NX CUSTOMIZATION FOR WINDOWS DATE: 6-JAN-2025

TABLE OF CONTENTS

DISCLAIMER

NXcustom is NOT an official Siemens product and has therefore no support!
When you have questions, please contact Gerrit and/or Eric (below) or address the Siemens Community for support!

IMPORTANT CHANGE!!

As of Januari 2025, NXcustom is not delivered anymore on every version change of NX.

The goal is to have the library template and the scripts to be version independent and therefore as of this version of NXcustom you can use for each NX version of NX2206 and newer (older versions may work also, but those are not tested, so please use it but be sharp on possible issues).

The distribution will not contain an NXxxxxlibrary anymore, but it will contain a template for such a library in the …\NXcustom\NXstartup\Configuration_Templates folder.

See the “

Setup" section in this document for more details on how to use it.

There will still be regular updates, but that will not be linked anymore to newer versions of NX, therefore there is no need to wait for an update of NXcustom when a new version of NX is released.

OVERVIEW

NX system administrators have a unique challenge in deploying and maintaining company standards and best practices for CAD/CAM/CAE throughout the enterprise. NXcustom provides a way to centralize these tasks with easy access to NX related environment variables, customer defaults, individual user settings, custom templates, ribbons, tables and libraries.

NXcustom allows users and system administrators to deploy NX customizations quickly and easily to multiple users. Placing the NXcustom folder on a mapped network drive will allow all users to use the same NX configuration simply by running the NXstart.bat file within the NXcustom\NXstartup folder. Single users can also benefit from using NXcustom on the local drive by setting up multiple environments that may be used to service the needs of multiple customers. All of this is done without modifying files in the NX install folders and without setting system and user environment variables, thus providing as-needed environment variables and preventing the possible loss of configuration data when NX is updated or uninstalled. NXcustom is used to launch the following applications that use the shared custom environment:

- NX (with and without Group-level customer defaults)
- CAM Express or SE CAM Pro
- NX Viewer
- Solid Edge to launch SE CAM Pro
- Machine Knowledge Editor (MKE) up until NX2212
- Power Drafting
- Teamcenter Integration for NX (TCIN)

The NX2412library folder provides a convenient location for customer defaults, environment variables, ribbon files, menuscripts, program files, libraries, templates, and CAM or CMM configuration files. These files are easily deployed globally to all users and can also be modified for functional work groups or individual users.

SETUP

IMPORTANT!

PATH NAMES AND SPECIAL CHARACTERS.
WHITESPACES: THOUGH NXCUSTOM CONSIDERS WHITESPACES IN PATH NAMES, THERE STILL MIGHT BE UNEXPECTED BEHAVIOR IN SOME CASES.

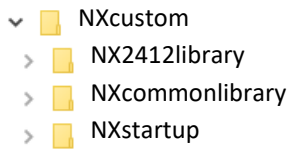
Special Characters: It is strongly advised to avoid any special characters in path names where NXcustom accesses information. Especially characters like: &, ^, %, $, #, @, !, * and any kind of brackets should be avoided.
When unexpected behavior occurs, please test it in a local location in a path with no special characters. If the issue disappears, the path name is most likely the issue.

INSTALLATION

To install NXcustom for the first time, place the NXcustom folder from the 7-zip file into a location available to all users, such as a mapped network drive or in individual user or group folder. Make sure the path does not contain any spaces. If so, reread the important notes in red above!

NXcustom is self-aware and knows where it resides on the local or network drive. In addition, NXcustom checks the registry for installed versions of NX and determines install location automatically. This eliminates the need for additional configuration steps.

Please copy “NXlibrary_template” from the …\NXcustom\NXstartup\Configuration_Templates folder and paste it to …\NXcustom and rename it to NXxxxxlibrary, where “xxxx” stands for the version you what to use it for.
For example, the result might look like:



You can also take a copy of library of an older NX version, but please do not use anything older than NX2312library to have at least the most compatible structure for your new version.

If you are coming from an older version than NX2312, you can still use the library, but you will be missing some of the latest changes in the configuration and it might be that you need to modify parts that are not fully in sync with the latest structure.

USING NXSTART.BAT AS GENERAL STARTUP SCRIPT FOR ALL MANAGED AND NATIVE VERSIONS

The NXstart.bat script allows starting up different version of NX and different variants of NX (ie. NX, Viewer, Mechatronics Concept Designer, CAM, etc.) and with different groups related to that environment, by making use of command line arguments that can be passed through to the “NXstart.bat” script.

Additionally a lot of logic is added to prevent the user from making mistakes and aiding in finding the issue to be resolved.

For configuring the environment, it is not required (and even strongly advised) to edit the NXstart.bat file. When presets need to be changed, the NXstart.conf file is the file where site-wide changes should be made.

When no arguments are given, the script will start the latest version of NX that is installed on the system that also contains a matching NXxxxxlibrary for that same version.
So if your NXcustom folder looks like:

- NXcustom
  - NX2306library
  - NX2312library
  - NX2412library
  - NXcommonlibrary
  - NXstartup

And the installed versions of NX on your system are: NX2306, NX2312 and NX2406, it will automatically launch NX2312 and not NX2412 because the highest version is limited by the installed NX2312 version.

But if your NXcustom folder looks like:

- NXcustom
  - NX2306library
  - NX2312library
  - NX2406library
  - NXcommonlibrary
  - NXstartup

And the installed versions of NX on your system are: NX2306, NX2312, NX2406 and NX2412 it will launch NX2406 and not NX2412 because now the highest version is limited by the NX2406library

So only if your NXcustom folder looks like:

- NXcustom
  - NX2306library
  - NX2312library
  - NX2412library
  - NXcommonlibrary
  - NXstartup

And the installed versions of NX on your system are: NX2306, NX2312, NX2406 and NX2412, it will automatically launch NX2412 because that is the highest common version.

Additionally, it is possible to set some default arguments in the accompanying “NXstart.conf” file and administrators can prevent users from using certain command line arguments from being changed.

COMMAND LINE ARGUMENTS OF NXSTART.BAT

The new NXstart.bat replaces all legacy batch files. Scripts that were there for the specific cases like specific versions, groups, applications and/or packages, and starting NX in managed mode are now all handled by the new NXstart.bat script. Eventually, the script will be getting a separate release timing than the NXxxxxlibrary folder that is linked to a specific NX version.
Since there are now a lot of command line options that can be given to the script, we try to keep this to a minimum and can even be none if you don’t need anything more than native NX. In other words, if you used in the past “NXxxxx.bat” to launch your NX that will work again in the new script.
All arguments to NXstart.bat will come in pairs.

VALID ARGUMENTS

Argument	Description
/v(ersion) [1847|…|2412|etc.]	Specifies the NX continuous release number you want to start Valid versions are those which are installed and have an NXxxxxlibrary in the NXcustom folder. Default = Highest common version
/a(pplication) [NX|MKE|SE|S3D|CMD|NONE]	Specifies the application that you want to start. NONE can be used to set the NX related variables and then exit without starting any application. Default = NX
/p(ackage) [NX|VIEW|NXCAM|MECHATRONICS |SECAM|LAYOUT|SIMCENTER3D |SIMVIEWER|MOTION|NX2DEDIT]	Specifies application package to use. Default = NX
/libname <library_name>	Specifies the named library to use with the version of NX to be started. Default = not set
/g(roup) <group_name>	Specifies a group. A folder with a matching name should exist in the ..\NXxxxxlibrary\CustomerDefaults\Groups folder, or ..\NXxxxxlibrary\CustomerDefaults\TC\Groups folder for managed mode. The matching UGII_GROUP_DIR will be set. Default = not set
/u(nits) [METRIC|INCH]	Specifies the routing units Default = METRIC
/s(essions) [SINGLE|MULTIPLE]	Specifies whether NX is allowed to start multiple sessions. So when the same version is executed multiple times: SINGLE Will connect to the same running session. MULTIPLE Will start a new NX session for each instance Default = SINGLE
/m(anaged) [TRUE|FALSE]	Specifies whether NX will be started in native or managed mode. TRUE Will start NX in managed mode. FALSE Will start NX in native mode. Default = FALSE
/w(orkdir) <start_directory>	Specifies the directory where the application will start. Use single or double quotes to encapsulate a path containing spaces. Default = some feasible paths are checked and set
/t(itle) <window_title>	Value to display in NX window title bar. Use single or double quotes to encapsulate a title containing spaces. Default = not set
/saltlic <license_server>	Specifies the SALT license server to use. Use port@hostname or port@ipaddress. Default = not set
/l(icserver) <license_server>	Specifies the UGS license server to use. Use port@hostname or port@ipaddress. Default = not set
/cdlmdlic <license_server>	Specifies the CDLMD license server to use. Use port@hostname or port@ipaddress. Default = not set
/tcenv <tc_env_name>	Specifies the environment name in the TCenv_<tc_env_name>.conf file For example “/tcenv production” results in using TCenv_production.conf to be read from the NXstartup folder
/h	Displays some help information

ARGUMENT EXAMPLES

- NXstart.bat /v 2312
  - Starts NX2312 with NXcustom (provided that NX2312 is installed and NX2312library exists)
- NXstart.bat /g CAD
  - Starts Highest version of NXxxxx that also as a NXxxxxlibrary available and uses the UGII_GROUP_DIR that points to NXxxxxlibrary\Customerdefaults\Groups\CAD (which must exist).
- NXstart.bat /v 2406 /a MKE
  - Starts the Machining Knowledge Editor with NX2406. NX2406 must be installed and NX2406library must exist, otherwise an error will be logged.
- NXstart.bat /v 2406 /libname Staging /m true
  - Starts NX2406 in managed mode with the Staging library. NX2406 must be installed and NX2406library_Staging must exist, otherwise an error will be logged.

PRESETS

To prevent the requirement for shortcuts that always need a set of arguments, it's possible to predefine a lot of the preferences in 3 ways.

1. USE OF THE NXSTART.CONF

This allows for preset of a set of arguments by the NX administrator. When the NXstart.conf file is added to the NXstartup folder, the NXstart.bat will parse this to make presets. These presets will overrule the defaults in the NXstart.bat script itself.

A template for this file is found in the "…\NXstartup\Configuration_Templates" folder.
Copy the file to the "…\NXstartup" folder.
See the comments in the file to determine what you might want to change.

At the bottom of this file there is the variable NXCUSTOM_USER_OVERRIDE_ARGS that manages which arguments are allowed to be overridden by users. Removing arguments from that list will prevent users from overriding with their own arguments.

The NXstart.conf file is intended to be configured by the administrators, and only the administrators should have write access to this file.

2. USE OF THE NXLIBRARY.CONF

This allows you to override some of the settings that are set in NXstart.conf specifically for a library. If a NXlibrary.conf is found in the NXxxxxlibrary folder these settings will be used.

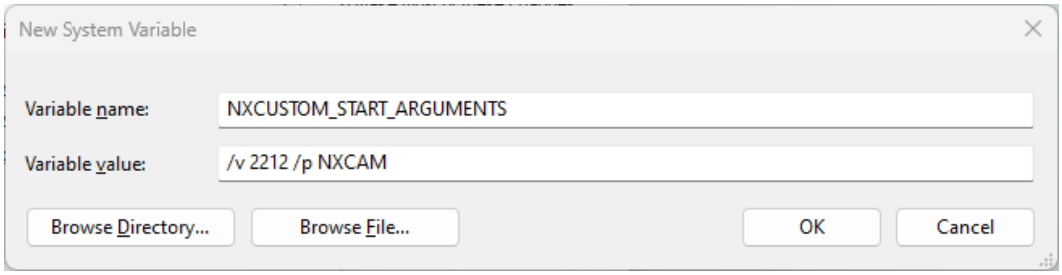
A template for this file is found in the "…\NXstartup\Configuration_Templates" folder.
Copy the file to the "…\NXcustom\\NXxxxxlibrary" folder for which you want to override some of the settings.
See the comments in the file to determine what you might want to change.

Be aware that some of the settings in NXstart.conf cannot be overridden at the library level. If you use one that is not allowed, you will get an error message.

3. USE OF THE NXCUSTOM_START_ARGUMENTS ENVIRONMENT VARIABLE

When a user sets the (system or user) environment variable NXCUSTOM_START_ARGUMENTS with some default arguments, those arguments will override the settings that are default or the ones set in the NXstart.conf.
This only works for the list of arguments that is allowed. Overridable arguments are defined by the NXCUSTOM_USER_OVERRIDE_ARGS variable in the NXstart.conf file.

Example:



ADDITIONAL OPTIONS IN THE NXSTART.CONF

In the NXstart.conf file there are some additional options that cannot be set with a command line argument.
Their function is described in more detail in the file. These are the following settings:

1. NXCUSTOM_MANAGED_CUSTOMER_DEFAULTS
   This allows for separation of the customer defaults for Managed and Unmanaged mode.
2. NXCUSTOM_USER_SETTINGS_LOCATION
   Defines the location where the User Files are stored.
3. NXCUSTOM_VERSION_IN_TITLE
   Allows to show a detailed NX version in the Title Bar of NX.
4. NXCUSTOM_LIBNAME_IN_TITLE
   Allows the library name (when named library is used) to display in the Title Bar of NX.
5. NXCUSTOM_LOAD_OPTIONS_IN_USER_DIR
   Allows to automatically save the Load Options file in the location as the User files.
6. UGII_UGDOC_BASE
   Option to refer to a local NX Documentation repository.
7. NXCUSTOM_REPORT_VARS
   Allows to Enable, Disable displaying the info (green window) or just the log file, where they are reported also.
8. NXCUSTOM_LOG_FILE
   Sets the name and location of the NXcustom log file
9. NXCUSTOM_USER_OVERRIDE_ARGS
   Allows for administrators to limit overriding of arguments by users to specified arguments.

PRIORITY OF ARGUMENTS

The arguments are processed in the following order:

1. Defaults of NXstart.bat
2. Settings in NXstart.conf
3. Settings in NXlibrary.conf of the library being used
4. NXCUSTOM_START_ARGUMENTS environment variable
5. Arguments passed to the NXstart.bat script upon execution

The last one set will override a possible same setting that was set earlier.

TC Configuration	Sessions	Title	NX Version	Library Name	
	Single		Auto		NXstart default
Production	Multiple	Prod	2306		NXstart.conf
Test		Test			NXlibrary.conf
	Single				Argument Var
			2312	Test	User Arguments
Test	Single	Test	2312	Test	Result

Example of priority.

ADDED SUPPORT FOR EXECUTING CUSTOM SCRIPTS WITH HOOKS

Where in earlier versions a script named NXstart_customer.bat was found in the NXstartup folder, allowed users to execute some specific code/logic before the application was launched, there is now a new mechanism called hooks, that allows you to inject code at several location in the script without the need to edit NXstart.bat.
At this moment there are 3 locations that allow for the execution of a script:

- After configuration evaluation (after_conf_eval). Here the entire configuration has been evaluated. This entry point would be a good location to add the NXcustom caching sample script.
- After reporting vars (after_report_vars). This is the old location where the NXstart_customer.bat file was executed. This would be just before the chosen application would be started.
- After application start (after_application_start). This is just before the scripts fully exits.

To activate those, make sure that one or more batch files exist in one of the subfolders of …\NXstartup\Hooks.
Some templates are provided in the …\NXstartup\ Configuration_Templates\Hooks_templates folder.

Those hooks allow for more complex environment configuration than the NXstart.conf file can support. If the additional complexity is needed, additional scripting should be done in one of the additional batch files, and preferably NOT the NXstart.bat file! The NXstart.bat file will call the batch files automatically when found in the hooks sub folders.

CONFIGURING THE NX ENVIRONMENT

All necessary environment variables can be added/edited in the NX_env.dat file in the NX2412library folder. If a variable is not set in this file, its default value will be read from the ${UGII_BASE_DIR}\UGII\ugii_env_ug.dat file in the NX installed directory.

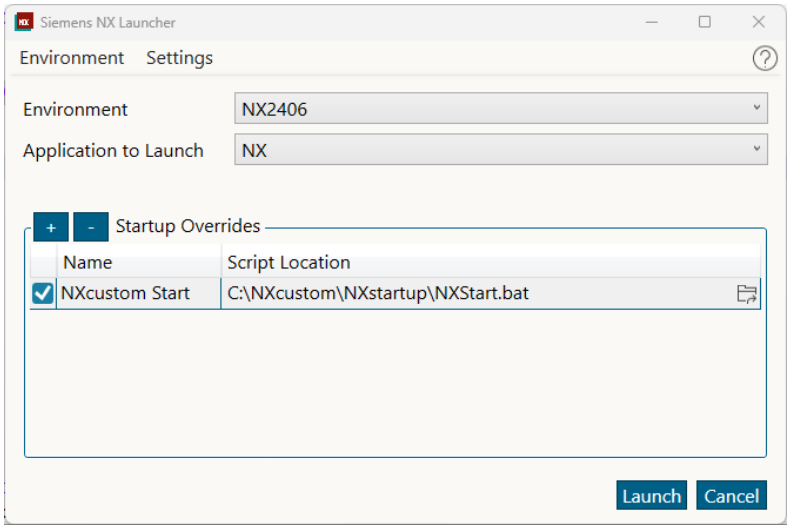
When you would want to change an environment variable to is not define in the NX_env.dat file, you can look it up in the ugii_env_ug.dat file and copy it (together with its description) into the NX_env.dat file to override the default value.

USE WITH SIEMENS NX LAUNCHER

As of NX2406 the NX Launcher can be used again with NXcustom. To use a NXcustom environment with NX Launcher just execute the NXstart.bat script from its original location.

Important!
Currently it has only been properly tested with a Native NX session! For Teamcenter controlled environments, it is not tested yet, so please check if it works correctly in this environment!

NXstart.bat automatically detects when it is executed through NX Launcher and will select the correct NXxxxxlibrary based on that version that NX Launcher is intending to use and will configure the environment accordingly. It also will omit the launch of any application, because the launcher decides which application is to be started after the execution of the script.



NX Launcher example

It is important that the folder structure of NXcustom is kept as intended and the NXstart file is executed from its original location, ie. :

- NXcustom
  - NX2306
  - NX2312
  - NX2412
  - NXcommonlibrary
  - NXstartup

Documentation on how to use/configure the Siemens NX Launcher see the documentation (application_launcher.pdf) in the doc folder under the installation folder.

Important!
When using the NX Launcher in combination with Teamcenter it is highly advised to DISABLE the “IsEnvironment“ by setting it to false in the “OEM_Environment_Def.xml” file.

```
<Environment IsEnvironment="false" Name="NX2406">
    <NXLocation>C:\PLM\NX2406</NXLocation>
</Environment>
```

Setting it to True also requires you to Enable Environments in Teamcenter and that can cause unexpected behavior if you do not know what you are doing.
Please read the documentation on Teamcenter/NX Environments to understand it's function and behavior.

UPDATE AN EXISTING NXCUSTOM ENVIRONMENT (HIGHER BASE VERSION)

The impact of upgrading an NX environment is highly dependent on the degree of configuration/customization. It is required that you have knowledge of the structure of the NX environment.

For Modeling usually all data files (Materials, Thread and Hole Tables) can be copied into the new environment, therefore, from NX1926 and beyond, there is a NXcommonlibrary folder that holds data that is NX version independent. Placing the data there reduces the need to copy that data into a newer version.
There is always a risk that some of the common data files, get changed in a new version. If so, this data must go back into the Version specific library (ie. NX2412library).

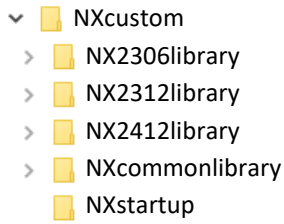
However, for certain applications, the configuration consists of data as well as scripting files and/or compiled code (dll's, etc.). Depending on what has been changed, the approach will be different.

To see what has changed between versions in the configuration, please visit:

https://docs.plm.automation.siemens.com/docs/nx_dev/nx_report/en_US/Whats_Changed.html?Version=2406

Some global guidelines on how to migrate your NXcustom environment to a higher base version, are described below:

1. Extract the content of the archive over the existing environment.
   The result should be something similar as below:



The new start script is added to the existing NXstartup folder. The NXstart.bat will be overwritten with updates as intended. Additionally, all the new/updated templates will be overwritten in the "…\NXstartup\Configuration_Templates" folder.

Your customized configuration files will not be touched.

2. Copy the NX_env.dat (and other custom *.dat) file(s) from the old NXxxxxlibrary into the NX2412library folder.
   a. Verify if there are no hard pointers to folder locations in the *.dat files but to use the ${NXCUSTOM_LIB} variable to point to the NXxxxxlibrary folder of an old version and its children.
3. Depending on the application, the best practice for migrating your environment can be different. Here some global steps:
   a. Modeling/Gateway
      i. Copy your customized "data" files from the old NXxxxxlibrary into the NX2412library folder
   b. Manufacturing CAM and CMM (See also the MANUFACTURING section below)
      i. Copy the OOTB resource content to the designated NX2412library sub folder.
      ii. Migrate your modified data (and possible scripting files) files into the new environment
      iii. …
   c. Re-compile NXopen applications (when required)
   d. Review the menu customization

   e. Other applications might require their own approach.
4. Start NX with the new NX2412.bat file and import the site and group customer defaults from your previous environment into your new environment.
5. Test if your environment is working correctly and that you have access to all the migrated data.

NXCOMMONLIBRARY

To further simplify the migration to a newer version, the NXcommonlibrary folder is introduced.
In this folder is a separate NX_common_env.dat file controls the settings that can be moved to this folder.

So, when you want to make a reference to the NXcommonlibrary in your "NX_common_env.dat" file use:

    ${NXCUSTOM_COMMON_LIB}

And when you want to make a reference to the NX2412library in your "NX_env.dat" file use:

    ${NXCUSTOM_LIB}

You are free to move other content of which you know you can use them across different versions to this folder also.
Make sure that the folders you add, are set into the "NX_common_env.dat" and removed from the "NX_env.dat" file.

For future versions, it is intended to keep pointing by default to this folder, so this data does not have to be duplicated when you work with multiple versions.

USER FILES

NX related user information is by default stored in the …\NX2412library\CustomerDefaults\Users folder within a subfolder of the user's name. This folder is created automatically and contains dialog memory, history, roles, and user level customer default files. NXcustom is using the UGII_USER_PROFILE_DIR variable to redirect the location of these user files. The location of the User Files and where UGII_USER_PROFILE_DIR will point to, can easily be changed in the NXstart.conf file.

IMPORTANT!!!!
Several users reported issues with starting of NX with NXcustom when used on AWS but also on other server locations.
The possible cause seems to be the fact that NX is downloading specific content in the UGII_USER_PROFILE_DIR and tries to unzip that content in the same location, some security policies seem to block that and therefore NX crashes.
To overcome this, NX should be allowed to unzip content in those folders.
When that cannot be overcome, it is advised to change the location of the User files.

ENFORCING USER PREFERENCES

Since NX1847 the User Preferences are not stored in the registry anymore. Therefore, tools that were used in the past to manipulate certain User preferences with the Reg command was not possible anymore.

Since NX1919 and beyond it is possible to overrule the User Preferences on Group or Site level.
Since the mechanism for managing UGII_SITE_DIR and UGII_GROUP_DIR is already in NXcustom, you can follow the description described in Customizing the NX installation (siemens.com).

PLUGINS

To add your own plugins to NX, you can use the …\NX2412Library\Plugins folder. As a sample the new "Compress Assembly" tool is in there. Please read the "README Plugins.txt" in the folder for more details.

Customizations previously contained in the NXxxxxlibrary\ToolBars or NXxxxxlibrary\Menuscript folder should now be reworked as a plugin and added to the Plugins folder. The NXxxxxlibrary\ToolBars and NXxxxxlibrary\Menuscript folders have been removed.

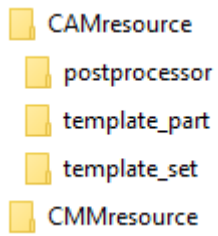
MANUFACTURING

The content that needs to be customized in your Manufacturing environment (the children of …\MACH\resource folder) then only the sub folders that you are going to change need to be copied to the …\NX2412library\CAMresource folder.

The NX2412library\NX_CAM_auto_env.dat file will automatically detect which folders exist in the NX2412library\CAMresource folder and will only include those in the custom environment. For all the folders that are missing, the ones from the original NX installation will be used.

This reduces the amount of data in the NXcustom environment, but also gives you a better overview of what is custom, and what is not.

When for example you would only use custom postprocessors and custom CAM templates, you only need to copy the postprocessor, template_part and template_set folder to the CAMresource folder ie.:



This automatic behavior is handled by the NX2412library\NX_CAM_auto_env.dat file which is included in the NX2412library\NX_env.dat file.
If you wish you can look at the files mentioned above to copy this behavior for other folders in the environment.

When you still want to use the entire content of the …\mach\resource folder in the CAMresource then look into the NX_env.dat file at the "CAM Resource Files" section and uncomment the line:

```
# UGII_CAM_RESOURCE_DIR=${NXCUSTOM_LIB}\CAMresource\
```

This will reinstate the legacy mechanism.

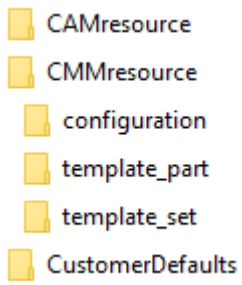
CMM INSPECTION

The content that needs to be customized in your CMM environment (the children of …\ CMM_INSPECTION\resource folder) then only the sub folders that you are going to change need to be copied to the NX2412library\CMMresource folder.

The NX2412library\NX_CMM_auto_env.dat file will automatically detect which folders exist in the NX2412library\CMMresource folder and will only include those in the custom environment. For all the folders that are missing, the ones from the original NX installation will be used.

This reduces the amount of data in the NXcustom environment, but also gives you a better overview of what is custom, and what is not.

When for example you would only use custom configuration files and custom CMM templates, you only need to copy the, configuration, template_part and template_set folder to the CMMresource folder ie.:



This automatic behavior is handled by the NX2412library\NX_CMM_auto_env.dat file which is included in the NX2412library\NX_env.dat file.
If you wish you can look at the files mentioned above to copy this behavior for other folders in the environment.

When you still want to use the entire content of the …\mach\resource folder in the CAMresource then look into the NX_env.dat file at the "CMM Resource Files" section and uncomment the line:

```
# UGII_INSPECTION_RESOURCE_DIR=${NXCUSTOM_LIB}\CMMresource\
```

This will reinstate the legacy mechanism.

OTHER FOLDERS IN THE NX2412LIBRARY SUB-FOLDER

AnnotationTemplates, DesignTools, Materials, ModelingStandards, Reuse, Showroom, Tables, etc. folders are configured empty, but ready to receive user data. Associated environment variables in the NX2412library\NX_env.dat file is being commented to point to the original installed folders to read specific files. If creating/editing custom template parts, it's recommended that the original files are first copied from the NX installed folder into the …\NXcustom\NX2412library sub-folders. Search the …\NXcustom\NX2412library\NX_env.dat file for the Template string and uncomment the line that specifies the Template folder.

See the README files in the …\NXcustom\NX2412library\sub-folders for setting up various libraries.

ASSOCIATING PART FILES

NXcustom can also be used when launching NX directly with a "*.prt" file.
The only restriction is that no additional arguments can be given to NXstart.bat, so when more preferences are required like the version of NX, etc. this has to be added to the NXstart.conf file.
The Teamcenter integration and allowing multiple sessions will be inactivated when NXcustom is launched by double clicking the part file.

To use NXcustom when starting NX from the rich client, the following script must be executed once on every client machine which needs this functionality. The following script in the …\Startup\Admin folder must be executed:

set_PRT_association.bat

To restore the original situation (default behavior), the following script must be executed once on every client machine which needs default functionality restored. The following script in the …\Startup\Admin folder must be executed:

restore_PRT_association.bat

Depending on the security policy, it might be required to execute this with administrative privileges.
The script will notify you when the command failed.

TEAMCENTER SUPPORT

NXcustom supports the launching of NX from the Rich Client (RAC), Active Workspace (AW) and directly as NXmanager mode.

LAUNCHING NX FROM THE RICH CLIENT

To start NX with Teamcenter Integration, simply run the NXstart.bat script with the arguments "/managed true" like shown below:

NXstart.bat /managed true

To use NXcustom when starting NX from the rich client, the following script must be executed once on every client machine which needs this functionality. The following script in the …\Startup\Admin folder must be executed:

set_link_from_RAC.bat

To restore the original situation (default behavior), the following script must be executed once on every client machine which needs default functionality restored. The following script in the …\Startup\Admin folder must be executed:

restore_link_from_RAC.bat

Depending on the security policy, it might be required to execute this with administrative privileges.
The script will notify you when the command failed.

LAUNCHING NX FROM ACTIVE WORKSPACE

To use NXcustom when starting NX from Active Workspace, the following script must be executed once on every client machine which needs this functionality. The following script in the …\Startup\Admin folder must be executed:

set_link_from_AW.bat

To restore the original situation (default behavior), the following script must be executed once on every client machine which needs default functionality restored. The following script in the …\Startup\Admin folder must be executed:

restore_link_from_AW.bat

Depending on the security policy, it might be required to execute this with administrative privileges. The script will notify you when the command failed.

LAUNCHING NX DIRECTLY IN NXMANAGER MODE

When you launch NXcustom directly in managed mode (NXstart.bat /managed TRUE), it is possible to define the Teamcenter configuration by making use of the "TCenv_xxxx.conf" file.

A template for this file is found in the "…\NXstartup\Configuration_Templates" folder.
Copy the file to the "…\NXstartup" folder and change "xxxx" to your environment name (ie. prod, uat or test)
Use the Teamcenter and NXintegration documentation to determine which variables to set

To select one of the TCenv_xxxx.conf files you need to add the "/tcenv xxxx" argument to NXstart.bat. Example:

- NXstart.bat /managed TRUE /tcenv prod

Or set those settings in the NXstart.conf file.

DRAFTING STANDARDS

NXcustom now refers to the OOTB Drafting Standards that come with the installation. When you save a custom Drafting Standard it will be placed in the:

For Site level in: …\NXcustom\NXxxxxlibrary\CustomerDefaults\Site\startup

For Group level in: …\NXcustom\NXxxxxlibrary\CustomerDefaults\"Groupx"\startup

For User level in: ${UGII_USER_PROFILE_DIR}\drafting_standards

Or when separate Teamcenter settings are used, in the:

For Site level in: …\NXcustom\NXxxxxlibrary\CustomerDefaults\TC\Site\startup

For Group level in: …\NXcustom\NXxxxxlibrary\CustomerDefaults\TC\"Groupx"\startup

For User level in: ${UGII_USER_PROFILE_DIR}\drafting_standards

NOTE: When the "drafting_standards" folder does not exist in the ${UGII_USER_PROFILE_DIR} folder, a user will not be able to save Drafting standards. This will prevent users from accidentally save his own user standards, and possibly not following the company standards.

This keeps the environment cleaner, and you only have to copy the content of your custom standards to the new NXcustom version, without bothering about changes in the OOTB content.

CONTACT

If you have questions or comments, please contact:

Gerrit Koelewijn or: Eric Magray

Gerrit.Koelewijn@siemens.com Eric.Magray@siemens.com