Legend™ for PDMLink™

Legend Best Practices Guide

Release 8.0

Author: Mark Winnie
I-Cubed

Last Updated: 07 August 2006

Document Control

Purpose

This document describes the typical implementation of Legend for PDMLink. This document is based on the implementation of the following version(s) of software.

Name	Release
Legend for PDMLink	8.0

References

- Legend for PDMLink User's Guide – This guide provides step-by-step instructions as well as installation and configuration information for using Legend for PDMLink.

In this document, the word *Legend* refers to the Legend for Windchill PDMLink product.

Contents

Legend Migration Scenarios

Scenario 1

Source: File system
Breadth: Latest iteration only
Objects: Create WTParts, EPMDocs and WTDocs, or create EPMDocs and WTDocs only

Overview

The simplest of all scenarios, the model files exist in the file system and the latest iteration of any file will be loaded.

Features

Legend features providing the most benefits are:

- Duplicate file filter – Configurable rules to select the most appropriate file when duplicates are identified along with complete reporting of duplicates.
- Duplicate object report – Reports of duplicate object names, including instances from different family tables and objects already in Windchill.
- VerDes – Supplemental tool to designate Pro/E parameters and verify family table instances.
- Reader Detail cleansing reports – A suite of reports, including object dependencies, metadata details, and destinations.

Caveats

Consistent file naming and standard parameters enhance the effectiveness. Inconsistent naming can increase preprocessing times. A standard set of parameters increases the value of the migrated objects.

Process

The general process includes the following:

- File source identification – Identify those directories containing the files to be migrated.
- Preprocessing – This is done if the family table instances are not verified or parameters need to be designated.
- Migrate files – Migration is normally done groups according to destination repository.

Scenario 2

Source: File system
Breadth: Latest iteration only
Objects: Preload WTParts and create EPMDocs and WTDocs

Overview

The scenario handles the case where the model files exist in the file system and the latest iteration of any file will be loaded, but there are metadata to be placed into IBA's on the WTPart objects that are not in Pro/E parameters.

Features

Legend features providing the most benefits are:

- Duplicate file filter – Configurable rules to select the most appropriate file when duplicates are identified along with complete reporting of duplicates.
- Duplicate object report – Reports of duplicate object names, including instances from different family tables and objects already in Windchill.
- VerDes – Supplemental tool to designate Pro/E parameters and verify family table instances.
- Reader Detail cleansing reports – A suite of reports, including object dependencies, metadata details, and destinations.

Caveats

The WTParts will need to be preloaded, possibly with the Windchill bulk loader. The WTParts should be loaded into the Legend user's personal Windchill cabinet. Again, consistent file naming and standard parameters enhance the effectiveness. Inconsistent naming can increase preprocessing times. A standard set of parameters increases the value of the migrated objects.

Process

The general process includes the following:

- File source identification – Identify those directories containing the files to be migrated.
- Preprocessing – This is done if the family table instances are not verified or parameters need to be designated.
- Preload WTParts – This is done outside of the Legend process.
- Migrate files – Migration is normally done groups according to destination repository.

Scenario 3

Source: CSV file
Breadth: All iterations of all revisions
Objects: Create WTParts, EPMDocs and WTDocs, or create EPMDocs and WTDocs only

Overview

The scenario is normally done when the files currently exist in another database, such as Pro/INTRALINK. The metadata and content are exported from the existing database. That export is then used to migrate into Windchill.

Features

Legend features providing the most benefits are:

- CSV source file – The files to migrate and their associated metadata are in a CSV file. Metadata from the CSV file is merged with metadata captured from the files, themselves, to produce a rich data migration.
- CSV attribute mapping – A GUI that provides the capability to map CSV metadata to Windchill attributes, and IBA's for file types other than Pro/E.
- Duplicate object report – Reports of duplicate object names, including instances from different family tables and objects already in Windchill.
- VerDes – Supplemental tool to designate Pro/E parameters and verify family table instances, not required for Pro/INTRALINK source databases.
- Reader Detail cleansing reports – A suite of reports, including object dependencies, metadata details, and destinations.

Caveats

Legend can call a user written routine to associate a given version of a dependent file to a given version of the file that depends upon it.

Process

The general process includes the following:

- Metadata export – Export the metadata from the existing database to a CSV file. This is done outside of the Legend process.
- Content export – Export the content files from the existing database to a CSV file. If the file names are encrypted, they will need to be renamed to their native file name. This is done outside of the Legend process.
- Preprocessing – This is done if the family table instances are not verified or parameters need to be designated.
- Migrate files – Migrate the files using the CSV file as the file source.

Scenario 3 Example

Pro/INTRALINK to PDMLink

This is a recommended migration strategy for migrating Pro/E files from Pro/INTRALINK to PDMLink.

Migration Steps

Migration can be broken up into several separate migrations.

- **A.** All released Pro/E library files.
- **B.** Released Pro/E files by project
- **C.** Pro/E files that are newer than the latest released iteration that has been migrated.

Export metadata: Export metadata from INTRALINK. Export all released Pro/Engineer files for the libraries or projects for the dataset being migrated.

Create staging directory: Create a local directory to hold the content files. If exporting more than one revision of a given object, a subdirectory should include the revision of the object. The file path in the metadata export file must agree with the actual directory of the content file.

Export content: Export the content files and place into their appropriate staging directory. The Pro/E supplemental program mdimpex can be used to export the content files.

Run Legend without loading: Run Legend using the metadata export file as input. Run the Legend steps up to and including the Reader. Save and exit Legend.

Review Legend Reader reports: Check data cleanliness. Review each report, identify anomalies and exceptions. Correct for anomalies and exceptions. Rerun Legend project Checking and Reading, then review, again.

Run Legend to load files: Run Legend Loader. Review loader reports. Review objects in PDMLink.

Project Overview

This guide gives detailed steps and information about migrating data with Legend for PDMLink.

Project Deliverables

The overall project deliverable is the configuration of Legend, a Migration Guide with detailed instructions for running Legend at this site, and a trained user who can run Legend.

Project Steps

Step 1 – Installation. Verify Legend installation based on customer requirements. Run verification test to check the success of installation.

Step 2 – Configuration. Configure Legend to handle the specific requirements of the customer data.

Step 3 – Customization. Create and test required VisualBasic custom routines.

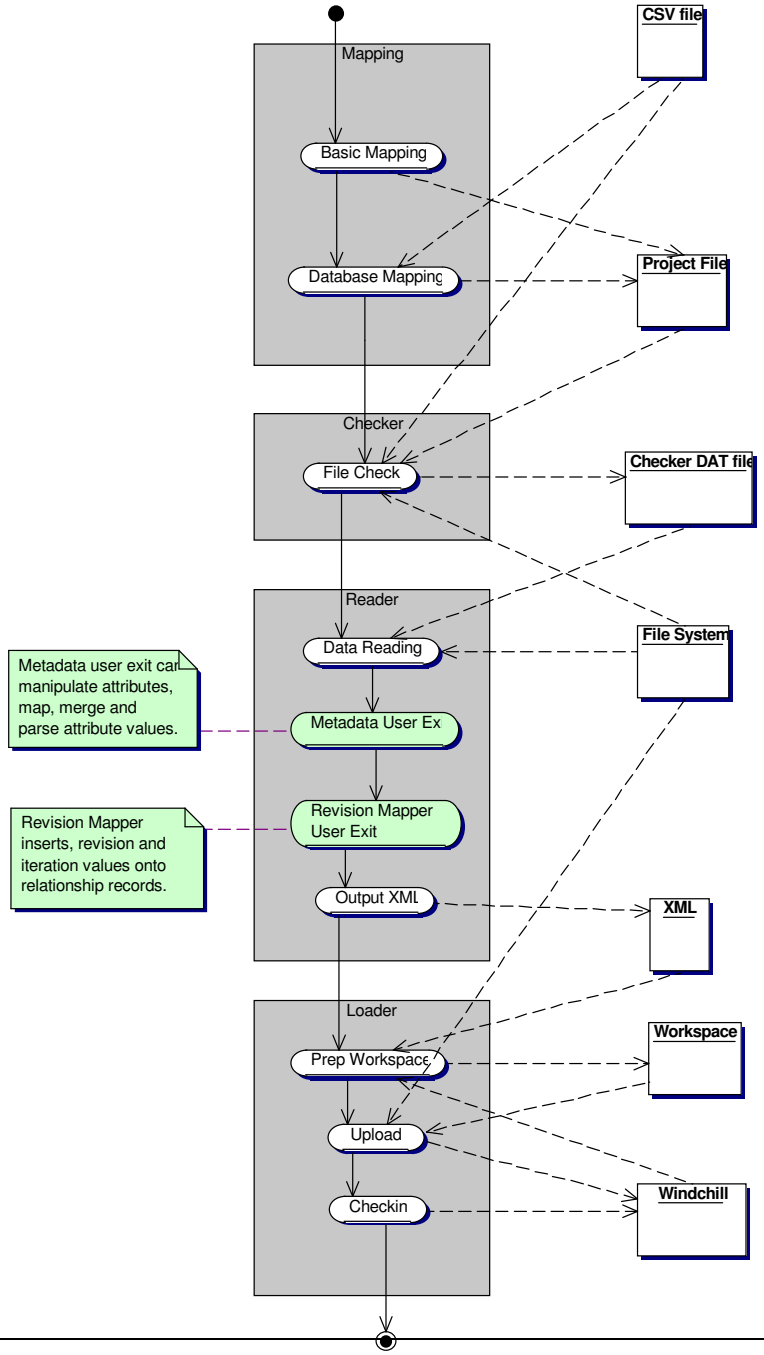
Step 4 – Database Export. Export from the database to a Comma-Separated Values (CSV) file and review the data.

Step 5 – Load Documents from a File System. Load files from the file system.

Step 6 – Load Documents from a CSV File. Load files into Windchill using CSV files created from a Database.

Legend Process Overview

The following diagram shows the Legend process along with the major files involved. The activities in green (with notes attached) are customizable steps within Legend.



Section 1 – Installation and Configuration

Step 1 - Installation

During this step, install Legend and the utilities that work with Legend.

1.1. Prerequisites

- **Legend Client Requirements**
  - Windows 2000 or XP
  - 500 MB of free disk space
  - AutoCAD 2004 for AutoCAD files
  - Inventor 2004 for Inventor files
  - SolidWorks 2004 or 2005 for SolidWorks files
  - Internet Explorer 6.0
  - Internet access
    - www.i-cubedcart.com (port 80)
    - legend-license.i-cubed.com:8080
  - Direct RMI and HTTP access to the Windchill server
- **Server Requirements**
  - Windchill PDMLink 8.0
- **Optional**
  - Visual Basic development platform, for customizations
- **Client Preparation**
  - Verify OS.
  - Verify free disk space.
  - Verify internet access. Internet access is used for installing Legend and acquiring Legend licenses. If not available, contact LegendSales@I-cubed.com for alternate access to licenses.
- **Server Preparation**
  -

1.2. Licenses

- Licenses should be purchased before you begin installation. If you do not have your license order information when you install, you will need to skip the licensing step of the installation, and then update your licenses at a later time. See below.
- The license information you need is:
  - Legend order number
  - E-mail address that created your license order
  - Your Windchill server host name
  - Your Windchill server domain name
- Your Legend licenses will be usable by any clients that are loading to this Windchill server.

1.3. Installation

Installation from Web site

- Go to www.LegendForPDMLink.com.
- Click **Downloads**.

- Click **Software Downloads Page.**
- Select **Download Installer** for your Windchill version.
- Run the Legend Installer. All modules will be installed, but only those modules for which you have a license will be activated.
- Follow the instructions displayed.
- If you have your license order information, enter when asked. If you do not, skip the licensing and see Updating a License, below.
- Review the _legend_vars.bat file and edit as required.
    - Verify the Windchill server URL is correct (WINDCHILLDIR)
    - Verify the CATIA PATH directories are correct (CATIA_PATH)
    - Add the following to force Pro/E instance status in PDMLink to verified
        - set UWGM_MARK_J01_INSTANCES_VERIFIED=t

Install from Existing Installation

- Copy complete Legend installation directory
- Edit _legend_vars.bat
    - Change *Legend load point* definition, if required
    - Change CATIA Path variable, if required
- Edit wtcad.ini
    - Change workspace name
- Remove <*Legend_loadpoint*>\workspaces
- Edit VerDes\VerDes.ini, correcting paths, if applicable.
- Run VerDes\register_server.bat, if applicable
- Remove all files from <*Legend_loadpoint*>\Projects
- Remove all files from <*Legend_loadpoint*>\logs.

1.4. Creating the CATIA V5 Abstraction Library

The CATIA Abstraction Library must be built onsite with the onsite license of CAA-RADE. To build the abstraction library, download the Legend CATIA abstraction library builder from http://www.i-cubedcart.com/Abstraction_builder/

Review the appendix Building the CATIA Abstraction Library for prerequisites nad details.

You will need to know the location of the tck_init.bat and CAADir.txt files. Typically, these files are located as indicated, below.

tck_init.bat - <catia install dir>\B<version>\intel_a\code\command
CAADir.txt - <catia install dir>\B<version>\

1.5. Setting the CATIA V5 Environment

The following should be performed if Legend will be loading CATIA files:

- Open a command prompt
- Change directories to the Legend loadpoint
- Run _legend_vars.bat
- Run setcatenv. This will create a new Catia V5 environment called "DefaultEnvironment"

- Run cnext. This will attempt to start Catia V5 using the DefaultEnvironment.
- Resolve any licensing errors that occur
- Run cnext again to ensure that Catia V5 properly starts up without errors
- Exit the command prompt and run Legend normally. Reading of Catia V5 files should now work properly.Navigate to <*Legend_loadpoint*>\jars

1.6. Installing classes on the Server

UNIX Server

- Navigate to <*Legend_loadpoint*>\jars.
- Copy LegendEnterprise.jar to the Windchill codebase directory.
- Unjar the LegendEnterprise.jar file.
- Restart the method server.

Windows Server

- Navigate to <*Legend_loadpoint*>\ServerInstall.
- Copy the file named "serverinstall.exe" to a location on the Windchill server.
- On the Windchill server machine, execute "serverinstall.exe."
- Read the Installation Introduction. This introduction explains that a set of classes will be copied into the Windchill Codebase.
- Click **Next** to continue.
- Choose the Windchill Codebase folder. You may either manually enter a location into the text box or browse for a location by clicking **Choose…** Also, you can restore the default installation location by clicking **Restore Default Folder**.
- Click **Next** to continue.
- Review the Pre-installation summary.
- Click **Install** to install the Legend for PDMLink classes to the Windchill Codebase.
- Read the installation summary information and click **Done** to complete the server-side portion of Legend for PDMLink Installation.
- Restart the Windchill method server.

1.7. Updating a License

To update a license file, you need access to port 8080 of the Legend license server. The following steps explain how to update licenses from that server.

- Click **Start**.
- Click **Programs**.
- Click **Legend**.
- Click **Update Legend License**.
- Enter Order # and Email Address from the Order Confirmation email for the purchased package received from www.i-cubedcart.com, and update the Windchill server name (if necessary).
- Click **Update**.

1.8. Switching to another server

To switch Legend to load into another server do the following:

- Licenses
  Copy licenses from <*loadpoint*>\licenses\<*host*> into <*loadpoint*>\licenses
- _legend_vars.bat

- Edit the Windchill URL environment variable
- Wtcad.ini
  - Verify the following are set correctly:
    - Workspace.name
    - DocumentLifecycle
    - DocumentLifecycleState
    - PartLifecycle
    - PartLifeCycleState

1.9. Configuration for Installation Verification

Edit the wtcad.ini file. The following options should be set:

- workspace.name= (set to a named workspace)
- upload.batch.size=1
- upload.retries=0
- PartLifecycle=
- PartLifecycleState=
- DocumentLifecycle=
- DocumentLifecycleState=

1.10. Verify the Installation

- Verify the Legend installation by running the sample set of data found in <*Legend_loadpoint*>\SampleData.
- The steps are outlined in Appendix A of the *Legend for PDMLink User's Guide*.
- If you receive a message that a DLL was not found, check the Legend load point directory to see if that DLL has been renamed with a bak extension. If it has, remove the bak extension.

1.11. Installing Supplemental Utilities

- VerDes
  - Go to www.LegendForPDM.com
  - Click **Downloads**
  - Click **VerDes**
- VerDes performs the following functions:
  - Verify instances
  - Designate parameters
- XML2CSV is a utility that reads Legend XML and creates two files viewable by Excel. XML2CSV processes the XML prior to it being trimmed due to missing dependencies and duplicate objects.
  - <project>_cleaner_data_passed.txt
    Contains metadata for each object to be created
  - <project>_cleaner_data_passed_relations.txt
    Contains relationships for each object to be created
- Using VerDes with Model Check in Pro/E
  - A modification to the verify_template is required to run VerDes with Pro/E when Model Check is being used.
  - A newline and space needs to be added to the file after the " ~ Activate `main_dlg_cur` `ProCmdModelSave.file` "line.

1.12. Upgrading Legend

When upgrading Legend with a new build of the installer do the following:

- From the current Legend directory run unregister_servers.bat
- Make a backup of the current installation by renaming the current Legend client installation directory, e.g. Legend_8.0_old
- Download and run the Legend installer from the downloads page of www.LegendForPDMLink.com or from the location given you by the Legend support staff.
    - Install Legend into the originally named install directory, e.g. Legend_8.0
    - Click Next when the license request screen appears
    - Then click Retrieve licenses later
- When client installation completes copy the following files and directories from the old install directory to the new install directory:
    - <Legend_loadpoint>\Legend_Start.bat
    - <Legend_loadpoint>\licenses\
    - <Legend_loadpoint>\Source\
    - <Legend_loadpoint>\Files\
    - <Legend_loadpoint>\VerDes\, if applicable
- Compare the _legend_vars.bat files from the old and new Legend directories. Add any new variables that were shipped in the new _legend_vars.bat file to the old _legend_vars.bat files.
- Copy the _legend_vars*.bat files from the old to the new directory.
- Compare the wtcad.ini files from the old and new Legend directories. Add any new INI options that were shipped in the new wtcad.ini file to the old INI file.
- Copy the wtcad.ini file from the old directory to the new directory.
- Copy and Source/*/*.dll files that you have customized into the Legend 8.0 installation directory, unless the Legend_Start.bat copies the files at startup.
- Edit the Legend.bat file and change the –Xmx512M to –Xmx1024M
- Install the Legend classes to the Windchill server, see "Installing classes on the Server", above.
- For CATIA, copy the following files from the old Legend adapters CATIA directory to the new Legend adapters CATIA directory
    - CATIA5Abstraction.dll

1.13. Upgrading PDMLink

Legend uses a jar files containing classes and properties files from the PDMLink server. Changes to PDMLink that may include customizations such as new lifecycle states, new DocTypes, extensions to WTPart or WTDocuments, new enumerated types, xconf changes, temp patches or MOR installations..

After making the above changes to PDMLink do the following:

- At the server, make the following jars if they were not made as part of the install. To make the jar files, open a Windchill shell, cd to codebase and enter the command:

    ant -f bin/swmaint.xml make_jar.config_jars

- Delete the <Legend_load_point>\cache\<pdmlink_server> directory and all files under it. Legend will download the appropriate files from the server the next time Legend runs.

Step 2 - Configuration

2.1. Basic Configuration

Edit the wtcad.ini file. The following options should be set:

- upload.timeout=60 (if large family tables of over 1000 instances)
- filter.duplicates=
- workspace.name= (set to a named workspace, this workspace should only be used by Legend and only from 1 client. If you have multiple clients, each Legend client should have its own workspace.)
- upload.batch.size=1
- upload.retries=0
- PartLifecycle=
- PartLifecycleState= (Leave this blank if you have pre-existing WTParts and you want the State of the WTPart to remain unchanged)
- DocumentLifecycle=
- DocumentLifecycleState=
- dms.object.exists=
- create.iba.definitions=false
- link.drawings.to.parts=
- checker.strict.versioning= (Pro/E only)
- revision.mapping.enabled=
- debug=false
- [*CAD system*] create.associated.parts=
- [*database*] process.exact.iteration=

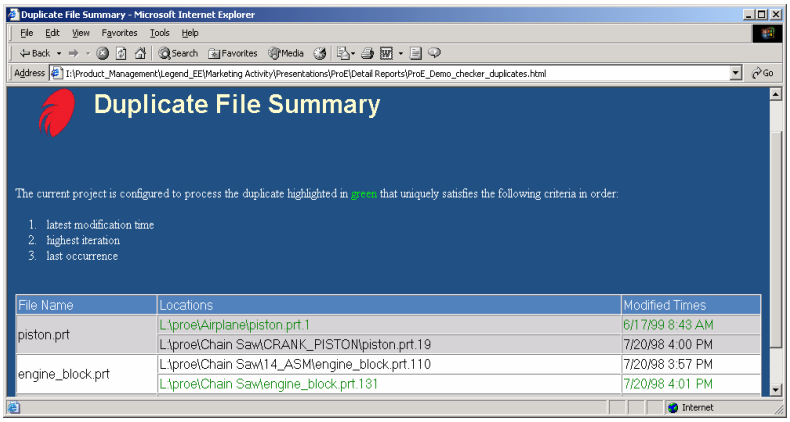
2.2. Filtering Duplicate Files

Use filter duplicates when loading only one version of all objects. Legend can identify duplicate files, select one of those files for processing based on a set of rules, and report which file will be processed and which ones are filtered out. To utilize this feature and to set the precedence of the selection rules, do the following:

- Edit the wtcad.ini file and set
  - filter.duplicates=true
  - process.highest.iteration=i
  - process.last.modified=j
  - process.last.occurrence=k

  where i, j, and k are integers representing rule precedence
- **process.highest.iteration** rule uses the Pro/ENGINEER file iteration number to select the file to process. Invoking this rule causes the file with the highest iteration number to be selected.
- **process.last.modification** rule uses the last modification date from the file system to select the file to process. Invoking this rule causes the file with the latest modification date to be selected.
- **process.last.occurrence** rule uses the order in which the files are processed to select the file to use. Invoking this rule causes the last file processed to be selected.

- Set the value for the rule you wish processed first to "1", set the value to "2" for the rule to process next and "3" for the rule to process last.
- When the Data Checker identifies duplicate files it will apply the rule that is configured with arecedence value of "1". If more than one file meets the rule criteria, the rule with precedence "2" will be applied. Rule "3" is applied if there are more than one file meeting criteria for both precedence "1" and "2" rules.



- The results of the Data Checker duplicate processing are shown in the Duplicate Files report. The order of rule precedence is shown at the beginning of the report. The duplicate file selected for processing is displayed in green. The duplicates filtered out are shown in black. See figure below.

- If the desired duplicates have been selected for processing, you can continue the Legend process. If the desired files were not selected, you can either change the rule precedence or remove the undesired files from the source directory. In either case, you must rerun the Data Checker.

2.3. Multiple Revisions and Iterations

When dealing with multiple revisions and iterations of files, use the following INI options.

Filter Duplicates

Turns duplicate file name filtering on or off. Files with the same name must represent unique revisions or iterations in PDMLink or they are filtered out before loading. Duplicate files are reported in the Duplicates Report of the Data Checker. This option must be set to false

`filter.duplicates=false`

Load Duplicates

Determines how objects with identifiers matching up with objects all ready in Windchill PDMLink should be loaded.

dms.object.exists=reject

The identifiers used for comparison vary based on one of the following scenarios:

- Loading a single revision of an object -- The identifier used for matching consists of the SP:DOC_NR metadata value from the read object's XML representation.

- Loading multiple revisions of an object -- The identifier used for matching consists of the SP:DOC_NR and SP:DOC_REV/SP:REV metadata values from the read object's XML representation
- Loading multiple iterations of a revision of an object -- The identifier used for matching consists of the SP:DOC_NR, SP:DOC_REV/SP:REV, and SP:DOC_ITERATION/SP:ITERATION metadata values from the read object's XML representation.

```
dms.object.exists=load
```

Valid settings are:

Load -- The objects will be loaded and will consequently update the existing Windchill PDMLink objects.

Reject -- The objects will not be loaded and consequently the existing Windchill PDMLink objects will not be affected.

Sort Read Objects

Setting that determines whether multiple revisions or iterations of objects should be reordered prior to loading.

```
sort.read.objects=true
```

Valid settings are:

True -- Reorders multiple revisions or iterations of objects prior to loading. The revisions or iterations are sorted lexicographically, in dictionary order, with additional checks implemented to handle rollovers, such as when going from revision Z to revision AA.

False -- Does not reorder multiple revisions or iterations of objects prior to loading. Set if:

(a) you want to keep the relative order of revisions or iterations as specified in a CSV file.

(b) you have implemented a custom series in PDMLink which may utilize an unusual sequence or incorporates multiple series.

Step 3 – Customization

Legend has several customization points for user-written Visual Basic routines. Edit the VB source to meet your specifications. See the Legend for PDMLink User's Guide chapter on Visual Basic plug-ins for more information.

3.1. Metadata User Exit

- The Metadata User Exit provides the capability to alter metadata values. You can add prefixes, parse out strings, and combine several pieces of data into one. You have the option to gather information from any accessible source to be used to combine or alter the metadata.
- The Metadata User Exit is called for each file after it is read during the Reader step.
- The Metadata User Exit sample source is located in <*Legend_loadpoint*>\Source\Metadata_User_Exit.
- Details of the calls available in the MetadataCustomizer are documented in the sample source code
- After editing the source, compile it into a DLL. In the VisualBasic editor select File / Make MetadataCustomizer.dll….
- Place the DLL into the <*Legend_loadpoint*> directory.

3.2. Custom Adapter

- The Custom Adapter provides the framework for you to create a Legend adapter for applications not supported out-of-the-box by Legend.
- The Custom Adapter is called when the Reader encounters a file with an extension defined in the custom.adapters option in the wtcad.ini file.
- The Custom Adapter sample source is located in <*Legend_loadpoint*>\Source\Custom_Adapter_Source.
- After compiling your user exit, place the DLL into the <*Legend_loadpoint*>\Adapters\Custom directory.
- Edit the <*Legend_loadpoint*>\wtcad.ini file:
  - Set the ProgID option in the [CAD Systems] section to your VB routine name
  - Set the option custom.adapters in the [CAD Systems] section to the file extensions to be processed by your custom adapter.

3.3. Associator

- The Associator provides the framework to associate related documents to parts, based on information other than the file name. For example, by default, when a Pro/E file and Word document with the same base name are together (abc.prt and abc.doc), Legend will make a link between the part ABC.PRT and the document abc.doc in PDMLink. If you have different rules for determining that relationship, they can be implemented in the Associator.
- The Associator is called by Legend after each file it reads during the Reader step.
- The Associator sample source is located in <*Legend_loadpoint*>\Source\VB_Associator_Source.

- After editing the source, compile it into a DLL. In the VisualBasic editor select File / Make Associator_DLL.dll….
- Place the DLL into the <*Legend_loadpoint*> directory.

3.4. Revision Mapper

- The Revision Mapper is used when multiple revisions of files are being loaded. The Revision Mapper provides the ability to use site defined information to define the appropriate revision of a given dependent file.
- The Revision Mapper is a DLL called by the Legend Reader prior to writing the XML file for the Legend Loader.
- The routine called by Legend is: FindRevisionForChild
- Inputs to FindRevisionForChild:
    - Object name – String – name of the object being processed. This is the instance name if the object is a family table instance.
    - Parent - String - absolute file name for the parent file (such as, path info + file title + extension + iteration). This is the generic file name if the object is the generic or an instance.
    - Children - String(array) - file title and extension of dependent file (such as, file title + extension). If the Child is a family table instance, the name of the child object is the generic name, thus the revision mapper table may need to be reprocessed to find the appropriate element in the table.
- Return from FindRevisionForChild:
    - String(array) - revision of each child in same order as Children
    - If migrating multiple iterations of any revision, the revision for each child should be in the form R_X, where R is the revision and X is the iteration.

3.5. Debugging User Exits

- Edit the wtcad.ini file and set debug=true in the [Logging] section.
- Run Legend through the Reader step.
- Save the Legend Project.
- Review the <*Legend_loadpoint*>\logs*Project_Name*_debug.log file.
- Entries should be made both before and after your user exit.
- A MetadataCustomizer log file is located in the <Legend_loadpoint>\logs directory.

Section II – Migration

Step 4 – Database Export

During this step, data from your existing database will be exported to a CSV file.

4.1. Legend CSV File Requirements (See Appendix B)

- The file must be comma delimited.
- The first line of the file contains the names of the source attributes. Each column name must be unique.
- The remainder of the file contains one line for each file to be processed.
- Each record represents one file.
- Column heading names do not have to match Windchill attribute names. They may be mapped in the Legend Database Mapping pane.
- The first field of the file is a special field named "**File Name*". The contents of this field are the existing file names with fully qualified paths and extensions.

4.2. Export Metadata

- Export the metadata from your database

4.3. Export Content

- Content files can be moved in to a staging area where they can be preprocessed, if necessary. This staging area should be broken into at least 2 subdirectories, one for Released files, one for WIP files.
- Create a directory listing of the content files with their fully qualified path name.

4.4. Merge CSV File with Content List

- Merge the list of files with metadata from the metadata export.

4.5. Data Review

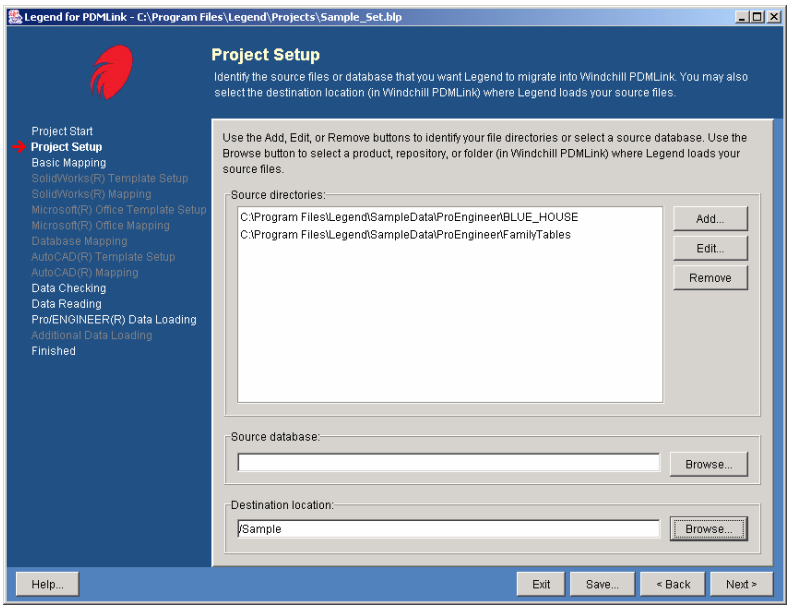
- Review the data and verify the CSV files are valid and complete.
- Verify the Pro/E files in the staging area are valid and complete by spot checking several directories.

Step 5 – Load Documents from a File System

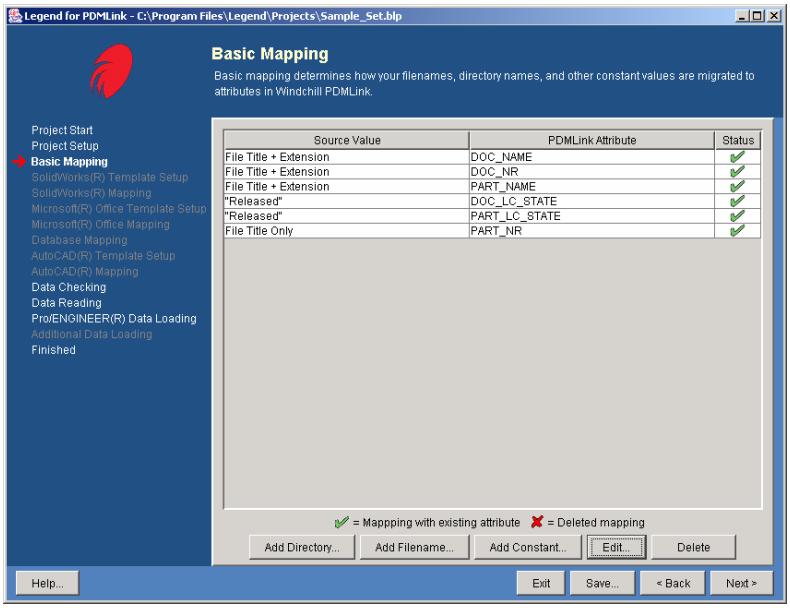
5.1. Legend Load

Load the Pro/E files to PDMLink -- adding content, setting IBAs, and creating relationships.

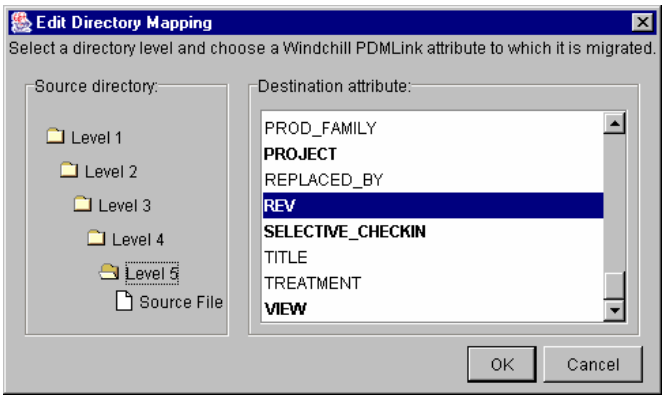
5.2. Legend Run



- If loading models to existing WTParts, Check Out the WTParts to a workspace other than Legend's workspace.
- Project Setup Source – Define either a set of source directories or a single CSV file. If defining a CSV file go to the next chapter.
  Destination – Select the target PDMLink repository for this project.

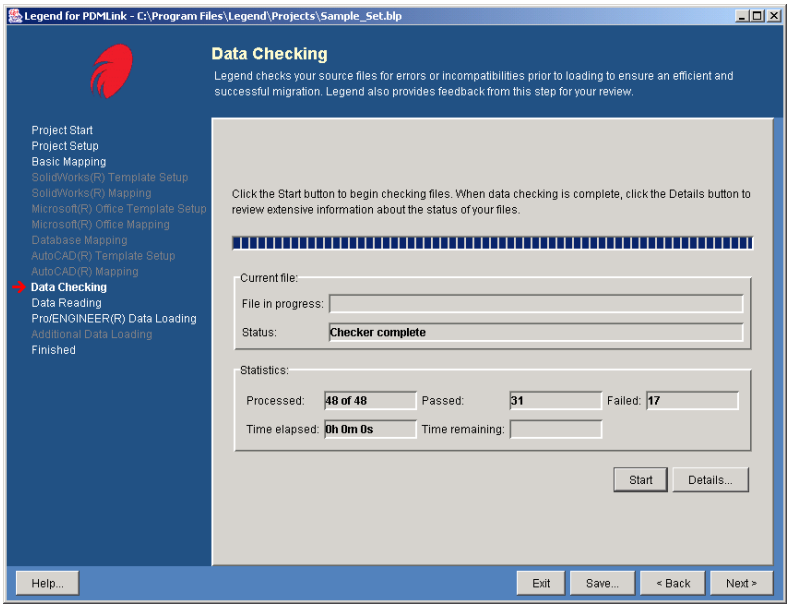


- Basic Mapping – maps the common elements to modeled attributes. The part name and number and document name and number are required. They are normally generated from the file name. But, they could also be defined by metadata, in which case, the definitions here are defaults in case the metadata was missing for some files.
  - Other attributes can be added as coming either from a constant, as shown.

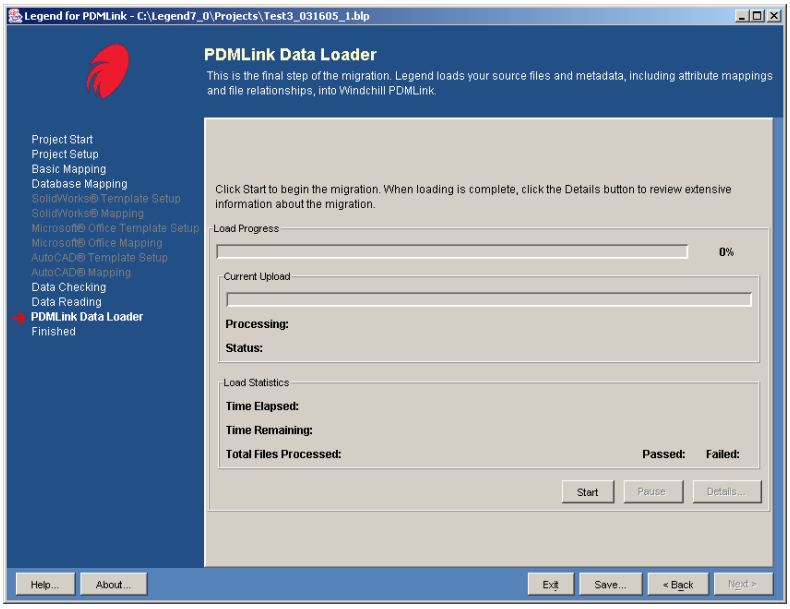


- Map a directory name to an attribute by clicking **Add Directory…**
  - The **Edit Directory Mapping** dialog appears.
  - Select the directory level you want mapped (must be 1 through 5). The levels are as follows:

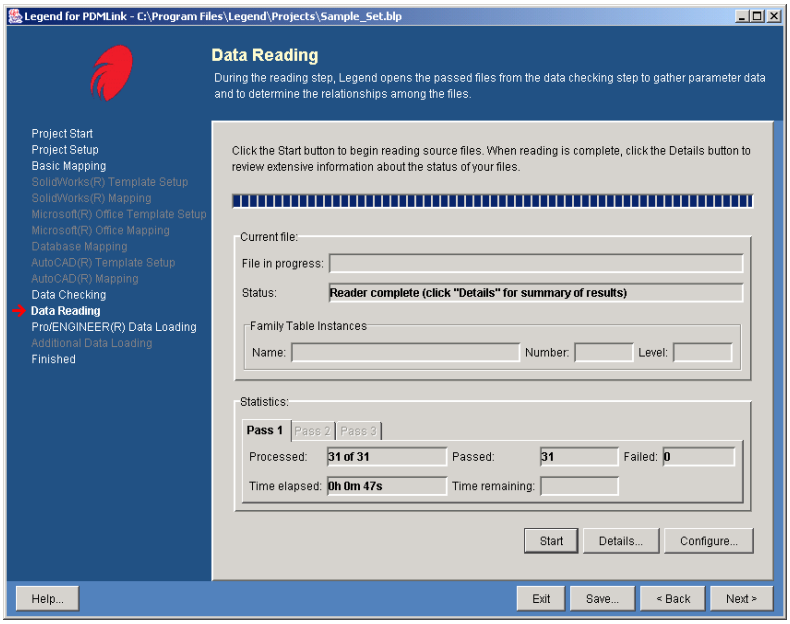
    C:\level_1\level_2\level_3\level_4\level_5
  - Select the **Destination** attribute. Be sure to select an item in **bold**, as these are modeled attributes. Legend is able to map only Pro/E parameters to IBAs (Destination attributes in normal font).



- Checking – The file directories are searched for files. The following exceptions may be reported:
  - File type not supported.
  - Duplicate file name.
  - Files that are unreadable.
  - The checker filters out those files it cannot handle and creates reports for your review. Click the **Details** button to review the reports. If, after reviewing the reports, you change file names or add or remove files to be processed, you will rerun the checker so Legend has an up-to-date list of what it will process.



- Reading – All of the files that passed the Checker will be opened and the following will be captured:
  - File interdependencies
  - Family tables
  - Designated parameters
  - Duplicate objects, including instances. The duplicates could be all with in this project, between this project and previous projects, or between this project and PDMLink.
- Check the detail reports for the following:
  - Metadata
  - Structure
  - Inconsistencies
  - Unique conditions
  - Target folders



- The PDMLink Data Loader will load the files into PDMLink using the Workgroup Manager File Adapter. During the load, the CADDocs and WTParts are created. Then:
  - Content is uploaded.
  - Attributes and IBAs are set.
  - Relationships are created.
  - The objects are checked into their target repository.

5.3. Legend Verification

- Review the message log for:
  - FAILURE" for files that did not upload
- Review the debug log for:
  - Exceptions. An exception message appears in the debug log in the midst of messages for a given file.

- The first record in the log for the loading of a given file starts with "Find epm document took …". The last record for loading a file is the file name.
- If there are more than one file listed at the end of the set of messages, the exception may apply to all files listed.

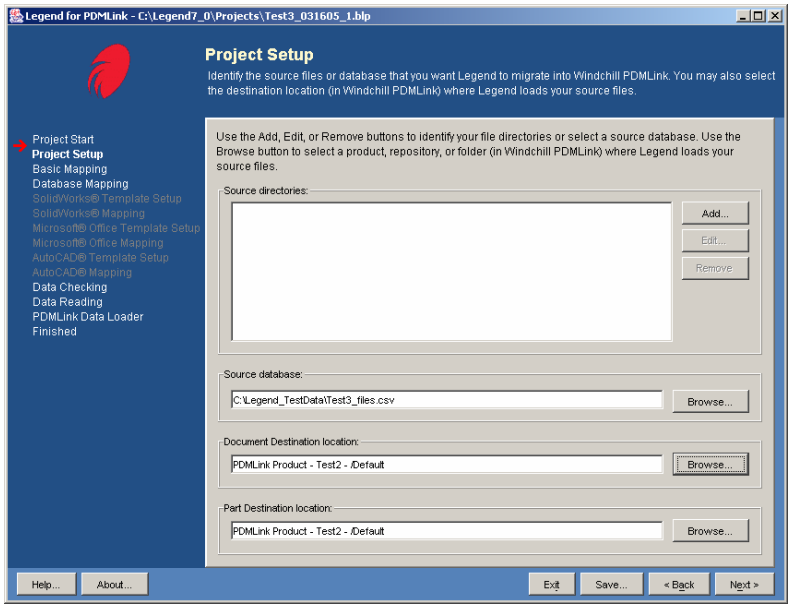
- Verify the objects can be checked out and checked back in with the Workgroup Manager or Wildfire.

5.4. Clean up

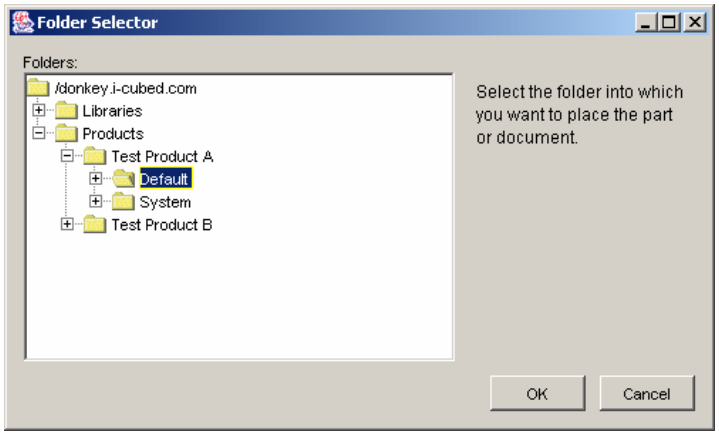
- If exceptions did occur
  - Verify if the objects were created in PDMLink
  - Verify the relationship to the WTPart

Step 6 – Load Documents from a CSV File

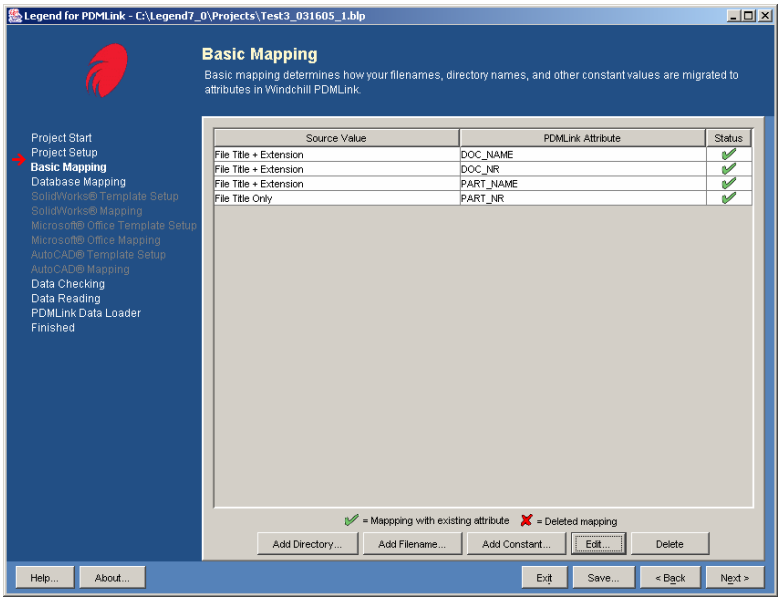
6.1. Legend Run



- Project Setup
  - Each Legend run is defined as a project. A project can be saved at any step in the process. The project can then be reopened and process restarted from where you left off.
  - Source - Define input CSV file (or input file directories if not using a CSV file.
  - Destination - Select the target PDMLink repository for this project. If you map an attribute from the CSV file to SP:FOLDER, the CSV mapping will override this Destination entry on the Project.



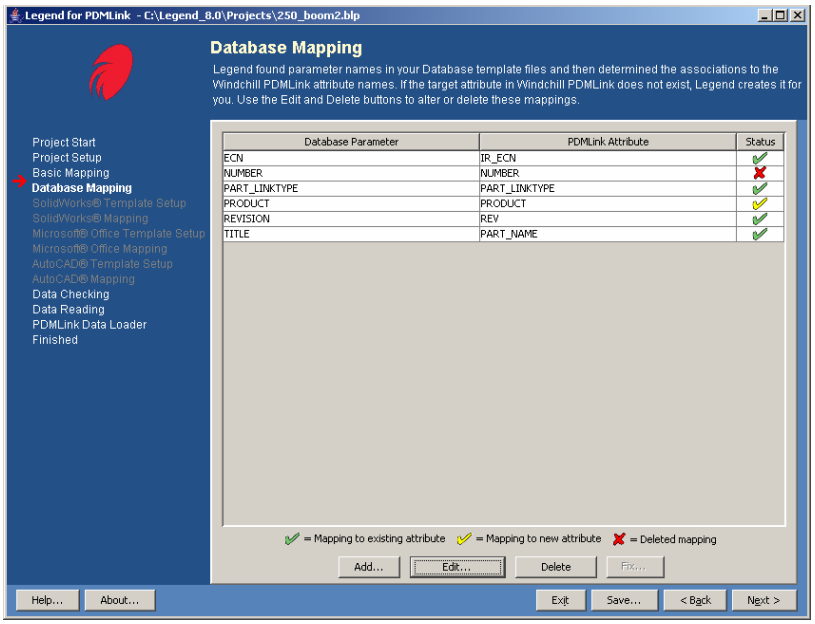
- When Selecting the destination folders, expand Library or Product icon, select the target library or product, then select the folder.



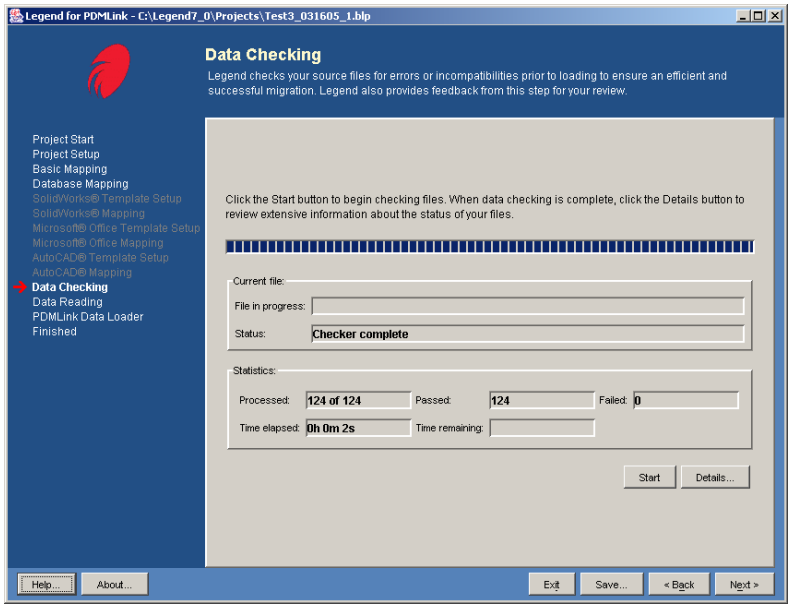
- Basic Mapping – Defines attributes from file names and directories. Select File Title Only for Part Number. All others can use the defaults.

Map a directory name to an attribute by clicking **Add Directory…**

- The **Edit Directory Mapping** dialog appears.
- Select the directory level you want mapped (must be 1 through 5). The levels are as follows: C:\level_1\level_2\level_3\level_4\level_5
- Select the **Destination** attribute. Be sure to select an item in **bold**, as these are modeled attributes. Legend is able to map only Pro/E parameters to IBAs (Destination attributes in normal font).

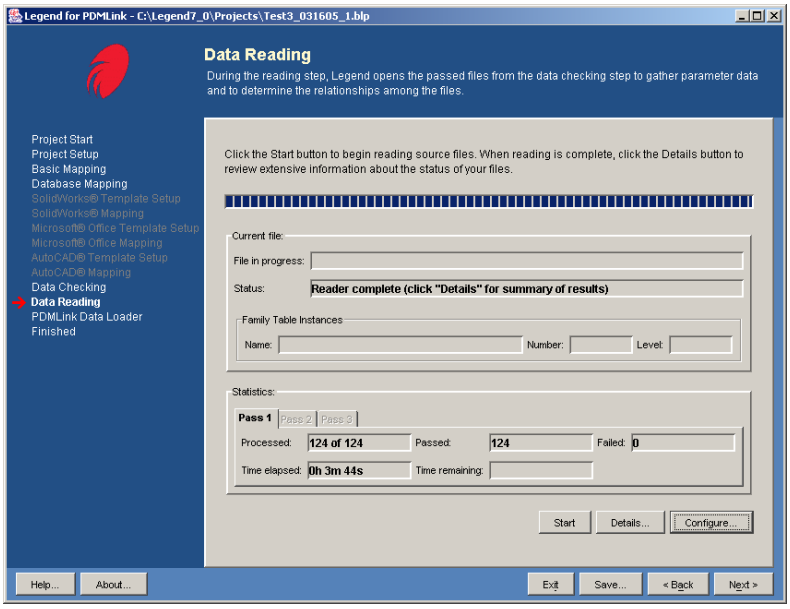


- Database Mapping -- Attributes in the CSV file can be mapped to Windchill PDMLink modeled attributes. In the database mapping edit dialog PDMLink modeled attributes are shown in **bold.** For Pro/ENGINEER files, soft attributes (IBAs) should not be set from values in the CSV file. Soft attributes are shown in normal text in the database mapping dialog. It is best to create the CSV file with attribute names the same as the PDMLink attribute names. If this is done, you will not have any database mapping entries to change.



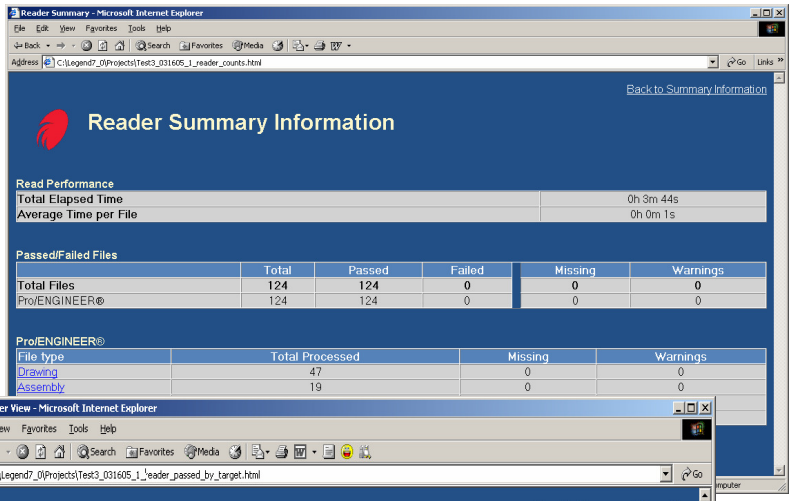
Checking – The file directories are searched for files. The following exceptions may be reported:

- File type not supported.
- Duplicate file name.
- Files that are unreadable.

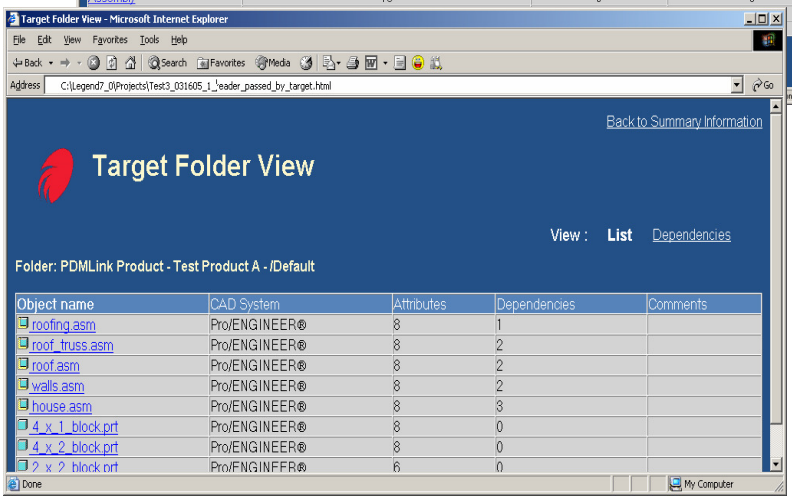


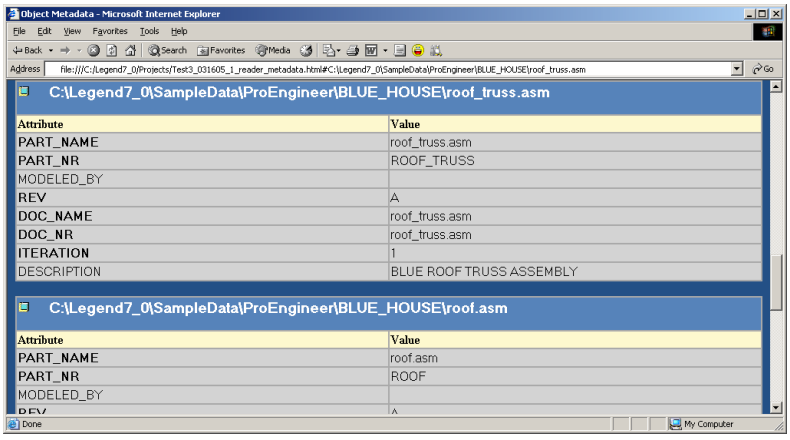
- Reading – All of the files that passed the Checker will be opened and the following will be captured:
  - File interdependencies
  - Family tables
  - Designated parameters
  - Duplicate objects, including instances. The duplicates could be all with in this project, between this project and previous projects, or between this project and PDMLink.

- The Reader Summary lists the files read by file type. Errors are also totaled in the Summary report.

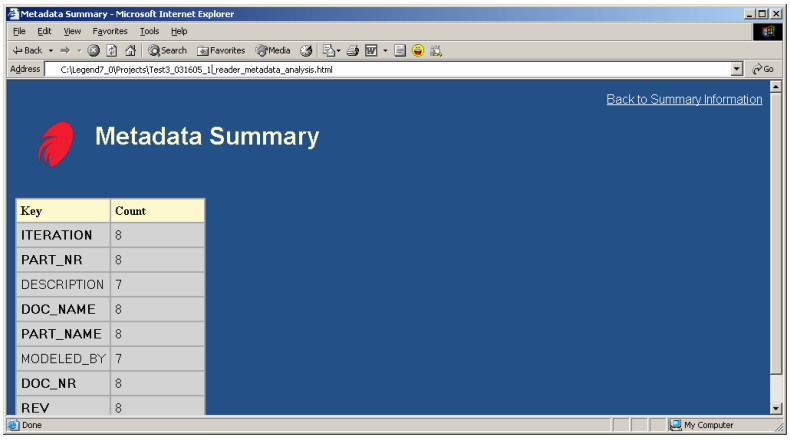


- The Target Folder View lists each object that will be migrated by the folder to which it will be created.

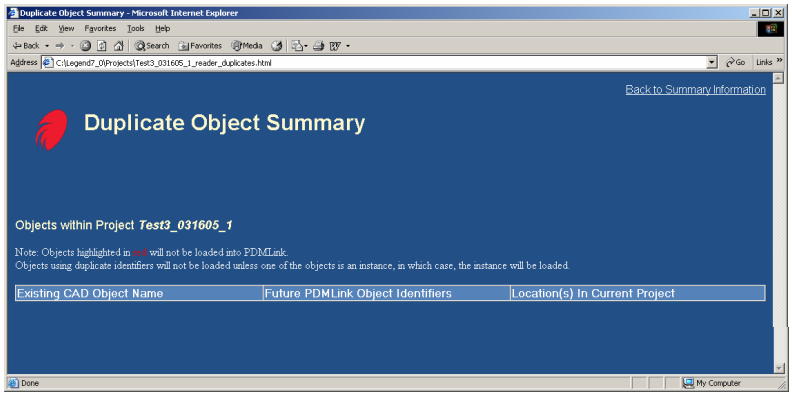




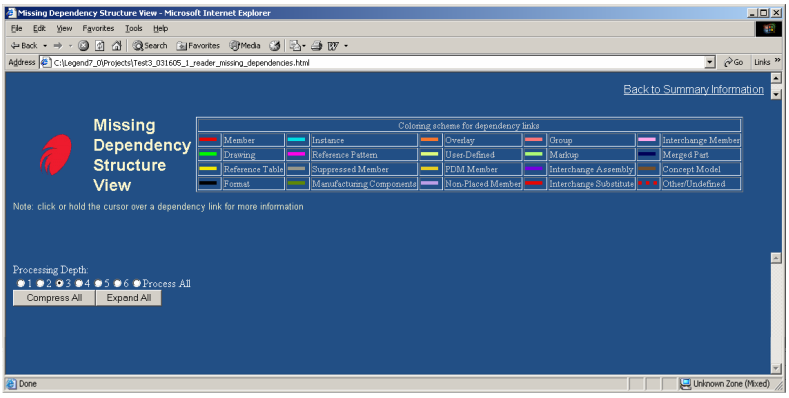
- Clicking the object hyperlink from the Target Folder view will display the metadata captured for that object. If selecting an instance, the generic will be displayed. You can scroll down to see the instances for the family table.



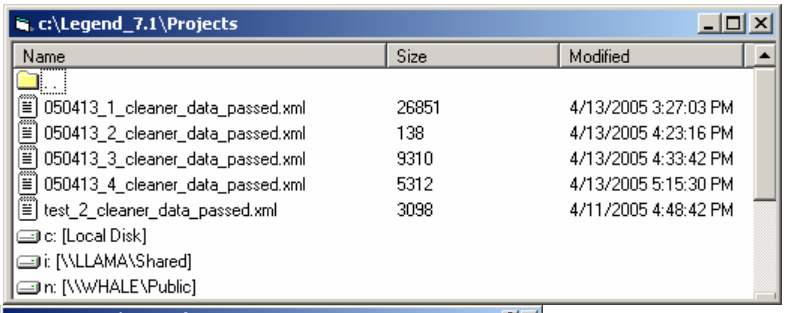
- The Metadata Summary displays each attribute captured from the CSV file as well as the designated parameters. Each attribute is followed by the number of objects that have that attribute. Note, the attribute can exist for an object but have no value.

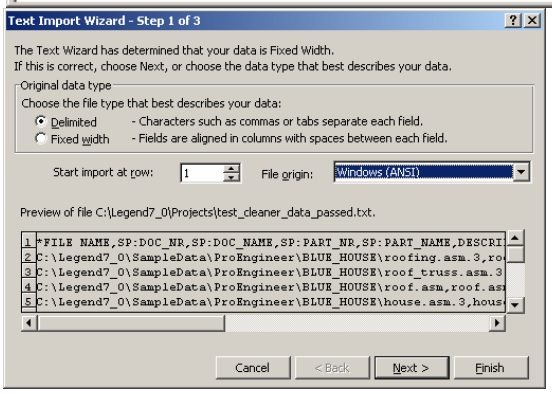


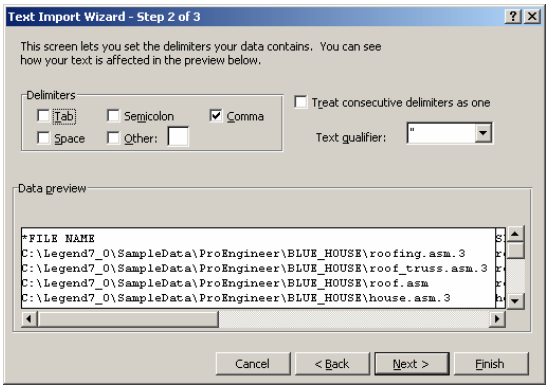
- The Duplicate Object Summary lists the objects that are duplicated either within this Legend project or within PDMLink. Objects listed in red will not be migrated.

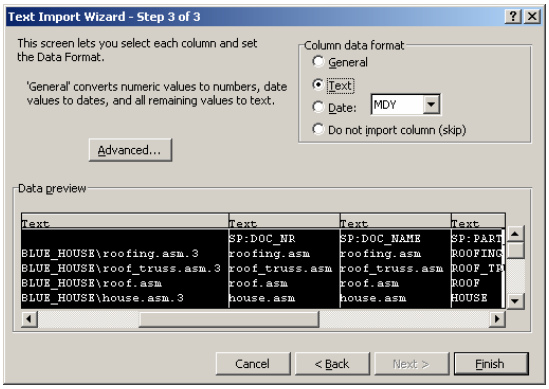


- The Missing Dependency View displays the missing dependent objects along with the files that depend upon them. Any object that depends upon a missing object, either directly or through a where-used hierarchy will be excluded from the migration.

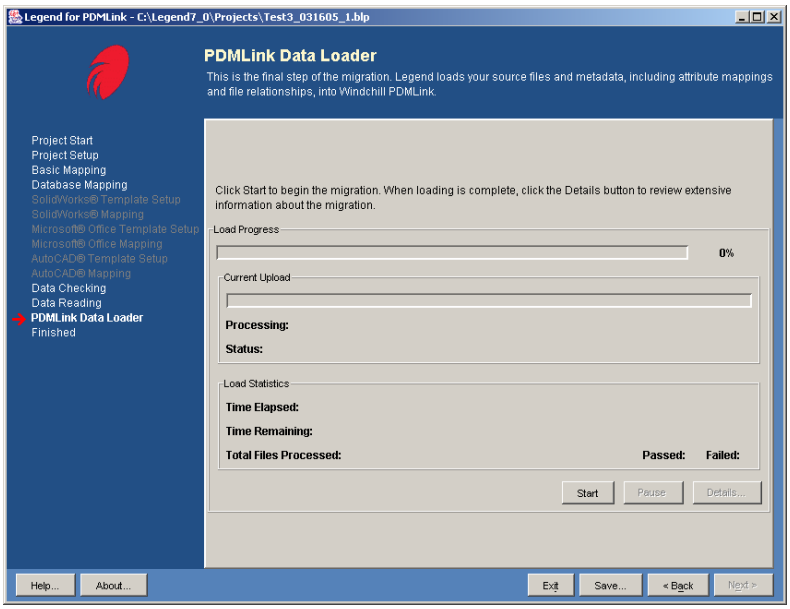








- Run <Legend>\Utils\XML2CSV to create Excel readable files of the captured information. XML2CSV processes the XML prior to it being trimmed due to missing dependencies and duplicate objects. Double click the cleaner_data_passed.xml file for the Legend project. XML2CSV will create two files.
  - Metadata report: <project>_cleaner_data_passed.txt
  - Relationship report: <project>_cleaner_data_passed_relations.txt
  - To open the files in Excel use the following options
    - Delimiter ","
    - Text Qualifier "
    - All column data formats as Text



- Load – Objects are created in Windchill, content is uploaded, and EPMDocument relationships are made. The load may slow down after a large number, approximately 5000, of files are loaded within one project.

6.2. Legend Verification

- Review the message log for:
  - FAILURE" for files that did not upload
- Review the debug log for:
  - Exceptions. An exception message appears in the debug log in the midst of messages for a given file.
  - The first record in the log for the loading of a given file starts with "Find epm document took …". The last record for loading a file is the file name.
  - If there are more than one file listed at the end of the set of messages, the exception may apply to all files listed.
- Verify the objects can be checked out and checked back in with the Workgroup Manager or Wildfire.

6.3. Clean up

- If exceptions did occur
  - Verify if the objects were created in PDMLink
  - Verify the relationship to the WTPart

Step 7 – Migration Tips

7.1. Processing Multiple Revisions from a File System

Slight variations are required when loading multiple versions of the data files. There are two cases possible, one iteration of any revision or multiple iterations of any revision.

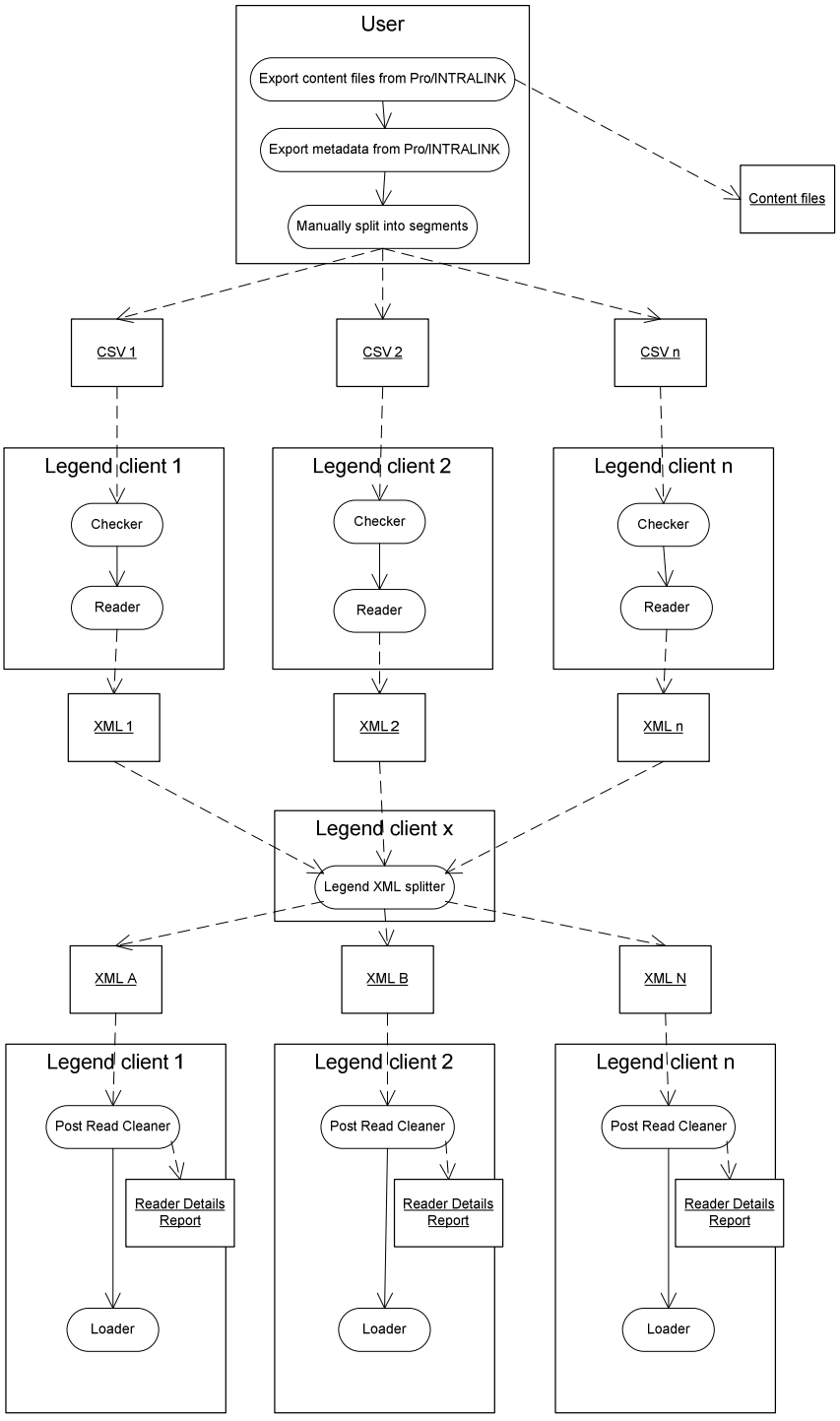
Be sure to review section 2.6 Multiple Revisions and Iterations.

- Data Prep – One iteration of any revision
  When creating the file system directories for the data files, create one directory that contains the revision number (or letter if using alpha revisions). This directory needs to be at the same depth from the drive letter for all data files within any one Legend project. Also, it must be no deeper than level five from the drive letter.
  Additionally, the Revision Mapper User Exit should be created to set the revision of a dependent file for a each revision of an assembly. See the Customization chapter, above.
- Data Prep – Multiple iterations of any revision
  This scenario is more complicated than one iteration of any revision and should be avoided if possible.
  When creating the file system directories for the data files, create one directory that contains the revision number (or letter if using alpha revisions) and one directory for each iteration within each revision. The revision directory for all data files needs to be at the same depth from the drive letter within any one Legend project. The iteration directory for all data files needs to be at the same depth from the drive letter within any one Legend project. Also, these directories must be no deeper than level five from the drive letter.
  Additionally, the Revision Mapper User Exit should be created to set the revision of a dependent file for each revision of an assembly. See the Customization chapter, above.
- Basic Mapping
  Run Legend as described in the Processing from a File System. Be sure to map the revision directory level to the DOC_REV attribute. If processing multiple iterations for a revision, be sure to map the iteration directory to the DOC_ITERATION attribute. To map a directory to an attribute:
  - Map a directory name to an attribute by clicking **Add Directory…**
    - The **Edit Directory Mapping** dialog appears.
    - Select the directory level you want mapped (must be 1 through 5). The levels are as follows:
      C:\level_1\level_2\level_3\level_4\level_5
    - Select the **Destination** attribute. Be sure to select an item in **bold**, as these are modeled attributes. Legend is able to map only Pro/E parameters to IBAs (Destination attributes in normal font).

7.2. Multiple Legend Clients

- Each Legend client must use a differently named workspace.
- If using the XML Splitter for multiple clients, the path to the content files should be the same for each client.
- All iterations of a given document must be loaded in iteration order. The Legend Reader will order the XML in Revision, Iteration sequence within a Legend project. It is best if all iterations are run within the same Legend project on a single Legend client. This will be enforced by the XML Splitter.
- On the next two pages is a diagram and usage statements for the multiple client run. The bullets, below, describe the process.
- A single CSV file can be split, manually, into multiple segments and run on multiple Legend clients up to and including the Reader. Note, the missing dependency reports for these "partial" CSV datasets can be ignored, as they will be recreated later.
- Run Legend on each client against its specific CSV file segment. Use the same Legend project name on each client.
- Run Legend all the way up to the Loader pane, but do not start the load. Instead, Save and Exit the Legend project and continue with the steps, below.
- After the Reader completes on all of the clients, copy the <project_name>_cleaner_data_passed.xml onto one of the Legend clients. Be sure each file has a unique name or is in a different directory from other like named xml files.
- Use the XML Splitter to merge and split the XML into the desired number of XML files for loading. The split ensures each XML segment does not have any depencies on files in the other XML segments.
- Copy the XML Splitter output XML to the various Legend clients' Project directories. These XML files need to be named <project>_cleaner_data_passed.xml, where project is the Legend project name used on this given client to run the Reader.
- Run the PostReadCleaner on the split xml segments to recreate the Reader data cleanliness reports.
- Run Legend on each client and load the data into PDMLink.

- Activity Diagram of a multiple client read and multiple client load



- Run the XML Splitter from the Legend installation directory
  - Usage:

Utils\XMLSplitter ***m xml1.xml ... xmln.xml***

Where:

m is the number of xml segments after the split

xml1.xml ... xmln.xml the xml files to be merged together before calculating the split. Be sure to specify the xml files named <project_dir>\<project_name>_cleaner_data_passed.xml

Example:

Utils\XMLSplitter 3 ..\Projects\pocket_knife_cleaner_data_passed.xml ..\Projects\blue_house_cleaner_data_passed.xml

If you're using multiple input XML files be very careful about duplicate parts. For example the airplane and pocket_knife projects both contain a part called pin.prt. As a result they are seen as a contiguous dataset joined by the part called pin.prt even though they are different parts.

- Run the Post Read Cleaner from the Legend installation directory

The Post Read Checker creates the Reader missing dependency report. After splitting the XML you should run the Post Read Checker to validate the data in the XML file.

  - Usage:

Utils\ PostReadCleaner <input xml file> <output xml file> <missing_dep output xml file> <project>

Where:

input xml - is the fully qualified filename of the input xml

output.xml - is the output file to be read by the loader. It will automatically be put into the \Projects dir and will have <project> prepended

missing_dep.xml – is the missing_dependency xml file name. It will automatically be put into the \Projects dir and will have <project> prepended

project - The name of the Legend project

Example:

Utils\PostReadCleaner house_cleaner_data_passed.xml reader_data_passed_output.xml reader_data_missing_deps.xml house

all console output will go into the \logs\console.log file

7.3. Processing Pro/E Family Tables

Legend processes family tables on a file basis. Whether coming from a files system or from a CSV file, the generic file is passed to Legend as the object to be processed. Legend, using the File Adapter, gathers the instance information from the generic's file header. Instances inherit their attributes from the generic.

- Family tables are processed on a file basis
- Instances are determined by the generic's file header
- Family table instances must be verified before running Legend
- Instances inherit attributes from the generic
- Instances will be at the same revision as their generic

- Metadata Customizer user exit is called for the generic but not the instances
- Revision Mapper user exit is called for each instance

7.4. Pro/E Designated Note Parameters

- Earlier releases of Pro/E allowed notes to be designated. These designated notes can cause an upload error during the load step. If that occurs, the notes will need to be undesignated before loading those models.

7.5. Workflow Tasks

To turn off submit tasks:

- Go to **Life Cycles** under **Administrative Tools**
- Find the lifecycle associated with the object type whose lifecycle state is changing. For example, most objects such as EPMDocuments are associated with the PDMLink Basic lifecycle by default. ECN/ECR objects are typically associated with the various "PDMLink Change" lifecycles.
- Highlight the appropriate lifecycle, and click **Check Out**. Then click **View**.
- The **View Life Cycle** dialog appears, with a diagram of the phases and gates in the workflow. Click on a phase in the diagram, and then go to its **Workflow** tab.
- If submit tasks are enabled, the phase and gate processes will show "submit" and "review" as their associated workflows.
- Clear these fields using the **Clear** button.
- Check in the lifecycle.

After migration, reactivate the submit tasks in a similar fashion.

7.6. PDMLink External Vaults

When using external vaulting, it is best to set up both the shared domains and the user domain for external vaulting. This will prevent a BLOB from being crated when the content is first uploaded. It will also improve performance when moving to the shared vault.

Notes

General

- Do not edit the wtcad.ini file while Legend is open. If you do, your changes will be lost.
- To prevent Submit workflows during a load do the following: (The workflows may need to be in the /System folder for this to take affect)
  - Login to PDMLink as administrator
  - Go to the "Administration" page
  - Select **Lifecycles**
  - Select **EPMDoc life cycle**
  - Select **Released** in the graphic
  - Uncheck **Phase Process**
  - Save the workflow
  - Repeat steps 4 through 7 for PDMLink Basic
- To reset the Submit workflows after a load do the following: (The workflows may need to be in the /System folder for this to take affect)
  - Login to PDMLink as administrator
  - Go to the "Administration" page
  - Select **Lifecycles**
  - Select **EPMDoc life cycle**
  - Checkout and Update the workflow
  - Select **Released** in the graphic
  - Select the Process Phase and select the **Submit workflow**
  - Save the workflow
  - Repeat steps 4 through 7 for PDMLink Basic
- During Loads, the server must be monitored for:
  - Available disk space
  - method server 'health'

Performance notes

The Windchill Performance and Tuning Guide should be reviewed and followed as appropriate. Below are some suggestions, many of which can be found in the Tuning Guide.

-Disable WVS while loading:
wvs.properties publish.service.readytopublish.enabled=false and/or wvs.enabled=false

-Update Oracle Statistics:
sqlplus system/manager
execute dbms_stats.gather_schema_stats('wcadmin', DBMS_STATS.AUTO_SAMPLE_SIZE, CASCADE =>TRUE);

-Set some tuned Oracle parameters (http://www.ptc.com/WCMS/files/19870/en/19870en_file1.pdf page 71). A good starting point would be the following:

optimizer_index_caching=90
optimizer_index_cost_adj=10

db_cache_size=200000000
large_pool_size=15000000
log_buffer=20971520
pga_aggregate_target=200000000
_b_tree_bitmap_plans=FALSE

Also take a look at http://www.ptc.com/WCMS/files/23484/en/23484en_file1.pdf as it has some WMProE2001 tuning suggestions (I think the Legend Loader uses WMProE2001?)

-Resize any datafiles prior to them having to continually autoextend

-Setup to vault content

-Restart Oracle, Methodserver, continue loading for a day and get a feel for how the system is performing.

If performance is still poor, set the following in wt.properties

wt.method.serverMethodTiming=true
wt.adapter.verbose=true

restart the MethodServer, load some data or perform actions which you feel are slow, run the gather_info_script, and return the report.txt, Methodserver.log, and time/description of the actions you were performing.

At some point, you may want to consider installing the Oracle Tune-Up package.

TPI 132477 (http://www.ptc.com/cs/tpi/132477.htm) contains most if not all of these.

HTTPS

You have to load the certificate into your jre with

keytool -import -alias pdmlink -file erlf570a.crt -storetype jks -keystore %JAVA_HOME%/jre/lib/security/jssecacerts
keytool -import -alias cadclient -file erlf570a.crt -storetype jks -keystore %WT_HOME%/jre/lib/security/cacert.jks

Oracle Statistics

- Periodically, update Oracle stats from the method server host as follows:
  - Open a command prompt
  - Enter:
    SQLPLUS name/password
  - For Oracle 9 enter:
    execute dbms_stats.gather_schema_stats ('NAME', DBMS_STATS.AUTO_SAMPLE_SIZE, CASCADE =>TRUE);
  - For Oracle 8 enter:
    execute dbms_stats.gather_schema_stats ('NAME', CASCADE =>TRUE);
    where 'NAME' is the upper case value of the wt.pom.dbUser entry of the db.properties file in $WT_HOME\db

AutoCAD

- How to edit AutoCAD attribute tag names, interactively, in AutoCAD 2000. This only affects new inserts not existing inserts.
  - Menu option – Modify / In-place XREF and Block Edit / Edit Reference (or key in REFEDIT command)
  - Select **block insert** from graphics window
  - Select **block** in the **Reference Edit** dialog
  - Select **Display attribute definitions for editing** check box
  - Click **OK** in **Reference Edit** dialog
  - With left mouse button, drag a selection box around the title block
  - Press ENTER to end the selection process
  - Key DDEDIT
  - Select the attribute tag name on the **graphics** window
  - Edit the tag name in the **Edit Attribute Definition** dialog
  - Click **OK** in the **Edit Attribute Definition** dialog
  - Repeat steps h though k for all tag names to be edited
  - Click the last icon in the right of the **Refedit** dialog. This icon is the **Save back changes to reference** function.
  - Click **OK** to **All references edits will be saved**
  - Save the drawing file

Clobber

- Clobber is installed in the <*Legend_loadpoint*>\Utils directory. Clobber deletes objects in PDMLink. Run instructions follow:
  - 1. Delete by workspace.
    - clobber workspace:<*workspace_name*>
    - Example: clobber workspace:legend_temp_workspace
  - 2. Delete by folder
    - clobber /*folder_name*
    - Example: clobber "/ExchangeContainer - Site - /catadmin/legendPDMLink_ProductCaterpillar_TEST_Product /legendLibraryCAT_Library"
  - 3. Delete single object
    - clobber <*object_number*>
    - Example: clobber d-26038.asm
  - Notes on clobber:
    - Clobber removes all objects from any baselines or workspaces to which they may belong prior to deletion, and checks them in if they are checked out. It also decouples family instances from the table to which they belong. In other words, deleting a generic using a delete single object command will leave all its instances in PDMLink, no longer in a family. Clobber removes all versions of all objects. If you delete a workspace, then all versions of objects in that workspace will be removed regardless

of whether they belong to that workspace. The same applies to folders.

- The PostLoadCleaner moves missing dependents that are left in the PDMLink workspace folder to a shared repository. Run instructions are in the Cleanup section of the Processing step. To create the PostLoadCleaner
    - Copy your start.bat file in the Legend load point to the <*Legend_loadpoint*>\Utils directory
    - Change the call in PostLoadCleaner.bat
        - from: com.icubed.le.legend.enterprise.LegendEnterprise
        - to: com.icubed.le.legend.common.PostLoadFinalizer %1 %2

Troubleshooting

Overview

The first level of troubleshooting is to review the Legend detail reports. These reports are html pages that are produced by Legend's Checker, Reader and Loader. These are described in the Reports section, below.
If further definition is required, review the logs files shown in the Logs section. You might also consider reviewing the method server log on the Windchill server.

Process

The process differs slightly depending upon step that encountered the problem. In all cases, review the reports and logs as shown below. Identify the problem. Correct the problem and rerun.

Checker

- Review the Checker Detail report.
- Review the error log.
- Review the console log.

Reader

- Review the Reader Detail reports, Reader Summary and Target Folder. Review the comment field of these reports.
- Review the error log. The exception "Can't find bundle for base name com.icubed.le.legend.common.FailuresReporterRB, locale en_US" can be ignored.
- Review the jmsfileadapter log.
- Exit Legend, set the debug option to "true" in the wtcad.ini file and rerun the Reader.
- Review the debug log.
- Enter an incident at www.LegendForPDMLink.com support.

Loader

- Review the Loader Details report. Note, the loader failed and passed counts may be in error (Legend incident 050308-000000).
- To find errors, review the debug log. Search for the word "Exception". Following the "Exception" message, find the first occurrence of "Process batch entire process took …" Immediately after the "process batch…" message will be one or more file names listed. This is the file(s) that were affected by the exception.
- Fatal errors should appear in the message log. Search for the keyword "Failure".
- Review the error log.
- Review the console log.
- Exit Legend, set the debug option to "true" in the wtcad.ini file and rerun the Reader. Note you may need to remove the objects from the workspace.
- Enter an incident at www.LegendForPDMLink.com support.

Reports

These HTML reports are located in <*Legend_loadpoint*>\Projects.

- **Data Checking report [*project_name*_checker_summary.html] -** Displays information on the number of files that Legend counted in the Source directories.
- **Data Reading report [*project_name*_reader_summary.html] -** Displays information on how many files were processed, passed, and failed during the reading stage. The customer can view a detailed list of the files that passed and a detailed list of the files that failed.
- **Data Loading report [*project_name*_proe_loader_summary.html] -** Displays information on how many files were processed, passed, and failed. The report also gives details on the relationship building among the files. Managers view this report as very important on production runs.

Logs

These log files are located in <*Legend_loadpoint*>\logs. Note the project name is pre-pended to these files only when the Legend project is saved.

- **Message log [*project_name*_message.log] –** Contains information for post reader and loader activity. Files that failed to load can be found in this log by search for the word "Failure".
- **Error log [*project_name*_error.log] –** Contains error messages and trace backs of exceptions.
- **Debug log [*project_name*_debug.log] –** Contains debug messages as well as all error and trace back of exceptions. Setting the wtcad.ini option debug=true can cause very large debug logs to be created. Note, when reviewing loader entries, the first record in the log for the loading of a file starts with "Find epm document took …" The last record for loading a file is the file name.
- **JMS file adapter log [jmsfileadapter.log]** – A cumulative log of activity between Legend and the CAD Readers.
- **Client error log [clienterror.log] –** A cumulative log of activity between the file adapter and the Legend Reader. This log exists in the Legend load point.
- **UWGM log [<Legend_load_point\uwgm.log]** – Log of UWGM activity.
- **Failed log [*project_name*_loader_log_failed.log]** – Contains list of files that failed the load process.
- **Passed log [*project_name*_loader_log_passed.log]** – Contains list of files that failed the load process.
- **ProcessWatchdog log [pwatchdog.txt]** – Contains information on timed out processes.
- **MetadataCustomizer.log** – Contains an entry for each object processed. Each entry shows the metadata values before and after the MetadataCustomizer user exit.
- **License update log [updateui.log] –** Contains a log of license update activity.

Additional Files

- **Legend.bat –** Is the batch file used to start Legend. This file contains class paths and additional arguments to the Legend client. Updating the Legend license will update this file.
- **Updatelicense.bat –** Is the batch file used to update a Legend license. This update procedure will contact the Legend license server across the internet, retrieve the license files and update the start.bat file. See the Installation and Configuration section, above.

- **Register_servers.bat –** Is the batch file, called by the installation program, to update the Windows registry with the Legend components. If you copy the Legend client installation to another directory, you must run register_servers.bat.
- **Project file [...\Projects*project_name*.blp] –** Contains information critical to the named Legend project. This file should not be edited unless directed by I³ Tech Support.
- **Failed XML file [...\logs*project_name*_loader_data_failed.xml] –** Contains the XML of those files that failed to load. If the error condition that caused the load failure is corrected, this file can be used to load the files that failed, initially. To reuse this file:
    - Save the Legend project, if you have not already done so
    - Copy this file to the …\Projects directory
    - Rename the initial load xml from *project_name*_cleaner_data_passed_proe.xml
    - Rename the loader data failed xml file to *project_name*_cleaner_data_passed_proe.xml
    - Start Legend
    - Open the existing project
    - Click **Next**. If you are not on the Pro/E Loader step click **back** to go to the Pro/E Loader step
    - Click **Start**

Technical Support

I-Cubed technical support self-service portal is located at

- http://www.i-cubed.com/support/support.php

After purchasing Legend you will receive support accounts for your users and access to the self-service portal.

Use the self-service portal to create a support case, view pending cases, and search our knowledgebase.

More information is available on our support website at http://www.i-cubed.com/support/support.php

Appendix A

Legend Panel Selections

	Project Setup	
	Source Database	(File System)
	Destination Location	?
	Basic Mapping	
	File Title and extension	CAD_NAME
	File Title and extension	DOC_NAME
	File Title and extension	DOC_NR
	File Title and extension	PART_NAME
	File Title only	PART_NR
	Attribute Mapping	

	Project Setup	
	Source Database	*.csv
	Destination Location	?
	Basic Mapping	
	File Title and extension	CAD_NAME
	File Title and extension	DOC_NAME
	File Title and extension	DOC_NR
	File Title and extension	PART_NAME
	File Title only	PART_NR
	Database Mapping	
	(all others)	(delete)

Appendix B

CSV File Specification

The file must be comma delimited.

- The first line of the file contains the names of the source attributes.
  - The first field of the file is a special field named "**File Name*". The contents of this field are the existing file names with fully qualified paths and extensions.
  - Column heading names do not have to match Windchill attribute names, but, if they do, there will be less Database Mapping required.
- The remainder of the file contains one line for each file to be processed.
- Each record should have metadata including:
  - Part number (SP:PART_NR)
  - Part name (SP:PART_NAME)
  - Document name (SP:DOC_NAME)
  - Revision (SP:REV) – sets both part and doc revs
    (SP:DOC_REV)
    (SP:PART_REV)
  - Lifecycle state (SP:PART_LC_STATE)
    (SP:DOC_LC_STATE)
- Metadata items can relate other information such as
  - Target repository (SP:FOLDER) – sets both part and doc folders
    (SP:DOC_FOLDER)
    (SP:PART_FOLDER)

  When specifying Folder, the format of the value is:
  <context_type> - <context_name> - <folder>
  examples:
  PDMLink Product – Test2 - /Default
  Library – Hardware - /Default/Fasteners
  An example of the format of the destination folder is also displayed in the Legend Project Setup pane.

  - Description (SP:DESCRIPTION)
  - Iteration (SP:ITERATION)
- The SP:ITERATION value is used only to sort the iterations within a given revision. The actual iteration number of the EPMDocument will be assigned a sequential number according to the order they are loaded.
- Values in the CSV file do not have to be exactly as they will be in Windchill. They may be manipulated in Legend's Metadata User Exit. This is helpful when target repository names are different from names in the current database or when some values, such as part numbers, need to be prefixed or parsed. See Customize in Section 3.

Appendix C

Checker Reports

The Checker report will indicate the following errors:

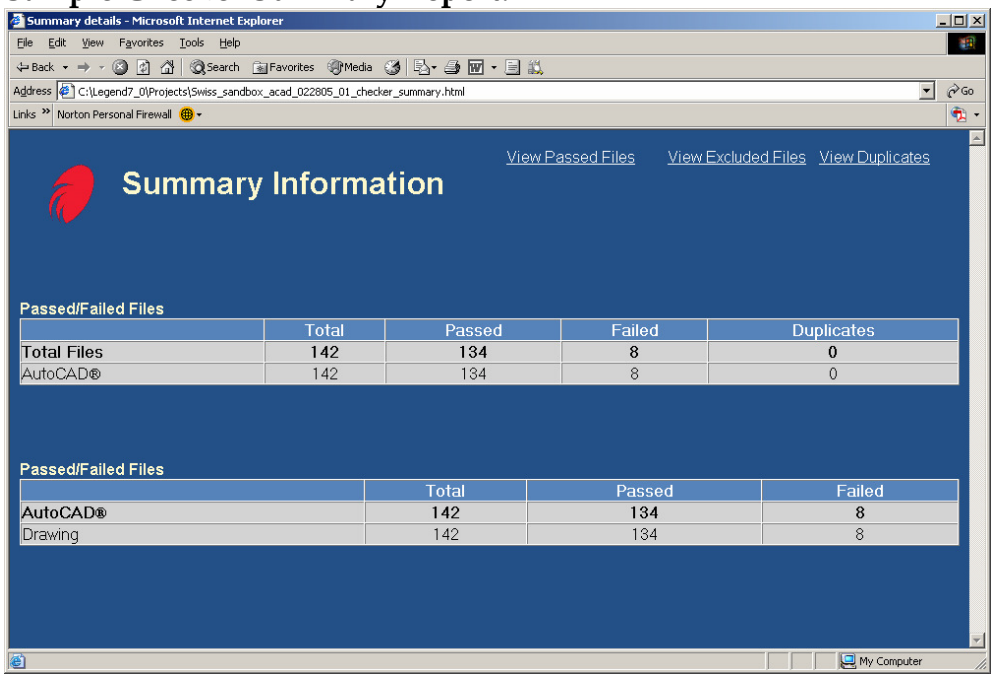
- Unsupported files types
- Duplicate file names within the set of files processed
- Files with a zero length

Legend report files are named with the name of the Legend project (a descriptive name for the dataset) followed by the report name.

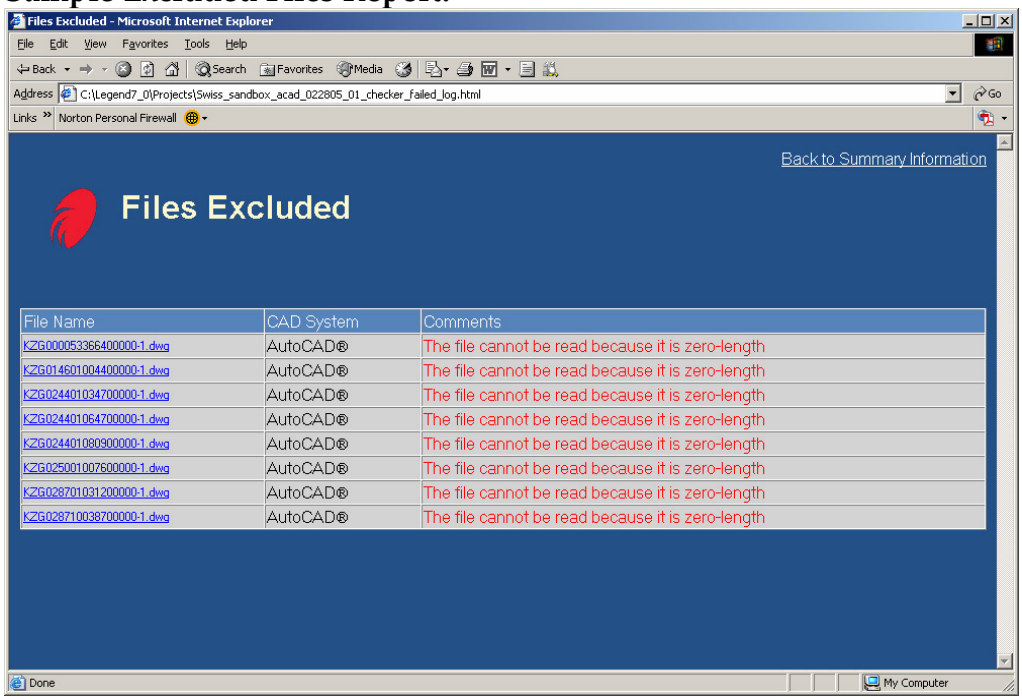
To view the Checker reports open the file <Legend_project>_checker_summary.html.

Error Type	Responsibility	Shown in Report	Description
Unsupported file type	Load Group	Excluded Files	**Indication:** Legend is not configured to handle the given file type. **Action:** For CAD files, ensure the Legend license is correct. For non-CAD files, edit the wtcad.ini file and verify the Custom Adapter will handle the file properly.
Duplicate file names	Business Group	Duplicates	**Indication:** More than one file exists in the migration dataset with the same name. For Pro/E files, the comparison is done excluding the file iteration. **Action:** Remove the unwanted duplicate from the set of files to be migrated.
Files with zero length	Business Group	Excluded Files	**Indication:** An empty file was sent as part of the migration dataset. This may be an error in the sites file system or in the zip file when the migration dataset was shipped. **Action:** Replace the zero length file with a valid file within the set of files to migrate.

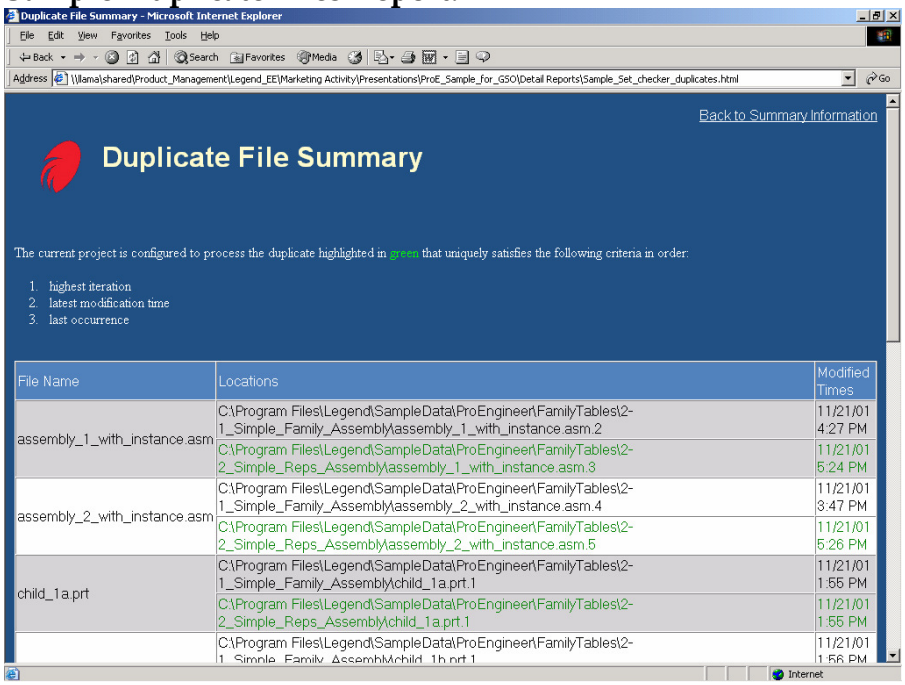
Sample Checker Summary Report:



Sample Excluded Files Report:



Sample Duplicate Files Report:



Appendix D

Reader Reports

The Reader reports will indicate the following errors:

- Duplicate Objects – already in PDMLink
- Missing Dependencies
- Invalid target folder
- Reader Errors

Legend report files are named with the name of the Legend project (a descriptive name for the dataset) followed by the report name.

To view the Reader reports open the file <Legend_project>_reader_summary.html.

Error Type	Responsibility	Shown in Report	Description
Duplicate Objects	Multiple Site Business Groups	Duplicate Objects	**Indication:** An object with the same document number already exists in PDMLink or multiple objects within this dataset have the same document number. The object from this set will not be loaded. This is can be an indication of an object shared with another site, or a name collision between multiple sites. **Action:** Compare the object that is in PDMLink with the object from this set. If it is a shared object, all sites using that object should move the file into the shared set, if it is two or more different objects with the same number; determine which site(s) needs to rename the object within their data set.

Error Type	Responsibility	Shown in Report	Description
Missing Dependencies	Business Group	Missing Dependency Structure View	**Indication:** The objects shown with a red exclamation point "!" (Those seen by clicking the Compress All button) are required by one or more files in this load. These objects are neither in this dataset nor in PDMLink. All files in a where-used hierarchy of the missing object will be excluded from the migration. For large datasets this report may take a few minutes to finish displaying. **Action:** Locate the missing files and include them in the migration dataset. The missing object could be an instance in a family table, in which case the family table should be included in the migration dataset.
Invalid Target Folder	Load Group & Business Group	Target Folder	**Indication:** The Target Folder view lists all of the objects that will be created in sections according to the target folder destination. If the Folder is followed by the phrase "*will be created during load*", the folder does not exist in PDMLink. Target folders can also be reviewed in the metadata verification spreadsheet. See Metadata Verification below. **Action:** The application analyst must review the folder name and determine if the folder should have been created in PDMLink. If so, he should create the folder in accordance with the PDMLink Business Configuration. If the folder name is in error, the SME needs to correct the folder name in the migration data submitted.
Reader Errors	See below		See Reader error table below

Reader Error Report Table

Reader Error	Responsibility	Shown in Report	Description
Failed because object is missing from the dataset	Business Group	Reader Summary	**Indication:** The file is a missing dependent. **Action:** See Missing Dependencies, above.

Reader Error	Responsibility	Shown in Report	Description
Failed because object depends upon an object that is missing	Business Group	Reader Summary	**Indication:** The file depends upon a file that neither exists in the migration dataset nor in PDMLink. **Action:** See Missing Dependencies, above.
Will not load because its model name is duplicated	Load Group & Business Group	Reader Summary	**Indication:** There is more than one object with the same document number. Legend may be mis-configured. **Action:** Load Group: Check options filter.duplicates and dms.object.exists and determine if they are set consistently and appropriately for this migration dataset. Business Group: Review the metadata for the objects and determine is the document number is set correctly. If they are duplicates remove the duplicate file. If the document numbers are incorrect notify the Load Group who can check the rules for determining the document number.
Will not be loaded because its identifier is already used in PDMLink	Multiple Site Business Groups	Reader Summary	**Indication:** An object with the same number already exists in PDMLink. **Action:** See Duplicate Objects, above.
The CAD object was not successfully read	Load Group	Reader Summary	**Indication:** Legend could not open or read the file. This message also applies to document files. The use of the word CAD in the message should be ignored. There could be a problem with the file or with the Legend setup. **Action:** If all files of a given type are failing the problem is likely with the adapter. Run Legend's UnregisterServers.bat file followed by the RegisterServers.bat file. If only a small number of files are failing, review the <Legend_loadpoint>\clienterrorlog.txt file and determine the cause of the error. If more help is needed contact Legend support.
An error occurred while processing this duplicate	Business Group	Reader Summary	**Indication:** A family table is missing and its instances are used by some assemblies. **Action:** Find the missing family table and add it to the migration dataset.

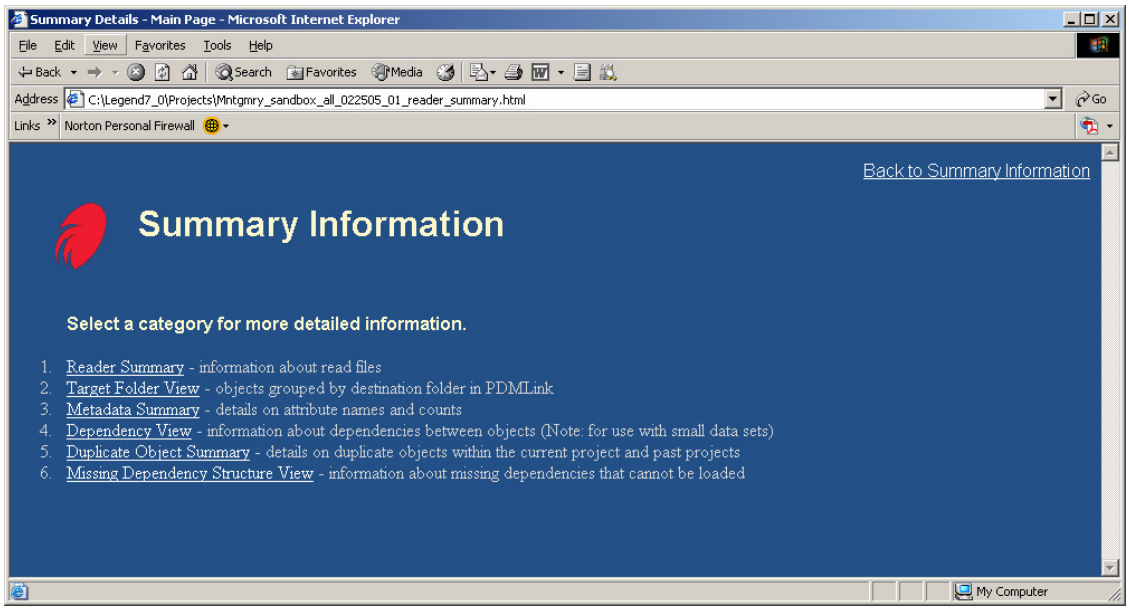
Reader Error	Responsibility	Shown in Report	Description
Other	Load Group	Reader Summary	**Indication:** An error exists. **Action:** Review the logs and reports as defined in the trouble shooting section of the Legend Migration Guide. If more help is needed contact Legend support.

The Reader report will also show the following information:

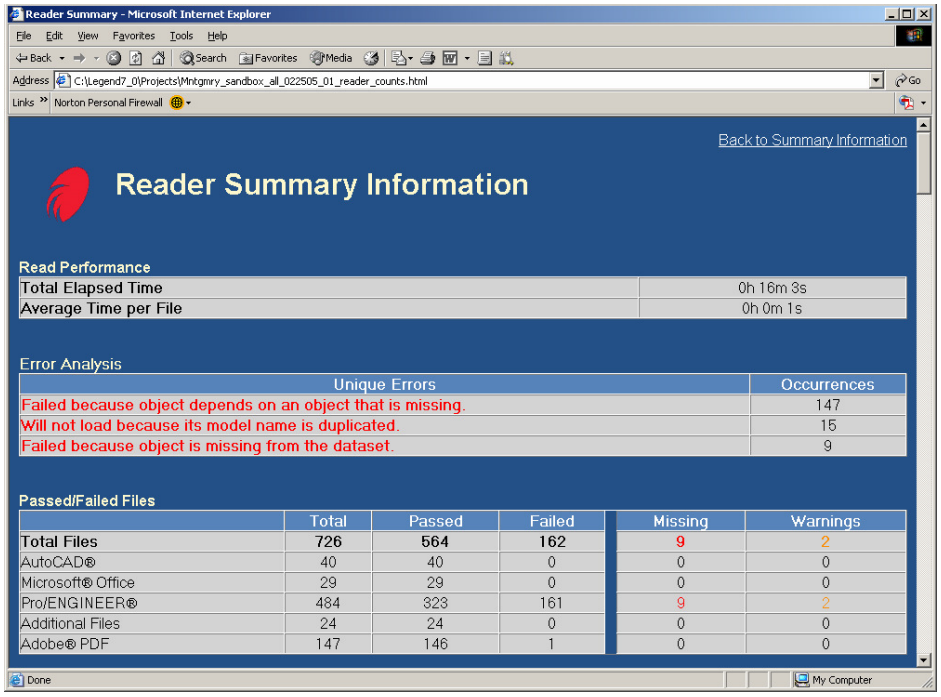
- File counts
- Target folder location
- Metadata counts
- File interdependencies

Information Type	Responsibility	Shown in Report	Description
File Counts	Load Group & Business Group	Reader Summary	The number of files processed, passed and failed broken down by file type.
Target folder location	Load Group & Business Group	Target Folder View	The list of objects that will be loaded grouped by the PDMLink Product, Library and folder. Clicking on the object name hyperlink will display the attributes that are being migrated for that object.
Metadata counts	Business Group	Metadata Summary	The attribute name (in the Key column) and the number of objects that have an attribute defined. A null value for an attribute will be counted. See the metadata spreadsheet for detailed information for the attributes.
File interdependencies	Business Group	Dependency View	A graphical hierarchy showing file interdependencies. This diagram shows the type of dependency between the files. For large datasets this report may take a few minutes to finish displaying.

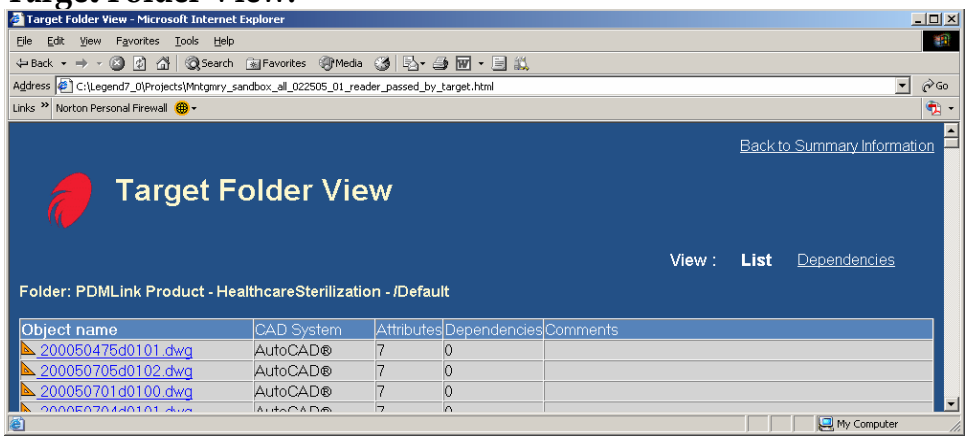
Reader Reports:



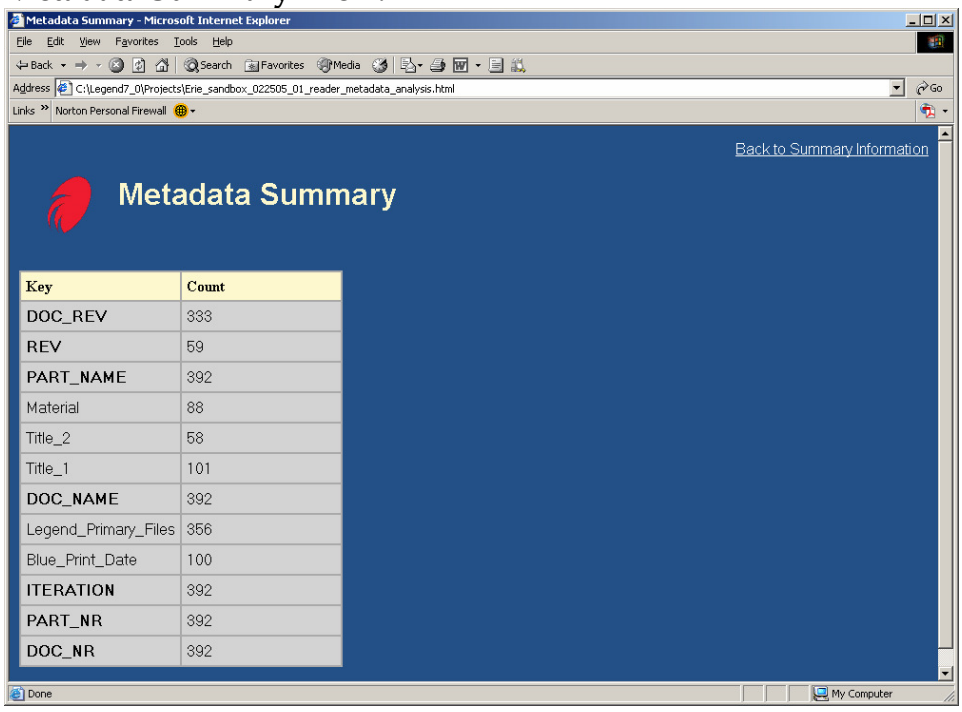
Reader Summary:



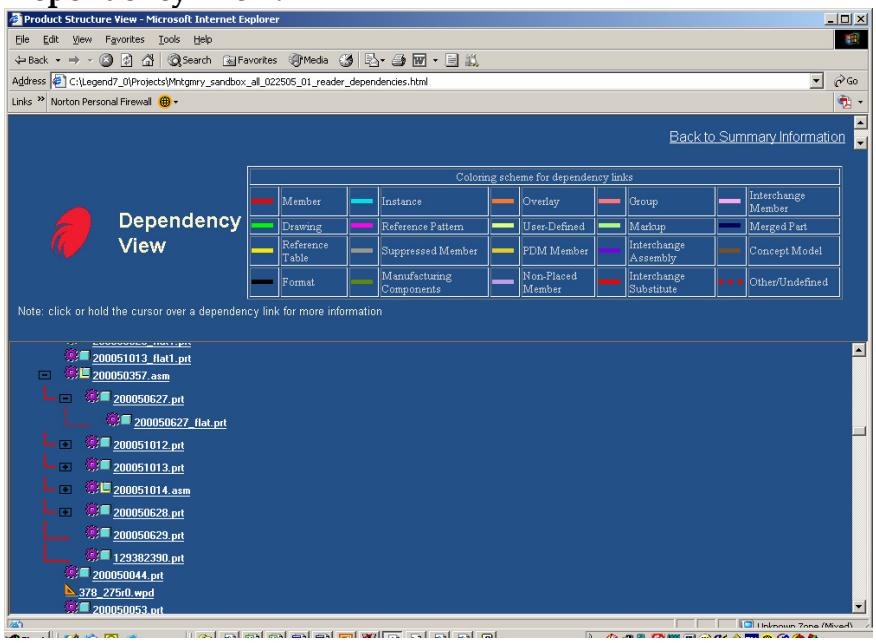
Target Folder View:



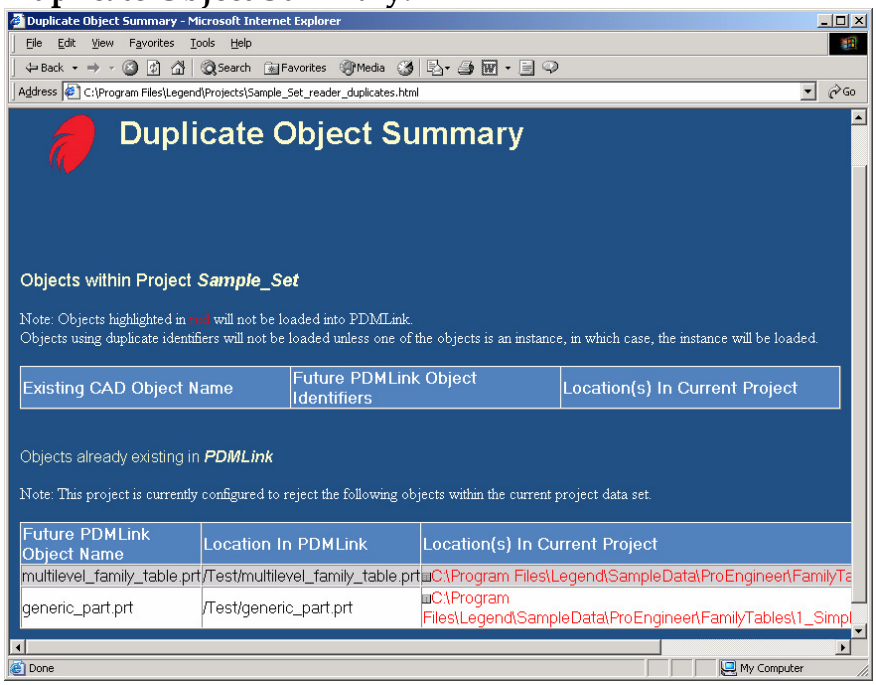
Metadata Summary View:



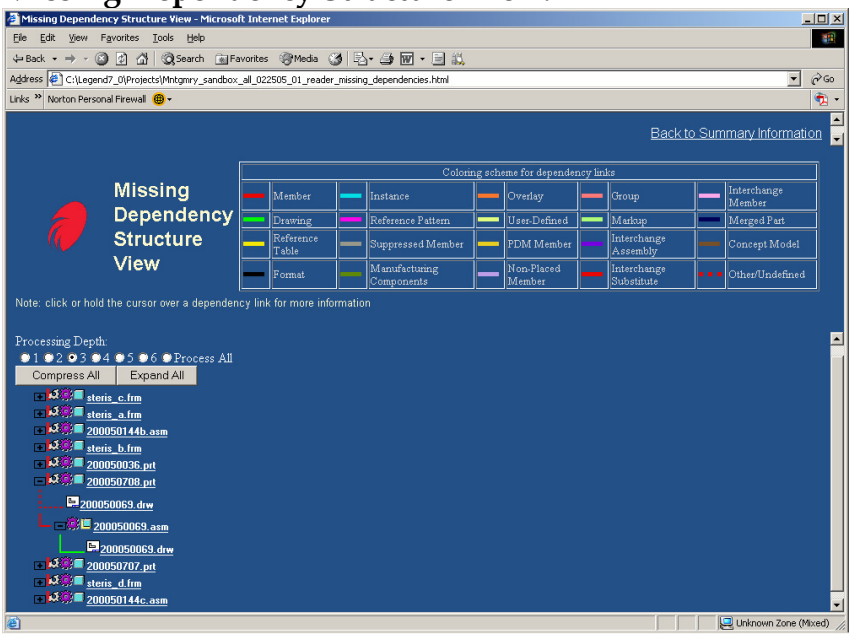
Dependency View:



Duplicate Object Summary:



Missing Dependency Structure View:



Appendix E

Registry Server and related

Requirements to load Legend:
The Work Group manager for Pro/ENGINEER needs to be installed on the PDMLink Method Server.

After installing the MS, edit the registryclient.ini file located under $WT_HOME/codebase/cfg/site.
Enter the hostname of the registry server and the mode.
Save file.

Rebuild jar files via running custom install of WGM from CDROM/setup or downloaded software. Choose only to make client jars. Alternative is to build wmcore.jar, wmproe.jar and wmpref.jar with:
windchill wt.tools.boot.MakeJar jar=xx.jar

Build legend_wt.jar on the Method Server.

Stop MS, clear Tomcat cache by deleting the directory \ptc\tomcat\work\Standalone.

Install WGM client locally a Windows workstation.
Create a cache location on the client; this will be shared with the Legend client.
Copy the legend_wt.jar to the WGM cache location.

Install legend client.

Appendix F

PDMLink Performance considerations

There are a number of considerations that should be taken in regards to Windchill and Oracle performance for a bulk load. Some of these include:

- Queue processing
- Content vaulting
- Replication servers
- Oracle statistics
- WVS CAD workers
- Windchill properties

Some Windchill properties that affect bulk loading performance are listed in the spreadsheet noted here (See PDMLink settings)

Appendix G

Legend Patch Installation

Legend patches are currently delivered via e-mail. Patches can include several types of files. Different file types require a different procedure for installation. Installation instructions delivered with the patch should be followed. Typical instructions are presented here.
Prior to installing a patch, it is best to create a backup copy of the files being replaced.
If you have more than one Legend client, install and verify the patch on one client before installing on other clients.

Installing JAR fles

Jar files should be placed on each Legend client in the <*Legend_loadpoint*>\jars directory. In addition, if the LegendEnterprise.jar is included in the patch, the following steps need to be followed to ensure the Windchill server is updated.

- Copy the LegendEnterprise.jar file to the Windchill server codebase directory.
- Expand the LegendEnterprise.jar file into the Windchill codebase directory using one of the following:
  - Unix: cd ../codebase
jar -xvf LegendEnterprise.jar
  - Windows: Open LegendEnterprise.jar with WinZip
Extract all files, Using folder names, into the codebase.

Installing DLL files

DLL files are normally placed into the <*Legend_loadpoint*> directory. After installing the DLLs run the following:

- <*Legend_loadpoint*>\register_servers.bat

Installing EXE files

EXE files are placed into the <*Legend_loadpoint*>\Adapters directory, with the exception of the GenericCADAdapter which is placed in the <*Legend_loadpoint*> directory. After installing the EXE files run the following:

- <*Legend_loadpoint*>\register_servers.bat

INI options

Some patches include support for new INI options. Add the new options into the appropriate section of the wtcad.ini file. Set the value for the new options according to the instructions received with the patch.

Verifying the Patch

Run Legend using a known set of data, such as the Sample Set installed with Legend. Also, run Legend with a dataset that the patch is to handle differently. Verify the patch is processing as expected.

Appendix H

Multiple Site Configuration

The batch file below can be used to start Legend with different configurations for different sites. It also, can be used to load to the Dev, Test, or Production servers without reconfiguring Legend.

```
@echo off
setlocal
rem Legend Startup script
rem Server setup (for running Legned to test and production servers)
rem   edit the server section of this srcipt to match the servers in use
rem   create a _legend_vars_<server>.bat file for each server in the Legend loadpoint
rem   edit the legend.bat and register_servers.bat  files and comment the call to _legend_vars.bat
rem
rem Site setup (for loading sites with different configurations and customizations)
rem   edit the site section of this script to facilitate the sites from which you will load
rem   create a subdirectory for each site in <legend_loadpoint>\Source\
rem   copy the <legend_loadpoint>\Source\* directories into the Source\<site> subdirectories
rem   customizations for each site need to be done in each site's subdirectory
rem   optionally, create a wtcad.ini file for each site and edit this script to use the appropriate ini file

:MENU_SRVR
cls
:MENU_SRVR2
echo '
echo '
echo '        Legend Startup Menu
echo '
echo '
echo '        Select Server
echo '
echo '           1. Test
echo '           2. Production
echo '
echo '           9. Cancel
echo '
set /P LEGSRVR=Enter Option:

if %LEGSRVR%==1 GOTO :SRVR1
if %LEGSRVR%==2 GOTO :SRVR2
if %LEGSRVR%==9 GOTO :EOF

cls
echo '
echo '
echo ' Invalid option
GOTO :MENU_SRVR2

:SRVR1
SET LEGSRVR=test_server.domain.com
GOTO :MENU_LOC

:SRVR2
SET LEGSRVR=prod_server.domain.com
GOTO :MENU_LOC

:MENU_LOC
cls
:MENU_LOC2
echo '
echo '
```

```
echo '
echo '
echo '
echo '        Select Site
echo '
echo '          0. Default
echo '
echo '          1. Site 1
echo '          2. Site 2
echo '
echo '          9. Cancel
echo '
set /P LEGLOC=Enter Option:

if %LEGLOC%==0 GOTO :LOC0
if %LEGLOC%==1 GOTO :LOC1
if %LEGLOC%==2 GOTO :LOC2
if %LEGLOC%==9 GOTO :EOF

cls
echo '
echo '
echo ' Invalid option
GOTO :MENU_LOC2

:LOC0
SET LEGLOC=Default
GOTO :VALIDATE

:LOC1
SET LEGLOC=Site_1
GOTO :VALIDATE

:LOC2
SET LEGLOC=Site_2
GOTO :VALIDATE

:VALIDATE
cls
:VALIDATE2
echo '
echo '
echo '
echo '  You have selected:
echo '
echo '          Server:  %LEGSRVR%
echo '
echo '          Site:    %LEGLOC%
echo '
echo '
echo '
echo '  Is this corrrect? (y/n)
echo '
echo '
SET /P OPT=Enter Option:

if %OPT%==Y GOTO :SETSRVR
if %OPT%==y GOTO :SETSRVR
if %OPT%==Yes GOTO :SETSRVR
if %OPT%==YES GOTO :SETSRVR
if %OPT%==yes GOTO :SETSRVR
if %OPT%==N GOTO :MENU_SRVR
if %OPT%==n GOTO :MENU_SRVR
```

```
if %OPT%==NO GOTO :MENU_SRVR
if %OPT%==No GOTO :MENU_SRVR
if %OPT%==no GOTO :MENU_SRVR

cls
echo '
echo ' Invalid answer
GOTO :VALIDATE2

:SETSRVR
call _legend_vars_%LEGSRVR%.bat
echo Copying license files %LEGSRVR%
copy /Y %INSTALLDIR%\licenses\%LEGSRVR%\*.dat %INSTALLDIR%\licenses

:SETLOC
rem unregister servers
echo Unregistering DLLs
call %INSTALLDIR%\unregister_servers.bat

rem if %LEGLOC%==Default GOTO :DFLT_LOC

rem copy customized dlls
echo Copying customized DLLs for %LEGLOC%
copy /Y %INSTALLDIR%\Source\%LEGLOC%\Custom_Adapter_Source\*.dll %INSTALLDIR%\adapters\custom
copy /Y %INSTALLDIR%\Source\%LEGLOC%\VB_Associator_Source\*.dll %INSTALLDIR%
copy /Y %INSTALLDIR%\Source\%LEGLOC%\VB_MetadataCustomizer_Source\*.dll %INSTALLDIR%
copy /Y %INSTALLDIR%\Source\%LEGLOC%\VB_Revision_Mapper_Source\*.dll %INSTALLDIR%
GOTO :REG_DLLS

:DFLT_LOC
echo Copying default DLLs for %LEGLOC%
copy /Y %INSTALLDIR%\Source\Custom_Adapter_Source\*.dll %INSTALLDIR%\adapters\custom
copy /Y %INSTALLDIR%\Source\VB_Associator_Source\*.dll %INSTALLDIR%
copy /Y %INSTALLDIR%\Source\VB_MetadataCustomizer_Source\*.dll %INSTALLDIR%
copy /Y %INSTALLDIR%\Source\VB_Revision_Mapper_Source\*.dll %INSTALLDIR%
GOTO :REG_DLLS

:REG_DLLS
rem register servers
echo Registering DLLs
call %INSTALLDIR%\register_servers.bat

rem start Legend
echo Starting Legend
call legend.bat

:EOF
endlocal
```

Appendix I

Building the CATIA Abstraction Library

Windows Pre-requisites

- CATIA V5
- CATIA V5 CAA RADE at same SP level as CATIA V5
- CATIA API at same SP level as CATIA V5
- Microsoft Visual Studio C++ 6.0

Windows client:

1. Install CATIA V5 at a given release and SP level
2. Install CATIA V5 CAA RADE from the same level
3. Install the CATIA API at the same level
4. Install Microsoft Visual Studio C++ 6.0
    a. Select Typical install
    b. Check Register Environment Variables
5. Search Visual Studio installation directories for
    a. Mfc42u.lib
    b. Mfcs42u.lib
6. If installed, go to next step, else copy both files from the Visual Studio CD to the client's <Visual_Studio_loadpoint>\VC98\MFC\Lib directory
7. Run CATVBTLICENSER.EXE from the <Dassault_Systemes_loadpoint>\T13\intel_a\code\bin to select the RADE license.
8. Run CATVBTSETUP.EXE and select all options then Install
9. Start Microsoft Visual Studio
    a. Select Tools / Customize / Add-Ins & Macros
    b. Select all CAA V5 Add-Ins
    c. Close
    d. Click OK on all confirmation windows
10. Close out of Visual Studio and Restart Visual Studio to verify CAA setup
    a. Select File / New CAA V5 Workspace
    b. Set working directory & tool level Next
    c. Create new generic framework
    d. Set Name: (Test)
    e. Click OK on all confirmation windows
11. Download the CATIA_V5_Builder.exe from http://www.i-cubedcart.com/Abstraction_builder/
12. Execute CATIA_V5_Builder.exe
13. Copy the CATIA5Abstraction.dll file to the <Legend_loadpoint>\Adapters\Catia5\<version> directory

Appendix J

Exporting files from Pro/INTRALINK

Checkout the required files in Pro/INTRALINK from the commonspace to a workspace

Configuration requirements are: dependencies all, include generics, include instances

(if exporting Project files you will probably want to include drawings as well)

If a library is already loaded you would want to get the library parts out of the .csv file, so you will want to exclude the library parts from the workspace view as well

Once the files are in the workspace then -

Change the workspace view to display generics only

Print the workspace to a text file (.txt)

Open the .txt file in Excel, then do the required edits to make the .csv

Change the header line

Filename, REV, DOC_LC_STATE, FOLDER

Edit the FOLDER entry to match the folder schema of PDMLink

Version is changed to REV

Release Level is changed to DOC_LC_STATE

Save as .CSV

Export the Pro/ENGINEER generics from the workspace to the appropriate directory

Index