

COM/ReconNX Release 376 and following

Features, Installation & Administration Manual



Copyright

© 2019 T-Systems International GmbH.
All rights reserved. Printed in Germany.

Contact

T-Systems International GmbH
GDC Product Lifecycle Management
Fasanenweg 5
70771 Leinfelden-Echterdingen
Germany

Manual History

Version	Date
1.0	June 2018
1.1	July 2019

Your Comments are Welcome

Please feel free to tell us your opinion; we are always interested in improving our publications. Mail your comments to:

T-Systems International GmbH
GDC Product Lifecycle Management
Fasanenweg 5
70771 Leinfelden-Echterdingen
Germany

mail: support.comfox@t-systems.com

Preface

About this Manual

This manual describes the features of COM/ReconNX, and it provides installation and configuration information for COM/ReconNX. Before using this guide, be sure you understand:

- the Microsoft Windows operating system
- the administration of the NX system

Related Documents

T-Systems Licman 2.0 Installation Manual.

Trademarks

NX is a registered trademark of Siemens PLM.

Names of other products mentioned in this manual are used for identification purpose only and may be trademarks of their companies.

Table of Contents

CHAPTER 1	7
OVERVIEW	7
FEATURES	7
SYSTEM AND SOFTWARE REQUIREMENTS	7
CHAPTER 2	8
FEATURES OF COM/RECONNX	8
USE CASE 1: SCAN	9
<i>Options for use case 1: Scan</i>	<i>10</i>
USE CASE 1B: FLAT SCAN	11
USE CASE 2: RECONCILE	11
<i>Options for use case 2: Reconcile</i>	<i>12</i>
<i>PLMXML specification for use case 2: Reconcile</i>	<i>13</i>
CHAPTER 3	14
INSTALLATION / TROUBLESHOOTING	14
INSTALLATION STEPS	14
<i>Licman license</i>	<i>14</i>
TROUBLESHOOTING	15
RETURN CODES	15
APPENDIX	17
SAMPLE PLMXML RESULT	17

CHAPTER 1

Overview

This chapter provides basic information about the features and installation of the COM/ReconNX batch processing tool.

Features

COM/ReconNX is a batch processing tool, which has two major 'use cases':

- Scan a NX assembly structure and create a PLMXML file representing this structure, including transformations and meta data.
- Reconcile an NX prt/jt structure according to the 'master structure' being defined in a PLMXML file.

For details, refer to the major chapter 'Features of COM/ReconNX'

System and Software Requirements

Depending on the NX Release COM/ReconNX is built for, you will need to have installed the appropriate NX Client version, e.g. NX 11.x.x.x.

It will require the following NX licenses:

- gateway
- solid_modeling
- drafting
- assemblies

Note that parallel processing will require this license package for each single process (even when being run with just one user)

COM/ReconNX is currently supported on the following operation systems:

- Windows 7 (64 Bit)
- Windows 10 (64 Bit)

It is recommended to have at least 8 GB RAM available, because loading NX data takes about 10-15x of the file size in memory ('1 GB on disk will take 10 GB in RAM')

CHAPTER 2

Features of COM/ReconNX

COM/ReconNX is a batch processing tool, which has two major 'use cases':

Scan an NX assembly structure and create a PLMXML file representing this structure, including transformations and meta data.

Reconcile an NX prt/jt structure according to the 'master structure' being defined in a PLMXML file.

For both use cases, COM/ReconNX can handle structures with following file types:

prt

jt

*_jt.prt (JT wrapper files)

Use case 1: Scan

In 'scan' mode, COM/ReconNX will only **read** NX data, it will not alter any file.

COM/ReconNX opens a root assembly prt, and recursively traverses the assembly structure. All info will be written to a **result PLMXML file**. A sample for such a PLMXML file can be found in the Appendix, [Sample PLMXML result](#)

Optionally, the input 'file' can be a **directory**. In this case COM/ReconNX will search all root assembly prts in this directory, and will create a 'multi-root' assembly structure.

Optionally, **drawing** prts in the input directory can be analyzed for dependencies to the assembly structure.

Note, that a 'flat scan' of referenced files is possible via the option `-ScanFlat`, see following 'Use case 1b: flat scan'

For the scan of meta data, COM/ReconNX distinguishes between 'standard' meta data (PARTNUMBER, NOMENCLATURE, REVISION) and the 'other' properties (within the 'part properties' of a prt):

 <No Category>	
..... DEFINITION	
..... NOMENCLATURE	Arretierbolzen
..... PARTNUMBER	ARR_20110628_AB
..... REVISION	0001,1

By default, all **properties** are read, and subsequently written to PLMXML like follows:

```
<Part id="id8" name="ARR_20110628_AB" type="assembly" instanceRefs="id27">
  <UserData id="ud8">
    <UserValue title="Product Identifier" value="ARR_20110628_AB"></UserValue>
    <UserValue title="Product File Name" value="ARR_20110628_AB.prt"></UserValue>
    <UserValue title="Product Revision" value="0001,1"></UserValue>
    <UserValue title="Product Label" value="Arretierbolzen"></UserValue>
    <UserValue title="CXP_NODETYPE" value="ASM-PRT"></UserValue>
    <UserValue title="USERPROP::DEFINITION" value=""></UserValue>
    <UserValue title="USERPROP::NX_MaterialMissingAssignments" value="TRUE"></UserValue>
    <UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
  </UserData>
</Part>
```

Filtering of 'other' properties is possible via the options `MetaDataAddtlPropsPart` and `MetaDataAddtlPropsProduct`, see below

Options for use case 1: Scan

Note, that

```
<install dir>\go\scanNXToPLMXML.opt
```

contains a **sample option file** for this use case.

The following options are relevant for this use case.

Options marked with (+) must be set, but have only 'internal semantics'

Option values in [] are default, string-value options with (*) are optional

-InFile

Full path of the input prt (or directory) being subject to scan

-OutFormat

Should be set to PLMXML

-Reverse

Set to 'Yes' to activate use case 1 (scan)

-OutFile

Full path of the output PLMXML file

-Verbosity

Trace/Info/[Warning]/Error : verbosity of reporting

-CapsVerbosity

Trace/Info/[Warning]/Error : verbosity of reporting of the additional capper library

-DrawAssocAnalysis

[Yes]/No: analyze drawings prts in the input directory (-InFile),
and try to assign them to the scanned assembly structure

-MetaDataAddtlPropsPart (*)

-MetaDataAddtlPropsProduct (*)

Additional properties to be scanned from geo/asm prts.

Value is either:

* ('star') - scan all user properties and write them as such to PLMXML

Or : a separated list of user property name pairs like

(NX_MaterialMissingAssignments:NXMMA) (...,...) (...,...)

In this case, a property NX_MaterialMissingAssignments would be read
(if present), and written to PLMXML with the property name NXMMA

Use case 1b: Flat Scan

A very quick possibility to scan a set of prt documents for all its dependencies is given via the option **-ScanFlat**. If this option is set, all prts of the **-InFile** directory are scanned for their dependencies (prt,jt). The complete list of the dependent files (without path) is written to a simple text file, no assembly structure or meta data will be scanned.

This use case only works in combination of the following options:

```
-ScanFlat=Yes
-Reverse=Yes
-InFile=<directory subject to flat scan>
-OutFile=<complete path to result txt file>
-OutFormat=TXT
```

Contents of a sample output file:

```
A2046220035.prt
A2046220035_1.prt
A2046220035_2.jt
```

Note, that

```
<install dir>\go\scanNXToTXT.opt
```

contains a **sample option file** for this use case.

Use case 2: Reconcile

In 'reconciliation' mode, COM/ReconNX modifies NX assemblies according to the 'master structure definition' within a PLMXML file..

The following picture illustrates the way COM/ReconNX works:

- 1) COM/ReconNX will **read** a 'master structure' being defined in a PLMXML file.
- 2) It will then try to open all NX files being referenced in the 'master structure' (being located in the same input directory)
- 3) It will propagate the 'master structure' onto the NX structure.
- 4) The modified NX files will be saved

This propagation applies to the following functionalities:

- Adjust part numbers, instance names, and other meta data
- Adjust file names (via 'Assembly Clone')
- Add and remove instances
- Create assembly prts where appropriate
- Adjust transformations
- Remove broken links

Options for use case 2: Reconcile

Note, that

```
<install dir>\go\synchronizeNXFromPLMXML.opt
```

contains a **sample option file** for this use case.

The following options are relevant for this use case.

Option values in [] are default, string-value options with (*) are optional

-InFile

Full path of the input PLMXML file ('master structure')

-OutFormat

Should be set to NX

-Verbosity

Trace/Info/[Warning]/Error : verbosity of reporting

-CapsVerbosity

Trace/Info/[Warning]/Error : verbosity of reporting of the additional capper library

-CPTemplate (*)

Where new assembly prts need to be created,
use a given file as template; specify the full path.

-ActivateSendTo

Yes/[No]: Allow rename of files

-CleanBrokenLinks

Yes/No/[PerMaster]: Cleanup broken links in the NX assembly.
For PerMaster, all broken links which have no equivalent
in the master structure get deleted

PLMXML specification for use case 2: Reconcile

While PLMXML is standardized for a long a time, a lot of different flavours are around.

COM/ReconNX will correctly interpret the assembly structure being defined in PLMXML in many cases. However, there are several rules which have to be taken into account, if **meta data reconciliation** should work as designed.

Basically these rules can be seen in the PLMXML result files from *Use Case 1: Scan*.

The mapping of PLMXML UserValue titles to NX standard properties is as follows

Product Identifier	Part Number
Product Revision	Revision
Product Label	Nomenclature
Product Definition	Definition

If a **rename of files** is desired (see option `ActivateSendTo`), the target file names must be defined in an additional UserValue:

Product File Name New	< target file name, no path>
-----------------------	------------------------------

The UserValues titles can be specified with or without trailing '::'

CHAPTER 3

Installation / Troubleshooting

Installation steps

COM/ReconNX is a NXOpen C++ batch application, it requires to be run in a proper NX environment.

For the installation of COM/ReconNX, you just have to unpack the delivered package to an appropriate directory `<install dir>`, either on a client machine, or on a network drive.

(It is recommended, that `<install dir>` does not contain blanks!)

After having done this, please have an eye on the script

```
<install dir>\go\COMReconNX.bat
```

This script is the main entry point for starting COM/ReconNX. Typically, the tool is started from the command line like:

```
COMReconNX.bat <Full path of option file>
```

Before using the script and an option file, you have to adjust a few places in COMReconNX.bat and in the option file being used.

Within COMReconNX.bat , you will find places marked with **ADJUST_HERE**. These are :

```
rem Setup the path INSTALL_DIR to the COM/Recon install path
set INSTALL_DIR=C:\tmp\180322_COMRecon

rem Setup the NX installation path,
rem typically sth like >C:\Progra~1\Siemens\NX11.0<
set UGII_BASE_DIR=C:\PROGRA~1\Siemens\NX11~1.0
```

For adaptations in the option files, please refer to the chapter 'Features of COM/ReconNX'

Licman license

COM/ReconNX also requires a Licman license setup, please refer to the *Licman 2.0 Installation Manual*.

The Licman key being used by COM/ReconNX is: **612 or 614**

This is important when requesting a license from T-Systems.

Troubleshooting

If CxpBatch.exe fails to start with the message of a missing dll, this may be due to the following setup errors:

- 1) No NX is installed on the computer, where COM/ReconNX is installed
- 2) NX is installed with a wrong release level
- 3) The start script <install dir>\go\COMReconNX.bat is not set up correctly:

Most likely the environment setup is not correct; Hint: change the first line

```
@echo on
```

and check the output of the script.

You should also check the **[TROUBLESHOOTING]** hints in the start script

```
<install dir>\go\COMReconNX.bat
```

Another source of issues is the lack of appropriate Windows C++ redistributable packages.

All necessary packages can be found in the directory

```
<install dir>\vcredist
```

Return Codes

COM/Recon returns a value depending on the warnings/errors encountered during processing. The return codes are:

```
0      OK
202    Warning
203    Error, but data processing to the end
204    Severe Error = Abort with control;
        COM/Recon does not finish data processing to the end
```

With RC 203, it is recommended to check the error(s) in the log and the result data

With RC 204, there is definitely something wrong, the result data should not be used

APPENDIX

Sample PLMXML result

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<PLMXML xmlns="http://www.plmxml.org/Schemas/PLMXMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.plmxml.org/Schemas/PLMXMLSchema
    PLMXMLPDMSchema.xsd"
  xmlns:vis="PLMXML-VisExtensions"
  schemaVersion="1.0"
  author="COM/ReconNX Conversion to PLMXML"
  time="11:53:18"
  date="2018-02-09">

  <!-- created by COM/ReconNX Version 376.1, Fri Feb 09 11:53:18 2018 -->
  <!-- # (c) T-Systems 2018 -->

  <ProductDef id="pd1" name="Dummy Product Definition">
    <InstanceGraph id="igl" rootInstanceRef="id99999999">
      <Description>Assembly structure for id1</Description>

      <Instance id="id99999999" partRef="id1">
        <Description>ARR_20110628_TOP</Description>
        <Transform id="tr99999999">1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1</Transform>
      </Instance>

      <Part id="id1" name="ARR_20110628_TOP" type="assembly" instanceRefs="id16 id17 id18
        id19 id20 id21 id22">
        <UserData id="ud1">
          <UserValue title="Product Identifier" value="ARR_20110628_TOP"></UserValue>
          <UserValue title="Product File Name" value="ARR_20110628_TOP.prt"></UserValue>
          <UserValue title="Creation label" value="COM/ReconNX Version 376.1"></UserValue>
          <UserValue title="Product Revision" value="0001,1"></UserValue>
          <UserValue title="Product Label" value="Örretierung_Üssm"></UserValue>
          <UserValue title="CXP_NODETYPE" value="ASM-PRT"></UserValue>
          <UserValue title="USERPROP::DEFINITION" value=""></UserValue>
          <UserValue title="USERPROP::GRPVRSN" value="stlformat_laden V13.0"></UserValue>
          <UserValue title="USERPROP::NX_MaterialMissingAssignments"
            value="TRUE"></UserValue>
          <UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
          <UserValue title="USERPROP::RNI_PARTTYPE" value="ASM"></UserValue>
          <UserValue title="USERPROP::SECTION-COMPONENT" value="NO"></UserValue>
        </UserData>
      </Part>
    </InstanceGraph>
  </ProductDef>
</PLMXML>
```

```

<UserValue title="USERPROP::SF_BENEN" value="Arretierung"></UserValue>
</UserData>
<Representation id="repl" format="prt" location="ARR_20110628_TOP.prt">
</Representation>
</Part>

<Instance id="idl6" partRef="id2">
<Description></Description>
<Transform id="tr16">1.0000000000 0.0000000000 0.0000000000 0.0000000000
0.0000000000 1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000</Transform>
</Instance>

<Instance id="idl7" partRef="id3">
<Description>ARR_20110628_S 1</Description>
<Transform id="tr17">1.0000000000 0.0000000000 0.0000000000 0.0000000000
0.0000000000 1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000</Transform>
</Instance>

<Instance id="idl8" partRef="id6">
<Description>ARR_20110628_ZS 1</Description>
<Transform id="tr18">-0.8371691043 0.0000000000 -0.5469441386 0.0000000000
0.0000000000 -1.0000000000 0.0000000000 0.0000000000 -0.5469441386 0.0000000000
0.8371691043 0.0000000000 0.1155733899 0.0309000000 0.0344073869
1.0000000000</Transform>
</Instance>

<Instance id="id22" partRef="id14">
<Description>ARR_20110628_FG 1</Description>
<Transform id="tr22">1.0000000000 0.0000000000 0.0000000000 0.0000000000
0.0000000000 1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000</Transform>
</Instance>

<Part id="id2" name="A2046390494_FP001_1_jt">
<UserData id="ud2">
<UserValue title="Product Identifier" value="A2046390494_FP001_1_jt"></UserValue>
<UserValue title="Product File Name"
value="A2046390494_FP001_1_jt.prt"></UserValue>
<UserValue title="CXP_NODETYPE" value="GEO-PRT"></UserValue>
<UserValue title="USERPROP::Creation label" value="T-Systems COM/FOX Version
2.5.1"></UserValue>
<UserValue title="USERPROP::NX_MaterialMissingAssignments"
value="TRUE"></UserValue>
<UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
<UserValue title="USERPROP::Name" value="A2046390494_FP001_1"></UserValue>
<UserValue title="USERPROP::UGII_JT_DETAILS"
value="2|1130315333|1|1|1|X:\rofrank\data_nx\i708_rotations_freemarket\orig\A204639
0494_FP001_1.jt"></UserValue>
<UserValue title="USERPROP::UGII_JT_FILE_VERSION" value="7|0"></UserValue>

```

```

<UserValue title="USERPROP::UGII_JT_PMI_IMPORTED" value="1"></UserValue>
<UserValue title="USERPROP::UGII_JT_WIREFRAME_REF_GEOMETRY_IMPORTED" value="1"
type="int"></UserValue>
</UserData>
<Representation id="rep2" format="prt" location="A2046390494_FP001_1_jt.prt">
</Representation>
</Part>

<Part id="id3" name="ARR_20110628_S" type="assembly" instanceRefs="id23 id24">
<UserData id="ud3">
<UserValue title="Product Identifier" value="ARR_20110628_S"></UserValue>
<UserValue title="Product File Name" value="ARR_20110628_S.prt"></UserValue>
<UserValue title="Product Revision" value="0001,1"></UserValue>
<UserValue title="Product Label" value="Schrauben"></UserValue>
<UserValue title="CXP_NODETYPE" value="ASM-PRT"></UserValue>
<UserValue title="USERPROP::DEFINITION" value=""></UserValue>
<UserValue title="USERPROP::NX_MaterialMissingAssignments"
value="TRUE"></UserValue>
<UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
<UserValue title="USERPROP::RNI_PARTTYPE" value="ASM"></UserValue>
</UserData>
<Representation id="rep3" format="prt" location="ARR_20110628_S.prt">
</Representation>
</Part>

<Instance id="id23" partRef="id4">
<Description>ARR_20110628_SS 1</Description>
<Transform id="tr23">1.0000000000 0.0000000000 0.0000000000 0.0000000000
0.0000000000 1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000</Transform>
</Instance>

<Instance id="id24" partRef="id4">
<Description>ARR_20110628_SS 2</Description>
<Transform id="tr24">0.9398610203 0.3415571146 0.0000000000 0.0000000000 -
0.3415571146 0.9398610203 0.0000000000 0.0000000000 0.0000000000 0.0000000000
1.0000000000 0.0000000000 0.0666268312 -0.0076368432 0.0000000000
1.0000000000</Transform>
</Instance>

<Part id="id4" name="ARR_20110628_SS" type="assembly" instanceRefs="id25">
<UserData id="ud4">
<UserValue title="Product Identifier" value="ARR_20110628_SS"></UserValue>
<UserValue title="Product File Name" value="ARR_20110628_SS.prt"></UserValue>
<UserValue title="Product Revision" value="0001,1"></UserValue>
<UserValue title="Product Label" value="Senkschraube"></UserValue>
<UserValue title="CXP_NODETYPE" value="ASM-PRT"></UserValue>
<UserValue title="USERPROP::DEFINITION" value=""></UserValue>
<UserValue title="USERPROP::NX_MaterialMissingAssignments"
value="TRUE"></UserValue>

```

```

<UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
<UserValue title="USERPROP::RNI_PARTTYPE" value="ASM"></UserValue>
</UserData>
<Representation id="rep4" format="prt" location="ARR_20110628_SS.prt">
</Representation>
</Part>

<Instance id="id25" partRef="id5">
<Description>ARR_20110628_SS_1_1</Description>
<Transform id="tr25">1.0000000000 0.0000000000 0.0000000000 0.0000000000
0.0000000000 0.0000000000 1.0000000000 0.0000000000 0.0000000000 -1.0000000000
0.0000000000 0.0000000000 0.0250000000 0.0150000000 -0.0005000000
1.0000000000</Transform>
</Instance>

<Part id="id5" name="ARR_20110628_SS_1">
<UserData id="ud5">
<UserValue title="Product Identifier" value="ARR_20110628_SS_1"></UserValue>
<UserValue title="Product File Name" value="ARR_20110628_SS_1.prt"></UserValue>
<UserValue title="Product Revision" value="1"></UserValue>
<UserValue title="Product Label" value="senkschraube"></UserValue>
<UserValue title="CXP_NODETYPE" value="GEO-PRT"></UserValue>
<UserValue title="USERPROP::BENENNUNG" value="Senkschraube"></UserValue>
<UserValue title="USERPROP::DEFINITION" value=""></UserValue>
<UserValue title="USERPROP::EINHEIT" value="Stck"></UserValue>
<UserValue title="USERPROP::GRPVRSN" value="section_component V13.0"></UserValue>
<UserValue title="USERPROP::NUMMER" value="DIN 963 M10x16"></UserValue>
<UserValue title="USERPROP::NX_MaterialMissingAssignments"
value="TRUE"></UserValue>
<UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>
<UserValue title="USERPROP::RNI_PARTTYPE" value="PRT"></UserValue>
<UserValue title="USERPROP::SECTION-COMPONENT" value="NO"></UserValue>
<UserValue title="USERPROP::WERKSTOFF" value="Stahl"></UserValue>
</UserData>
<Representation id="rep5" format="prt" location="ARR_20110628_SS_1.prt">
</Representation>
</Part>

<Part id="id6" name="ARR_20110628_ZS" type="assembly" instanceRefs="id26">
<UserData id="ud6">
<UserValue title="Product Identifier" value="ARR_20110628_ZS"></UserValue>
<UserValue title="Product File Name" value="ARR_20110628_ZS.prt"></UserValue>
<UserValue title="Product Revision" value="0001,1"></UserValue>
<UserValue title="Product Label" value="Zylinderstift"></UserValue>
<UserValue title="CXP_NODETYPE" value="ASM-PRT"></UserValue>
<UserValue title="USERPROP::DEFINITION" value=""></UserValue>
<UserValue title="USERPROP::NX_MaterialMissingAssignments"
value="TRUE"></UserValue>
<UserValue title="USERPROP::NX_MaterialMultipleAssigned" value="FALSE"></UserValue>

```

```
<UserValue title="USERPROP::RNI_PARTTYPE" value="ASM"></UserValue>
</UserData>
<Representation id="rep6" format="prt" location="ARR_20110628_ZS.prt">
</Representation>
</Part>

</InstanceGraph>
</ProductDef>
</PLMXML>
```