TRANSLATE for Creo - NX

USER GUIDE

TRANSLATE for Creo - NX

Release Version: 25.2

Revision: 1.0

Issued: 04/11/2022

Contents

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'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 NX CADverter

The Creo to NX CADverter is a direct database converter between Creo and NX. 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 NX, or NX 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 NX or Creo data this can be done by using the integrated User Interface, which is included with CADverter
- The conversion process can be run Interactively from the Creo session, in Batch Mode or using the Unified Interface
- Command line interface allows process integration

Primary Product benefits?

- Direct conversion between Creo and NX 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 Creo to NX 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

Using the Product

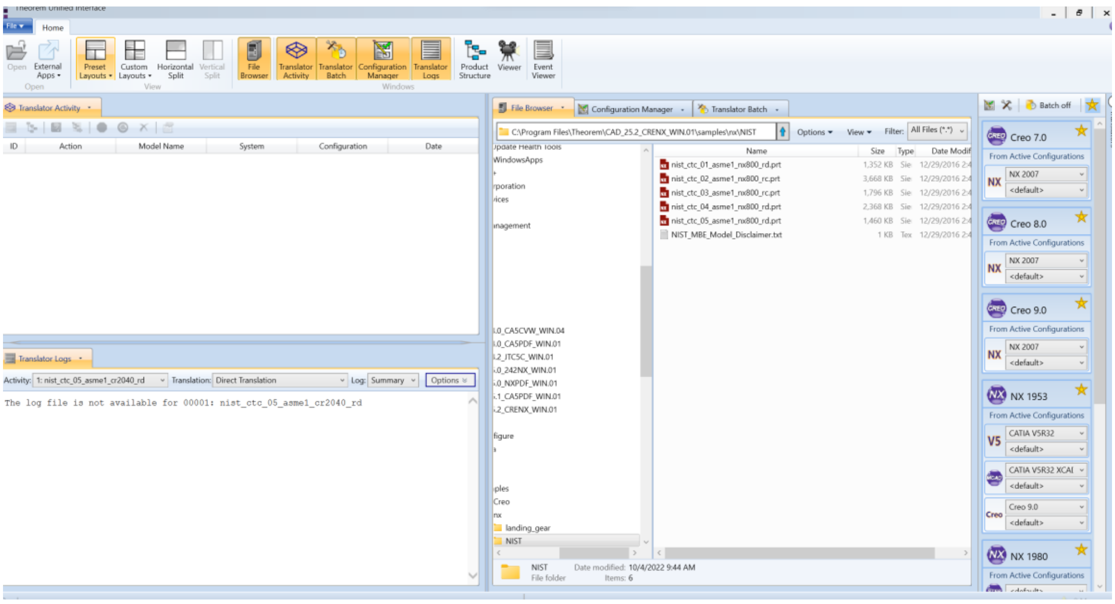
Default Translation – via the Unified Interface

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

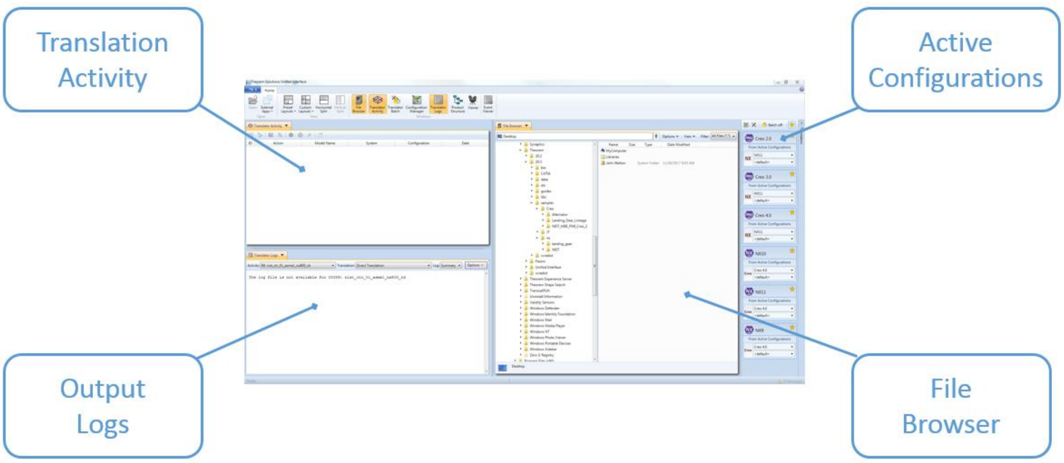
Alternatively, the Unified Interface can be run via a Windows Explorer selection in:

<UI_installation_directory>\bin\Unified_Interface.cmd

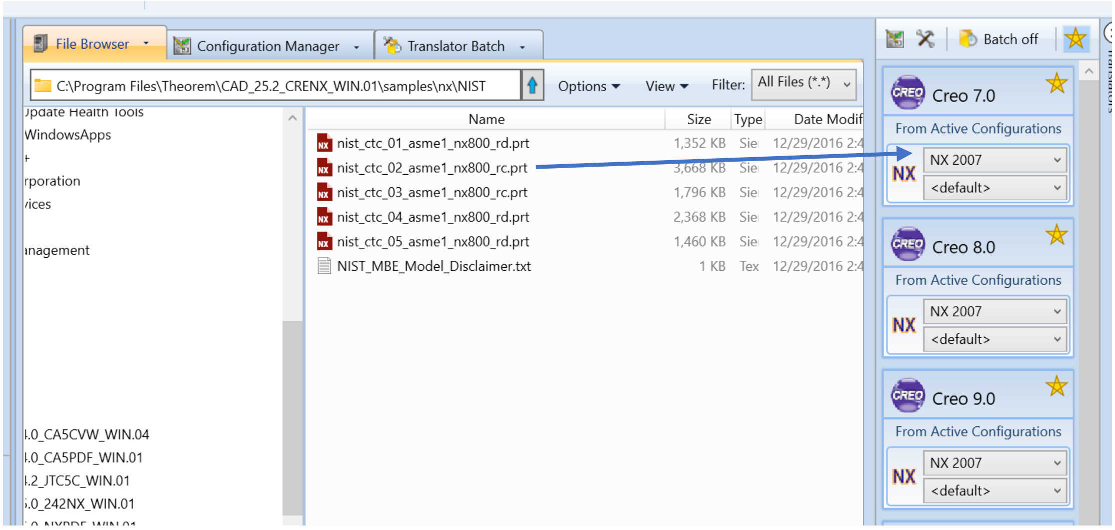
The following interface will be launched:



The default layout is split into 4 primary areas, which can be altered to the user's preference:

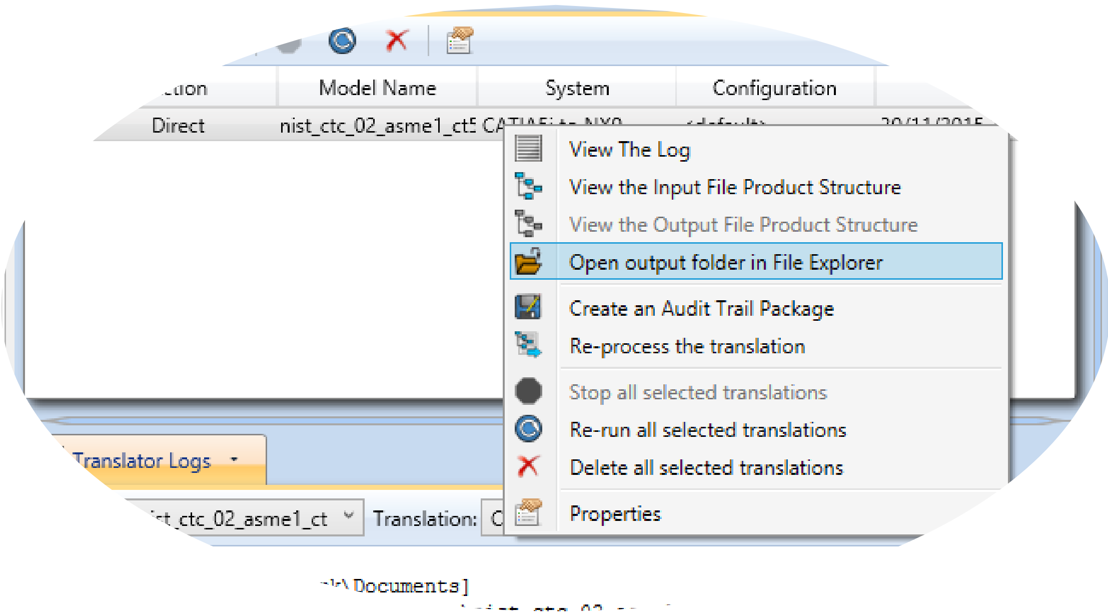


The simplest way to translate from Creo to NX is to drag a file from the file Browser Pane on to the Active Configurations for the translation you require.



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:



Default Translation – via the Command Line

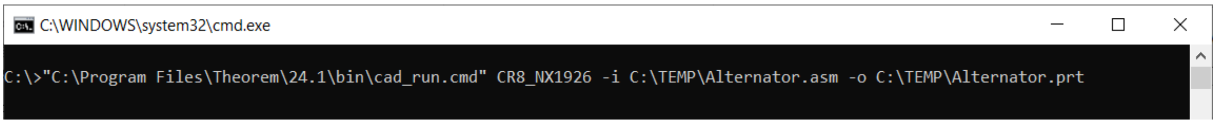
Running a translation via the command line can be carried out via the ***cad_run.cmd*** file located in the ***<installation_directory>\bin*** directory. The format of the command is as follows when translating from Creo to NX:

<Translator_installation_directory>\bin\cad_run.cmd CR[X]_NX[YY] –i <input_file> -o <output_file>

The format of the command is as follows when translating from NX to Creo:

<Translator_installation_directory>\bin\cad_run.cmd NX[YY]_CR[X] –i <input_file> -o <output_file>

Note! *Replace the [X] seen in the example with the version of CREO you are using and the [YY] with the version of NX you are using. E.g. for CREO 8 change to CR8 and for NX 1926 change to NX1926:*



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

C:\Temp\alternator.prt

CADverter Customization

The Theorem translator 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 Creo to NX

Within the Configuration Manager panel of the Unified Interface, arguments that can be specified when publishing Creo data into NX are grouped into the following areas:

- Creo Read – Those arguments that affect how data is read from NX
- NX Write – Those arguments that affect how the data is written to NX
- 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 | NX 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 reading of PMI. (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

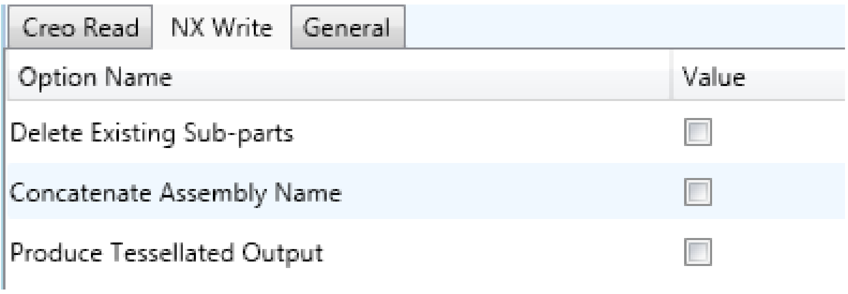
Read Sub Assembly Colours	Read colour information set on sub-assembly level (Default is off)
	• Command Line Syntax:
	  ○ sub_assy_colours
	  ○ sub_assy_colours_off (default)

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.*

NX Write Arguments

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



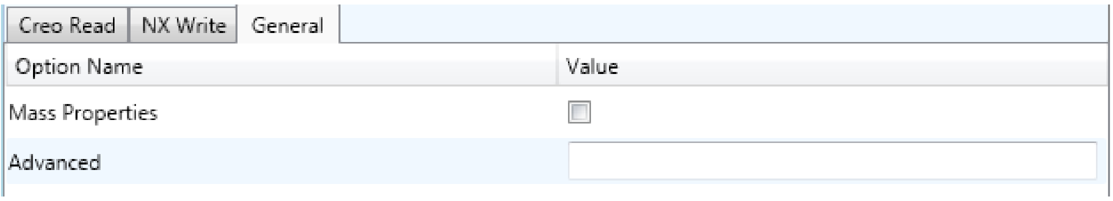
Each of these options is described below:

Option	Description
Delete Existing Sub-parts	Delete existing sub-parts. Default is OFF. • *Command Line Syntax* ◦ *delete_parts*
Concatenate Assembly Name	Concatenates assembly name. Default is OFF. • *Command Line Syntax* ◦ *Concat_assy*
Produce Tessellated Output	Create a tessellated JT file instead of an NX file. Default is OFF.

	• Command Line Syntax ○ *tess_output*

Creo to NX General Arguments

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



Each of these options is described below:

Option	Description
Mass Properties	Mass properties (volume/area CofG) are read and any applied materials, using this option, in cases where a part has multiple solids, volume and area values are summed, but CofG data is invalid. • Command Line Syntax ○ *mprops*
Advanced	Allows any of the Command Line Advanced arguments documented to be passed to the Unified Interface invocation.

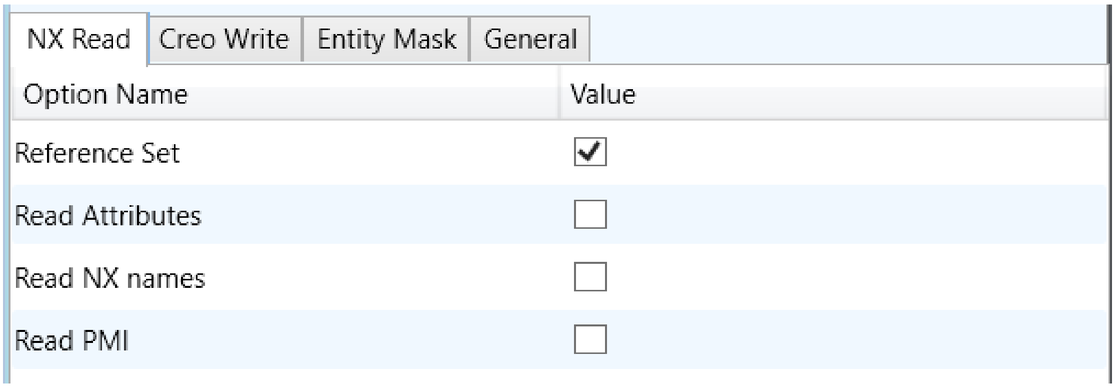
Common Options for NX to Creo

Within the Configuration Manager pane of the Unified Interface, arguments that can be specified when publishing NX into Creo data are grouped into the following areas:

- NX Read – Those arguments that affect how data is read from NX
- Creo 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

NX Read Arguments

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



Each of these options is described below.

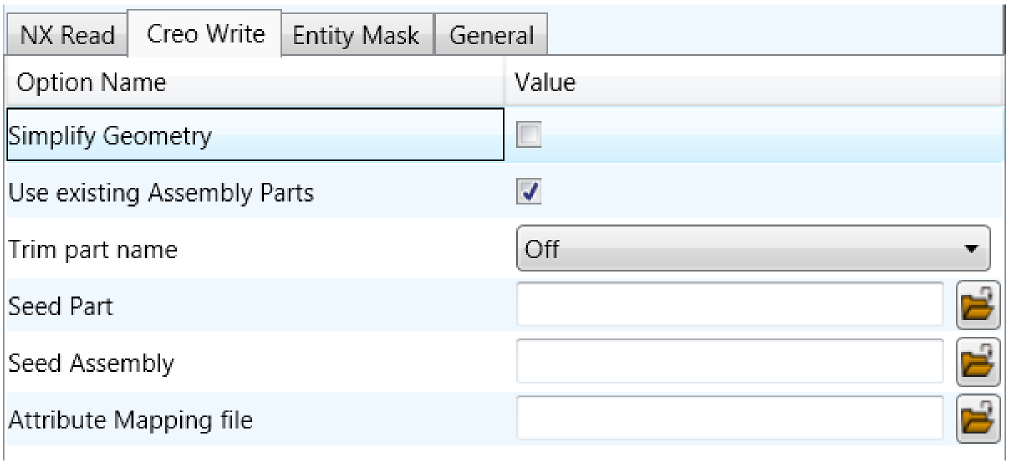
Option	Description
Reference Set	Enabled reference set processing. Default is OFF • Command Line Syntax: ○ *ref_set – to turn on*
Read NX Attributes	Read NX detail user attributes. Default is OFF. • Command Line Syntax: ○ *read_attrs*

Read NX names	Read NX entity names, if they exist. Default is OFF.
	• Command Line Syntax:
	  ○ *no_read_name – default*
	  ○ *read_name – to turn on*

Read PMI	Read 3D PMI. Default is OFF. • Command Line Syntax: ○ *read_pmi write_stroked_pmi*

Creo Write Arguments

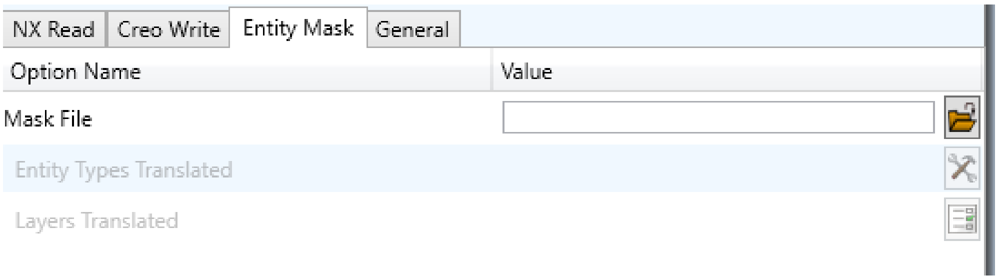
The image below shows the Creo Write arguments that are available, with their default settings:



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*

NX to Creo Entity Masking Arguments

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



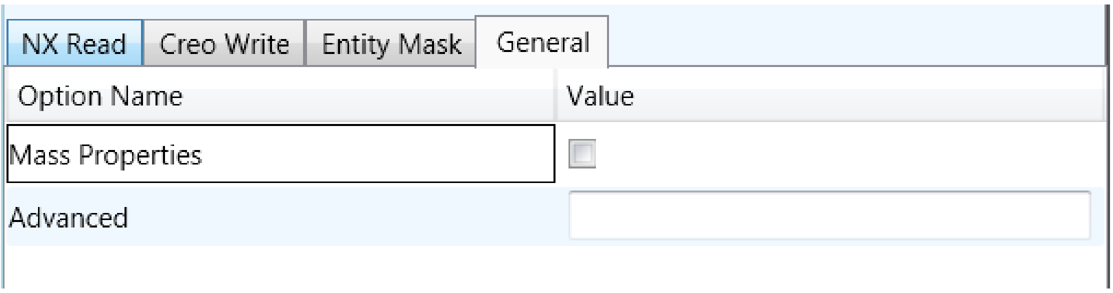
Each of these options is described below:

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 <filename>*
Entity Types Translated	Specifies a selection list from which to select which entity types are to be processed. The following types are available: "SOL" - Masks any 3D entity "SKIN" - Masks any 2D entity "CUR" - Masks any 1D entity "POI" - Masks any 0D entity "AXIS" - Masks Axis Systems "ISOL" - Masks Isolated facetted solids "CCRV" - If on creates a CCRV curve for wire frame edges that have more than one supporting curve "TEXT" - Masks PMI Text ▪ 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***

NX to Creo General Arguments

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



The option is described below:

Option	Description
Mass Properties	Mass properties (volume/area CofG) are read and any applied materials, using this option, in cases where a part has multiple solids, volume and area values are summed, but CofG data is invalid. • Command Line Syntax ○ *mprops*
Advanced	Allows any of the Command Line Advanced arguments documented below to be passed to the Unified Interface invocation

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 to NX Advanced Arguments

Creo Read 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
Regenerate Creo data	Allow the processing of data marked as requiring regeneration in Creo • Command Line Syntax ○ *process_unregenerated*

NX Write Argument	**Description**
Parasolid Tolerant Modelling	Enables Parasolid tolerant modelling. Default is ON. • Command Line Syntax ○ *nopstolmodel – to turn off*
Factor	A secondary option used with Parasolid Tolerant Modelling. Allows a factor to be defined. Default is 3. • Command Line Syntax ○ *pstolmodel 3*

Force Body Creation	Force the creation of bodies. Default is ON.
	• Command Line Syntax
	  ○ *check – to turn off*
	  ○ *nocheck*

Attempt body healing	A secondary option used with Force Body Creation. Tries to heal the forced body. Default is ON. • Command Line Syntax ○ *no_heal_NX – to turn off*
Body healing factor	The factor to be applied to Attempt Body Healing. Default is 0.0095. • Command Line Syntax ○ heal_NX 0.0095
Sew Parasolid Bodies	Enabled Sewing of Parasolid Bodies. Default is ON. • Command Line Syntax ○ *nosew – to turn off*
Tolerance	A secondary option for Sew Parasolid Bodies giving the tolerance level to use. Default is 0.1. • Command Line Syntax ○ *pssew 0.1*

Keep all bodies	A secondary option used with Sew Parasolid bodies allowing all bodies to be kept (no matter how small) that may be created as a result of sewing Default is OFF.
	• Command Line Syntax
	  ○ *no_keep_all_bodies – default*
	  ○ *keep_all_bodies – to turn on*

Prepare CSG Primitives	Prepare CSG Primitives Tolerance (input part units). ▪ Command Line Syntax

	▪ *csg_prep*
Change CSG Shift	Change CSG Shift Distance (input part units). • Command Line Syntax ○ *csg_shift*
Fix CSG Primitives	Fix CSG Primitives. Default is OFF. • Command Line Syntax ○ *csgfix*
Improve Accuracy of Facetted Solid	Improve Accuracy of Faceted Solids. Default is ON. • Command Line Syntax ○ *no_fsol_fix – to turn off*
Explode Solids to Faces	Explode Solids to Faces. Default is OFF. • Command Line Syntax ○ *split_brep*
Split Discontinuous Surfaces	Split Discontinuous Surfaces. Default is ON. • Command Line Syntax ○ *brep_prep* ○ *no_brep_prep – to turn off*
Fix Degenerative Edges	On face create failure, check and fix any degenerate edges. Default is ON. • Command Line Syntax ○ *fix_degen* ○ *no_fix_degen – to turn off*
Specify a Face Edge Tolerance	Specify an edge tolerance to be used when creating faces. Default is ON. • Command Line Syntax ○ *Please see Edge Tolerance below*
Edge Tolerance	A secondary option used with Spicify a Face Edge Tolerance where the tolerance value is assigned. Default is 0.000006. • Command Line Syntax ○ *face_edge_tol 0.000006*
Fix small features in solids	Remove small edges, sliver and spike faces from solid bodies. Default is OFF. • Command Line Syntax ○ *ps_fix_small – to turn on* ○ *no_ps_fix_small - default*
Fix small features in open solids	Remove small edges, sliver and spike faces from open solids. Default is OFF. ▪ Command Line Syntax ▪ *ps_fix_osol – to turn on* ▪ *no_ps_fix_osol - default*
Simplify Geometry	Simplify Geometry. Default is OFF. • Command Line Syntax ○ *Simplify_solids – to turn on*
Create empty NX Parts	Creates empty NX parts for missing input data • Command Line Syntax ○ *create_empty_part*

NX to Creo Advanced Arguments

NX Read Option	Description
all_layers	Read entities from ALL layers even if the layer is invisible
fit_view	Fits the visible geometry to the screen size (e.g. zooms the view), regardless of users source system settings. The default option is *'fit_view_as_saved',* which reads the view according to the settings in the source CAD system.
heal_ug_read	Attempt an NX heal on the (native) body - using the NX API. Note! This should be used in combination with *regen_brep*
part_layer	If set, Process As Saved part layers. If unset process All layers
read_blank	Reads any blanked entities. *(Default is off)*
read_assy_pmi	Allows assembly component level PMI to be read. *(Default is off)*
read_views	This flag turns on the processing of 3D Views even if *'read_pmi'* is off. If *'read_pmi'* is ON, then *'read_views'* is switched ON automatically. This can be overridden by specifying *'no_read_views'* If *'read_pmi'* is OFF, then *'read_views'* is OFF by default. This can be overridden by specifying *'read_views'*
ref_set_part_name	If set, append the Reference Set name to created part
regen_brep	Allows brep data to be regenerated from the topology using the NX API. Note! It is only advised to use this option if solid information cannot be written into 3D PDF.
no_create_datum_plane_border	Disables the creation of datum plane borders. The default is to create 4 datum plane bounding lines to mimic NX section lines

Creo Write Option	Description
Collapse Assembly Structure	If the Input CAD data contains any assembly structure, then by default assembly structure will be created in the Output CAD format. Running this option causes the assembly structure to be "exploded" into a flat single component file. • Command Line Syntax ○ *noditto*
Convert surfaces to NURBS	Read surfaces as NURBS surfaces (else read in native form). Default is ON. • Command Line Syntax: ○ *Noprep – to turn off*
Convert Edge Curves to NURBS	Read edge curves as NURBS curves (else read in native format). Default is ON. • Command Line Syntax: ○ *rd_native_edge – to turn off*

Group All Geometry	Creo default is to create one part per Solid body. This option will allow all Geometry to be written into one part.
	• Command Line Syntax
	    ○ *mult_feat_on*
	    ○ *mult_feat_off* (default)

Group Open Solids into a Single Creo Part	This option will allow all "Open Solid" (Quilt) Geometry to be written into one part.
	• Command Line Syntax
	    ○ *mult_open_on*
	    ○ *mult_open_off* (default)

Group Closed Solids into a Single Creo Part	This option will allow all "Closed Solid" Geometry to be written into one part.
	• Command Line Syntax
	    ○ *mult_brep_on*
	    ○ *mult_brep_off* (default)

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.

NX 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 NX 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

assy_script *Invokes the LAP process.*

as_leaf_node_args *Translator arguments to 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 "read_pmi noprep"

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\23.3\bin\cad_run.cmd" ^
NX1899_CR6 ^
-i "C:\Temp\Input\ landing_gear\landing_gear_assembly.prt" ^
-o "C:\Temp\Output\landing_gear\landing_gear_assembly.asm" ^
ref_set read_pmi  assy_script ^
as_leaf_node_args "read_pmi noprep"
```

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 honoured.

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 NX read or CREO write arguments, or both.

NX 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 NX 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 NX

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

assy_script	*Invokes the LAP process.*
as_leaf_node_args	*Translator arguments to 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 "read_pmi surf_to_nurbs edge_to_nurbs"

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\20.3\bin\cad_run.cmd" ^
CR6_NX1899 ^
-i "C:\Temp\Input\Alternator\alternator.asm" ^
-o "C:\Temp\Output\Alternator\Alternator.CATProduct" ^
read_pmi no_datum_curves no_quilts no_datum_surfaces  assy_script  ^
as_leaf_node_args "read_pmi 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 NX 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 NX 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 NX 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.

NX 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.