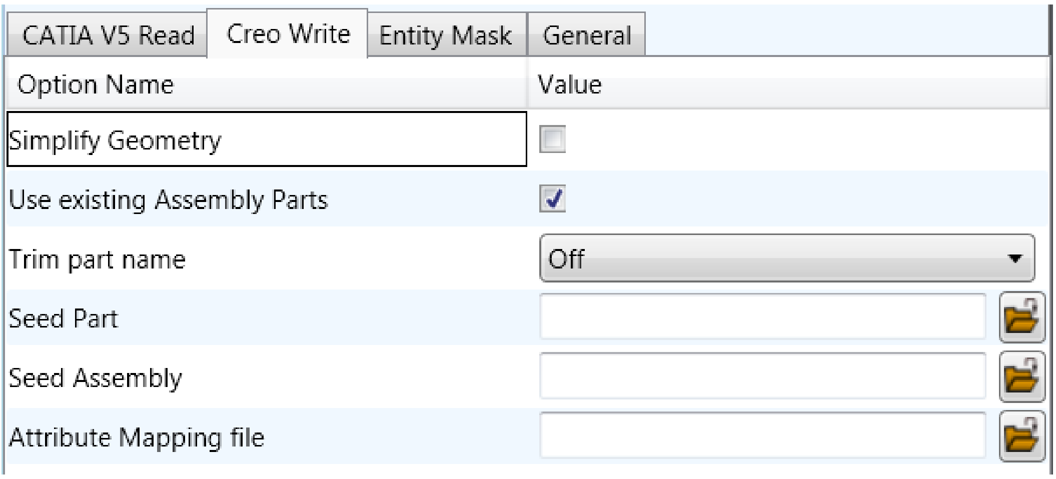
Option	Description
Retain Assembly Structure	Maintain assembly structure in target file *(Default is ON)* • Command Line Syntax: ○ *offditto (no structure, single part created)*

Read PMI	Enables PMI data read from the V5 file. *(Default is OFF)*. - Command Line Syntax: ○ *dont_read_pmi – to turn off*
PMI Level	A secondary argument to 'Read PMI' and allows control of the level of PMI to be read. *(Default is ALL when 'Read PMI' is marked as ON.)* - Options Available (command line syntax in italics and square brackets next to the option) ○ All - [read_pmi] ○ Part Level - [read_part_pmi] ○ Assembly Level - [read_assy_pmi] ○ Assembly Set (From CATPart) - [read_part_assy_pmi]
Read Captures	A secondary argument to 'Read PMI' and allows the control over whether captures are read as part of the process. *(Default is OFF)* when 'Read PMI' is marked as ON. - Command Line Syntax: ○ read_captures – to turn on ○ dont_read_captures – to turn off
Read FTA Reference Geometry	Enables reading of FTA Reference Geometry *(Default is OFF)* - Command Line Syntax: ○ read_geometry – to turn on
Maintain CATIA V5 Instance Names	Honours CATIA V5 *Tools->Options->Infrastructure->Product Structure->Nodes Customization* panel settings *(Default is OFF)*  Options Options General Display Compatibility Parameters and Meas Devices and Virtual R Infrastructure Product Structure Cgr Management ENOVIAvpm Nodes Customization Product Struct Configure... Configure... Configure... Reference Product □ Customized display #PN# Product instance, reference loaded ☑ Customized display #SN#(#SD#)#EN#(#SC#)#SRP# Product instance, reference unloaded ☑ Customized display #IN# [#SRP#] - Command Line Syntax: ○ ditto_naming v5 – to turn on
Read Face Colours	Read the face colours *(Default is Off)* - Command Line Syntax: ○ read_face_colours – to turn on
Read Face Opacity	Read the opactity of the face *(Default is Off)* - Command Line Syntax: ○ read_materials – to turn on

©Theorem Solutions 2023

CREO Write Arguments

The image below shows the Write CREO arguments that are available, with their default settings:



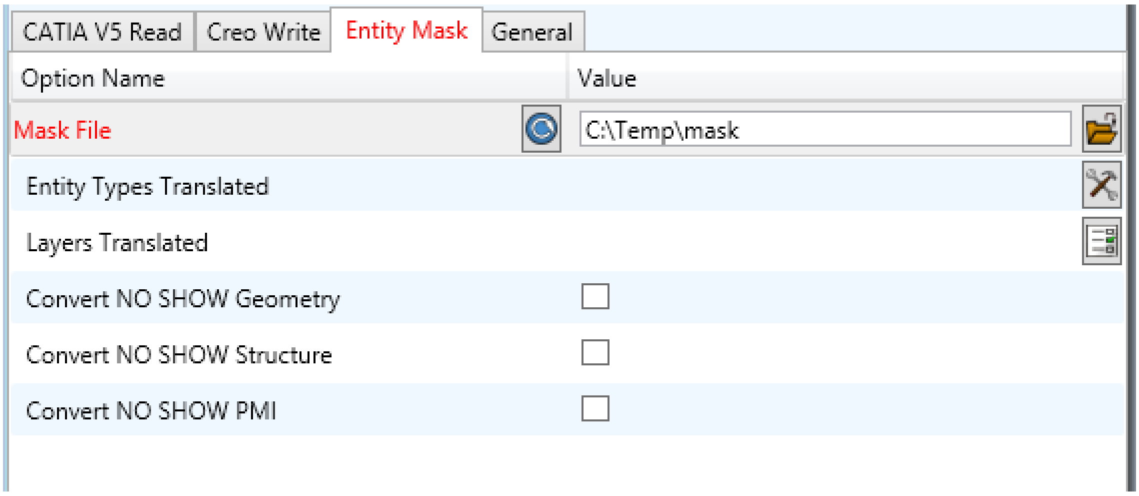
Option Name	Value
Simplify Geometry	□
Use existing Assembly Parts	☑
Trim part name	Off ▾
Seed Part	
Seed Assembly	
Attribute Mapping file	

Each of these options is described below:

Option	Description
Simplify Geometry	Attempt to write Analytical Geometry where possible *(Default is Off).* • Command Line Syntax ○ *simplify*
Use existing Assembly Parts	If the output Folder already contains output Files of the same name, do not Overwrite those Files *(Default is On).* • Command Line Syntax ○ *use_parts*
Trim part name	If the part name is too long, use the first X characters and the final (30 - X) characters to reach the 30 character limit. *(Default is off).* • Command Line Syntax ○ *chop_name X*
Seed Part	Use a Creo Seed Part file when creating the Creo output • Command Line Syntax ○ *seed_prt*
Seed Assembly	Use a Creo Seed Assembly file when creating the Creo output • Command Line Syntax ○ *seed_asm*
Attribute Mapping File	Select a standard property mapping file • Command Line Syntax ○ *cad_prop_map_file*

CATIA V5 to CREO Entity Masking Arguments

The image below shows the Masking arguments that are available, with their default settings:



Option Name	Value
Mask File	C:\Temp\mask
Entity Types Translated	
Layers Translated	
Convert NO SHOW Geometry	□
Convert NO SHOW Structure	□
Convert NO SHOW PMI	□

Each of these options is described below:

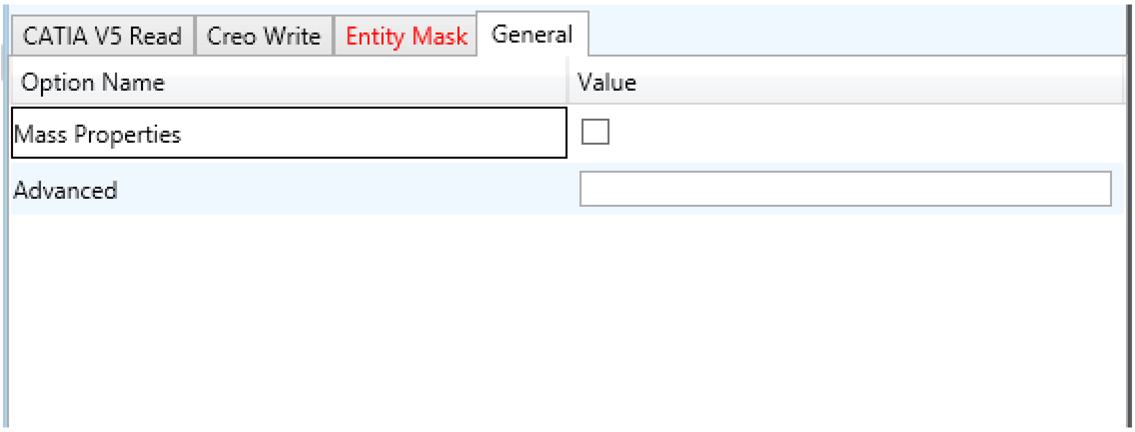
Option	Description
Mask File	Specifies the Mask File to be written to, that can be referenced by future translations. A Mask file MUST be specified if masking is required. The first line in this file is OFF ALL ENT: • Command Line Syntax: ○ *Mask <path\filename>*
Entity Types Translated	Specifies a selection list from which to select which entity types are to be processed. The following types are available: "POI","CUR",”SKI”,"SOL",”ISO”,"TEX","AXI" • Command Line Syntax: ○ *Add any of the above to the specified mask file, one entry per line prefixed by the word ON, e.g.:* ***ON POI*** *to ensure they are considered in the translation*
Layers Translated	Specifies a selection list from which to select which layers are to be processed. • Command Line Syntax: ○ *A single entry of* ***ON ALL LAY*** *Must precede any Layer Mask command.* ○ *Add a list or range of numbers representing layer to be processed to the specified mask file to ensure they are NOT considered in the translation* *e.g.:* ***OFF LAY 114,149,166,167,168***

©Theorem Solutions 2023

Convert NO SHOW Geometry	Enables Hidden geometry to be processed *(Default is Off)* • Command Line Syntax: ○ *Add the following entry to the Mask file* ***ON NOSHOW***
Convert NO SHOW Structure	Enables Hidden Assembly Structure to be processed *(Default is Off)* • Command Line Syntax: ○ *Add the following entry to the Mask file* ***ON NOSHOW STR***
Convert NO SHOW PMI	Enables Hidden PMI to be processed *(Default is Off)* • Command Line Syntax: ○ *Add the following entry to the Mask file* ***ON NOSHOW PMI***

CATIA V5 to CREO General Arguments

The image below shows the General arguments that are available, with their default settings:



Option Name	Value
Mass Properties	□
Advanced	

Each of these options is described below:

Option	Description
Mass Properties	Allows Mass Property information to be read from the source data and written as attributes to the PDF document. Default is OFF. ○ Command Line Syntax: ▪ *mprops*
Advanced	Allows any of the Command Line Advanced arguments documented below to be passed to the Unified Interface invocation

©Theorem Solutions 2023

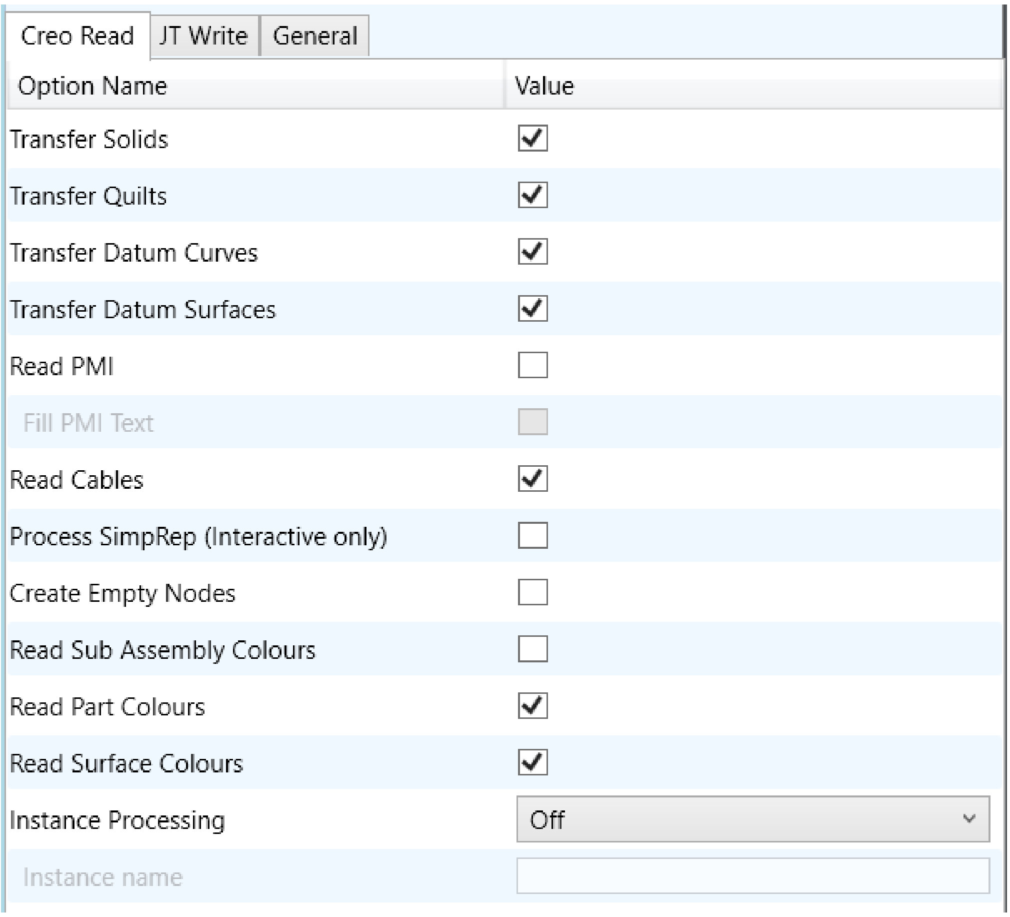
Common Options for CREO to CATIA V5

Within the Configuration Manager pane of the Unified Interface, arguments that can be specified when publishing CATIA V5 data into CREO are grouped into 4 areas:

- CREO Read – Those arguments that affect how data is read from CATIA V5
- CATIA V5 Write – Those arguments that affect how the data is written to CREO
- General – Those arguments that are common to ALL Publishing activities regardless of source data

CREO Read Arguments

The image below shows the CREO Read arguments that are available, with their default settings:



Creo Read | JT Write | General

Option Name	Value
Transfer Solids	☑
Transfer Quilts	☑
Transfer Datum Curves	☑
Transfer Datum Surfaces	☑
Read PMI	☐
Fill PMI Text	☐
Read Cables	☑
Process SimpRep (Interactive only)	☐
Create Empty Nodes	☐
Read Sub Assembly Colours	☐
Read Part Colours	☑
Read Surface Colours	☑
Instance Processing	Off
Instance name	

Each of these options is described below.

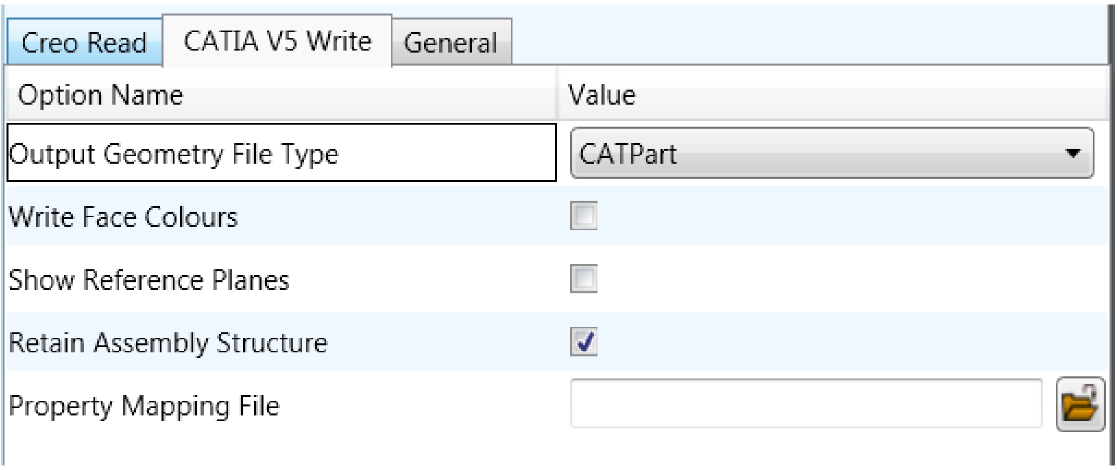
Option	Description
Transfer Solids	Enables solid processing. (Default is on). • Command Line Syntax: ○ no_solids – to Disable
Transfer Quilts	Enables quilt processing. (Default is on). • Command Line Syntax:

	○ no_quilts – to Disable
Transfer Datum Curves	Enables Datum Curve processing. (Default is on). • Command Line Syntax: ○ no_datum_curves – to Disable
Transfer Datum Surfaces	Enables Datum Surface processing. (Default is on). • Command Line Syntax: ○ no_datum_surfaces – to Disable
Read PMI	Enables Datum Surface processing. (Default is off). • Command Line Syntax: ○ read_pmi
Fill PMI Text	Enabled when 'Read PMI' is selected. Improves the quality of PMI, but increases output size and processing time. (Default is off) • Command Line Syntax: ○ fill_pmi_text
Read Cables	Enables the reading of Cable data from Creo. (Default is off). Note that in the Creo Configuration Editor, the setting display_thick_cables should be set to yes. • Command Line Syntax: ○ read_cables
Process Simprep	Enables the reading of a specified Simplified Representation. This is only available when processing data interactively and using the option via the Configuration Manager. • Command Line Syntax: ○ *process_simprep*
Create Empty Nodes	If a part is missing, create an empty leaf node for it. (Default is off) • Command Line Syntax: ○ create_empty_part
Read Sub Assembly Colours	Read colour information set on sub-assembly level (Default is off) • Command Line Syntax: ○ sub_assy_colours ○ sub_assy_colours_off (default)
Read Part Colours	Read colour information set on parts. (Default is on) • Command Line Syntax: ○ part_colours (default) ○ part_colours_off
Read Surface Colours	Read colour information set on surfaces. (Default is on) • Command Line Syntax: ○ surface_colours (default) ○ surface_colours_off
Instance Processing	Process a defined instance. Contains 3 options: (Not to be used in conjunction with Process Specified Instance). (Default is off). • Off • List all instances in log file - List all instances of Family table to the progress file. ○ *Command Line Syntax* ▪ *instance LIST_ALL* • *Process Specified Instance – Only process the specified instance. The Text Box Instance Name will become active.* ○ *Command Line Syntax*

	○ *instance [instance_name]*
Instance Name	Enter the instance name to process. *Only activates when 'Process Specified Instance' selected.*

CATIA V5 Write Arguments

The image below shows the CATIA V5 Write arguments that are available, with their default settings



Creo Read | CATIA V5 Write | General

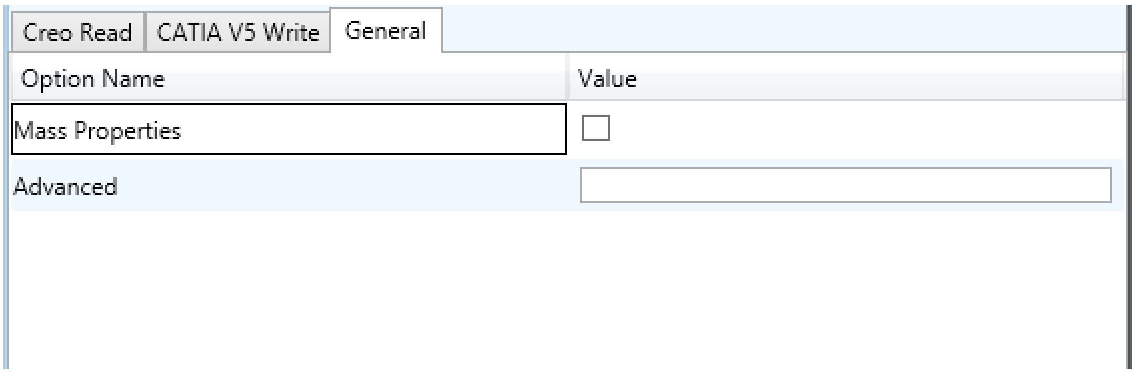
Option Name	Value
Output Geometry File Type	CATPart
Write Face Colours	☐
Show Reference Planes	☐
Retain Assembly Structure	☑
Property Mapping File	

Each of these options is described below:

Option	Description
Output Geometry File Type	Output Geometry file type *(Default is CATPart - a CATProduct will be created for an assembly)* • Command Line Syntax ○ *output_type CATPart (CATPart)* ○ *output_type model (Model)* ○ *output_type cgr (Cgr)* ○ *output_type igs (Igs)* ○ *output_type CATShape (CATShape)* ○ *create_cgr (Tessellated)*
Write Face Colours	Processes Colours as applied to individual faces *(Default is OFF)* • Command Line Syntax ○ *face_colour (defult)* ○ *solid_colour (use colour as applied to solids)*
Show Reference Planes	Creates reference planes. Default is OFF. • Command Line Syntax ○ Show: *dont_blank_planes*
Retain Assembly Structure	Maintain assembly structure in derive output *(Default is ON).* • Command Line Syntax ○ *off_ditto*
Property Mapping File	Assign a property mapping file • Command Line Syntax ○ *cad_prop_map_file [Path to file]*

CREO to CATIA V5 General Arguments

The image below shows the General arguments that are available, with their default settings:



Option Name	Value
Mass Properties	□
Advanced	

The option is described below:

Option	Description
Mass Properties	Allows Mass Property information to be read from the source data and written as attributes to the PDF document *(Default is off)* ○ Command Line Syntax: ▪ *mprops*
Advanced	Allows any of the Command Line Advanced arguments documented below to be passed to the Unified Interface invocation

©Theorem Solutions 2023

Command Line Advanced Arguments

Any of the Advanced arguments can be added to the Command Line Invocation or to the General > Advanced field when run from within the User Interface.

Creo Arguments

Creo Argument	Description
pmi_pcurves	Store non planar PMI graphics (leaders not in the plane of the annotation)
unique_occ	Read multiple occurrences, eg support for occurrence PMI associations
read_assy_pmi	reads PMI in lower level assembly parts
ignore_std_views	Disable reading views with standard names (TOP, LEFT, etc).
ignore_view_list <file>	supply a list of view names to be ignored
no_exploded_views	Disables exploded views.
part_level_views	*Default: **off*** Enable the processing of Part Level Views within an assembly.
part_level_views_moved	Process part level views and move into assembly space, such that only one part instance's views are displayed.
part_level_pmi	*Default: **off*** Enable the processing of PMI on parts within an assembly.
view_part_name	Uses the part name in the view names to help identify the views when selected in PDF
views_geom_exploded <on | off | hybrid | hybrid2>	*Default: **hybrid*** Geometry grouped into assembly nodes for views is by default done on a view bases, which re-uses geometry where possible. This option creates a node for every item of geometry so that the views can hide/show them in the views. This can (depending upon the data and views) reduce the resulting file size and shorten the translation time, BUT in some cases the PDF is slow to respond due to the VIEW limitations. *exploded - off* - create a node per view with all geometry / wire etc for that view *explode - on* - create nodes for very solid / wireframe etc that can be referenced by views *hybrid (default)* - mixture of explode on/off - solids being exploded and wireframe grouped (best compromise) *hybrind2* - as hybrid with points also grouped

CATIA V5 Arguments

CATIA V5 Arguments	Description
convert_curves	Converts curves to NURBS form
convert_surfaces	Converts surfaces to NURBS form
face_opacity	Read face opacity
no_face_colour	Sets the default to SOLID colours
noshow **noshow_geom** **noshow_struct** **noshow_pmi**	Reads hidden geometry / structure / pmi
output_mbd	Allows sub-part specification tree information to be read and presented to 3D PDF as product structure information. This option also enables the *‘part_level_views’* and *‘part_level_pmi’* 3D PDF options
Read_geometry_edges	CATIA V5 has a display mode that allows the display of shaded surfaces and edges. This option allows the translator to mimic this for FTA construction geometry by promoting the edge curves to standalone wireframe

Large Assembly Processing

Overview

Large Assembly Processing (LAP) is available via command line only. Theorem's LAP mechanism was designed to be a method to efficiently converting large assemblies into their target format.

This process differs from the standard CAD-to-CAD conversion method. The standard method opens the entire assembly into memory and converts the assembly into the target system. Whereas LAP employs a method to convert each assembly leaf node independently and then processes the top-level assembly using the already created component files.

LAP Advantages:

- Consumes fewer system resources, RAM and CPU, thus allowing for larger assemblies to be processed on the same hardware.
- Can be used as a troubleshooting mechanism as files which failed to convert can be re-translated with different options. This requires the assembly process to be run again to ensure the assembly contains these components.

LAP Disadvantages:

- The use of LAP comes with the overhead of loading the libraries and licensing to convert each file independently. For this reason, the user may find that this process takes more time to complete.

LAP Process Overview

The LAP process begins with the user writing a “Wrapper Script” This script is a windows batch file that calls the initial translation using the *assy_script* argument. When this script runs, it will output a number of Viewer files and a component script (batch file *.bat) to the specified output folder. More details of the component script are provided in the applicable section.

A Viewer file is a Theorem proprietary file used as a temporary container for passing CAD information from the source to the target system. These files are maintained through the translation process. This can be useful if the translation is stopped and needs to be restarted. Using the existing Viewer files, the translation will continue from where it was stopped.

The component script, created by the wrapper script, contains a single line translation for each part file with the required translator arguments. This script can also be used to map the Viewer file names to the specified output file.

Once the part files have been created, the assemblies will be processed. Using the part files that were created when the component script ran, the assemblies are built using the transforms from the source CAD file. Thus ensuring all component files are at their proper location.

©Theorem Solutions 2023

CATIA V5 to Creo

Review the “LAP Process Overview” section before continuing. This section outlines process in detail.

The wrapper script will create three component batch scripts. The first is created to process CATIA V5 data to Viewer files. The name of this script is *input_filename*.vwr.bat. The second component script is to process the Viewer files to CREO. The name of this file is *input_filename*.vwr_2PROE.bat. The third script is *input_filename*.vwr_MOD.bat and is used during the update process, discus later in this section.

Command Line Arguments

Argument	Description
assy_script	*Invokes the LAP process*
as_leaf_node_args	*Translator arguments can be supplied for each leaf node conversion. Provides a method to supply different translator arguments which apply only to the component part level.* *e.g.: as_leaf_node_args “convert_curves convert_surfaces”*

Example Wrapper Script:

Note: The ^ symbol denotes continue to next line. Shown here to make the script easier to read, else it is a single line call to the translator.

```
C:\Program Files\Theorem\24.1\bin\cad_run.cmd ^
CATIA530_CR6 ^
C:\Temp\Input\Rear_Suspension\Rear_Suspension.CATProduct ^
C:\Temp\Output\Rear_Suspension\Rear_Suspension.prt ^
read_pmi_1  assy_script ^
as_leaf_node_args "read_pmi_1 dont_fill_pmi_text
dont_fill_pmi_arrows"
```

Updating the Assembly

The following methods can be used to update an already translated assembly into the same output folder. Any new files will be created. All Transforms will be re-processed, so component placement will be correct.

To begin, delete or rename the Creo files and the corresponding Viewer files to be re-translated. Set the environment variable TS_REM_OUT_EXISTING_PARTS=1 either in the command prompt window used to run LAP or in the wrapper script. Then run the wrapper script. For this process, the *input_filename*.vwr_MOD.bat will be used to recreate the required Viewer files. Also, the *input_filename*.vwr_2PROE.bat will be recreated to only translate those new Viewer files to CREO Part files.

When the translation is complete, the required component files have been recreated as well as the assembly files. The assembly files are processed to ensure any modification to the component transforms are honored.

©Theorem Solutions 2023

Troubleshooting Files

One advantage to LAP is that each output file creates its own log file. This may make it easier to troubleshoot that individual file if it is found that it did not translate properly. The user must decide what arguments are to be added and if they are CATIA read or CREO write arguments, or both.

CATIA Read Arguments Only – Begin by deleting the Viewer file and the CREO part file. Edit the Wrapper script to apply new arguments, and then run the translation.

CREO Write Arguments Only – Begin by deleting or renaming the CREO file in the output folder. The *input_filename*.vwr_2PROE.bat file can be edited to comment out each part file which does not require a new translation. Then edit the translator options for those files to be processed again. Run the *input_filename*.vwr_2PROE.bat script.

Alternatively, the required component translation can be copied and pasted into a command prompt window. The translator arguments edited in this window and the translation processed.

Both CATIA Read and CREO Write Arguments - Begin by deleting the Viewer file and the CREO part file. Edit the Wrapper script to apply new arguments. Be sure to use the '*as_leaf_node_args*' to apply options, where required, to only the component part files. Then run the wrapper script to begin the translation.

Creo to CATIA V5

Review the "LAP Process Overview" section before continuing. This section outlines process in detail.
The wrapper script will create the component batch script. This script will contain the command line translation for all required part files.

Command Line Arguments

Argument	Description
assy_script	*Invokes the LAP process*
as_leaf_node_args	*Translator arguments can be supplied for each leaf node conversion. Provides a method to supply different translator arguments which apply only to the component part level.* *e.g.: as_leaf_node_args "convert_curves convert_surfaces"*

Example Wrapper Script:

Note: The ^ symbol denotes continue to next line. Shown here to make the script easier to read, else it is a single line call to the translator.

```
C:\Program Files\Theorem\24.1\bin\cad_run.cmd ^
CR6_CATIA530 ^
C:\Temp\Input\Alternator\alternator.asm ^
C:\Temp\Output\Alternator\Alternator.CATProduct ^
no_datum_curves no_quilts no_datum_surfaces assy_script ^
as_leaf_node_args " surf_to_nurbs edge_to_nurbs "
```

If the translation is to be re-processed, the existing Viewer files will be used. To re-process the translation while overwriting the existing Viewer files, the following environment variable must be set:
TS_NO_USE_ASSY_TEMP_FILES=1

Updating the Assembly

The following methods can be used to update an already translated assembly into the same output folder. Any new files will be created. All Transforms will be re-processed, so component placement will be correct.

To begin, delete or rename the CATIA files to be re-translated. Either use the environment variable listed previously to overwrite the exiting Viewer file, or delete the Viewer file prior to running the translation. If unsure which viewer file to remove, the Component Batch script will map each Viewer file to an output CATIA filename. Then run the translation again.

Troubleshooting Files

One advantage to LAP is that each output file creates its own log file. This may make it easier to troubleshoot that individual file if it is found that it did not translate properly. The user must decide what arguments are to be added and if they are CREO read or CATIA write arguments, or both.

CREO Read Arguments Only – Begin by deleting the Viewer file or setting the environment variable mention previously. Edit the Wrapper script to apply new arguments, and then run the translation.

CATIA Write Arguments Only – Find the file to be re-translated in the Component script. Edit the translator arguments. Either comment out all other lines (add REM to the beginning of the line) or copy the translation call and paste it into a command prompt window. Run the translation.

Both Read and Write Arguments – Begin by deleting the Viewer file or setting the environment variable mention previously. Edit the Wrapper script. Be sure to add the required arguments to the as_leaf_node_args portion. Run the translation.

©Theorem Solutions 2023

Appendix A – CATIA V5 Configuration

Introduction

This Appendix details how to define and configure the CATIA V5 and Theorem environment to work together.

Conventions

Release of CATIA V5

To indicate a release of CATIA V5 the notation <XX> shall be used. This needs to be replaced with the specific release to be used i.e. 28, 29, 30, 31.

Platform specific directory

Within the installation directory of CATIA V5 there is a platform specific directory i.e. win_b64. This directory shall be referred to as ***<OSDS>*** in this Appendix.

Theorem Installation directory

The Theorem translator installation directory is set at installation time in the translator ***ts_env.bat*** file. This directory shall be noted as ***<%TS_INST%>*** in this Appendix.

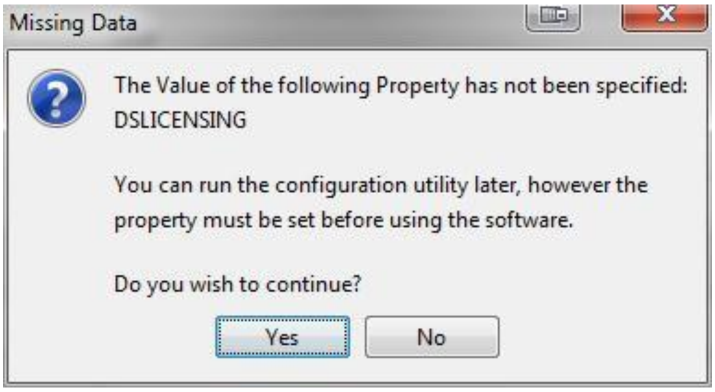
CATIA V5 Installation Directory

Upon installation of a CATIA V5 product the user will be asked to specify the installation directory. This is the directory which contains the platform specific <OSDS> directory.

Having selected the CATIA V5 installation directory via the browse button, the installation process will record the location of the CATIA V5 installation directory in the ts_env.bat file. This file is located in the Theorem translator installation directory. If the location of CATIA V5 subsequently changes, the translator can be guided to the changed location by modifying this file using a text editor to modify the ***ts_env.bat*** that is located in the translator installation directory.

For CATIA V5 R22 and lower releases, a choice is available of using the older LUM Legacy licensing system or the new Dassault Licensing. To enable this enter LEGACY in the DSLICENSING option.

If no entry is included for DSLICENSING a warning dialog will be displayed which warns of the empty field. Selecting ***Yes*** to continue will allow the installation to continue.


Missing Data

The Value of the following Property has not been specified:
DSLICENSING

You can run the configuration utility later, however the property must be set before using the software.

Do you wish to continue?

Running CATIA V5 Translators

Before running the translator the user must run CATIA V5 interactively at least once to configure the CATIA V5 environment and license settings. This can be achieved by running the catia5r<XX>_start script as follows:

%TS_INST%\bin\catia5r<XX>_start.cmd

Once CATIA has been run the Translator can run as described in the relevant product User Guide.

CATIA V5 Environment DIRENV & ENV

The default location for CATIA V5 to store its global environment files is in the global directory:

Windows XP:

C:\Documents and Settings\All Users\Application Data\DassaultSystemes\CATEnv

Windows 7, 8 & 10:

C:\ProgramData\DassaultSystemes\CATEnv
Or
%APPDATA%\CATEnv

You can find this location by running:

%CATIAV5_INST%\<OSDS>\code\bin\setcatenv -h

The environment files are named in the form ***CATIA.V5RN.B<XX>.txt***

If when installing CATIA V5 the default environment file location was replaced with another location then this location needs to be indicated to the CADverter by defining in the ***ts_env.bat*** the environment variable CATIAV5_DIRENV:

set CATIAV5_DIRENV=/some/directory

If the Theorem installation is needed to support multiple releases of CATIA. Then the user can define release specific locations using:

set CATIAV5R<XX>_DIRENV=/some/directory

The Theorem translator will attempt to create its own environment file called ***TheoremCatia5R<XX>.txt.*** The user must therefore have write permission to the CATEnv directory. If this is not possible an existing environment file can be specified using the variable **CATIAV5_ENV**. e.g.

set CATIAV5_ENV=CATIA.V5R26.B26

Note: The extension ***.txt*** is not required. The user can specify a release specific name using CATIAV5R<XX>_ENV e.g. set CATIAV5R25_ENV=CATIA.V5R26.B26

©Theorem Solutions 2023

Checking the CATIA V5 Environment

A script is provided to check that the CATIA V5 environment is set up correctly. In a command window run the command script:

%TS_INST%\bin\checkcatia5r<XX>env.cmd

Checking the Theorem Shared Library

A script is provided to ensure that the CATIA V5 environment is compatible with the Theorem shared library. In a command window run the command script:

%TS_INST%\bin\checkcatia5r<XX>cadverter.cmd

A successful output is an indication that the location for CATIA V5 has been specified to the Theorem translator correctly and that the correct version of the Theorem CATIA V5 translator products have been installed.