

**precisely**

# Precisely EnterWorks

## Classic Administration Guide

Version 10.4.x



## Notices

Copyright 2007, 2023 Precisely.

### Trademarks

"EnterWorks" and the "EnterWorks" logo are registered trademarks and "Enable PIM", "EnterWorks PIM", "EnterWorks Process Exchange" and "EnterWorks Product Information Management" are trademarks of Precisely.

### Third-party Acknowledgments

Windows, .NET, IIS, SQL Server, Word, and Excel are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Java and all Sun and Java based trademarks are trademarks or registered trademarks of the Oracle Corporation in the United States and other countries.

Oracle is a registered trademark and Oracle 10g is a trademark of Oracle Corporation.

Pentium is a registered trademark of Intel Corporation in the United States and other countries.

JBoss is a registered trademark of Red Hat, Inc.

All other trademarks and registered trademarks are the property of their respective holders.

All icons and graphics, with the exception of the "e." logo, were obtained from West Coast Icons and Design at <http://www.bywestcoast.com>.

# Table of Contents

Document Conventions .....	9
Document Terminology .....	10
Overview .....	11
Logging into EnterWorks Classic.....	11
Quick Start Guide .....	13
Home Page Report Widgets.....	14
Administrative Widgets .....	14
Creating a Shortcut Widget.....	14
Creating a System Health Widget .....	17
Creating a Channel Readiness Widget .....	21
Switching To Other Group Views .....	22
Managing Taxonomies and Hierarchies.....	23
Registering a Hierarchy .....	23
Exporting a Hierarchy.....	25
Importing a Hierarchy .....	26
Copying a Hierarchy .....	29
Creating a Restricted Hierarchy .....	30
Adding a Hierarchy Node or Changing its Metadata .....	32
Managing Code Sets.....	34
Templates.....	38
Import Templates.....	39
Export Templates .....	39
Exchange Templates .....	39
View Exchange Templates .....	39
Create or Edit Exchange Templates .....	39
Syndication Templates.....	42
View Syndication Templates.....	43
Create or Edit Syndication Templates.....	44
Publication Templates .....	51

Multiple Language Support.....	52
Session Language .....	52
Language List.....	52
Access the Language List.....	52
Manage Supported Languages .....	53
Reorder the List of Supported Languages .....	55
Enable or Disable a Supported Language .....	56
Delete a Supported Language.....	58
Defining a Multi-language Attribute.....	58
Editing Multi-language Attribute Values .....	59
Translating Data Model Entities.....	61
Export Data Model Objects for Translation .....	61
Import Translated Data Model Objects .....	67
Translating EnterWorks Labels and Messages .....	68
Validation Levels and Rules .....	75
Create or Edit a New Validation Rule .....	75
Validation Rule Types.....	82
Assign/Unassign Existing Validation Rule to an Existing Attribute.....	88
Defining Attribute Validation Rules for Validation Levels .....	89
Assigning Multiple Attributes to a Validation Rule.....	92
Setting a Syndication/Export's Validation Level .....	94
Setting a Record's Validation Level.....	94
Managing Digital Assets.....	95
Bulk Importing Digital Assets .....	95
Batch Importing Digital Asset Metadata.....	97
Category Attributes.....	100
Manage Category/Dynamic Attributes from the Taxonomy List .....	100
Manage Category/Dynamic Attributes from Repository View.....	108
Using Staging and Production Repositories.....	111
Staging vs. Production Repositories .....	111

Manually Promoting Records .....	113
Manually Promoting a Saved Set Definition .....	114
Manually Promoting a Repository .....	114
Review difference from Production.....	115
Package Promotions .....	116
Package Promotion Repository.....	117
Defining a Data Package .....	118
Job Scheduling .....	120
Scheduling Imports .....	121
Scheduled Imports Repository.....	121
Scheduled Import Jobs Repository .....	130
Scheduled Import Configuration Properties.....	131
Scheduling Exports.....	131
Scheduled Export Repository.....	132
Scheduled Export Jobs Repository.....	141
Scheduled Export Configuration Properties .....	142
Creating a Scheduled Repository export .....	143
Creating a Scheduled View/DataMart Export.....	146
Creating a Scheduled Template Export .....	150
Implementing Pre- and Post-Processing of Scheduled Imports and Exports.....	157
Pre/Post-Processing Block .....	169
Scheduling Promotions .....	175
Scheduled Promotions Repository .....	176
Promotion Jobs Repository.....	180
Scheduled Promotion Configuration Properties .....	180
Channel Readiness .....	181
Data Model Management.....	181
Managing Folders.....	181
Managing Profiles .....	182
Creating and Editing Profiles.....	182

Adding an Attribute to a Repository.....	192
Managing Repositories .....	193
Manage a Repository from the Model Feature .....	193
Manage a Repository from the Content Feature .....	198
Manage a Repository’s Properties.....	199
Manage a Repository’s Trigger Properties .....	203
Manage a Repository’s Promotion Properties .....	207
Repository Link Management.....	215
Configuring a Repository’s Edit Screens .....	229
Managing Attributes .....	238
Making an Attribute a Search Filter .....	238
Managing Snapshot Tables.....	241
Enable or Disable Multi-language Snapshot Table Support .....	242
Add or Remove Attributes from a Snapshot Tables .....	243
Rebuilding, Dropping, or Repopulating a Snapshot Table.....	243
Auto-Generated Sequence Maintenance.....	244
Defining the Sequence Attribute .....	244
Adding a Sequence Definition.....	245
Adjusting the Starting Sequence Number .....	247
Managing the Custom Controls to select linked data .....	249
Migrating EnterWorks.....	251
Migration Process Overview .....	252
Migrate Out Data Model Components .....	253
Migrate In Data Model Components .....	255
System Administration.....	256
UI Customization.....	256
Customize the EnterWorks Classic UI Title Bar.....	256
System Architecture.....	258
EnterWorks Environments.....	258
Controller and Workers Services .....	258

Configuration Files .....	258
Elasticsearch Configuration .....	267
System Management .....	267
Stopping and Starting EnterWorks Services .....	267
Changing the EPIM Database Password .....	270
Users, Groups, and Security Management .....	270
Groups.....	270
Users .....	279
Attribute Security Filter Management.....	286
Record Security Filter Management .....	290
Configuring a Group’s Home Page Dashboard .....	294
Define a Home Page in the Classic UI .....	294
Define a Dashboard in the New UI .....	295
Workflow.....	296
Workflow Properties.....	297
EnterWorks PIM Options .....	299
Work Item Types.....	304
Open Workflow Task Manager .....	311
Monitor Workflow Process.....	311
Manage Workflow Process .....	311
Purge a Workflow – DO NOT USE .....	314
Add a Task to the Worklist.....	316
Logs and Debugging .....	317
Logging Levels .....	317
Log Format .....	318
Using the Logs for Debugging .....	318
Debugging Logs for EPX Workflows .....	319
Configure Cached Repository Permissions Tracing Log Messages.....	320
System Maintenance .....	320
Clear Data Cache.....	320

Configure System Maintenance Tasks ..... 321



# Document Conventions

This EnterWorks document uses the following typographic conventions:

Convention	Usage
pathnames	Pathnames are shown with backslashes, as for Windows systems.
Courier New font	Denotes sample code, for example, Java, IDL, and command line information. May be used to denote filenames and pathnames, calculations, code samples, registry keys, URLs, messages displayed on the screen.  If <i>italicized</i> and in angle brackets (< >), it denotes a variable.
<b>Calibri Font (bold)</b>	When used in body text, it denotes an object, area, list item, button, or menu option within the graphical user interface; or a database name or database-related object. (Examples: the <b>Save</b> button; the <b>Product</b> tab; the <b>Name</b> field; the <b>SKU</b> repository)  Can also be used to denote text that is typed in a text box. (Example: Type “ <b>trackingNo</b> ” in the Name field)
<a href="#">Blue underlined text</a>	Words, phrases or numbers in blue are active links that can be clicked. Clicking these active links will bring the user to the required information, steps, pages, chapters, or URL.

# Document Terminology

This document uses the following terminology:

Convention	Usage
"EnterWorks" and "Enable"	The EnterWorks Enable product is now referred to simply as "EnterWorks". Some system components and images in this document may still retain the name "Enable".
"master" and "slave"	In computing environments, the term "master" has historically been used to describe a process that controls other processes. The controlled processes have been referred to as "slaves". Winshuttle no longer finds these terms acceptable. We are in the process of replacing them with "controller" and "workers" in our documentation, displayed content, and underlying architecture. In the meantime, for further information or instructions on changing the terms in is played content, please contact your EnterWorks account representative.

## Overview

This document covers procedures for administering an organization's EnterWorks system using the EnterWorks Classic User Interface (UI). It covers configuration activities such as, modifying the Taxonomy, code sets, data models, Hierarchies, users, groups, and security. Some of these activities can be assigned to one or more advanced users, but it is recommended that data maintenance activities be given to advanced users. Any structural changes should be handled by the System Administrator. This document also covers system maintenance activities, such as archiving log files, monitoring scheduled jobs, and stopping/starting services.

A System Administrator (a user assigned to the Administration security group) will have access to all EnterWorks system features and functions, but not all functionality is covered in this document. For information regarding the EnterWorks components, installing EnterWorks, using EnterWorks, system maintenance, system architecture, and installing or using EnterWorks plugins, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Logging into EnterWorks Classic

To log into EnterWorks Classic, perform the following steps:

1. Open the web browser.
2. Enter the EnterWorks Classic URL :  

```
http://<servername>/webcm/
```
3. Select a **Language**, Enter your **Login ID** and **Password** and click the **Logon** button.
4. The EnterWorks startup screen appears.

Depending upon the user's permissions, the startup screen could have different features available. The screen shown above contains all features and functions. The Home Page tab is shown on system entry. Home Pages are user group specific; the screen above shows an example of three possible report widgets or internal web applications.

EnterWorks® Enable System Administrator

Home System Administrator Log Out Quick Links

**Content**

- Office
  - Item
  - Relationships
  - ItemProd
  - Office Product
  - Office Product Prod
  - Product Line
  - Product Line Prod
  - Brand
  - BrandProd
  - Customer
  - CustomerProd
  - Manufacturer
  - ManufacturerProd
- Import & Export
- Publication
- Assets
- Reports
- Model
- Model Configuration
- Folders
- Data Connectors
- Users and Groups
- Security Filters
- Migration
- System

**Item Status**

**Item Status**

Status	Count
Valid and in production	3706
Valid but not in production	10
Severe	8259
Warning	1214
Not Validated	358

**Shortcuts**

- Add New Product
- Add New Item
- Open Item
- Copy Item
- Template Creation

**Repositories**

Office

Name

Repository	Type	Total	Valid	Errors	Not Validat
Item	Stagin	13547	3716	9473	358
Relationships		26217	26216	0	1
ItemProd	Produc	8111	7930	2	179
Office Product	Stagin	6163	6080	17	66
Office Product Pr	Produc	6937	6936	1	0

Reload Page 1 of 1 50 View 1

# Quick Start Guide

The following Quick Start Guide for EnterWorks is not meant to describe all functionality, but to identify the most used. The following sections go into more detail and provide step-by-step instructions.

## EnterWorks Enable™ Quick Start Guide

**Assets**  
Maintain digital images, documents, and other digital assets associated with your data

**Repositories**  
Click to open a table of product information in the right frame

**Imports & Exports**  
Bulk import or extract your product data

**Publications**  
Create publications like catalogs or sales sheets from your product data

**Feature Bar**  
(Permissions determine features shown)

**Search Area**

**Record View**

**Annotations:**

- Click to close/shrink Features Bar
- Refresh tab function
- Switch between open functions in the tab bar
- Select all records on the page
- Select the full set of records
- Log out of Enable
- Change record view
- Refresh Workspace
- Update User Profile
- Quick access to your functions
- Create a personal view of the data
- Record Activities
- Record Toolbar
- Click on any blue heading to sort
- Shows if the record has a validation error
- Shows the total number of records in list after any filters are applied
- Page and page navigation
- Select to show additional records per page
- Limit records shown on right using search, hierarchy, or saved set filter tabs
- Select individual records for action
- Click to close/shrink Search Area
- Use these advanced searches/filters to limit records shown on right pane
- Current searches/filters applied shows here

#	Error	Image	Brand Id	Brand	Status
1	Ind		33908	APC	Active
2			33909	Apple	Active
3			33910	Belkin	Active

View 1 - 10 of 49

Revised - 12/04/2014


# Home Page Report Widgets

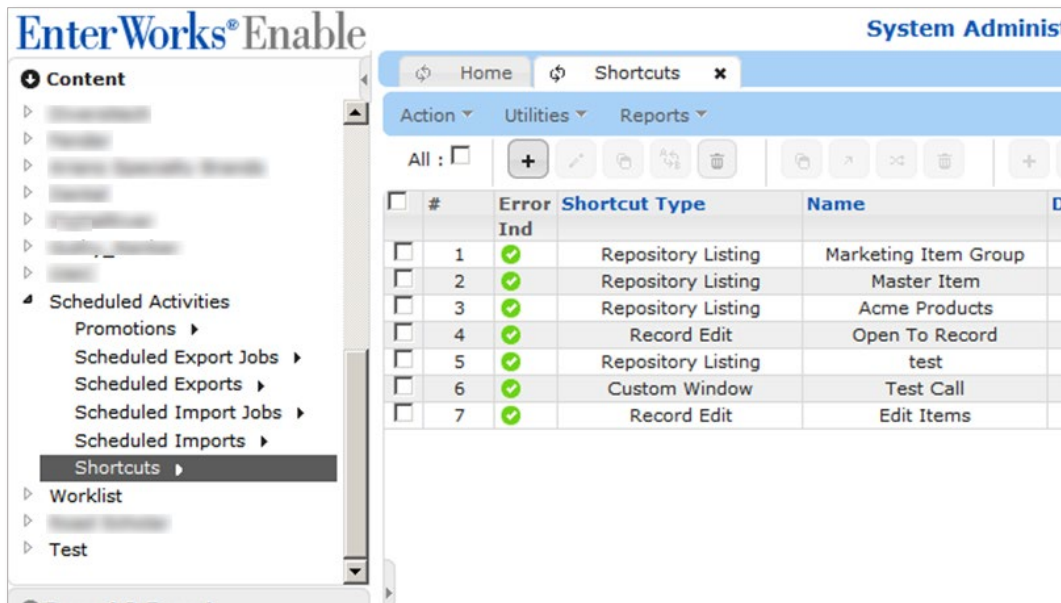
This section describes report widgets designed for administrative use, plus how to set up default home pages for each user group. When a user first signs on, the default home page associated with the user group assigned will be shown. The user can then modify their home page, as desired, without changing the default home page for the user group. For information about configuring user-oriented widgets, see the Precisely EnterWorks online help at <https://support.precisely.com>.


## Administrative Widgets

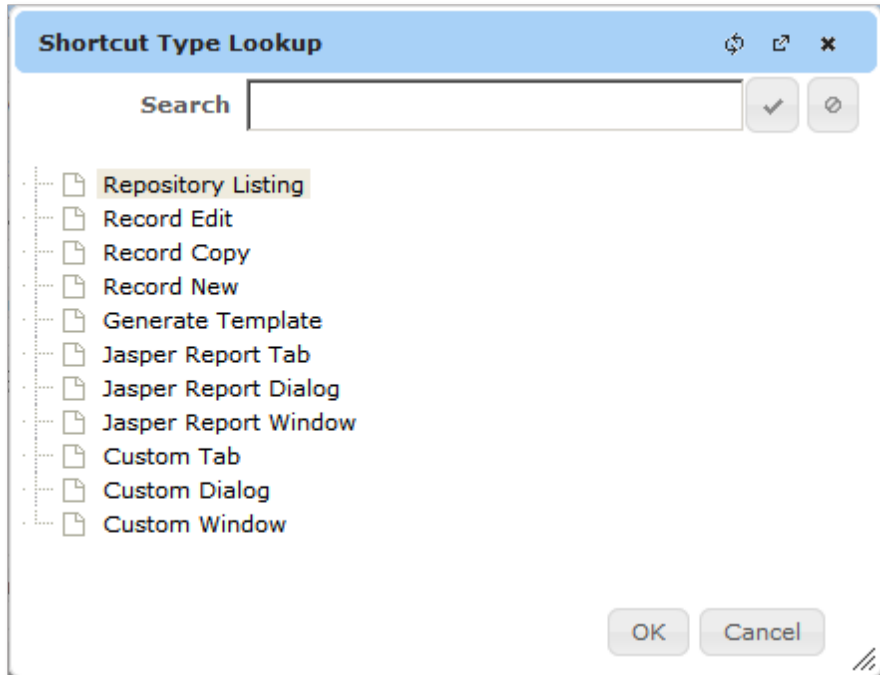
### Creating a Shortcut Widget

The Shortcuts widget can be used to set up quick links to certain functions so users can access them via the Home Page instead of navigating from the Feature Bar. These shortcuts are customized by editing the Shortcuts repository as shown below. The Shortcuts repository is found in the Scheduled Activities or the Workflow folder in the Feature bar.

1. Open the Shortcuts repository.
2. Click the Create a new record button  .



3. The **Add New record** window appears.
4. In the **Shortcut Type** field, click the lookup button  on the far right to open the dropdown. Select the Shortcut Type.



5. Click the **OK** button.
6. Enter a **Name** for the Shortcut which will appear in the widget as a link.
7. Enter a **Description** for the Shortcut (optional).
8. Select a **Group Name** (optional) which will limit the shortcut to a Shortcut Widget to any user within the group.
9. The **Id** will be generated by the system when the record is saved.
10. The default **Display Order** is last; this can be changed in this field by entering a number.
11. Click on **Details** tab to enter the details for the Shortcut Type. The required fields will change based on the **Shortcut Type**. The example below shows the Repository Listing details.

The screenshot shows a configuration window with three tabs: Summary, Details, and States. The 'Repository' section is expanded, showing several fields with dropdown menus:

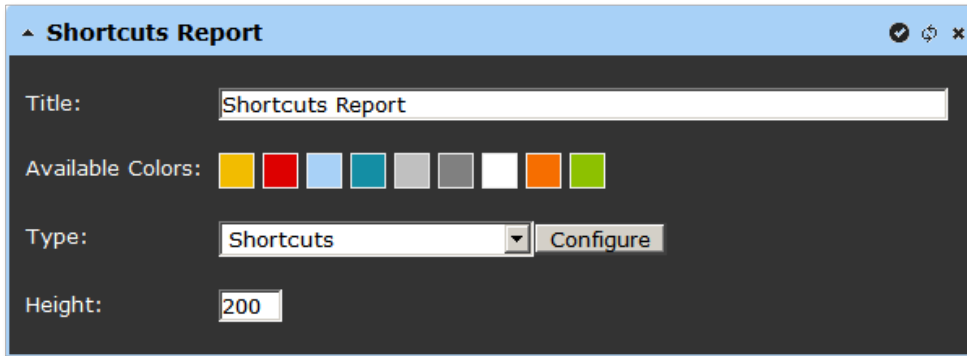
- Repository Name:** PIM\_Item\_Staging
- Preference Listing:** (empty)
- Saved Set:** (empty)
- Saved Search:** (empty)
- Search Attribute:** (empty)
- Search Operator:** (empty)
- Search Value:** (empty)

At the bottom right, there are 'Save' and 'Cancel' buttons.

- a. Select the **Repository Name** to identify the repository that will open up when the widget link is clicked.
  - b. Select the **Preference Listing** to indicate the Preference View that will be shown when the listing is opened (optional). If no Preference Listing is selected, the default preference will be shown.
  - c. Select a **Saved Set** and/or **Saved Search** to apply to the repository before displaying the records (optional).
  - d. Select a **Search Attribute**, **Search Operator**, and **Search Value** to execute on the repository before displaying the records (optional).
  - e. Click the **Save** button.
12. When finished creating the Shortcut records, add the Shortcut Widget to the home page (and/or group home page) as shown below.
- a. Click the plus **+** button on the top-right of the Home Page to add a new widget.
  - b. Enter a **Title** for your report widget.



- c. Select the **Type** = “Shortcuts” from the drop-down list.
- d. Select a different color for the report widget (optional).
- e. Change the **Height**, if required to show all the shortcut links (optional).
- f. There is no configuration window for this widget.



- g. Click the check button on the top right and the **Shortcut** widget will be shown.



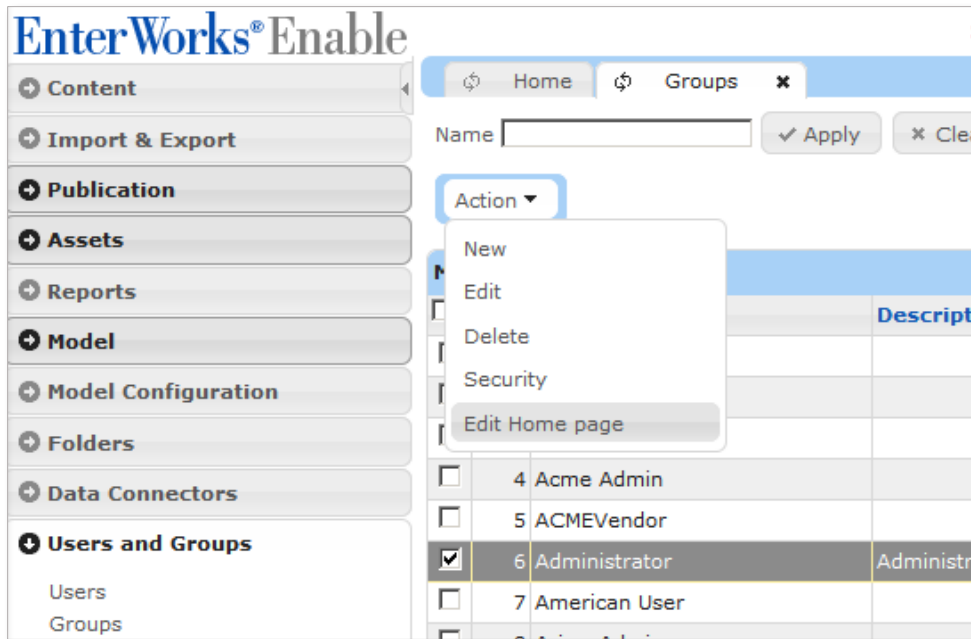
- h. Click the **Save** button on the top-right of the home page to save the new widget.

## Creating a System Health Widget

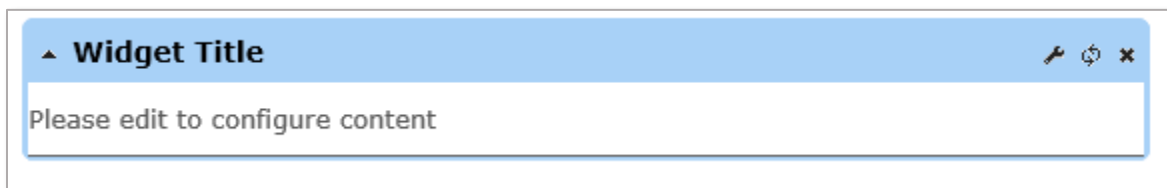
This widget is designed to help a Systems Administrator monitor the components of the EnterWorks system. Therefore, it is recommended this widget be added to the Administrator home page.


1. Open the **Users & Groups** Feature on the Feature bar.

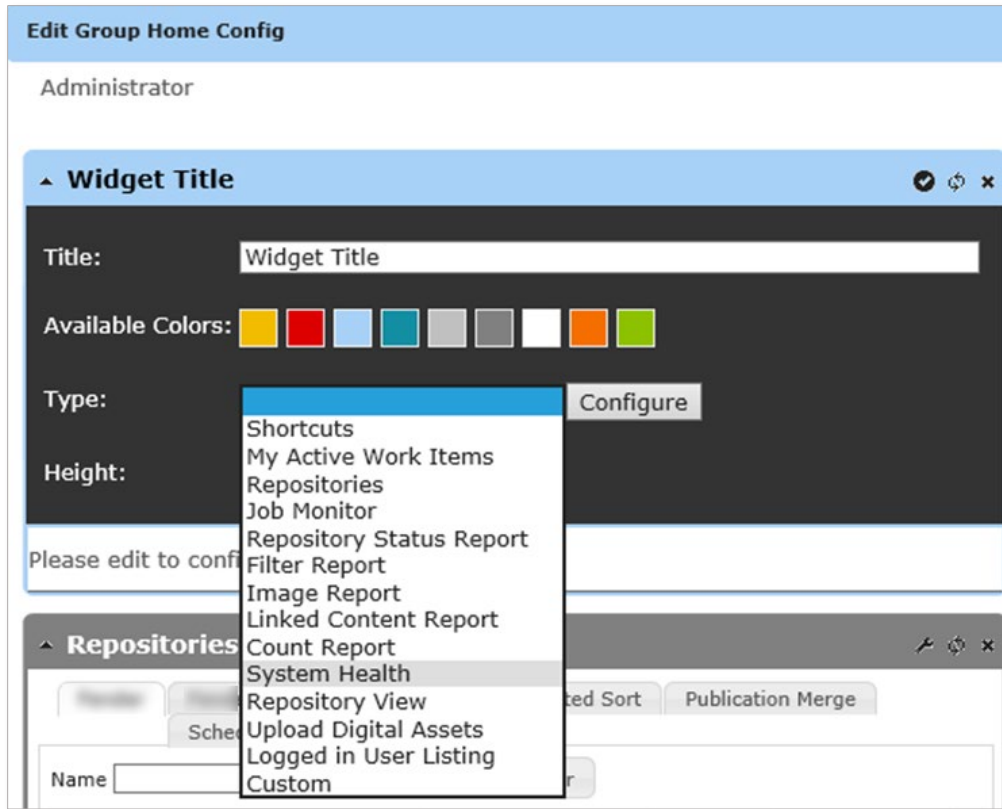
2. Open the **Groups** Function.
3. Select the **Administrator** group record.
4. Click on the **Action** drop-down list and select the **Edit Home Page** option.



5. Click the plus **+** button in the top-right of the Home Page to add a new widget.
6. The new widget will be added to the bottom of the home page.



7. Click the wrench  button in the upper right corner of the window. The Widget editor will open.



8. Enter a **Title** for your report widget.
9. Select a different color for the report widget (optional).
10. Select the **Type** = “**System Health**” from the drop-down list.
11. Change the **Height**, if required (optional).
12. Click the **Configure** button (the first time it will show automatically).

**Configure - System Health**

**EPIM**

- Tomcat Master
- Jboss Master
- Jboss Slaves
- Database
- Job Monitor
- Web File System Access
- Application File System Access

**Enable**

- Enable Web

**EPX**

- Tomcat
- Jboss

EPX User  
admin

EPX Password  
•••••

- BIC Manger

Workflow Process Name  
Ping-Enable

Starting Point Activity Name  
Start

End Point Activity Name  
End

File System

Save Cancel

13. Change default settings (optional).

14. Click the **Save** button.

**System Health Report**

Title: System Health Report


Available Colors:

Type: System Health ▼ Configure

Height: 200

15. Click the check button on the top right and the **System Health Report** widget will be shown:

System Health Report		
SERVICE	STATUS	MESSAGE
EPIM_TOMCAT_MASTER	✓	OK
EPIM_DATABASE	✓	OK
EPIM_JBOSS_MASTER	✓	OK
EPIM_JBOSS_SLAVE	✓	2 total slaves.
EPIM_JOB_MONITOR	✓	0 running jobs. 7 queued jobs.


16. Click the **Save** button  on the top-right of the home page to save the new widget on the Administrator's home page.

## Creating a Channel Readiness Widget

To configure a Channel Readiness widget:

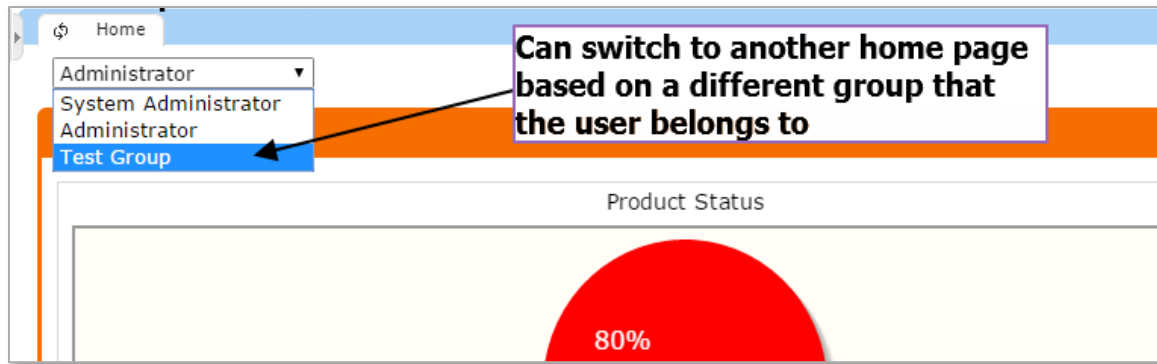
1. In the **Feature** bar, open the **Users and Groups** folder and select **Groups**.
2. Select the group that is going to get the widget.
3. Click on the **Action** drop-down list and select the **Edit Home Page** option.
5. Click the + symbol in the top right of the **Edit Group Home Config** window to add a new widget.
6. Click the wrench icon at the top right of the new widget to open the configuration menu for the new widget.
7. Edit the configuration options:
  - **Title:** Enter a title for the widget.
  - **Type:** Select **Channel Readiness**.

- If you are configuring the widget for use by users with administrative privileges, select 'yes' for Admin, otherwise select 'no'.
- If the widget is for inbound channels select 'yes' for inbound, otherwise select 'no'.
- **Height:** Enter 600 or more.

8. Click the Checkmark symbol at the top right of the widget to save the configuration.
9. Click the disk icon  at the top right of the Edit Group Home Config window to save the Home Page.
10. The widget should now be visible for the users in the Channel Manager group.

## Switching To Other Group Views

Users may belong to several groups. To go to the home page of a different group you belong to, select the group from the down list on the top left of the home page tab.



## Managing Taxonomies and Hierarchies

This section describes:

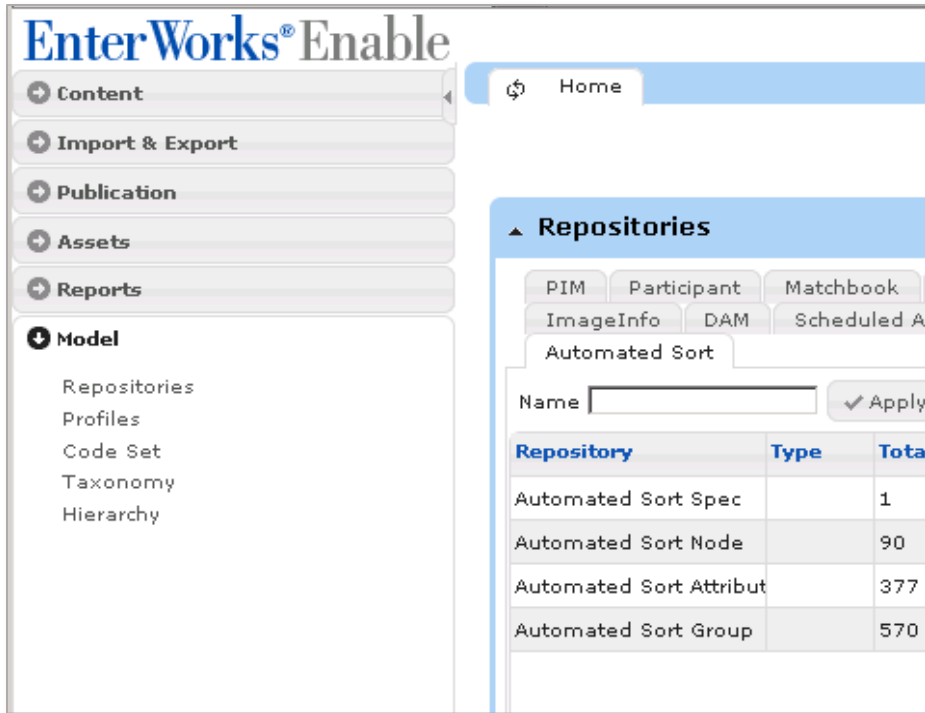
- Registering a Taxonomy or Hierarchy for use.
- Importing and exporting Hierarchies to, from and within environments (for example, from a Dev server to a QA server, copying a Hierarchy).
- Copying Hierarchies.
- Creating a restricted Hierarchy.
- Managing a node's metadata.

For instructions on defining and modifying Taxonomies or Hierarchies, see the Precisely EnterWorks online help at <https://support.precisely.com>.

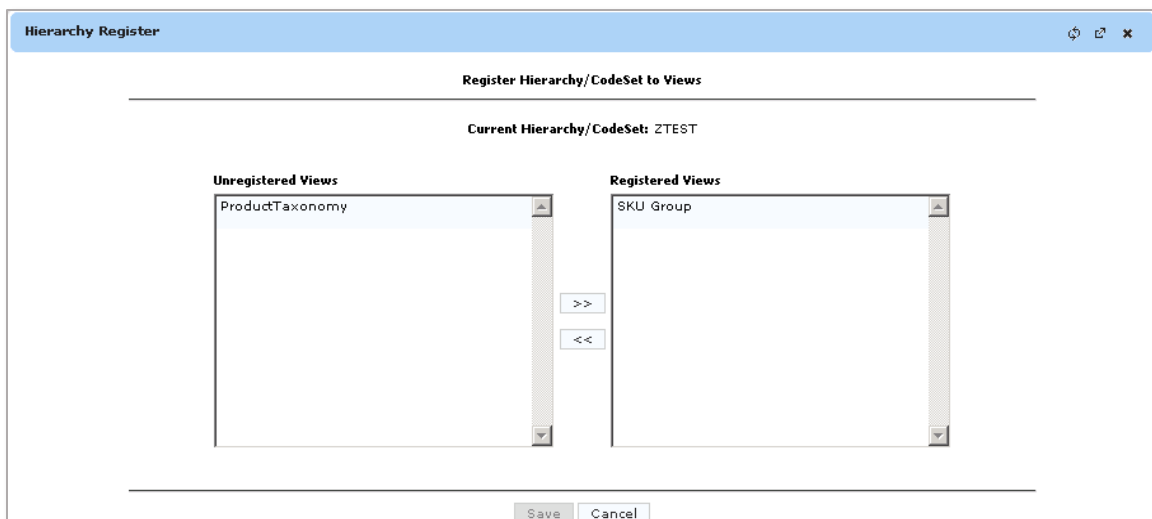
### Registering a Hierarchy

To register a Hierarchy:

1. Navigate to the **Model** Feature in the **Feature** bar and click on the **Hierarchy** (or **Taxonomy**) function option.



2. Select the desired Hierarchy.
3. Under the **Utilities** drop-down menu, select the **Register** option.
4. Select the desired repository and use the arrow button to move it to the **Registered Views** list.



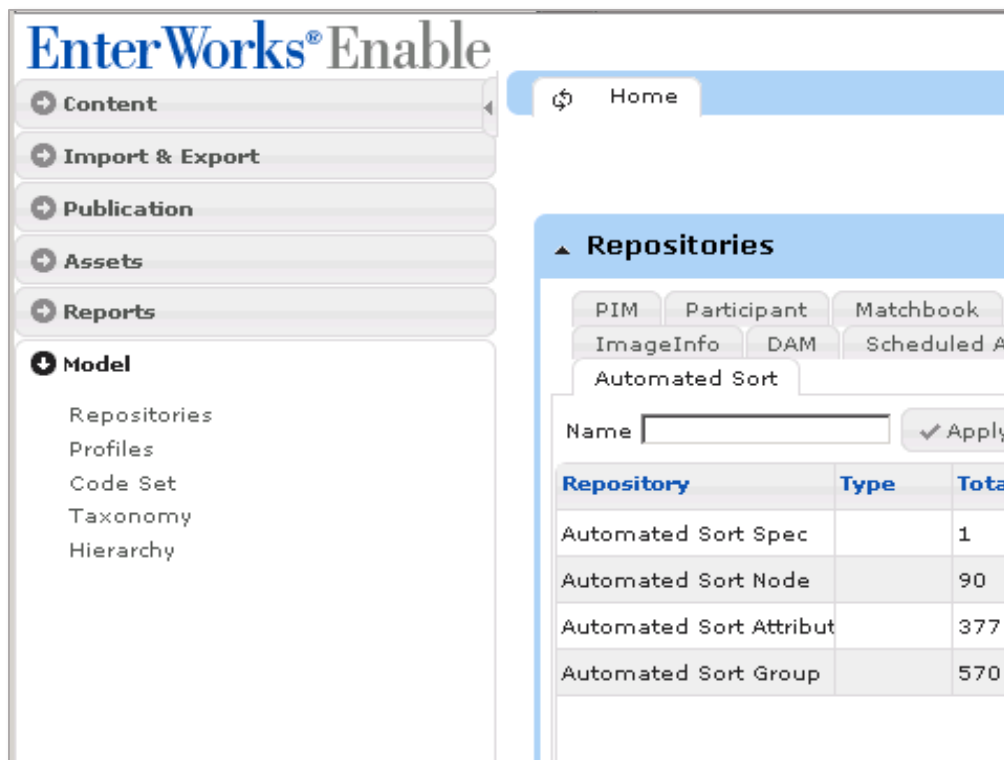


5. Click the **Save** button.

## Exporting a Hierarchy

To export a Hierarchy:

1. Navigate to the **Model** Feature in the **Feature** bar and click on the **Hierarchy** function option.



The screenshot shows the EnterWorks Enable interface. On the left is a navigation menu with categories: Content, Import & Export, Publication, Assets, Reports, and Model. The Model category is expanded, showing sub-items: Repositories, Profiles, Code Set, Taxonomy, and Hierarchy. On the right, the 'Home' breadcrumb is visible, followed by a 'Repositories' section. This section contains several tabs: PIM, Participant, Matchbook, ImageInfo, DAM, Scheduled A, and Automated Sort. Below the tabs is a 'Name' input field and an 'Apply' button. A table displays the following data:

Repository	Type	Total
Automated Sort Spec		1
Automated Sort Node		90
Automated Sort Attribut		377
Automated Sort Group		570

2. Select the Hierarchy that you want to export.
3. Select the **Export** option under the **Utilities** drop-down menu.
4. Select the desired format of the exported Hierarchy.
5. The name of generated file will be shown in a pop-up window. Click the file name link to download the generated export file.

6. Open and view the generated export file.

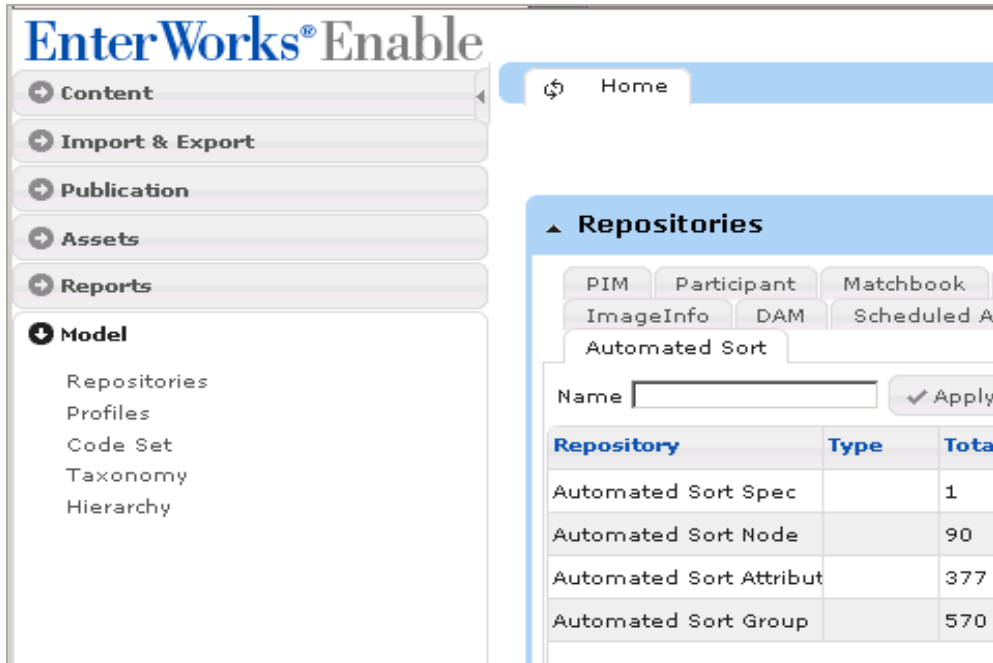
```
Code,Description,Parent Code,Is_Nav,Ind,Code Owner,User Group,IND,Sequence,Code Id
Abrasive,Abrasive,,1,,1,1338556
Abrasive.Non woven Abrasive,Non woven Abrasive,Abrasive,0,,2,1315170
Abrasive.Non woven Abrasive.Non woven Belt Abrasive,Non woven Belt Abrasive,Abrasive.Non woven Abrasive,0,,3,13153:
Abrasive.Non woven Abrasive.Non woven Disc Abrasive Parts & Accessories,Non woven Disc Abrasive Parts & Accessories,Abr.
Abrasive.Non woven Abrasive.Non woven Disc Abrasive,Non woven Disc Abrasive,Abrasive.Non woven Abrasive,0,,5,13152:
Abrasive.Non woven Abrasive.Non woven Pad Abrasive Parts & Accessories,Non woven Pad Abrasive Parts & Accessories,Abras
Abrasive.Non woven Abrasive.Non woven Pad Abrasive,Non woven Pad Abrasive,Abrasive.Non woven Abrasive,0,,7,1315423
Abrasive.Non woven Abrasive.Non woven Roll Abrasive Parts & Accessories,Non woven Roll Abrasive Parts & Accessories,Abr.
Abrasive.Non woven Abrasive.Non woven Roll Abrasive,Non woven Roll Abrasive,Abrasive.Non woven Abrasive,0,,9,13152:
Abrasive.Non woven Abrasive.Non woven Sponge Abrasive,Non woven Sponge Abrasive,Abrasive.Non woven Abrasive,0,,10,:
Abrasive.Non woven Abrasive.Non woven Star Abrasive,Non woven Star Abrasive,Abrasive.Non woven Abrasive,0,,11,1315:
Abrasive.Non woven Abrasive.Non woven wheel Abrasive Parts & Accessories,Non woven wheel Abrasive Parts & Accessories,Abr
Abrasive.Non woven Abrasive.Non woven wheel Abrasive,Non woven wheel Abrasive,Abrasive.Non woven Abrasive,0,,13,13:
Abrasive.Non woven Abrasive.Steel wool,Steel wool,Abrasive.Non woven Abrasive,0,,14,1315452
Abrasive.Non woven Abrasive.Steel wool,Steel wool,Abrasive.Non woven Abrasive,0,,15,1315452
```

## Importing a Hierarchy

The easiest way to determine the columns of an imported Hierarchy file is to export the Hierarchy as shown above. The exported file can then be modified and reimported to update the Hierarchy. This is useful if there are a large number of changes that need to be made.

To import a Hierarchy:

1. Open the **Feature** bar, open the **Model** folder, and select **Hierarchy**.



2. The **Hierarchy** tab will open. Select the Hierarchy that you want to import into and select the **Import** option under the **Utilities** drop-down menu.
3. Check the Import options to make sure the import will function as you desire. Click the **Next** button.

**Print Hierarchy (Aftermarket)**

Name:

Group:

Description:

Description in Other Languages:

Include Owner:

- Yes Default Owner:
- No

Path Code:

- No Path
- Full Path
- Relative Path

Path Format:

- Delimited
- Fixed

Source:

- Detail Editor
- Import from a file
- Auto Build from Selected Attributes

Import Options:

- What should be done with codes that exist in the source file but not in the hierarchy?
  - Add new code
  - Do not add new code
- What should be done with codes that do not exist in the source file but are in the hierarchy?
  - Keep existing code
  - Remove existing code
- What should be done with codes that exist in both the source file and in the hierarchy?
  - Replace code details with source file details

4. Add or remove levels, if required.
5. Click the **Next** button.

**Print Hierarchy (Aftermarket) - Level Editor**

Level	Name
<input type="checkbox"/> 1	<input type="text" value="Level 1"/>
<input type="checkbox"/> 2	<input type="text" value="Level 2"/>

Buttons: Add, Remove, Back, Next, Cancel

6. Click the **Browse** button and navigate to the input file.

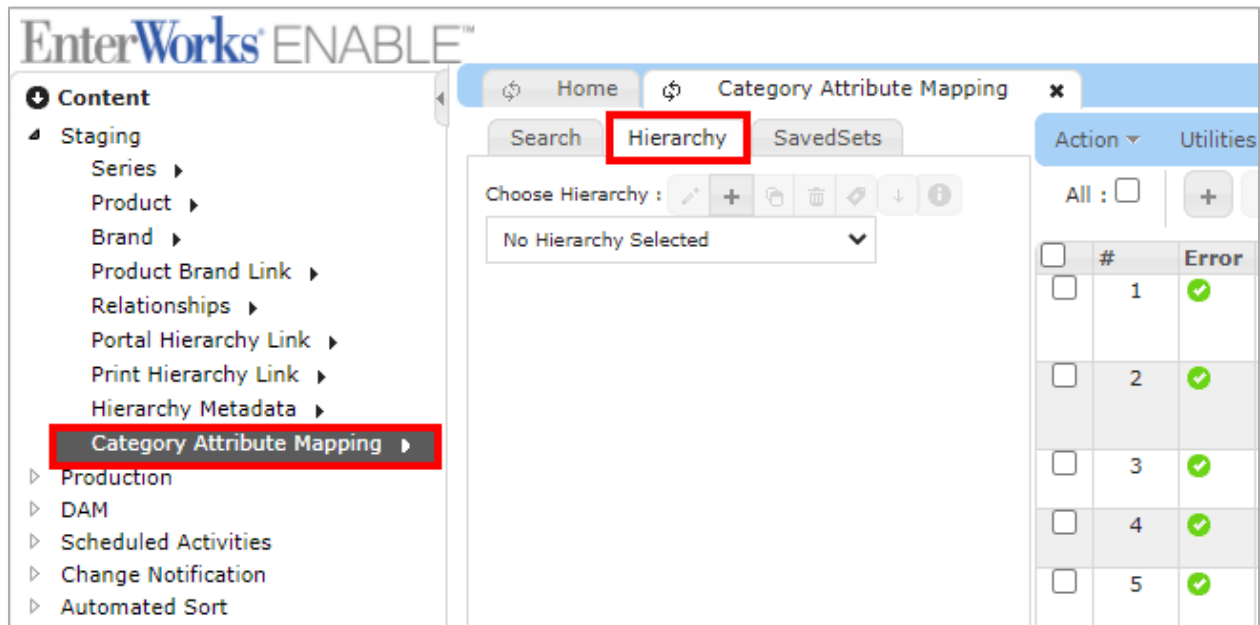
7. Click the **Open** button.


## Copying a Hierarchy

Existing Hierarchies and all repository items attached to their nodes can be copied to a new Hierarchy by users with administrative privileges. After the copy is complete, each Hierarchy is separate and changes to one Hierarchy do not affect the other Hierarchy.

To copy a Hierarchy:

1. Open the repository associated with the Hierarchy.
2. Click on the **Hierarchy** tab, (located between **Search** and **Saved Sets** tabs).



3. Select a Hierarchy from the **Choose Hierarchy** dropdown.
4. Click the copy  icon.
5. A window will appear that asks you to enter a name for the new Hierarchy. Enter it and click **Copy**.

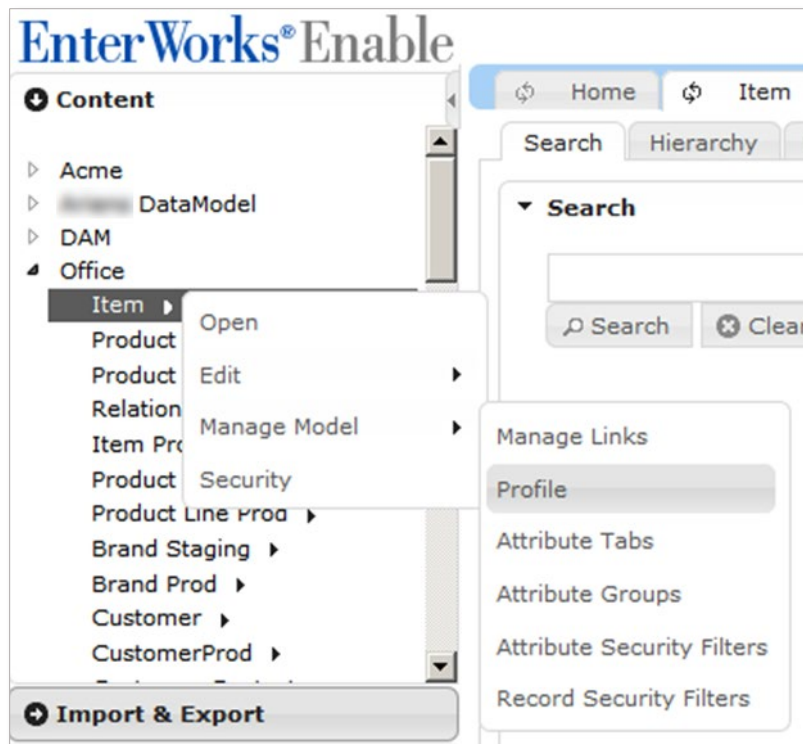
## Creating a Restricted Hierarchy

A restricted Hierarchy is similar to a Taxonomy. An attribute in the repository is used to store a record's Hierarchy node assignment, so a record can only be assigned to one node in the Hierarchy. However, the record can be assigned to nodes in multiple restricted Hierarchies. Each restricted Hierarchy's node assignment is stored in its own repository attribute. Restricted Hierarchies cannot have category attributes.

A Hierarchy can be set up by any user that has been given permission, but only an Administrator can make the Hierarchy restricted.

To make an existing Hierarchy into a restricted Hierarchy, create an attribute on the repository to capture the Hierarchy node:

1. On the **Feature** bar, open the folder of the desired repository, then click the arrow to the right of the repository's name.
2. Select **Manage Model**.
3. Select the **Profile** option.



4. Click the **Next** button.



The screenshot shows the 'Profile (New)' window. At the top, there are fields for 'Name' (containing 'Item'), 'Description', 'Type' (a dropdown menu set to 'EPIM'), and 'Is Valid' (a dropdown menu set to 'True'). Below these fields are two tabs: 'Profile Properties' and 'Profile Property Rules'. The 'Action' menu is open, displaying a table with the following columns: 'Name', 'Data Type', 'Default Value', 'Code Set', and 'Display Order'. At the bottom of the window, there are four buttons: 'Back', 'Next', 'Save', and 'Cancel'. The 'Next' button is highlighted in blue.

5. Click the **New** option under the **Action** menu.

6. The **Define Attribute Details** screen will appear.

The screenshot shows the 'Define Attribute Details (Attribute ID)' window. It has several sections. On the left, there are fields for 'Common Attribute' (a dropdown menu set to 'Attribute ID[BIGINT]'), 'Name' (containing 'Attribute ID'), 'Restricted Name', 'Description' (containing 'ID of the Automated Sort Attribute to which this group is assigned.'), 'Group' (a dropdown menu with a 'Manage' button), 'Associated Group' (a dropdown menu with a 'Manage' button), 'Code Set' (a dropdown menu with a 'Manage' button), 'Category Attribute Association' (a dropdown menu with a 'Manage' button), 'Special Function Ind' (a dropdown menu), and 'Control Type' (a dropdown menu). On the right, there are fields for 'Data Type' (a dropdown menu set to 'BIGINT'), 'Default Value', and 'Data Precision' (containing '[ex:00000=5]'). Below these fields are several checkboxes: 'Is Multi Language', 'Is Category Attribute', 'Is Required' (checked), 'Is Primary Key', 'Is Repeatable', and 'Is Business Key'. At the bottom, there are two buttons: 'OK' and 'Cancel'.

7. Enter a **Name** for the field that will contain the Hierarchy node.


8. An attribute name is a translatable field. To add translations, simply select the multi-language icon  to open the multi-language editor.
9. Enter a **Description** (optional) to better describe the attribute.
10. An attribute description is a translatable field. To add translations, simply select the multi-language icon  to open the multi-language editor.
11. Change the **Data Type** to VARCHAR.
12. Enter a numeric value in the **Data Size** to fit the largest node name.
13. Select the **Group** that the attribute will be assigned to.
14. Select the **Associated Group** (optional) for the attribute.
15. Select the restricted Hierarchy name under the **Code Set** drop-down list.
16. In the **Special Function Ind** dropdown, select **Category Node Link**.
17. Click the **OK** button.
18. Click the **Save** button.

## Adding a Hierarchy Node or Changing its Metadata

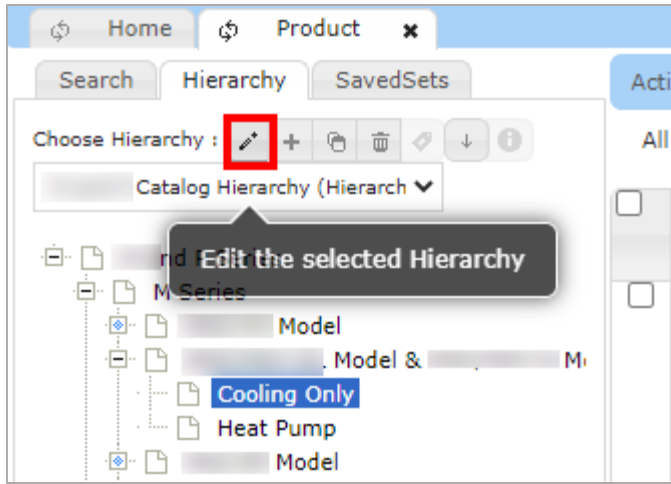
Metadata properties can be added or changed from a repository's Hierarchy tab. It is necessary that the "codeSet\_metadata" type repository is already set up and attached to the Hierarchy.

- Profile name (suggested): <name>HierarchyProperties
- Profile type: CODE\_SET\_METADATA
- Metadata repository name (suggested): <name>HierarchyProperties

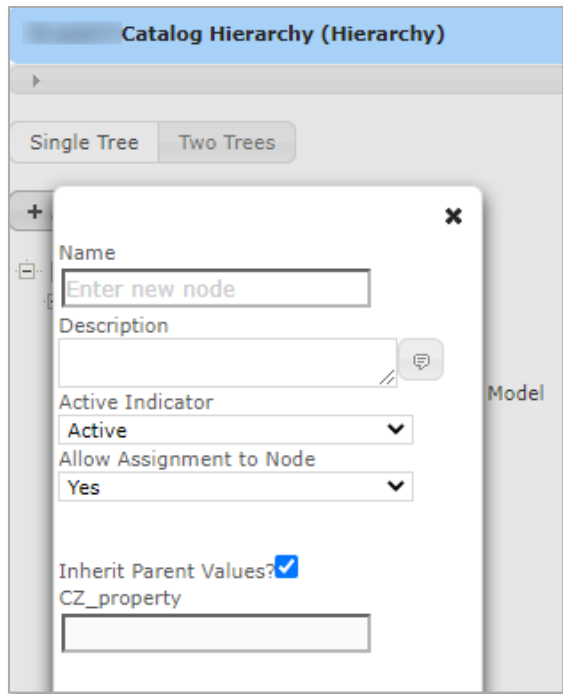
To add a Hierarchy node or change an existing node's metadata:

1. On the **Feature** bar, open the folder of the desired repository, click the arrow to the right of the desired repository's name, and select **Open**.
2. Click on the **Hierarchy** tab, (located between **Search** and **Saved Sets** tabs).
3. Select the desired Hierarchy from the **Choose Hierarchy** dropdown.
4. Drill down to the desired node and select it.
5. Click the pencil icon  at the top of the **Hierarchy** tab.





6. The **Hierarchy** page pops up.
7. To add a node, select the desired parent node and click the **Add** button. A new node will appear in the node editor.
8. To edit a node's metadata, double click the desired node. The node will appear in the node editor.
9. The node editor shows the contents of the node's `<name>HierarchyProperties` record.



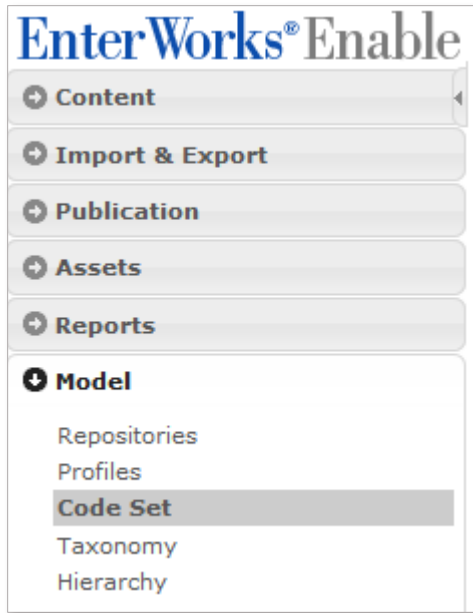
10. Edit the nodes metadata and click **Save**.
11. Open up the `<name>HierarchyProperties` repository to verify.

## Managing Code Sets

Code Sets are created by System Administrators. Data in Code Sets is usually maintained by one or more users. For details on maintaining Code Set data, see the Precisely EnterWorks online help at <https://support.precisely.com>.

To create or delete a Code Set.

1. Click on the **Model** feature in the Feature bar.
2. Click on the **Code Set** function under the **Model** feature to show all the defined code sets.

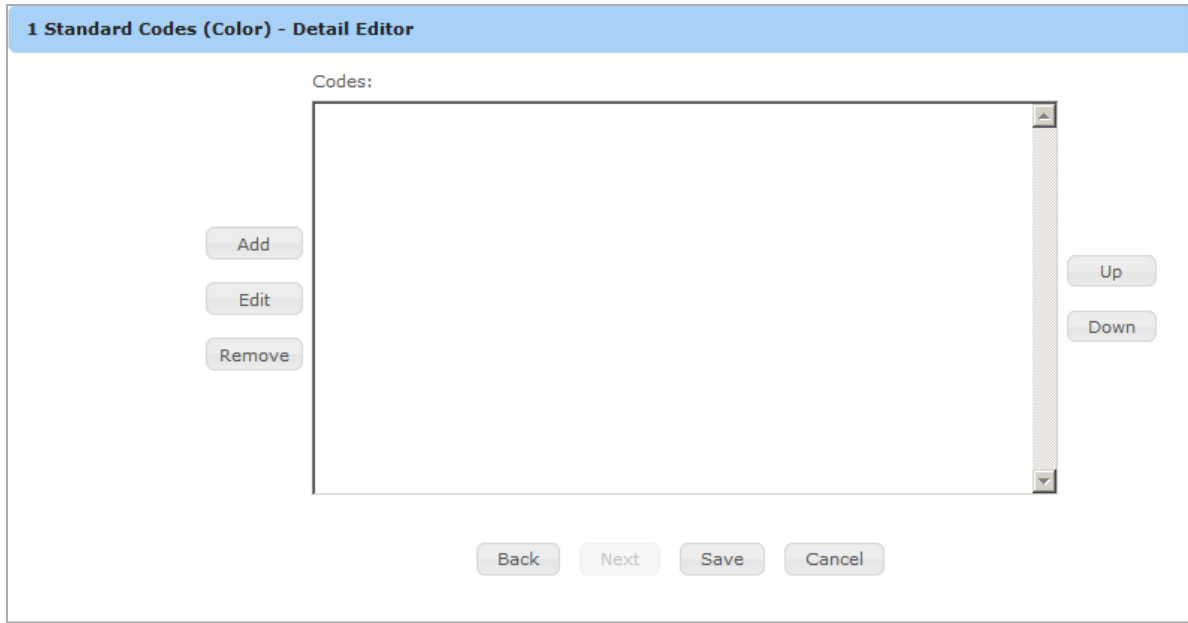


3. To delete a Code Set, select it, open the Action dropdown menu and select **Delete**. Skip the rest of the steps in this list of instructions.
4. To add a Code Set, open the **Action** dropdown and select **New**. A new Code Set will appear in the **Code Set Editor**. Proceed with the steps to edit a Code Set.
5. To edit a Code Set, select it, open the **Action** dropdown menu and select **Edit**. The Code Set will appear in the **Code Set Editor**.

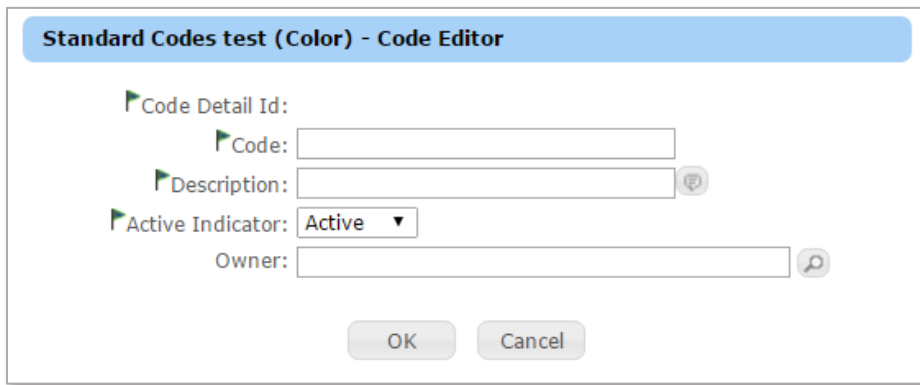
6. Edit the following fields as desired:

- **Name:** The name of the Code Set.
- **Folder:** The folder to hold the Code Set.
- **Description:** (Optional.) A description of the Code Set.
- Select an optional default **Owner** for the Code Set maintenance.
- Identify the **Source** of the initial data.
  - **Detail Editor:** Create a Code Set manually.
  - **From an Existing Set:** Copy Code Set values from an existing Code Set that is in the same folder as the new Code Set.
  - **Import from a file:** Import the Code Set values from a file.

7. Click the **Next** button.



8. Click the **Add** button to enter a code for the new Code Set.



9. Enter the **Code** and the **Description**.

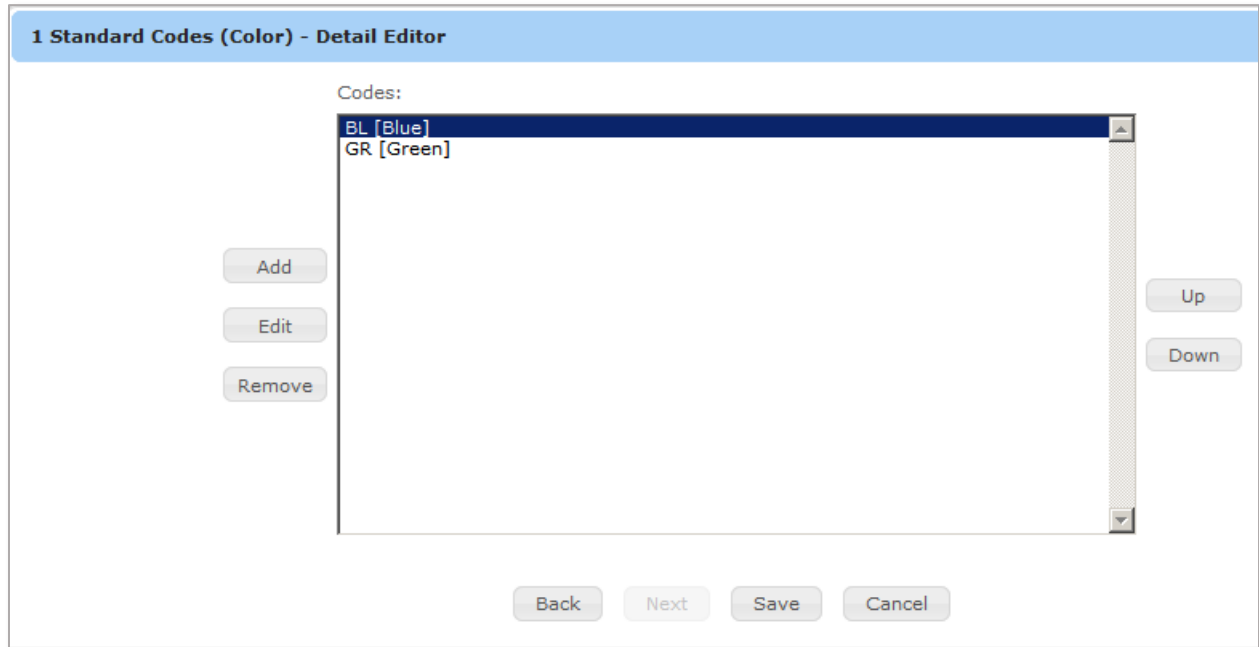
10. If **Include Owner** was selected, an owner can be defined for each specific code.

11. Click the **OK** button.

12. To modify a code, select the code and click the **Edit** button.

13. To remove a code, select the code and click the **Remove** button.

14. To re-sequence a code, select the code and click the **Up** or **Down** button to move it in the list.



15. Click the **Save** button to save changes or the **Cancel** button to cancel changes. If you have just added a new Code Set, it is ready to be assigned to a profile's attributes.

## Templates

Templates may be used in both scheduled and manually triggered activities. For information regarding the use of syndication templates during scheduled activities, see [Job Scheduling](#).

When a template is used to import data to or export data from multiple repositories, those repositories must be linked and you need to specify one repository as the root repository. Templates cannot import or export data from unrelated repositories.

While advanced users can be given permissions to create templates, it is recommended that a system administrator assists users creating advanced templates and helps manage existing templates.

## Import Templates

The Import Template function is used to develop templates that can be used to import data into one or more EnterWorks repositories. The easiest way to set up an import template is to create an export file (see [Export Templates](#)) and then use it to create the import template. It is recommended that the System Administrator verifies and tests a user's import template to make sure it is defined and works correctly. It should be tested with one or two records in your development environment before it is used in your production environment.

While the EnterWorks Classic UI can be used to configure and use import templates, it is recommended that you use the EnterWorks New UI. For details, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Export Templates

Export templates can be used to export data and digital assets from one or more linked repositories, in a variety of formats.

While export templates can be configured and used in the EnterWorks Classic UI, it is recommended that you use the EnterWorks New UI. For more information, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Exchange Templates

Exchange templates are bidirectional. Once you create an exchange template, you can use it to import and export data into one or more linked repositories. Exchange template exports started from the UI output structured XML output. Exchange template exports started by the API return JSON output.

## View Exchange Templates

To view existing exchange templates, from the Feature Bar, open the **Import & Export** tab, and select **Exchange Templates**. The list of existing exchange templates will be displayed.

## Create or Edit Exchange Templates

Exchange templates are viewed, defined, and edited in the **EnterWorks Classic UI**, in the same manner in which Export templates are created, with the exception that the only available output is XML Exchange.

To create or edit an exchange template:

1. Open the Exchange Templates list.

**NOTE:** The **Exchange Template** list can be filtered by entering text into the **Name**, **Root Repository**, and/or **Type** fields at the top of the Exchange Template tab. Once you have entered the text you want to use for a filter, click the **Apply** button.

- a. The **Action** drop-down menu contains:

- i. **New:** Select to create a new template.
- ii. **Edit:** Select to edit the selected templates.
- iii. **Edit Mappings:** Select to define new attribute mappings or to edit existing mappings.
- iv. **Delete:** Select to delete the selected template.

- b. The **Utilities** drop-down menu:

This option provides the ability to **Import** an existing template or to **Export** an existing template, if required. These options are used to move import or export templates from one environment to another, such as, exporting a template from the Development environment and importing it into the QA environment.

2. Select the **New** option under the **Action** drop-down list.
3. The **Exchange Template Editor** screen will be displayed. Edit the options as desired.
4. **Name:** (Required; Multi-language) A name for the Exchange Template.
5. **Description:** (Optional; Multi-language) A description for the template.
6. **Output Type** will be preselected as **XML Exchange**.
7. **Validation Level:** (Optional) Select the minimum validation level that records must meet to be included.
8. **Root Repository:** Select the root repository.
9. Use the **Advanced Call Out Function** and **Advanced Call Out Class Path** for export file transformations (optional).



10. **Available Link Relationships:** Click the **Edit** button to select linked repositories that have attributes you want to include in the template. The **Edit Included Links of Template** editor will appear.
11. Use the + and – buttons to the left of a repository’s name to show or hide that repository’s child repositories. Select the repositories you would like to include. In order to select a child repository, its parent repository must be selected.



12. When you select a repository, the Edit Included Links of Template editor will appear. Another way to open the Edit Included Links of Template editor is to click the **Edit Details** button to the right of the repository’s name. The editor is used to identify all the attributes you want included in the export. The attributes will be exported as key value pairs. If you want to include category attributes, edit the options:
  - **Include All Category Attributes:** All category attributes will be included.
  - **Category:** You can enter the taxonomy value of a node or you can use the double arrow dropdown to the right of the field to open the taxonomy in tree form and

select the node you want to open. When you select a node, that node's attributes will be listed in the **Available Attribute** list.

- **Name Filter:** If you enter text into this field, the Available Attribute list will be filtered to show only attributes whose name begins with the text.
  - **Available Attributes and Selected Attributes:** To include attributes in the template, select attributes from the **Available Attributes** list and use the right-arrow buttons to move them to the **Selected Attributes** list. To exclude attributes from the template, select them in the **Selected Attributes** list and use the left-arrow button to move them to the **Available Attributes** list.
13. When you are finished selecting attributes to include in the template, click the **Save** button to go back to the **Edit Included Links of Template** links.
  14. When you are finished adding all the attributes from the Included Links that you want to include in the template, click **Return** to go back to the **Exchange Template Editor**.
  15. **Output File Alias Format Type:** This will be preselected as **None**. Leave it as it is.
  16. **Output File Layout:** Click the **Edit** button to arrange the attributes in the output file.
  17. The Manage Output File Layout editor will appear. The attributes included in the template will be listed. You can use the **Action** dropdown list to modify the order of the attributes and their column aliases.
    - **Set Alias:** Select an attribute then select **Set Alias** from the Action dropdown list. An editor will open that allows you to edit the column header for the selected attribute.
    - **Set Order:** To change the order in which attributes are arranged in the output file, select **Set Order** from the **Action** dropdown list. An editor will appear that will let you select attributes and use the **Move Up** and **Move Down** buttons to change their order in the list. When you are finished arranging the attributes, click the **OK** button to return to the Manage Output File Layout editor.
  18. When you are finished managing the output file layout, click the **Return** button to return to the Exchange Template editor.
  19. Click the **Save** button to save your changes and return to the **Exchange Template** list.

## Syndication Templates

Syndication templates are useful when you need to define a specific format for the output file, or you want to transform or validate data before it is exported. The transformation tools allow

to apply Advanced callout functions and to use VTL formulas to apply conditional logic and calculate column values based on values from one or more attributes.

Templates may be used in both scheduled and manually triggered activities. For information regarding the use of syndication templates during scheduled activities, see [Job Scheduling](#). It is recommended that a system administrator assists users in managing syndication templates.

## View Syndication Templates

To view existing syndication templates:

1. From the Feature Bar, open the **Import & Export** tab, and select **Syndication Templates**.
2. A list of the existing Syndication Template folders is displayed across the center screen. Click a folder to open it and display its existing Syndication Templates.

The screenshot shows a web application interface for managing export templates. At the top, there are tabs for 'Home', 'Import Templates', and 'Export Templates'. Below the tabs, there are several buttons and dropdown menus, including 'Action', 'Utilities', and 'Setup Job'. The main content area is titled 'Templates' and contains a table with the following data:

<input type="checkbox"/>	#	Name	Root Repository	Type
<input type="checkbox"/>	1	3187 Test Export	Fender - SKU	Excel 2003
<input type="checkbox"/>	2	Material_Query_For_Initial_Lo	Staging - Material	Excel 2007
<input type="checkbox"/>	3	Product Taxonomy Attributes	Staging - Material	XML
<input type="checkbox"/>	4	Product Taxonomy Attributes	Production - MATERIAL Produc	XML
<input type="checkbox"/>	5	-test export	Bike - SKU	Excel 2007

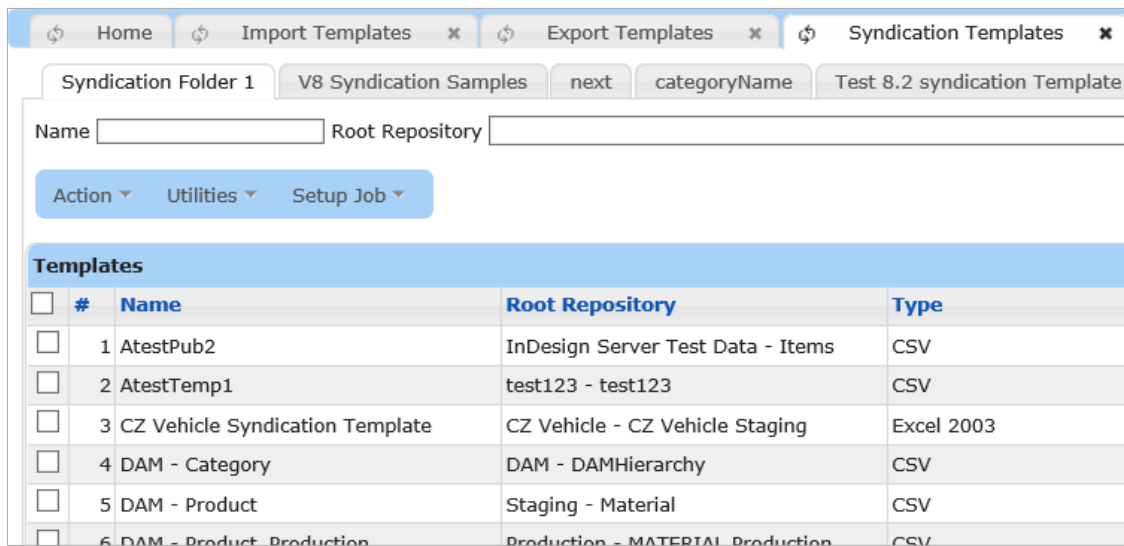
*Open Folders to Show Export Templates*

## Create or Edit Syndication Templates

Syndication templates are viewed, defined, and edited in the **EnterWorks Classic UI**, in the same manner in which Export templates are created, with the exception that the only available output formats are CSV, XLS, and XLSX.

To create or edit a syndication template:

3. Open the Syndication Templates list.



<input type="checkbox"/>	#	Name	Root Repository	Type
<input type="checkbox"/>	1	AtestPub2	InDesign Server Test Data - Items	CSV
<input type="checkbox"/>	2	AtestTemp1	test123 - test123	CSV
<input type="checkbox"/>	3	CZ Vehicle Syndication Template	CZ Vehicle - CZ Vehicle Staging	Excel 2003
<input type="checkbox"/>	4	DAM - Category	DAM - DAMHierarchy	CSV
<input type="checkbox"/>	5	DAM - Product	Staging - Material	CSV
<input type="checkbox"/>	6	DAM - Product Production	Production - MATERIAL Production	CSV

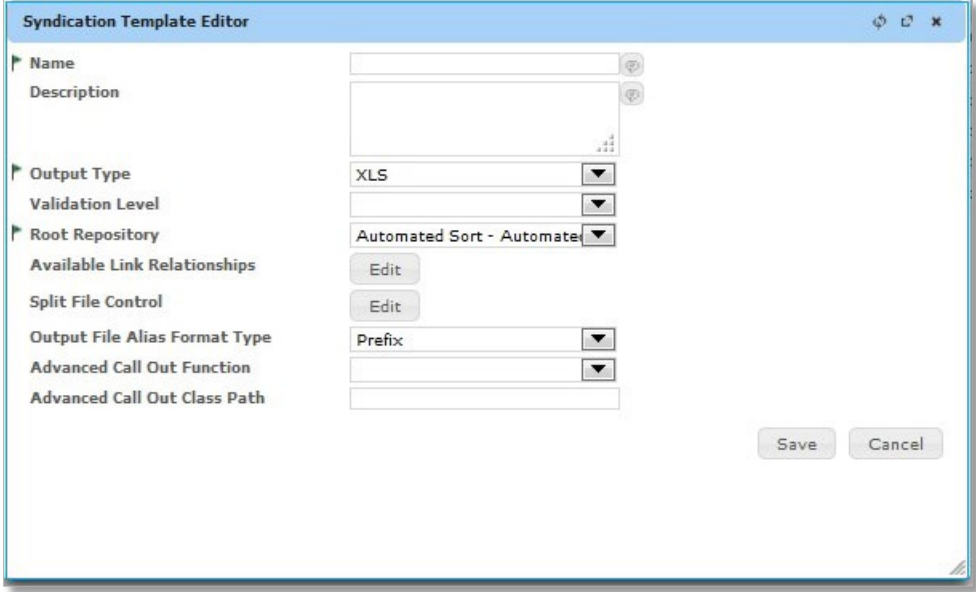
List of Syndication Templates

**NOTE:** The **Syndication Template** list can be filtered by entering text into the **Name**, **Root Repository**, and/or **Type** fields at the top of the Exchange Template tab. Once you have entered the text you want to use for a filter, click the **Apply** button.

- a. The **Action** drop-down menu contains:
  - i. **New:** Select to create a new template.
  - ii. **Edit:** Select to edit the selected templates.
  - iii. **Edit Mappings:** Select to define new attribute mappings or to edit existing mappings.
  - iv. **Delete:** Select to delete the selected template.
- b. The **Utilities** drop-down menu:

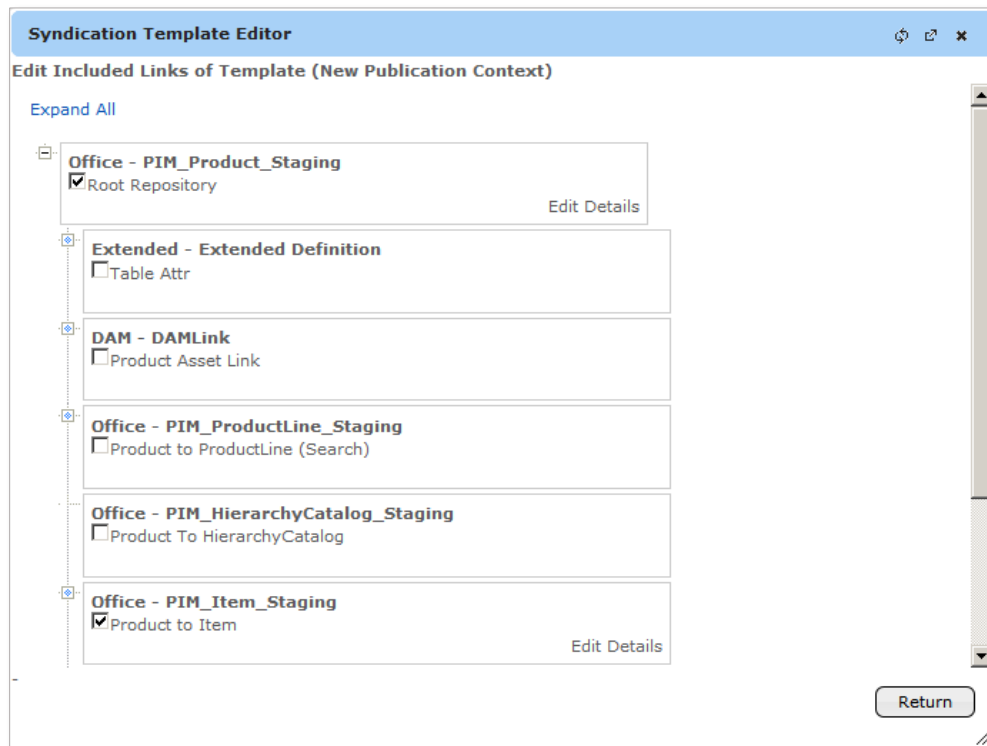
This option provides the ability to **Import** an existing template or to **Export** an existing template, if required. These options are used to move import or export templates from one environment to another, such as, exporting a template from the Development environment and importing it into the QA environment.

4. Select the **New** option under the **Action** drop-down list.
5. The **Syndication Template Editor** screen is then shown.



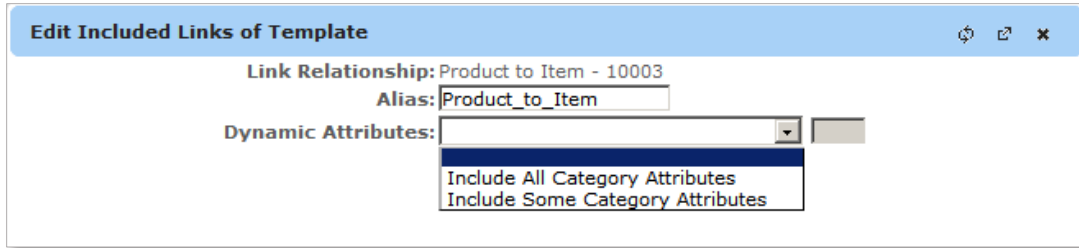
6. **Name:** (Required; Multi-language) A name for the Syndication Template.
7. **Description:** (Optional; Multi-language) A description for the template.
8. Select the **Output Type** from the drop-down list. The options are:
  - a. XLS
  - b. XLSX
  - c. CSV (common separated values)
9. **Validation Level:** (Optional) Select the minimum validation level that records must meet to be included.
10. **Root Repository:** Select the root repository from the drop-down list.
11. **Available Link Relationships:** Click the **Edit** button to select linked repositories that have attributes you want to include in the template. The **Edit Included Links of Template** editor will appear.

12. Use the + and – buttons to the left of a repository’s name to show or hide that repository’s child repositories. Select the repositories you would like to include. In order to select a child repository, its parent repository must be selected.



13. Click the **Edit Details** button to optionally identify that all or some of the dynamic (category) attributes should be appended to end of the export file in key value pair columnar format.

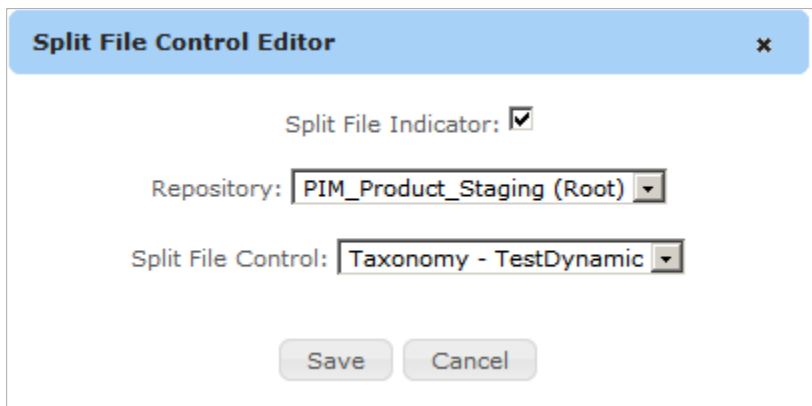
- **Include All Category Attributes:** All category attributes will be included.
- **Include Some Category Attributes:** You can how many attributes to include.



14. Click the **Save** button (above) to go back to the **Syndication Template Editor** links:

15. Click **Return** to go back to the **Syndication Template Editor**:

16. EnterWorks can create an output file for each Taxonomy node rather than putting all the content in one output file. Each file will contain the specified global attributes plus the node's category attributes. To do so, click the **Edit** button next to the **Split File Control** label. An editor window will appear. Select **Split File Indicator**, the repository, and the Taxonomy control attribute. These values may already be filled in for you.



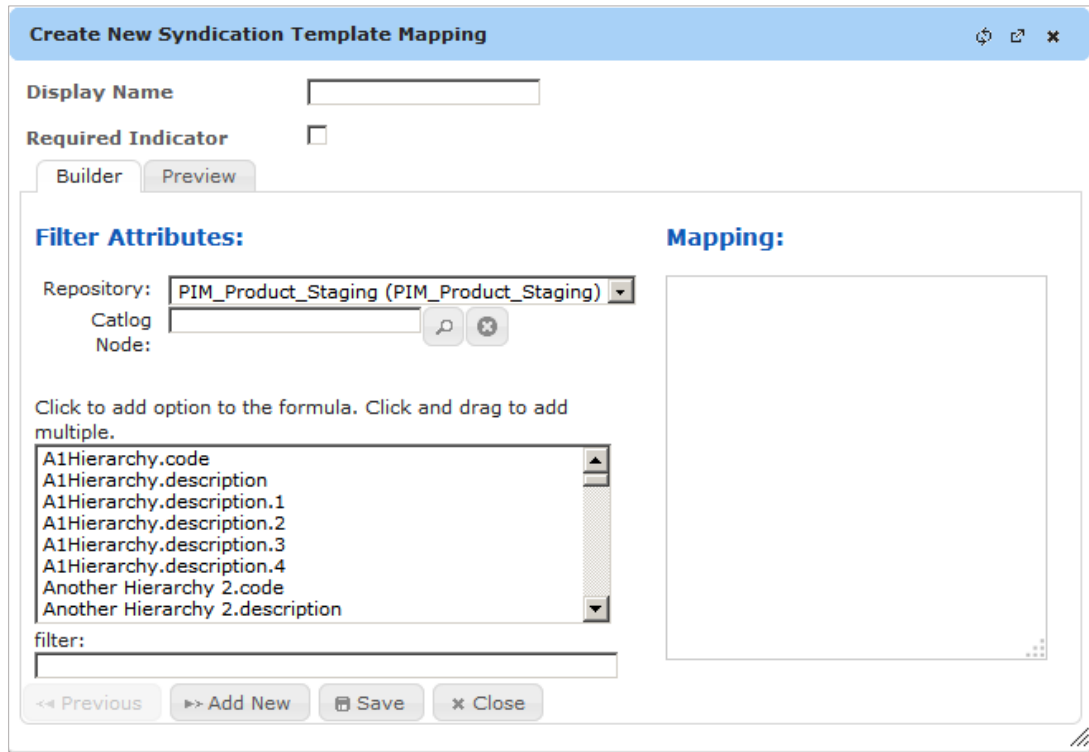
17. Click the **Save** button to go back to the **Syndication Template Editor**.

18. Use the **Advanced Call Out Function** and **Advanced Call Out Class Path** for export file transformations (optional).

19. Click the **Save** button to save your changes, then close the window to go back to the **Syndication Template** list.

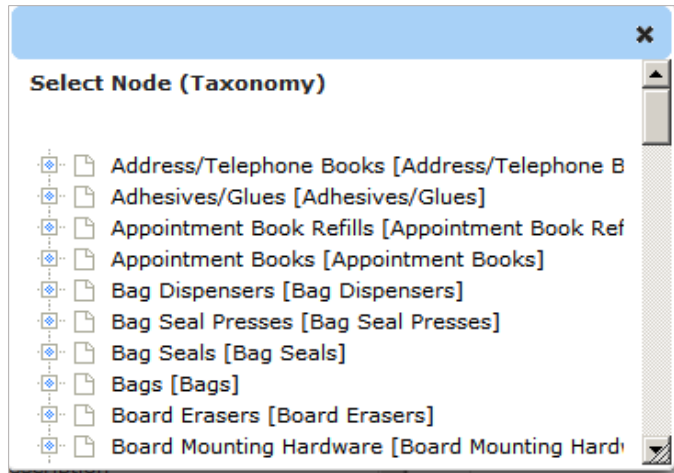
20. Select the Syndication Template that was just created.

21. Select the **Edit Mappings** option from the **Action** drop-down list to define (or modify) the export file and identify where the data for the export is coming from in EnterWorks. The Syndication Template Mapping Editor will appear.
22. Select the **New** option from the **Action** drop-down list to define a new column mapping.
23. The **Create New Syndication Template Mapping** screen is shown.



24. Enter the export column name in the **Display name** field.
25. Check the **Required Indicator** checkbox, if applicable. This indicates that the attribute must have a value to output to the output file. If a value is not present, the record will be exported to the error output file. This setting is useful if an attribute is required by an outside syndication target but it is not required internally to EnterWorks.
26. Select the **Repository** that contains the field to export.
27. Select the **Catalog Node** that contains dynamic attributes, if applicable. Click the magnifying glass to show a list of nodes:





28. The list can be filtered by using the **Filter** field below the attribute list.

29. The Mapping editor allows you to build the VTL (Velocity Template Language) formula that will be used to create the attribute's output. Select the source attribute (field) that will populate the target column. The field will be displayed in the Mapping editor.

For more information about VTL, see the *EnterWorks Advanced VTL Mapping Guide* at <https://support.precisely.com/>.

**Note:** To concatenate fields together, choose multiple fields to include in the Mapping window, then add any separation characters, if required. The fields can be reordered within the window.

30. Click the **Preview** tab to see what the mapping will look like. Use the fields on the bottom to enter test data into the fields.

**Edit Syndication Template Mapping**

Display Name:

Required Indicator:

**Rendered Text:**

2233577 - Office Equipment

**Attribute Formula:**

`${PIM_Product_Staging.Product Line Id} - ${PIM_Product_Staging.Product Line}`

**Fill In Values For the Attributes:**

Found Attribute Name	Value
1. PIM_Product_Staging.Product Line	<input type="text" value="Office Equipment"/>
2. PIM_Product_Staging.Product Line Id	<input type="text" value="2233577"/>

31. Click the **Builder** tab and click the **Save** button to go back to the **Syndication Template Editor** screen.

**Syndication Template Editor**

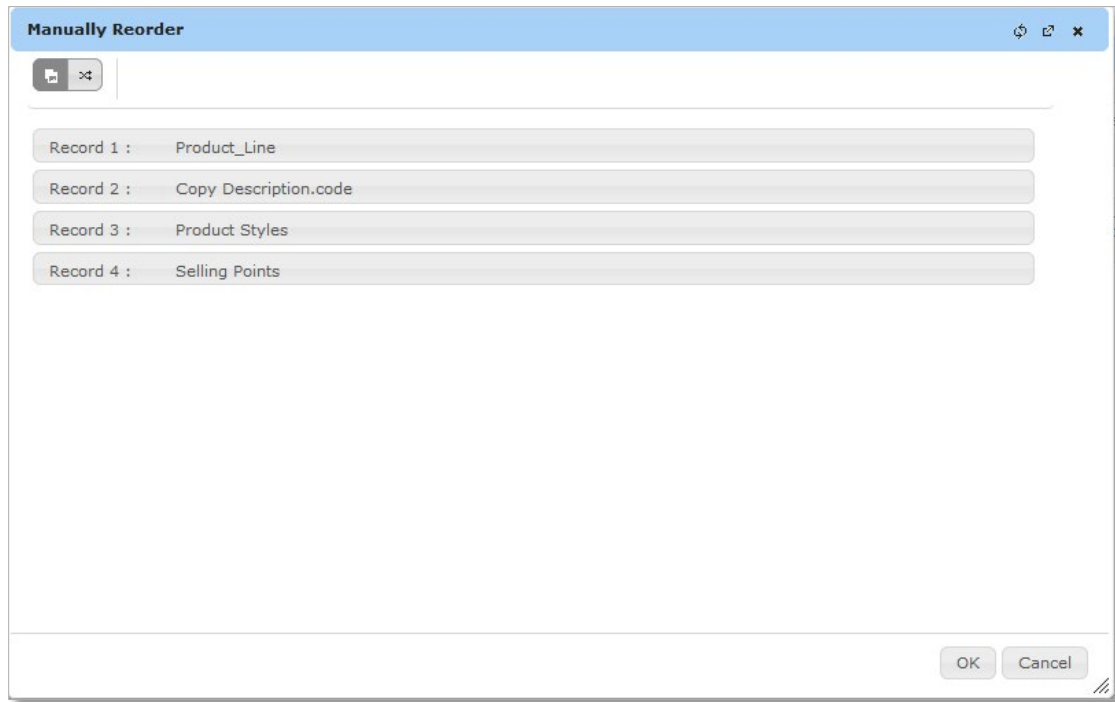
Target Attribute Name:  Mapping:  Required:

**Watsco Syndication Template Attribute Mapping List**

#	Target Attribute	Display Name	Is Data Required	Mapping
<input type="checkbox"/> 1	Product Number	Product Number	Yes	<code>\${Product.Id}</code>
<input type="checkbox"/> 2	Manufacturer	Mfr	Yes	<code>\${Item.manufacturer}</code>
<input type="checkbox"/> 3	Country Of Origin	COO	No	<code>\${Item.Country}</code>
<input type="checkbox"/> 4	Price	Price	No	

Page 1 of 1 50 View 1 - 4 of 4

32. One or more existing mapping items can be edited, copied, or deleted by selecting one or more mapping line items and selecting the desired action from the **Action** drop-down menu.
33. To reorder the fields in the export template, select the **Set Sequence** option from the **Action** drop-down list. A window will appear that lets you reorder the fields.



34. Drag and drop the fields to set the order of the fields in the export file.
35. Click the **Ok** button.

## Publication Templates

For information regarding publication templates, please see the *EnterWorks Enable 9 Publications User Manual* and the Precisely EnterWorks online help at <https://support.precisely.com>.

# Multiple Language Support

## Session Language

During login to the New UI, the user can select the language that will be used in their EnterWorks session by choosing the desired language from the **Language** dropdown list in the Login window. The language chosen is referred to as the “session language”. The list of languages supported are defined in the Language List.

The session language for the Classic UI is always English (EN-US).

While most aspects of the EnterWorks UIs are displayed in the session language, some are not. To display all UI elements in the session language, set your browser to translate the webpages to the desired language.

Currently EnterWorks does not support the following languages:

- Tasawaq language
- Languages that have an apostrophe as a grouping separator, such as de-CH (Switzerland) and de-LI (Liechtenstein).

## Language List

The Language List defines the set of languages EnterWorks will maintain and it indicates which language is the default language.

## Access the Language List

To access the **Language List**:

1. There are two ways to access the Language List:
  - In the **Feature** bar, open the **System** feature and select **Languages**.
  - Or, open the **Quick Links** drop-down list on the top-right of the screen and select **System Configuration**. The **Data Model Management** tab will open. Under the heading **Model**, select **Languages**.
2. The **Language List** screen will appear.

<input type="checkbox"/>	#	Name	Extension	Description	Is Default	Enable
<input type="checkbox"/>	1	English	en	Default Language	Yes	Yes
<input type="checkbox"/>	2	French	fr	French Language		Yes
<input type="checkbox"/>	3	Spanish	es	Spanish Language		Yes
<input type="checkbox"/>	4	German	de	German Language		Yes
<input type="checkbox"/>	5	Danish	da	Danish Language		Yes
<input type="checkbox"/>	6	Dutch	nl	Dutch Language		Yes
<input type="checkbox"/>	7	Estonian	et	Estonian Language		Yes
<input type="checkbox"/>	8	Finnish	fi	Finnish Language		Yes
<input type="checkbox"/>	9	Italian	it	Italian Language		Yes
<input type="checkbox"/>	10	Latvia	lv	Latvia Language		Yes
<input type="checkbox"/>	11	Lithuanian	lt	Lithuanian Language		Yes

## Manage Supported Languages

Depending on system configuration and user permissions, a user can enter language translations for attribute values and system-level objects.

To manage a supported language:

1. [Open the Language List.](#)
2. From the **Language List** screen, click on the **New** option under the **Action** dropdown list to add a new language option.
  - Current Languages can be edited using the **Edit** option.
  - Current Languages can be deleted using the **Delete** option.
  - The Language list can be sequenced using the **Sequence** option.

- Multiple Languages can be enabled or disabled using the **Enable/Disable Languages** option.

NOTE: Only Active Languages can be sequenced. Deactivated languages are displayed first.

3. The **Language Editor** will appear.

The screenshot shows the 'Language Editor' dialog box. It has a title bar with the text 'Language Editor' and standard window controls (refresh, maximize, close). The main content area includes three text input fields: 'Name', 'Extension', and 'Description'. Below these fields is an 'Enable' checkbox. In the center, there are two list boxes: 'Available Groups' on the left and 'Selected Groups' on the right. Between these list boxes are four arrow buttons: a double right arrow (>>), a single right arrow (>), a single left arrow (<), and a double left arrow (<<). The 'Available Groups' list contains the following items: Administrator, Designers, Product Marketing, Product Planning, ReadOnly, ReadOnly with DAM Drop, and zz. At the bottom of the dialog are two buttons: 'Save' and 'Cancel'.

4. Edit the fields as desired:

- **Name:** The name of the language.
- **Extension:** Enter the Language Extension. Use a standard language code like ISO 639-1 or ISO 639-2/T in order to sufficiently differentiate the language for maintenance.
- **Enable** checkbox:

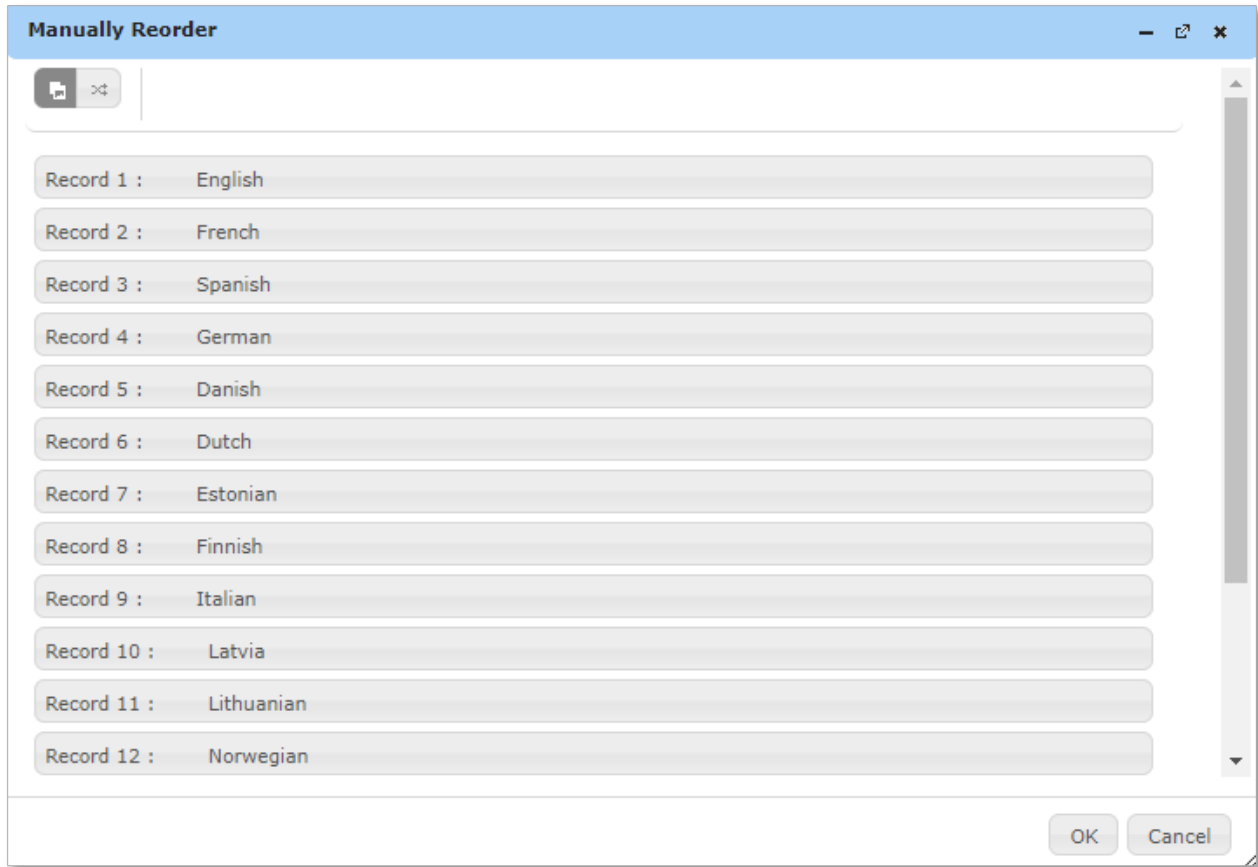
- **Checked:** The language is enabled. It will be included in the list of available languages in screens for editing multi-language fields.
  - **Unchecked:** The language is disabled. It will not be included in the list of available languages in screens for editing multi-language fields.
  - **Groups Lists:** To move a group from one list to the other, select it and click the left or right arrows to move it to the other list.
    - **Available Groups:** User groups in this list do not have the ability to update fields in this language.
    - **Selected Groups:** User groups in this list do have the ability to update fields in this language.
5. When you are finished editing the language, click **Save** to save your changes or click **Cancel** to leave the editor without changing your settings.

## Reorder the List of Supported Languages

NOTE: Only enabled Languages can be sequenced. Disabled languages are shown first.

To reorder the list of supported languages:

1. [Open the Language List](#).
2. Open the **Action** dropdown and select **Sequence**. The **Manually Reorder** pop-up will appear.



3. Just under the words “**Manually Reorder**” is a small toggle. If the left icon is selected, you will be able to drag and drop the languages into the desired order. If the right icon is selected, you will select a language and follow the instructions given to move it to the desired position. Hovering over the toggle icons brings up a description of the two methods of reordering languages.
4. When you are finished reordering the languages, click **Save** to save your changes or click **Cancel** to leave the editor without changing your settings.

### Enable or Disable a Supported Language

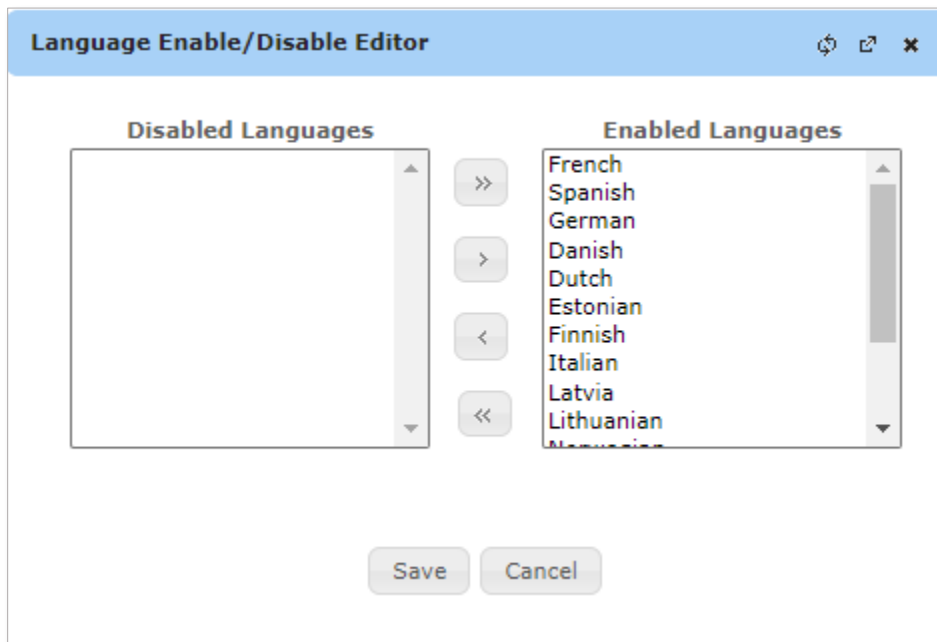
If a language is enabled, it will be included in the list of available languages in screens for editing multi-language fields.

If a language is disabled, it will not be included in the list of available languages in screens for editing multi-language fields.



There are two ways to enable or disable a language. For instructions on enabling or disabling a language through the **Language Editor**, see: [Manage Supported Languages](#). To enable or disable a language through the **Language Enable/Disable Editor**:

1. [Open the Language List](#).
2. Open the **Action** dropdown and select **Enable/Disable Languages**. The **Language Enable/Disable Editor** will appear.



3. **Groups Lists:** To move a group from one list to the other, select it and click the left or right arrows to move it to the other list.
  - **Disabled Languages:** These languages are disabled. They will not be included in the list of available languages in screens for editing multi-language fields.
  - **Enabled Languages:** These languages are enabled. They will be included in the list of available languages in screens for editing multi-language fields.
4. When you are finished enabling and disabling languages, click **Save** to save your changes or click **Cancel** to leave the editor without changing your settings.

## Delete a Supported Language

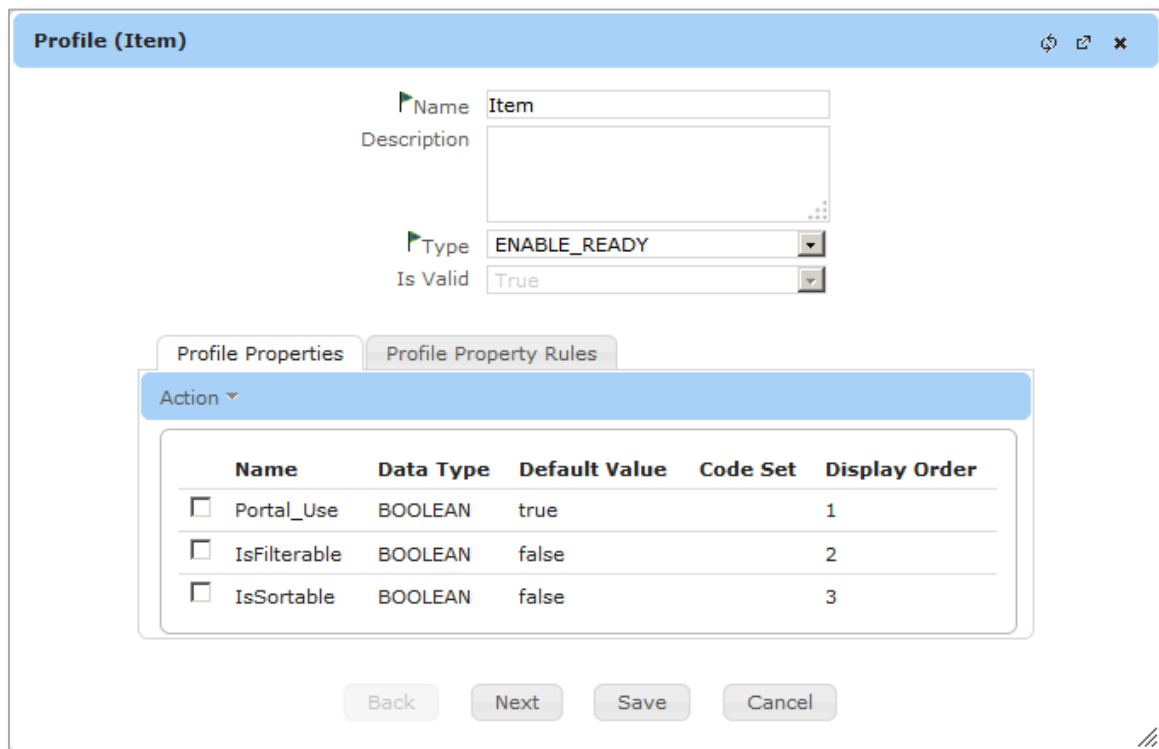
To delete a supported language:

1. [Open the Language List](#).
2. Select the language to be deleted.
3. Open the **Action** dropdown and select **Delete**.

## Defining a Multi-language Attribute

To define an attribute to be multi-language:

1. Select **Profiles** under **Model** from the Feature bar.
2. Select the Profile record that needs (or contains) an attribute that will have multiple languages entered, then in the **Action** dropdown menu, select **Edit**.
3. The Profile Editor will appear. Click the **Next** button to move to the Attribute definitions page.



The screenshot shows the 'Profile (Item)' editor window. The 'Name' field is 'Item' and the 'Type' is 'ENABLE\_READY'. The 'Is Valid' dropdown is set to 'True'. The 'Profile Property Rules' tab is active, and the 'Action' dropdown menu is open, showing a table of attributes.

	Name	Data Type	Default Value	Code Set	Display Order
<input type="checkbox"/>	Portal_Use	BOOLEAN	true		1
<input type="checkbox"/>	IsFilterable	BOOLEAN	false		2
<input type="checkbox"/>	IsSortable	BOOLEAN	false		3


Buttons at the bottom: Back, Next, Save, Cancel.

4. Find and select the Attribute that will have multiple languages, then in the **Action** dropdown menu, select **Edit**.
5. The **Define Attribute Details** editor will appear. Check the **Is Multi Language** checkbox.

The screenshot shows the 'Define Attribute Details (Description)' dialog box. The 'Name' field is set to 'Description'. The 'Data Type' is 'VARCHAR' and 'Data Size' is '50'. The 'Is Multi Language' checkbox is checked and highlighted with a red box. Other checkboxes include 'Is Global' (selected), 'Is Required', 'Is Primary Key', 'Seq Gen Ind', 'Is Repeatable', and 'Is Business Key'. The 'Group' is set to 'Bookkeeping - Summary'. At the bottom, there is an 'OK' button and a 'Cancel' button. Below the dialog box, there is a 'Define Attribute Rules' section with an 'Action' dropdown and a table with columns: Name, Rule, Severity, Level, Condition(s), and Properties.


6. Click the **OK** button to save your change.
7. You will be taken back to the **Define Attributes** editor. To save your changes, you must click the **Save** button. If you do not click the Save button before exiting this editor, your change to the Is Multi Language setting will not be saved.

## Editing Multi-language Attribute Values

If a field has been set up for multiple languages, it will have a multi-language  icon to the right of the field as shown below.

**Item Description**  
PEN,HIGHLIGHTER COMBO,BK

To edit the field translations:

1. Click the multi-language icon  to open the **Multi-Language Editor**. If the user is a Systems Administrator, all enabled languages will be modifiable, but a non-administrator will only be able to edit/modify the languages they have permission to edit.

**Multi-Language Editor**

**English (Default)**  
PEN,HIGHLIGHTER COMBO,BK

**Italian**

**Spanish**

**Chinese**

**German**

Save Cancel

The languages shown in the Multi-Language Editor are those that have been enabled. See [Enable or Disable a Supported Language](#).

If the user is a Systems Administrator, all enabled languages will be modifiable, but a non-administrator will only be able to edit/modify the languages they have permission to edit.

2. Enter the text you would like to appear for each language.
3. If EnterWorks has been configured to use Google Translate, the auto-translate icon will appear next to each language's content box. If you click the auto-translate icon for a language, the language's content box will be set to the default language's text that has been translated into the selected language.
4. Click **Save** to save your changes and exit or click **Cancel** to exit without saving your changes.

## Translating Data Model Entities

EnterWorks data model objects have translatable multi-language fields, such as name and description. For adding translating entire data model objects and its attributes use the Export and Import multi-languages functions described in this section.

### Export Data Model Objects for Translation

To access the Multi-Language function:

1. Select the feature **System** in the feature bar then select **Export Multi-Languages** from the list of functions. The **Multi-Language Export** dialog will appear.

**Export Multi-Languages**

## Multi-Language Export

Select Languages: English X French X Finnish X Chinese X

**Profile** Click names to transfer them to selected list.

Start typing to filter the Profile names

All Profiles <input type="button" value="Select All"/>	Selected Profiles <input type="button" value="Deselect All"/>
Product ⇒	Brand
Promotion ⇒	Price
PublicationMergeAttribute ⇒	
PublicationMergeNode ⇒	
PublicationMergeSpec ⇒	

**Code Set** Click on names to transfer them to selected list.

Start typing to filter the Code Set names

All Code Sets <input type="button" value="Select All"/>	Selected Code Sets <input type="button" value="Deselect All"/>
AudioFormat ⇒	Amazon Categorization
AuxInputs ⇒	Battery Packaging
AuxSwitchings ⇒	
Batteries ⇒	
Battery Types ⇒	

**Other Sub-systems** Click on names to transfer them to selected list.

Start typing to filter the names

All Sub-systems <input type="button" value="Select All"/>	Selected Sub-systems <input type="button" value="Deselect All"/>
Messages ⇒	Attribute Groups
Repository Folders ⇒	Repositories
Template Folders ⇒	
Templates ⇒	
User Preferences ⇒	

2. The Export Multi-Languages function has four sections

- **Select Languages:** This section allows you select which language(s) are available for translation. Only active languages are available for translation.
- **Profile:** Exports the names and descriptions of the selected Profiles, as well as all their attributes' name and description fields.
- **Code Set:** Exports the names and descriptions of the selected Code Sets, along with all the codes and code description fields.
- **Selecting Other Sub-Systems:** The Other Sub-systems section lists the rest of the data model components that can be translated, which include:
  - Attribute Groups
  - Attribute Tabs
  - Code Set Folders
  - Data Sources
  - File Definitions
  - Groups
  - Import Template Folders
  - Languages
  - Repositories
  - Repository Folders
  - Templates
  - User Preferences

(Messages are all the User Interface labels and messages. For instructions on translating Messages, see [Translating EnterWorks Labels and Messages.](#))

3. Once you are finished selecting language(s) and data model object(s), click the **Export** button at the bottom right of the page. This will generate a link that downloads a .csv file for the selected data model object(s).

A successful export will generate a link similar to the picture below. It will have a green background.

Exported to: [Model\\_Translation\\_1468842665158.zip](#)

An unsuccessful export will generate a link similar to the picture below and an error message. It will have a yellow background.

Exported to: [Model\\_Translation\\_1468842907943.zip](#)

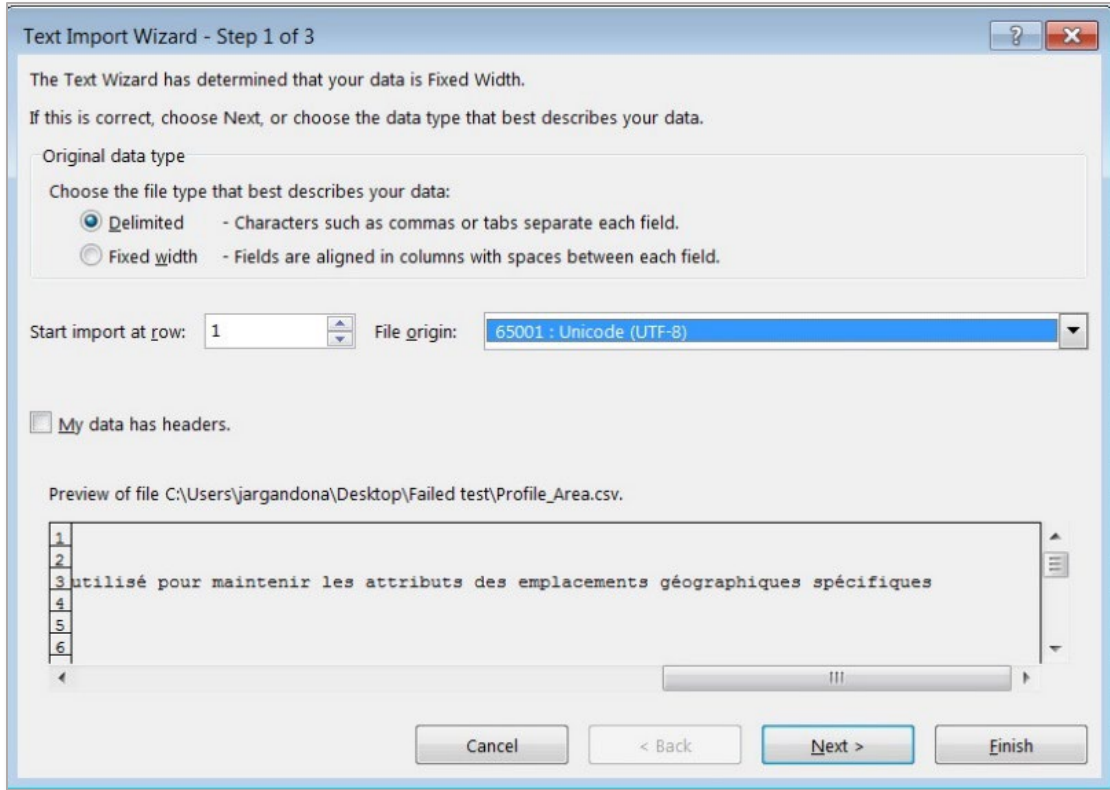
These objects could not be exported:  
Custom '68 Hand-Wound Single-Coil Strat(Code Set Details)  
Custom '68 Hand-wound Single-Coil Strat(Code Set Details)

4. Click the link to download the .csv zip file.

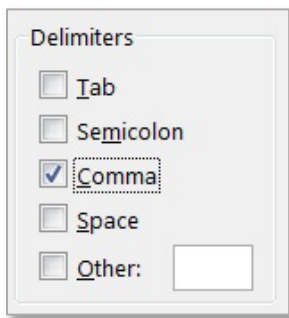
To add translations to these files:

1. Extract file(s) to your computer.
2. Open a new Excel file.
3. Select the **Data** tab.
4. In the **Get External Data** section of the data tab, select **From Text**.
5. In the file explorer that appears navigate to the extracted data model file(s).
6. In the **Text Import Wizard** that appears select the **Delimited** radio button and in the file origin drop down list select **Unicode (UTF-8)**. An example is shown below.





7. Click **Next** to move to the next page.
8. In the **Delimiter** section verify that only the **Comma** checkbox is selected.



9. Click **Finish**, place the data in the upper left corner cell (A:1), and click **Ok**.  
The image below is the general format for each exported data model object.

	A	B	C	D	E	F	G	H
1	Entity Type	Entity Name	Field Name	UUID	English	German		
2	Bprofile	Area	name	e9c83e08-49bc-4d38-ab6e-48d841c	Area			
3	Bprofile	Area	description	e9c83e08-49bc-4d38-ab6e-48d841c	Area description			
4	BformatAttr	Discontinued Date	name	9b138293-afb6-4b46-8658-c3b6f1d	Discontinued Date			
5	BformatAttr	Discontinued Date	description	9b138293-afb6-4b46-8658-c3b6f1d	89faa			
6	BformatAttr	Consumer Sellable	name	fef15be6-d416-4c32-9fc7-bf738d95	Consumer Sellable			
7	BformatAttr	Consumer Sellable	description	fef15be6-d416-4c32-9fc7-bf738d95	ebe3			
8	BformatAttr	Area	name	1f81b734-d55a-4610-83be-60d3ee4	Area			
9	BformatAttr	Area	description	1f81b734-d55a-4610-83be-60d3ee4	6a344			
10	BformatAttr	WorkFlowPost	name	b1b45969-4c1a-4b29-bab1-254d4c1	WorkFlowPost			
11	BformatAttr	WorkFlowPost	description	b1b45969-4c1a-4b29-bab1-254d4c1	25014c			
12	BformatAttr	WorkFlowStatus	name	01718272-40bb-4077-bdc0-e3d1d0	WorkFlowStatus			

10. From this point translate the default language column values into the language column(s) selected for translation. Pictured above is an example of German translations for a profile named Area and its attributes.

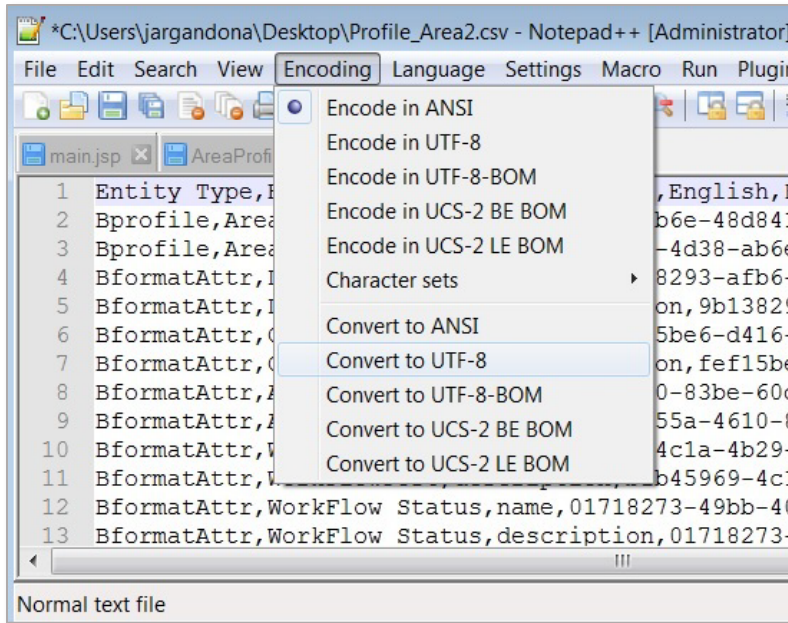
**Note:** The default language will always appear in its own column and display its values.

11. For languages that consist primarily of special characters, (such as Chinese, Korean and Japanese):

- a. Save file as Unicode text.
- b. Close Excel.
- c. Open the Unicode file with notepad++.
- d. Select all text and click Ctrl-F.
- e. Select the **Replace** tab.
- f. In the search mode section, select the **Extended** radio button.
- g. In the section **Find what** enter “\t” and in “Replace with” section enter “,” then click **Replace All**.

12. Save the file with the .csv extension.

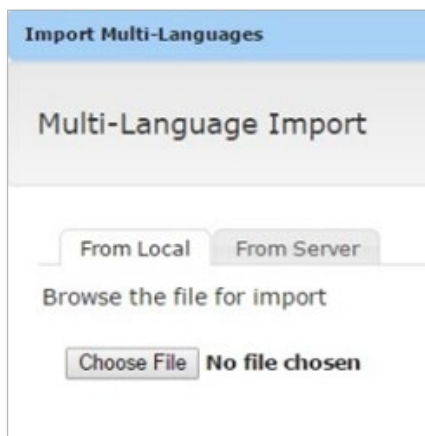
NOTE: The file must be saved in UTF-8 format. To verify, open the file in Notepad++ and check the encoding. Convert file to UTF-8 if it is not already and save the file again. An example is shown below.



## Import Translated Data Model Objects

To import translated data model object .csv files:

1. Select the feature **System** in the feature bar then select **Import Multi-Languages** from the list of functions. The **Multi-Language Import** dialog will appear.



2. To import a data model object .csv file from your local machine, select the **From Local** tab. To import the file from the application server, select the **From Server** tab.

3. Choose the desired .csv file, then click **Import**.
4. A new job is created to perform the import. Open the **Job Monitor** to check the status of the job.
5. When the import job has completed, you can view the job's log file. An example of a successful import log file is shown below.

```

multiLangImport_104022016_7_19_15_30_12-1 - Notepad
File Edit Format View Help
Import file D:\Enterworks\shared\export\Profile_Brand en.csv - Tue Jul 19 15:30:12 EDT 2016.
Import Bprofile with uuid: 5c6457de-7b64-483a-9d96-99ade73633f1 Brand en
Import BformatAttr with uuid: 9f821fc7-fe1c-4120-a07d-616948b03e7b WorkFlowPost
Import BformatAttr with uuid: 2c272b1b-f3f7-42cd-a9b0-1ea97af5dd3e WorkFlow Status
Import BformatAttr with uuid: 69585674-6c0d-4b1b-9aae-98d3272519d7 WorkFlow Comment
Import BformatAttr with uuid: 7e0d79ad-aad4-45bf-8b34-019d932d27f9 Website
Import BformatAttr with uuid: 92d9fdea-cb74-4fa5-bda9-69d7f0a30487 Status
Import BformatAttr with uuid: 38004ffe-d34e-4ff9-abe0-2a9d5c9d2b59 Short Description
Import BformatAttr with uuid: 71aba347-9106-49d8-a93e-5cd755cb7d24 Sequence Number
Import BformatAttr with uuid: 24240856-50ef-4c9f-9bdc-f9febba76f1e Long Description
Import BformatAttr with uuid: 09fddfb2-1b22-4d04-bda4-5bb71ba37f88 Logo
Import BformatAttr with uuid: 0a1fd08c-1750-40f0-84aa-03df10d5cc4e Founded
Import BformatAttr with uuid: c9eac979-547d-4288-bb18-a02bf9c2b546 Feature Bullets
Import BformatAttr with uuid: 013405ae-dd23-47a1-9039-d50604e817f0 Brand name
Import BformatAttr with uuid: 8022c4cb-a7e0-4a30-80de-dc873ba88af5 Brand ID
Import BformatAttr with uuid: 8ce640e4-c9c8-44a9-93b5-296dd15b379e web_brand_id
Finish import multi-language file D:\Enterworks\shared\export\Profile_Brand en.csv - Tue Jul 19 15:30:13 EDT 2016

```

6. Log file import failures may occur if the .csv file is formatted incorrectly. If the job's log file indicates there was an error, try repeating the steps formatting the .csv file.
7. If the import job was successful, clear the data cache and log out.
8. Log back in, selecting the language of the translated data model object.
9. Navigate to the new translated data model object to verify the results of your translation

For complete integration, change the browser specified language, as specified in [Browser Specific Language Settings](#).

## Translating EnterWorks Labels and Messages

EnterWorks provides the ability to translate all labels and messages into the active languages. Labels are the field names, headers, page names, and button labels. Messages include confirmation and error messages displayed as a consequence of user actions.

To translate all the labels and messages that make up the EnterWorks interface:

1. Select the feature **System** in the feature bar then select **Export Multi-Languages** from the list of functions. The **Multi-Language Export** dialog will appear.

**Export Multi-Languages**

## Multi-Language Export

Select Languages: English X French X Finnish X Chinese X

**Profile** Click names to transfer them to selected list.

Start typing to filter the Profile names

**All Profiles**  **Selected Profiles**

<ul style="list-style-type: none"> <li>Product =&gt;</li> <li>Promotion =&gt;</li> <li>PublicationMergeAttribute =&gt;</li> <li>PublicationMergeNode =&gt;</li> <li>PublicationMergeSpec =&gt;</li> </ul>	<ul style="list-style-type: none"> <li>Brand</li> <li>Price</li> </ul>
---	--

**Code Set** Click on names to transfer them to selected list.

Start typing to filter the Code Set names

**All Code Sets**  **Selected Code Sets**

<ul style="list-style-type: none"> <li>AudioFormat =&gt;</li> <li>AuxInputs =&gt;</li> <li>AuxSwitchings =&gt;</li> <li>Batteries =&gt;</li> <li>Battery Types =&gt;</li> </ul>	<ul style="list-style-type: none"> <li>Amazon Categorization</li> <li>Battery Packaging</li> </ul>
---	--

**Other Sub-systems** Click on names to transfer them to selected list.

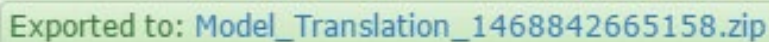
Start typing to filter the names

**All Sub-systems**  **Selected Sub-systems**

<ul style="list-style-type: none"> <li>Messages =&gt;</li> <li>Repository Folders =&gt;</li> <li>Template Folders =&gt;</li> <li>Templates =&gt;</li> <li>User Preferences =&gt;</li> </ul>	<ul style="list-style-type: none"> <li>Attribute Groups</li> <li>Repositories</li> </ul>
---	--

2. Select the desired languages for translation.
3. In the **Other Sub-systems** section, select **Messages**.
4. Click the **Export** button at the bottom right of the page. This will generate a link that downloads a .csv file for the selected data model object(s).

A successful export will generate a link similar to the picture below. It will have a green background.

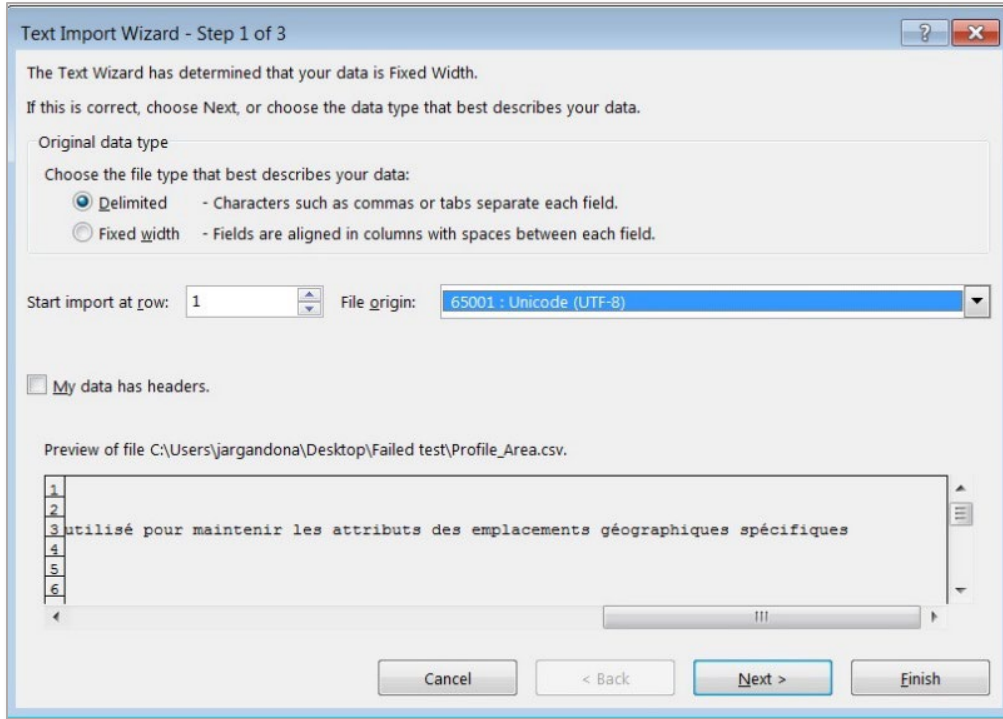
A screenshot of a green rectangular button with a thin border. The text inside the button reads "Exported to: Model\_Translation\_1468842665158.zip" in a blue, sans-serif font.

An unsuccessful export will generate a link similar to the picture below and an error message. It will have a yellow background.

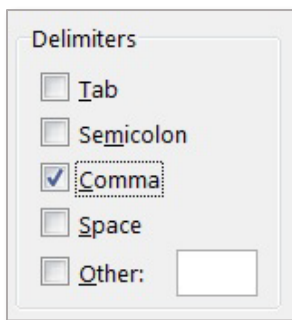
A screenshot of a yellow rectangular button with a thin border. The text inside the button reads "Exported to: Model\_Translation\_1468842907943.zip" in a blue, sans-serif font.

These objects could not be exported:  
Custom '68 Hand-Wound Single-Coil Strat(Code Set Details)  
Custom '68 Hand-wound Single-Coil Strat(Code Set Details)

5. Click the link to download the .csv zip file.
6. Open a new Excel file.
7. Select the Data tab.
8. In the **Get External Data** section of the data tab, select **From Text**.
9. Select the extracted message file.
10. In the **Text Import Wizard** that appears select the **Delimited** radio button and in the file origin drop down list select **Unicode (UTF-8)**. An example is shown below.



11. Click **Next** and in the following window, in the **Delimiter** section, verify that only the **Comma** checkbox is selected. An example is shown below.



12. Click **Finish**, place the data in cell A:1, and click **Ok**.

13. Add your translations for the language you have chosen to export.



	A	B	C
1	Message Key	English	German
2		#DO NOT MODIFY OR REMOVE THIS PORTION OF THE SOFTWARE. ANY MODIFICATION OR REMOVAL OF THIS PORTION OF THE SOFTWARE WILL	
3		#####	
4		## ePIM Date/Time Setting	
5		#####	
6		# ePIM's input date and time settings	
7	epim.input.date.format	MM/dd/yyyy	MM/dd/yyyy
8	epim.input.timestamp.format	MM/dd/yyyy HH:mm:ss	MM/dd/yyyy HH:mm:ss
9	epim.input.time.list	00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00	00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00 12:00
10		#####	
11		## ePIM Data Source Relation Operators	
12		#####	
13	epim.dataSourceRelationOperatorList	<>	<>
14		#####	
15		## ePIM Master Repository Types	
16		#####	
17		## REPOSITORY_TYPE_TAXONOMY_PRODUCTS_REPOSITORY_VIEW	
18	epim.master.repo.type.taxonomy.product.repository.view	Hierarchy Catalog Repository View	Hierarchie Katalog-Repository-Ansicht
19		#####	
20		## ePIM Link Sub Type Codes	
21		#####	
22		## LINK_RELATIONSHIP_SUB_TYPE_CODE_EXTENDED	
23		Extended	Verlangert
24	epim.link.sub.type.code.extended	Extended	Verlangert
25		# LINK_RELATIONSHIP_SUB_TYPE_CODE_ACCESSORY	
26	epim.link.sub.type.code.taxonomyView	Hierarchy View	Hierarchieansicht
27	epim.link.sub.type.code.accessory	Accessory	Zubehorteil
28	epim.link.sub.type.code.damlink	Digital Assets	Digitale Assets
29	epim.link.sub.type.code.manytomany	Many to Many (Parent-Child-Parent)	Viele zu viele (Parent-Child-Parent)

14. Save the file as Unicode (UTF-8) text. If the file name is changed, the new file name must include "Messages" in the file name.

15. For languages that consist primarily of special characters, (such as Chinese, Korean, and Japanese):

- Open the file in Notepad++.
- All the characters appear normally. If not then stop and start from the beginning
- Select all text and click Ctrl-F.
- Select the **Replace** tab.
- In the search mode section select the **Extended** radio button.
- In the section **Find what** enter "\t" and in **Replace with** section enter text ",", then click **Replace All**.
- In the **Find what** section enter ",'" and in the **Replace with** section enter ""'"," then click **Replace All**.
- Convert file to UTF-8 in the encoding menu section of Notepad++.
- Save file with .csv file extension.

16. Use the "Import Multi-Languages" function to import the translated file.

17. Inspect the Log file for the import job. An example of a successful import log file is shown below.

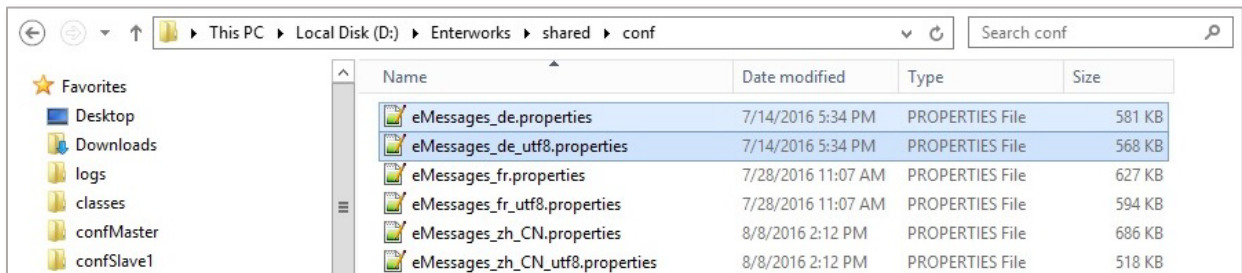


18. Log file import failures may occur if the .csv file is formatted incorrectly. If the job's log file indicates there was an error, try repeating the steps formatting the .csv file.

19. Remotely log into the application server and navigate to where the log file indicates the property files are generated. In this example above the location is on the second line:

```
<install_drive>:/Enterworks/shared/conf/eMessages_de.p  
roperties
```

20. Two files are generated from the import for each language copy both files that pertain to the language(s) imported, an example is shown below.



21. Paste the copied files into every Tomcat and Jboss instance. An example is shown below for a multiple Jboss instances.

```
<install_drive>:\Enterworks\EnableServer\tomcat\webapps\web  
cm\WEB-INF\classes
```

```
<install_drive>:\Enterworks\EnableServer\JbossController\st  
andalone\configuration\conf
```

```
<install_drive>:\Enterworks\EnableServer\JbossWorker<n>\standalone\configuration\conf
```

22. On the EnterWorks application, clear the data cache for the added translations to take effect.
23. For complete integration change the Browser specified language, (see [Browser Specific Language Settings](#)).

## Validation Levels and Rules

When attribute values are entered, EnterWorks automatically conducts implicit data validation, such as checking that the type of data being stored matches the attribute's defined data type, that the value's size falls within the attribute's defined data size, and ensuring that an attribute value is present if the attribute value is required. If more complex data validation is needed, EnterWorks provides the ability to define explicit validation rules and to define validation levels which specify which validation rules apply given a particular use of the data.

EnterWorks does support validation rules for repeatable attributes.

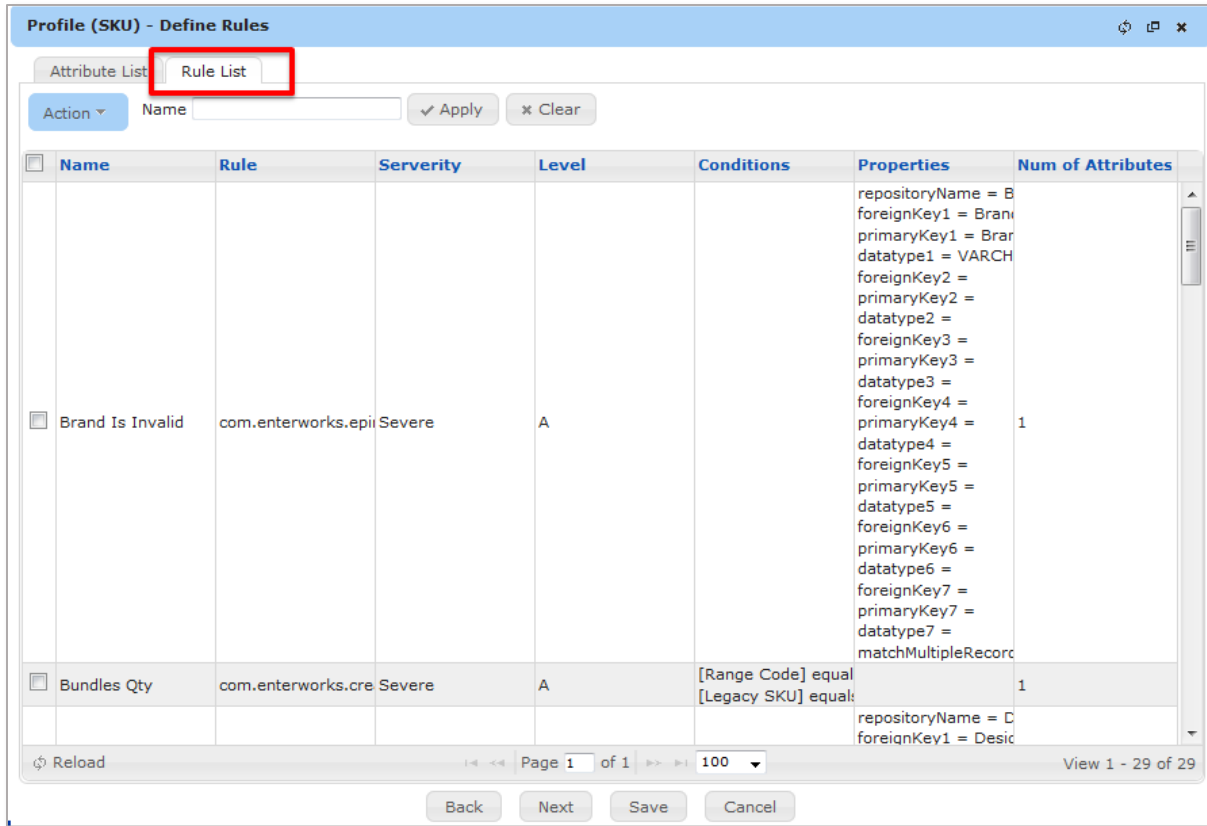
To learn more about validation levels and rules and how they are used, see the Precisely EnterWorks online help at <https://support.precisely.com>.

The following sections give instructions for configuring validation rules and performing data validation.

## Create or Edit a New Validation Rule

To create or edit a validation rule:

1. Log into EnterWorks Classic and navigate to the profile for the repository where you wish to establish the validation rule.
2. The **Profile Editor** will appear. Click **Next** to move to the next page of the profile editor.
3. Click the **Rule List** tab. This will take you to a list of existing validation rules:



4. Create a new rule by selecting **Action > New** or edit an existing rule by selecting it and selecting **Action > Edit**.
5. The **Define Rule Details (New)** window opens. Edit the following fields as desired.

**Define Rule Details (New)**

Name: CZ Multilanguage Test

Type: Simple

Severity: Warning

Level: A1

Rule:

Operator: Equals (=)

Value:

Condition(s):

Add Edit Remove

Any conditions valid  
 All conditions valid

OK Cancel

Define Rule Attributes

Action

<input type="checkbox"/>	Name	Group	Data Type	Attribute Type	Language
<input type="checkbox"/>	Product Name	Summary	VARCHAR(50)	Yes	Select Languages

Configuration Option	Required or Optional	Definition	For Rule Type(s)
Name	Required	This is the text that will appear on the record if it fails the validation rule.	All
Type	Required	Type of validation rule. (See <a href="#">Validation Rule Types.</a> )	All

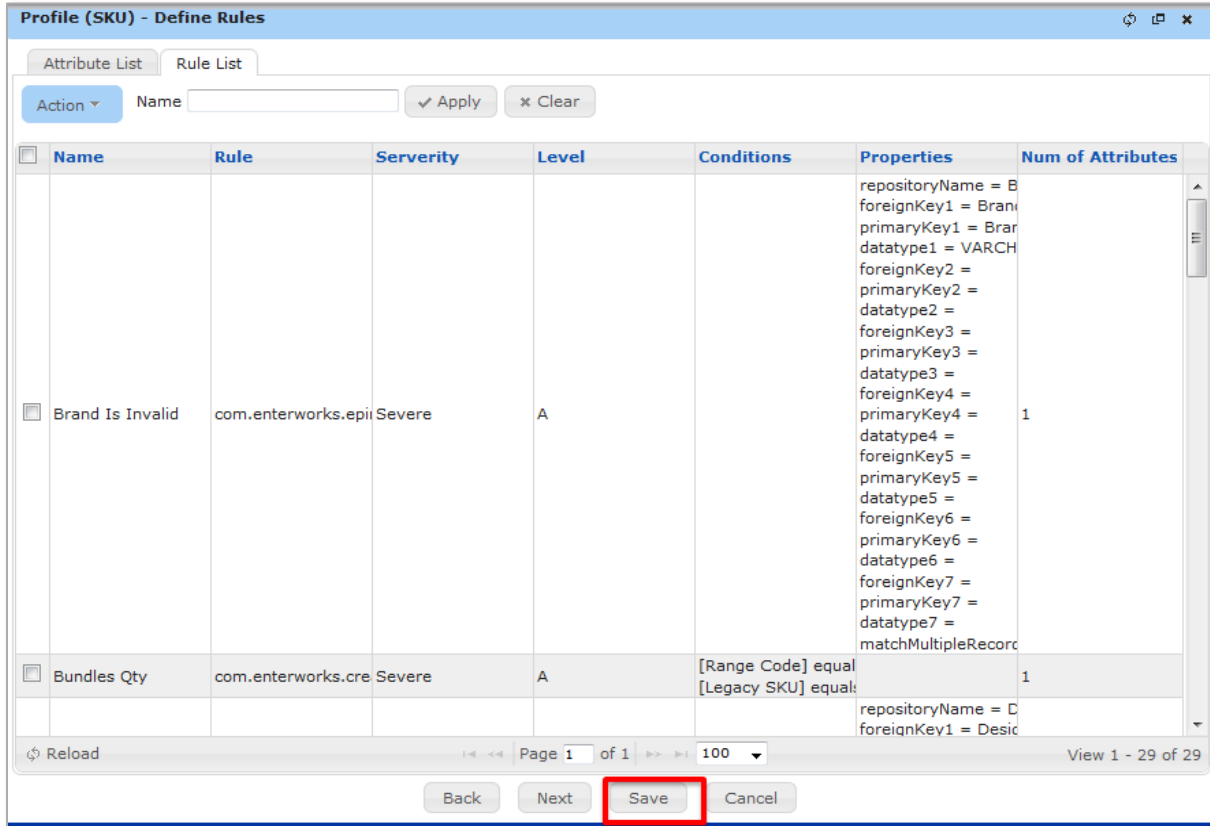
Configuration Option	Required or Optional	Definition	For Rule Type(s)
<b>Severity</b>	Required	<p>Specifies the severity of the validation error assigned to the attribute If the value of the assigned attribute violates this rule.</p> <ul style="list-style-type: none"> <li>○ <b>Warning</b> – Record will be flagged with an orange validation error icon but will be allowed to promote to production if the validation rule fails.</li> <li>○ <b>Severe</b> – Record will be flagged with a red validation icon. This setting will prevent a record from promoting to production if the validation rule fails.</li> </ul>	All
<b>Level</b>	If not specified will default to highest Validation Level.	The validation level this attribute is assigned to. Note that if the Level is not specified, it will default to the highest level.	All
<b>Operator</b>	Required	The operator compared with the assigned attribute.	<b>Simple</b>
<b>Value</b>	Required	The value the assigned attribute is compared to.	<b>Simple</b>
<b>Formula</b>	Required	How the assigned attributes will be altered.	<b>Calculate</b> <b>VTL-like</b>
<b>Call Out</b>	Required	The JAVA class process that will alter the assigned attribute. (See <a href="#">Callout</a> and <a href="#">Pre-Save Callout</a> .)	<b>Call Out</b> <b>Pre-save Call Out</b>

Configuration Option	Required or Optional	Definition	For Rule Type(s)
Class Path	Required	Specifies the class path for the JAVA class process that will alter the assigned attribute. (See <a href="#">Callout</a> and <a href="#">Pre-Save Callout</a> .)	Call Out Pre-save Call Out
Bulk Rule Text	Required	The text of the SQL query performed on the set of records being validated.	Bulk Callout

6. If you are creating or editing a Simple or Calculation rule, you can use the values of the record's other attributes in your **Value** or **Formula** setting. To do so, enter the character [. The names of the record's other attributes will be displayed in a dropdown. If you start to type the desired attribute's name, the list of attributes will be filtered accordingly.
7. **Condition(s):** Adding conditions provides you the ability to make the rule 'applicable' to the attribute based on values of another attribute. In the example below, we are setting up a rule that applies only to SKUs where Legacy SKU = No:
  - a. Click the **Add** button.
  - b. Select the attribute from the drop-down list, choose the appropriate operator and value, and then click the **OK** button.
8. If you have multiple criteria, specify whether ALL conditions must be true in order for this rule to be applicable or whether only one condition is necessary to be true.
9. **Define Rule Attributes:** Use the Action dropdown menu to assign or unassign the rule to attributes. Rules assigned to an attribute are available to enabled on the attribute.
  - **Assign:** Selecting this opens a list of attributes. Select one or more attributes you want the rule to be available for. Click **OK** to save your selections.

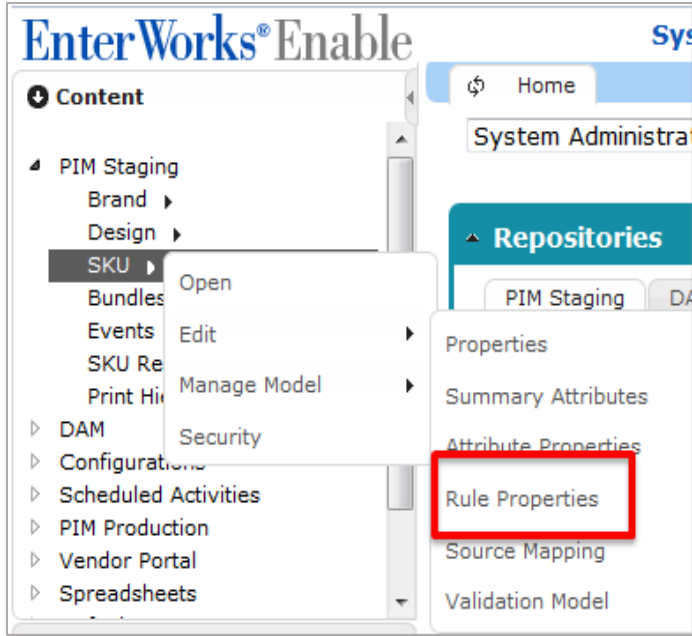
Once a rule is assigned to an attribute, if the attribute is multi-language, you can click the **Language** button to assign the rule to specific language versions of the attribute. If no languages are selected, the rule is assigned to the default language of the attribute.
  - **Detach:** To remove the rule from the list of rules that can be enabled for an attribute, select the attribute in the list, then select **Detach** from the **Action** menu.

10. Click the **OK** button to close the Rule Definition editor and return to the Profile editor.
11. Click the **Save** button.



12. If this is a new rule, we now need to assign it to (or “enable it for”) the desired repositories. Typically, validation rules are not enabled on production repositories. To do so, navigate to the **Rule Properties** page for the desired repository.





13. Select your new rule and use the down-arrow icon ▼ to assign the rule to the repository.

**SKU - Rule Properties**

Rule available:

**My New Rule (validation)**

UCC Prefix must be 00 for Displays (validation)

Rule selected:

Required - Severe (If Item Status Not 22) (validation)  
 Required - Severe (If Brand Manager Code Not P19) (validation)  
 Must Be Less Than 8 Characters Long (validation)  
 Required - Severe (If Range Code Not 92200) (validation)  
 Required - Severe (If PartType = M and Range Code Not 92200) (validation)  
 Required - Severe (If Part Type = P and Range Code Not 92200) (validation)  
 Design Is Invalid (validation)  
 Brand Is Invalid (validation)  
 Event Is Invalid (validation)  
 Bundles Qty (validation)  
 Range-Product Size-Ply Combo Invalid (validation)  
 Required - Warning (If Ecommerce Item Not V and Legacy SKU = No) (validation)  
 Required - Severe (If Shippable Unit = Y) (validation)  
 Required - Warning (If Ecommerce Item Not V, Legacy SKU = No, and Program Code = S) (validation)

Save Cancel

14. If this is a new validation rule it has now been successfully created, however, no attributes have yet been assigned to it. To assign attributes to the new rule, see [Assign/Unassign Existing Validation Rule to an Existing Attribute](#).

## Validation Rule Types

Rule Type	Definition	Notes
<b>Bulk Callout</b>	A bulk callout validation rule is a SQL query-based validation that is performed once on the set of records being validated.	See <a href="#">Bulk Callout</a> .

Rule Type	Definition	Notes
<b>Calculate</b>	Concatenates values from multiple attributes.	This is not actually a validation rule. The calculation is applied when a record is saved.
<b>Call Out</b>	Java-based process that performs a specific validation on the record (and the assigned attribute). Some callout validation rules are pre-defined and available for immediate use. The majority of the callout validation rules are developed for a specific EnterWorks implementation.	See <a href="#">Callout</a> .
<b>Hierarchy Validation</b>	Deprecated.	
<b>Normalization</b>	Deprecated.	
<b>Pre-Save Callout</b>	See <a href="#">Pre-Save Callout</a> .	See <a href="#">Pre-Save Callout</a> .
<b>Simple</b>	Simple validation rule where the designated operation (and possible value(s)) are applied to the assigned attribute.	This is the most common validation rule and is used to perform validations such as an attribute having to have a value or having to be empty, having a specific value (for example, Gross Weight must be greater than or equal to Net Weight, etc.)
<b>UCC GTIN Algorithm</b>	Verifies the value conforms to the UCC GTIN format.	
<b>UCC GLN Algorithm</b>	Verifies the value conforms to the UCC GLN format.	

Rule Type	Definition	Notes
<b>UCC UPC-12 Algorithm</b>	Verifies the value conforms to the UCC UPC-12 format.	
<b>UCC UPC-11 Algorithm</b>	Verifies the value conforms to the UCC UPC-11 format.	
<b>VTL-like</b>	Pre-Save Callout that executes a VTL script to update the record.	This is not a validation rule but a pre-save callout that will alter the repository record as it's being created or updated.

### *Bulk Callout*

A Bulk Callout validation rule is a SQL query-based validation that is performed once for the group of records being validated, such as:

- Individual record: the SQL is limited to just that record.
- Saved set: the SQL is limited to the records in the saved set.
- Full repository: the SQL query is run against the entire repository.

In each case, the results of the query are the records that violate the rule, which are listed in a temporary table. They are placed in the table so that when each record in the group is being validated, the temporary table is consulted instead of running the SQL again. When each record is validated for its other rules, the error table is consulted for any bulk callout violations. If there are any, their messages are added to the list of errors for that particular attribute.

The basic interface is that the callout needs to return one row for each record that has an error. The first column of the row is the internal ID of the record and the second column is the error message. The query itself needs to be formulated to find and return the offending rows.

Each query should be wrapped by the following outer query:

```
SELECT InternalRecordId, ErrorMessage
From (
```

```
<<... Actual query here >>  
) as T1 WHERE 1=1
```

Note that EnterWorks references repositories by using their name in the default language, which is English. Also, the column names of the error table cannot be changed.

In the examples given below:

- The repository named “Item\_Staging” contains columns whose restricted names are “Manufacturer”, “Manufacturer\_Name”, and “SKU\_Group”.
- The repository named “PIM\_MarketingAd\_Staging” has columns whose restricted names are “Marketing\_Ad\_ID”, “Marketing\_Ad\_ID”, and “Image\_Context”.

Example queries:

1. **Item\_Staging** records must have **Manufacturer**:

```
select InternalRecordId,  
  
cast(Manufacturer as nvarchar(30)) + ' has no reference in  
Manufacturer repository' as ErrorMessage  
  
from Item_Staging where Manufacturer is not null  
  
and InternalRecordId not in  
  
(select i.InternalRecordId from Item_Staging i inner join  
Manufacturer b on i.Manufacturer=b.Manufacturer_Name)
```

2. All **Item\_Staging** records require the same SKU Group:

```
select m.InternalRecordId, 'SKU Group:'+m.SKU_Group+' must  
have the same Taxonomy:'+m.Taxonomy as ErrorMessage  
  
from Item_Staging m, (select distinct a.InternalRecordId  
internalId from Item_Staging a inner join Item_Staging b on  
a.SKU_Group=b.SKU_Group  
  
where a.SKU_Group != 'DISCONTINUED' and a.Taxonomy !=  
b.Taxonomy) t1 where m.InternalRecordId=t1.InternalId
```

3. Records in **PIM\_MarketingAd\_Staging** must have a **Main** image:

```
select InternalRecordId,
```

```
'Marketing Ad: ' + Marketing_Ad_ID + ' has no Main image'  
as ErrorMessage  
from PIM_MarketingAd_Staging  
where InternalRecordId not in  
  
(select ma.InternalRecordId from PIM_MarketingAd_Staging ma  
inner join DAMLink dl on ma.Marketing_Ad_ID=dl.PK1  
where dl.Image_Context = 'Main')
```

If a bulk callout validation rule is to be conditionally applied, the conditions for that application must be encoded in the SQL – the rule conditions will be ignored. This is because those conditions are based on the current record, but the bulk callout must validate all records in the repository or saved set.

## Callout

A Callout is a Java-based process that performs a specific validation on a record (and the assigned attribute). Some callout validation rules are pre-defined and available for immediate use. The majority of the callout validation rules are developed for a specific EnterWorks implementation.

The following callout rules are pre-defined in EnterWorks:

- **AttributesCalc**
- **AttributesPriceUpdate**
- **AttributesSearchAndUpdate**
- **AttributesSearchAndCalculate**
- **CheckAttributeCodeSetName**
- **IsLengthLessThan**
- **MatchesPattern** – compares the value to a REGEX pattern. If the value does not match one of the defined patterns, the attribute is flagged as invalid. A common application is to define the different formats for a phone number (for example, 1 (909) 555-1212, (714) 555-1212 x 1234, 818-555-1212, etc.)
- **NoBlanks** – flags a value if it contains any spaces

- **NoInvalidCharacters** – flags a value as invalid if it contains any of the designated characters.
- **ValidReference** – flags a record as invalid if a corresponding record is not found in a linked repository (NOTE: This has been supplanted by a Bulk Callout which is a more-efficient to perform such a validation)
- **ValidCharacters** – flags a value as invalid if it contains anything but the designated characters
- **ValidDimensions**
- **PIMWebServiceGetRepoAttrData**
- **UpdateDistinctChildValues**
- **UpdateParentWithDistinctChildValues**
- **<custom>** - Custom callout validation rules can be implemented to perform any needed specialized validation, providing those callouts follow the prescribed interface in the form of a Java class. Examples of callout rules:
  - inheritance – the outcome of the validation is dependent upon the content of a linked parent record according to configured inheritance rules
  - comparison to data outside of EnterWorks, such as using an API to access a web service (NOTE: while such operations are possible, they are very expensive in terms of performance impact)
  - complex logic that cannot be expressed in SQL (bulk callout) or simple validation rule. NOTE: many of the callouts listed above originated as complex rules needed for an implementation but were recognized as having general applicability so were assimilated into the product.

### *Pre-Save Callout*

Pre-Saved Callouts are not a validation rule; they are used to perform a pre-save calculation on the record before the record is created or updated. For example, a Pre-Save Callout could conditionally define a default value for an attribute or calculate a primary key based on other attribute values. If it is a compiled class, it may define a primary key using data in an external table or repository, for instance, selecting a number from a pool of available numbers, then removing the number from the pool.

Pre-Save Callouts include:

- **VTLUpdatePreSaveCallout** – updates the record using the VTL script defined in the callout properties.
- **AttributeVTLCalculatePreSaveCallout**
- **PriceCopyPreSaveCallout**
- **AttributesSearchAndUpdatePrice**
- **GroovyPreSave** – executes a Java-based Groovy script defined in the callout properties.
- **NodeJSPreSave**
- **<custom>** - any Java class that follows the Pre-Save Callout interface can be implemented to perform a pre-save operation.

## Assign/Unassign Existing Validation Rule to an Existing Attribute

EnterWorks does support validation assigning validation rules to repeatable attributes.

To assign or unassign an existing validation rule to an attribute:

1. In the **Feature** bar, under the **Content** tab, select a repository based on the desired profile, click the arrow to the right of **Manage Model** and select **Profile**.
2. The Profile editor will open. Click **Next** at the bottom of the page to move to the Define Attributes page.
3. Find the attribute you wish to assign to a validation rule and double-click to open it in edit mode.
4. The top of the editor lists configuration options. Near the bottom of the editor is the **Define Attribute Rules** section. From the **Action** dropdown, choose **Assign**.
5. A list of rules will appear. Select one or more rules you wish to assign to the attribute. Note that if the attribute is multi-language, the rule may only apply to certain language versions of an attribute. When you are finished selecting the rules, click **OK** at the bottom of the page.
6. You are taken back to the attribute editor window where you will see the newly assigned validation rule.
7. To detach a validation rule, in the Define Attribute Rules section, select the rule, open the Action dropdown menu and select **Detach**.



**Define Attribute Details (# of Print Colors)**

Common Attribute: # of Print Colors per Panel[VARCHAR] Data Type: VARCHAR  
 Name: # of Print Colors Data Size: 50  
 Restricted Name: Description: Default Value:  
 Group: Product Detail - Marketing Manage  
 Associated Group: Manage  
 Code Set: Manage  
 Category Attribute Association: Manage  
 Special Function Ind:

Property Values  
 Target ASU Column (VARCHAR10)

OK Cancel

Define Attribute Rules

Action

<input type="checkbox"/>	Name	Rule	Severity	Level	Condition(s)	Properties
<input type="checkbox"/>	My New Rule	is not empty	Warning	A	[Legacy SKU] equals No	

8. Click **OK** to return to the Profile editor.
9. To save your changes, make sure to click **Save** in the Profile Editor.
10. Clear the data cache.

## Defining Attribute Validation Rules for Validation Levels

To define attribute validation rules for validation levels:

1. In the **Feature** bar, under the **Content** tab, select a repository based on the desired profile, click the arrow to the right of **Manage Model** and select **Profile**.
2. The Profile editor will open. Click **Next** at the bottom of the page to move to the Define Attributes page.

3. The **Attribute List** will appear. Double-click the desired attribute to open it in the Define Attribute Details editor.

The screenshot shows the 'Profile (acme\_Items) - Define Attributes' window. It features a tabbed interface with 'Attribute List' selected. Below the tabs are filters for 'Action' and 'Utilities', and a search area for 'Attribute Type' (set to 'All') and 'Name'. A table lists various attributes with columns for Name, Description, Group, Associated Code, Category, Attribute, Data Type, Default, # of Rules, and Is Primary. The 'Authorized Dealer' attribute is highlighted in yellow. At the bottom, there are navigation buttons: Reload, Back, Next, Save, and Cancel, along with a pagination indicator showing 'Page 1 of 1' and a page size of '100'.

<input type="checkbox"/>	Name	Descri	Groi	Associated G	Code S	Category	Attribu	Data Ty	Default	# of Rules	Is Prin
<input type="checkbox"/>	Active Indicator		Oper		YesNo			VARCHA			
<input type="checkbox"/>	Air Shippable Indicator		Oper		YesNoSp			VARCHA	Y		
<input type="checkbox"/>	Annual Estimate Sell Qty		Oper					DECIMAL			
<input type="checkbox"/>	Assembly Required Indicatc		Oper		Assemb			VARCHA			
<input type="checkbox"/>	Authorized Dealer		Marke					VARCHA			
<input type="checkbox"/>	Availability		Comr					VARCHA			
<input type="checkbox"/>	Available Sizes		Ident		Sizes			VARCHA			
<input type="checkbox"/>	Brand		Ident					VARCHA			
<input type="checkbox"/>	canonical_link		Marke					VARCHA		1	
<input type="checkbox"/>	Carton Per Pallet Qty		Carto					INTEGER			
<input type="checkbox"/>	Carton Per Tier Qty		Carto					INTEGER			
<input type="checkbox"/>	Carton Serial Number		Carto					VARCHA			
<input type="checkbox"/>	Color Code		Ident		Color C			VARCHA			
<input type="checkbox"/>	Common Base Qty		Comr					INTEGER			
<input type="checkbox"/>	Common Base Unit		Comr		CES Uni			VARCHA			
<input type="checkbox"/>	Common List Price		Comr					CURREN			
<input type="checkbox"/>	Common Selling Unit		Comr		CES Uni			VARCHA			
<input type="checkbox"/>	Country of Origin		Expor		Country			VARCHA			

4. Double-click the desired attribute to open it in the **Define Attribute Details** editor.
5. In the **Define Attribute Rules** section, select **Action > New**. The **Define Rule Details** editor will appear.

**Define Rule Details (New)**

Name:

Type:

Severity:

Level:

Rule: Operator:

Condition(s):

Add Edit Remove

Any conditions valid  
 All conditions valid

OK Cancel

Define Rule Attributes

Action ▾

<input type="checkbox"/>	Name	Group	Data Type	Is Global
<input type="checkbox"/>	Attribute Required for Level B			

- Define the rule for the attribute and select the **Validation Level** that the rule applies to.
- Click the **OK** button to return to the **Define Attribute Details** editor. The new rule will be shown in the rules list.

**Define Attribute Details (Available Sizes)**

Common Attribute:

Name:

Restricted Name:

Description:

Group:  Manage

Associated Group:  Manage

Code Set:  Manage

Category Attribute Association:  Manage

Special Function Ind:

Data Type:

Data Size:

Default Value:

Is Multi Language  
 Is Category Attribute  
 Is Required  
 Is Primary Key  
 Is Repeatable  
 Is Business Key

OK Cancel

Define Attribute Rules

Action ▾


<input type="checkbox"/>	Name	Rule	Severity	Level	Condition(s)	Properties
<input type="checkbox"/>	Attribute Required for Level B	is not empty	Warning	B		

- Click **OK** to return to the **Profile Editor**.

9. Click **Save** and a confirmation message will be shown.

**IMPORTANT:** For a new rule the following steps must be taken to activate it.

To activate the new rule.

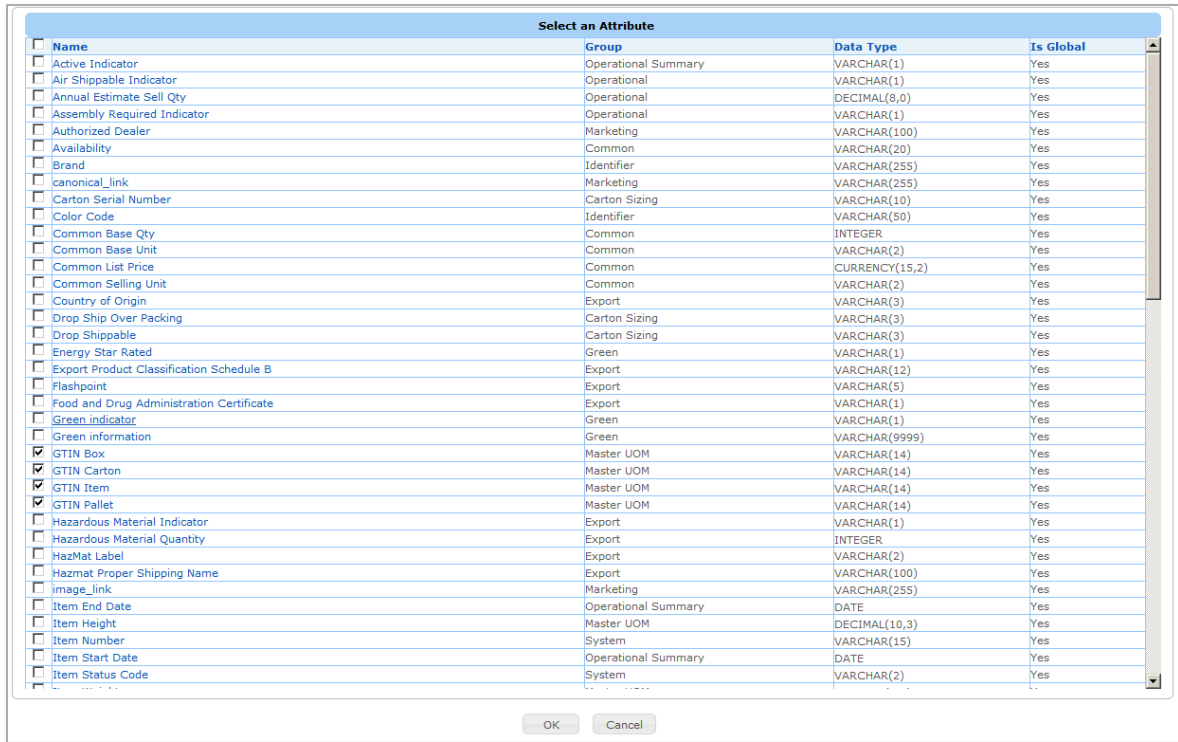
1. Open the Feature bar, right-click the repository, select **Edit**, then select **Rule Properties**.
2. The **Rule Properties** editor will open.
3. Select the new rule in the top **Rule Available** list box.
4. Click the Down  arrow to move the rule to the bottom list box.
5. Click the **Save** button.

## Assigning Multiple Attributes to a Validation Rule.

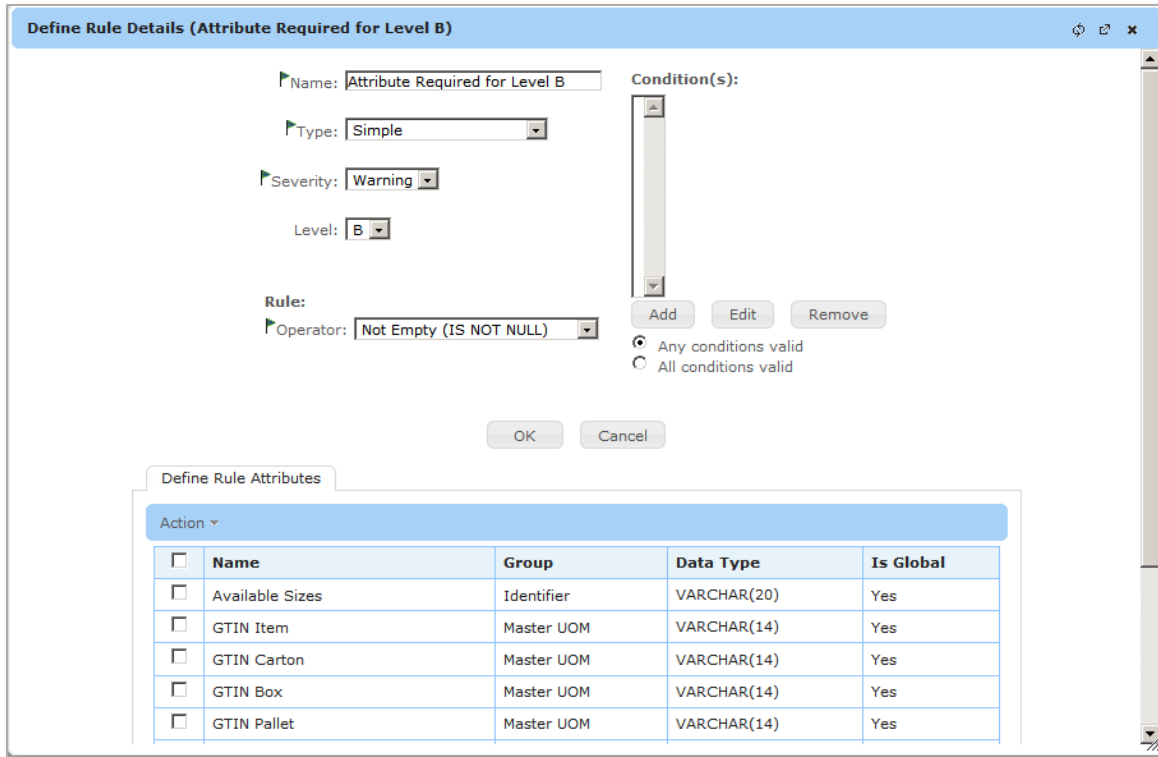
EnterWorks does support validation rules for repeatable attributes.

To assign multiple attributes to a validation rule:

1. Open the desired profile in the **Profile** editor, click **Next**, and open the **Rule List** tab.
2. Select the Validation Rule (or create a new one with **Action > New**).
3. Double-click the record or in the **Action** dropdown menu, select **Edit**.
4. The **Define Rule Details** editor will open.
5. In the **Define Rule Attributes** section, in the **Action** dropdown, click **Assign**.
6. A list of attributes will appear.



7. Click the checkbox next to the attributes to apply the same Validation Rule (with the same validation level).
8. The **OK** button is at the bottom of the list of attributes. You may need to scroll down to find it. Click the **OK** button and you will return to the **Define Rule Details** editor.
9. The new attributes for the Validation Rule will appear in the list.



10. Click **OK** to return to the **Rule List** tab on the **Profile** editor.
11. Click **Save** to save the changes to the Rule List.
12. A confirmation message will be shown.

## Setting a Syndication/Export's Validation Level

When defining a Syndication Template to use for an export or syndication, the default validation level is set to "A". This validation level can be changed for the template during setup or modification. For information about setting validation levels see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Setting a Record's Validation Level

The default validation level can be overridden for a specific record, if required. For information about setting validation levels, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Managing Digital Assets

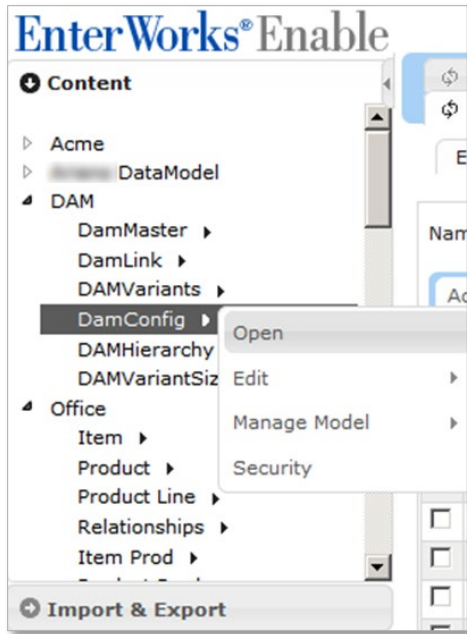
Digital assets typically consist of digital images and PDF documents, but may also include audio files, video files, Microsoft Office documents, and other types of files. An organization's digital assets, as well as the metadata associated with these digital assets, are stored in the DAM repositories. After the digital asset objects are stored, they can be associated with any repository that has been set up as Digital Asset enabled by the Systems Administrator. For instructions for managing digital assets through the New User Interface, see the Precisely EnterWorks online help at <https://support.precisely.com>.

This section only covers activities normally handled by a System Administrator.

## Bulk Importing Digital Assets

When dealing with a large number of Digital Assets to import, the basic import process could be lengthy. Another option called Bulk Uploading is available within EnterWorks and utilizes an automatic upload folder located on the EnterWorks Web Server.

The upload directory is defined in the `SourceFolder` property in the `DamConfig` repository. To access this value, open the `DamConfig` repository as shown below.



The first record shows the Source Folder's path on the server.

#	Error Ind	Sequence	Key	Value	ConfigGroup
1	✓	1	SourceFolder	C:\Enterworks\DAM_Drop	BulkUploadService
2	✓	2	DestinationFolder	C:\Enterworks\DAMRoot \Original	BulkUploadService
3	✓	3	UNCPath	C:\Enterworks\DAMRoot \Original	BulkUploadService

To import multiple digital assets:

1. Stage the files to be uploaded in a local directory.
2. To automatically assign the files to nodes in a DamHierarchy, define a directory structure that mirrors the DamHierarchy structure and place the files in the appropriate folder. Files placed in the upload directory will be uploaded without setting or changing the DamHierarchy assignments. New nodes will be created in the DamHierarchy structure if the directory structure does not match it exactly. For example, if a file is uploaded into a directory <drop\_folder>\Furniture\Office\Desks, the DamHierarchy setting for the assets being loaded will be set to the Furniture.Office.Desks node.

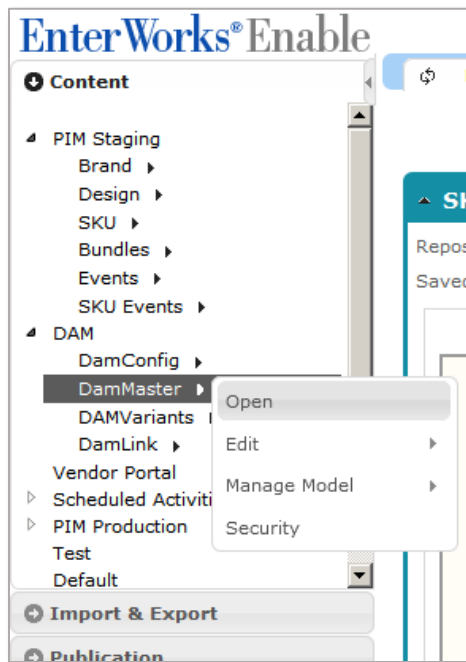


3. Log into the server and follow the source path to the Mass Upload folder (or access the upload folder if it has been configured as a shared directory on the network).
4. Copy the local files (including the directories) into the upload folder. The folder is checked every 15 seconds (which is configurable by the System Administrator) for new items, and when they are found they are automatically uploaded. Results of the upload processing can be viewed in the log file for the Asset Monitor process.

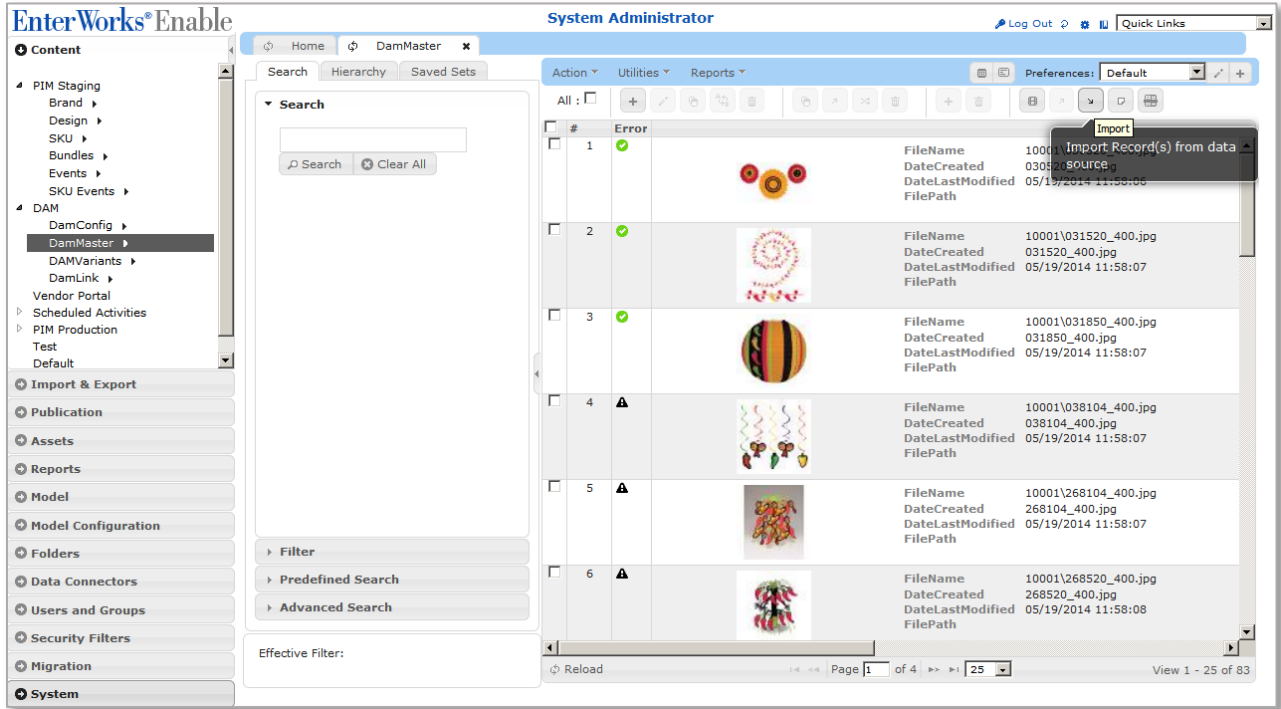
## Batch Importing Digital Asset Metadata

The Import function in the DamMaster also allows importing CSV or Excel files containing metadata for existing digital assets including links to other repositories. This is useful when you want to update information for many assets. This functionality is recommended only for System Administrators and should be tested on a development or QA box before running on the production server.

1. Under the **Content** feature in the Feature bar, open the **DAM** Repository Group and the DamMaster repository under this group.



2. Click the **Import Record(s) from data source** button on the toolbar third button from the right.



3. Select the **Import Type** and click the **Next** button. The Setup Import page will appear.

**Setup Import**

File Name:  DamMaster captions import.xlsx  
 The file upload size limit is set to 100 MB.

**Options**

File Format:  Delimited  Excel 2003  Excel 2007  
 File Encoding:

Save Records Imported From Source As Set  
 Saved Set Name:   
 Saved Set Description:   
 Send Above Saved Set To Workflow

Save Records In Repository but Not Imported From Source As Set  
 Saved Set Name:   
 Saved Set Description:

**Advanced Options**  
 NOTE: Only columns with names that match the profile column names will be mapped. The file must contain columns for all of the primary key fields.

- Map the fields from your file to appropriate target fields. **Note:** The fields will be automatically mapped if the column names in the file match the attribute names in the repository.

**Map Column Names to Format Attributes**

Taxonomy Name: MediaType  
 Taxonomy Node:  Map across all taxonomies  
   
 Update all records on partial key matching

Column Position	Column Name	Repository Format Attribute
1	FileName	FileName ** (Primary Key 1)
2	Caption	Caption

- Click **Finish**.

6. Monitor the import job on the **Job Monitor** page.

## Category Attributes

EnterWorks allows you to manage attributes based on their Taxonomy categorization. All attributes must first be defined in the profile for the repository. The user must be granted security to the underlying Taxonomy and profile to be able to edit the attributes.

Each Category Attribute must then be assigned to the nodes of the Taxonomy. This can be done from the list view for the repository containing the Category Attributes, from the Taxonomy editor under the Model feature, or from the Hierarchy Tab in the repository associated with the Taxonomy.

## Manage Category/Dynamic Attributes from the Taxonomy List

To manage category or dynamic attributes from the Taxonomy list:

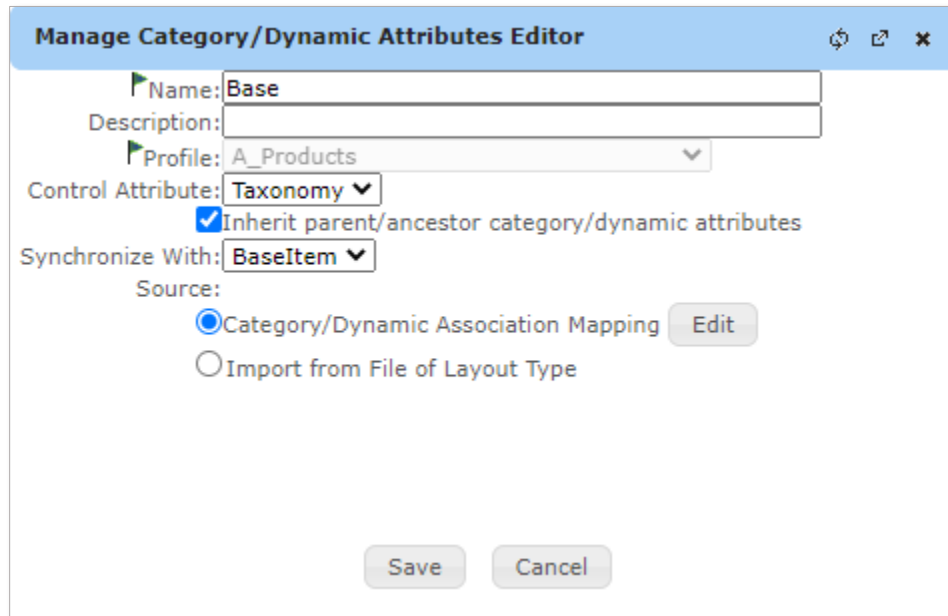
1. Log into EnterWorks.
2. Expand the left-side feature bar, and expand the **Model** Feature and click on the **Taxonomy** function. A list of the existing Taxonomies will appear.
3. Across the top of the screen (just under the Taxonomy tab's name) is the list of folders that contain the taxonomies. Open the folder containing the Taxonomy that is tied to the Category Attributes, if applicable.
4. Select the desired **Taxonomy** record, open the **Manage** dropdown, and select **Category/Dynamic Attribute Association**. The **Manage Category/Dynamic Attributes** editor will appear.



The screenshot displays the 'Manage Category Attributes Editor' interface. At the top, there is a blue header bar with the title 'Manage Category Attributes Editor'. Below the header, there is a blue button labeled 'Action' with a downward arrow. Underneath, there is another blue header bar titled 'Category Attribute Association'. Below this header is a table with the following structure:

<input type="checkbox"/>	#	Category Attribute Association Name	Profile
<input type="checkbox"/>	1	SKU Category Attributes	_SKU

5. Double-click the desired Category/Dynamic Attribute Association object to open it for editing, or select it, open the **Action** dropdown, and select the **Edit**. The **Manage Category/Dynamic Attributes** editor will appear.

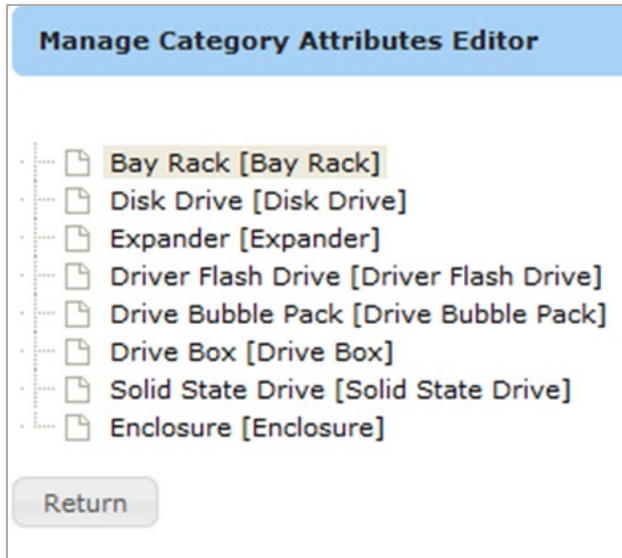


The screenshot shows a dialog box titled "Manage Category/Dynamic Attributes Editor". It contains the following fields and options:

- Name:** Base
- Description:** (empty text box)
- Profile:** A\_Products (dropdown menu)
- Control Attribute:** Taxonomy (dropdown menu)
- Inherit parent/ancestor category/dynamic attributes
- Synchronize With:** BaseItem (dropdown menu)
- Source:**
  - Category/Dynamic Association Mapping **Edit** (button)
  - Import from File of Layout Type

At the bottom of the dialog are **Save** and **Cancel** buttons.

6. Click the **Edit** button next to **Category/Dynamic Attribute Association Mapping**. The **Manage Category/Dynamic Attributes Editor** will appear.



7. Repeat the following steps for each Taxonomy node to be updated:
  - a. Drill down into the Taxonomy and select the node to be updated.
  - b. The **Category Attribute Association Mapping Editor** appears, showing the selected node attribute assignments along with any inherited assignments from parent nodes.

**Category/Dynamic Association Mapping Editor** ⚙️ ↗️ ✕

Selected Detail Code:IndiaMART.Furniture & Supplies [Furniture & Supplies]

**Category/Dynamic Attributes for the selected Detail Code**

	Attribute	Code Set (Optional)	Subset/All
<input checked="" type="checkbox"/>	Size	Size	Subset
<input type="checkbox"/>	Load Capacity		
<input type="checkbox"/>	No of Legs		
<input type="checkbox"/>	Seat Material		
<input type="checkbox"/>	Material		
<input type="checkbox"/>	Color	Colors	Subset

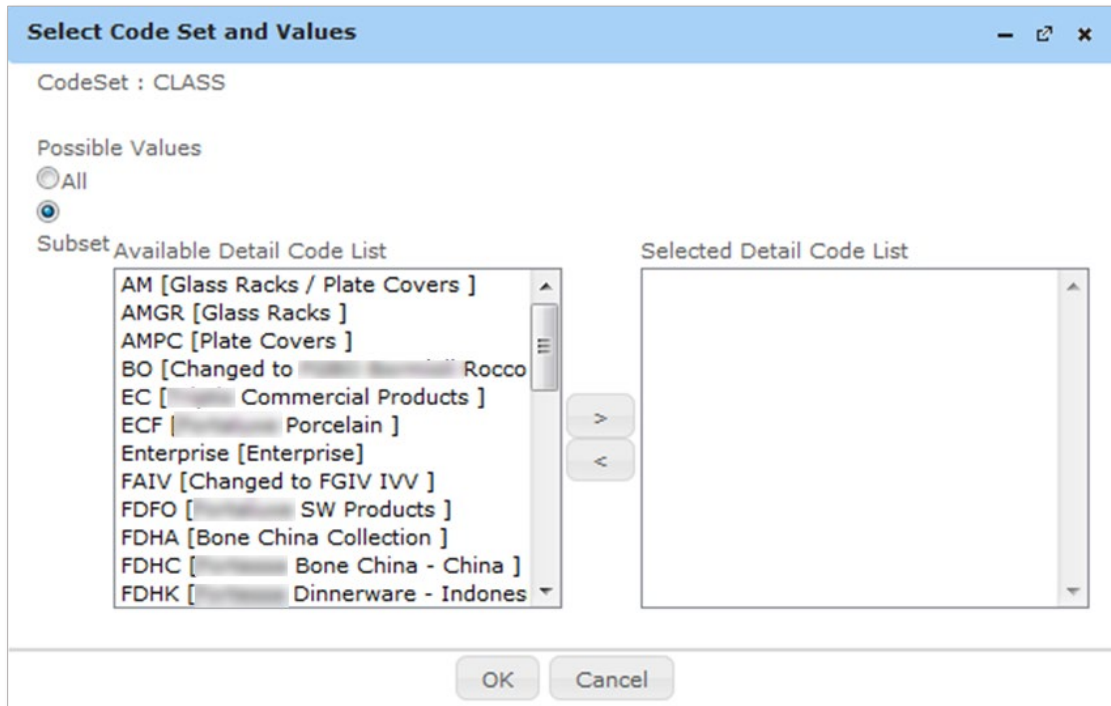
Manage
Delete
Possible Values

**Category/Dynamic Attributes Inherited**

Level	Parent/Ancestor Code	Attribute	Code Set (Optional)	Subset/All
		IndiaMART Member Since		
		Annual Turnover	Annual Turnover	All
		GST		
		Import Export Code (EIC)		
		Nature of Business	Nature of Business	All
		Company Overview		
		Contact Name		
		No of Employees	Company Size	All
		Website		
1	IndiaMART	Year Established	Established	All
		Address		
		Phone Number		
		Legal Status	Firms Legal Status	All
		Email Address		
		Product Usage		
		Packaging Details		
		Delivery Time		
		Min Order Quantity		
		Company Name		

Set Order
Close

- c. To remove an attribute assignment, check the checkbox next to the attribute and click **Delete**.
- d. To change the codes of an attribute's code set (if a code set is assigned to the attribute) that are allowed for this node, check the checkbox next to the attribute and click the **Possible Values** button at the bottom of the list.



- i. Select the **Subset** radio button.
  - ii. To assign codes, select the desired codes from the **Available Detail Code List** and click the right-arrow button to move them to the **Selected Detail Code List**. To unassign codes, use the left-arrow button to move codes from the **Selected Detail Code List** back to the **Available Detail Code List**.
  - iii. Click the **OK** button or **Cancel** button to return to the **Category/Dynamic Association Mapping** editor.
- e. To add an attribute assignment, click the **Manage** button. **The Select Category/Dynamic Attribute** window appears.



**Select Category/Dynamic Attribute**

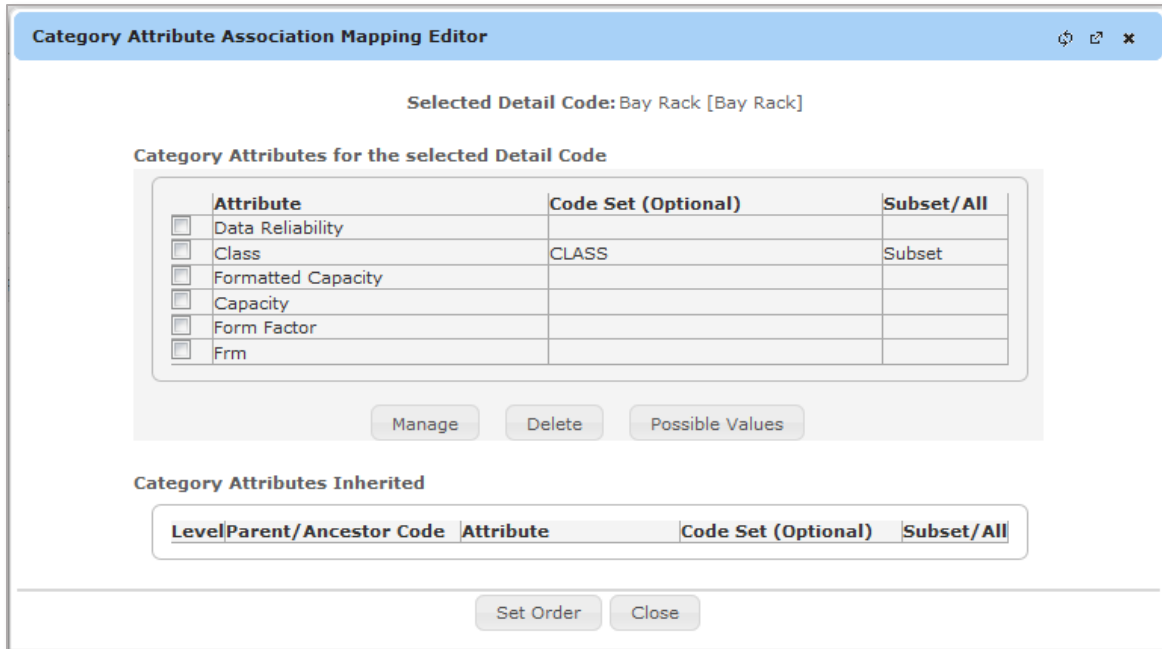
Attribute List

Action   Attribute Type  Name

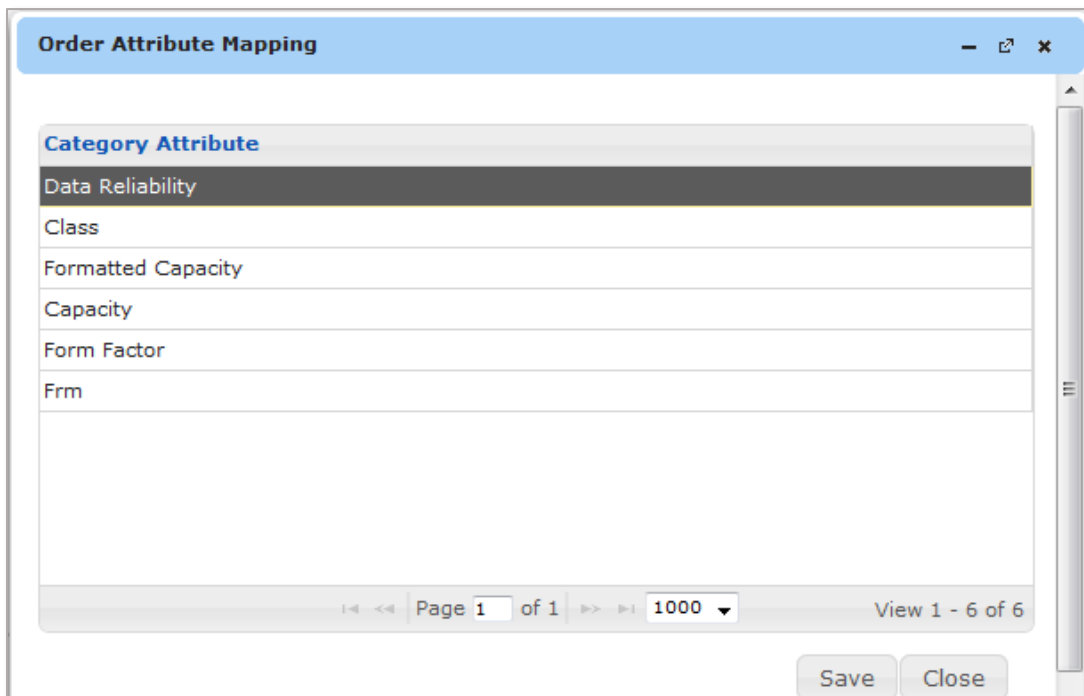
<input type="checkbox"/>	Name	Tab	Group	Code Set	Data Type	Is
<input type="checkbox"/>	a (Flanges)	Master Article	Master Article Raw		VARCHAR (50)	
<input type="checkbox"/>	Accessories	System	Master Article Cate	Accessories for Joi	VARCHAR (50)	
<input type="checkbox"/>	Actual Design Spec Sheet Submission Date	Project Mgt	Project Status		DATE	
<input type="checkbox"/>	Adaptive interface machine direction (ADINTMS)	Product Details	Category Attribute	Adaptive interface	VARCHAR (50)	
<input type="checkbox"/>	Additional Construction for Pipe	Master Article	Master Article Pipe	Additional Constr	VARCHAR (50)	
<input type="checkbox"/>	Address	Seller Details	Contact Details		VARCHAR (250)	
<input type="checkbox"/>	Age Grade	System	Specifications	SS_Age_Gender	VARCHAR (50)	
<input type="checkbox"/>	Allowable Operating Pressure	Master Article	Master Article Join		VARCHAR (50)	
<input type="checkbox"/>	Angular Defexion	Master Article	Master Article Join		VARCHAR (50)	
<input type="checkbox"/>	Annual Turnover	Seller Details	Overview	Annual Turnover	VARCHAR (20)	
<input type="checkbox"/>	App Compatible	Product Details	Appliance Categor	Yes/No	BOOLEAN	
<input type="checkbox"/>	Apparel Breathable	Product Details	Category Attribute		BOOLEAN	
<input type="checkbox"/>	Apparel Color	Product Details	Category Attribute	Apparel Color	VARCHAR (50)	
<input type="checkbox"/>	Apparel Conditions	Product Details	Category Attribute	Apparel Conditions	VARCHAR (50)	
<input type="checkbox"/>	Apparel Designed For	Product Details	Category Attribute	Apparel Designed	VARCHAR (50)	
<input type="checkbox"/>	Apparel Fabric	Product Details	Category Attribute	Apparel Fabric	VARCHAR (50)	
<input type="checkbox"/>	Apparel Fit	Product Details	Category Attribute	Apparel Fit	VARCHAR (50)	
<input type="checkbox"/>	Apparel Insulation	Product Details	Category Attribute	Apparel Insulation	VARCHAR (250)	
<input type="checkbox"/>	Apparel Length	Product Details	Category Attribute	Apparel Length	VARCHAR (50)	
<input type="checkbox"/>	Apparel Lining	Product Details	Category Attribute	Apparel Lining	VARCHAR (50)	

Page 1 of 4 100 View 1 - 100 of 381

- Select the desired attributes to be assigned and click the **Add Selected** button at the bottom of the editor. The selected attributes will be added to the list. Then click **Save** or **Cancel** to return to the **Category/Dynamic Association Mapping** editor.

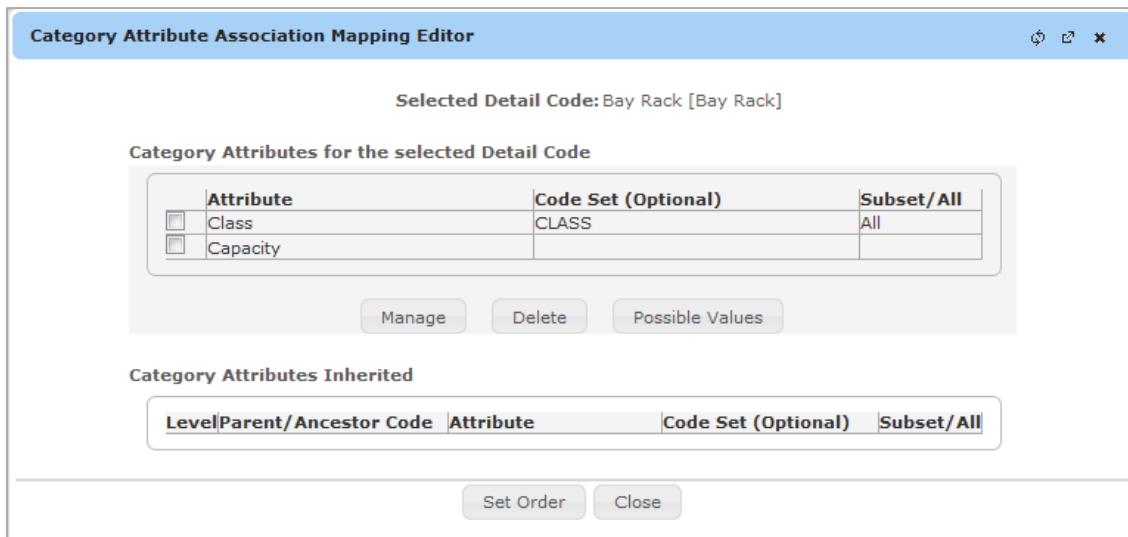


- To set the order of the assigned and inherited attributes, click the **Set Order** button at the bottom of the editor. The **Order Attribute Mapping** editor will appear.

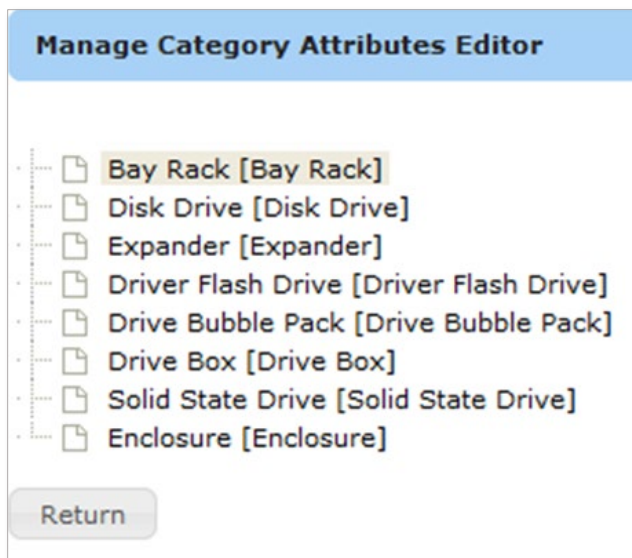


10. Drag and drop the attributes into the desired order.

11. Click the **Save** button, then click the **Close** button. You will be returned to the **Category/Dynamic Association Mapping** editor.



12. Click the **Close** button to return to the **Manage Category Attributes Editor**.



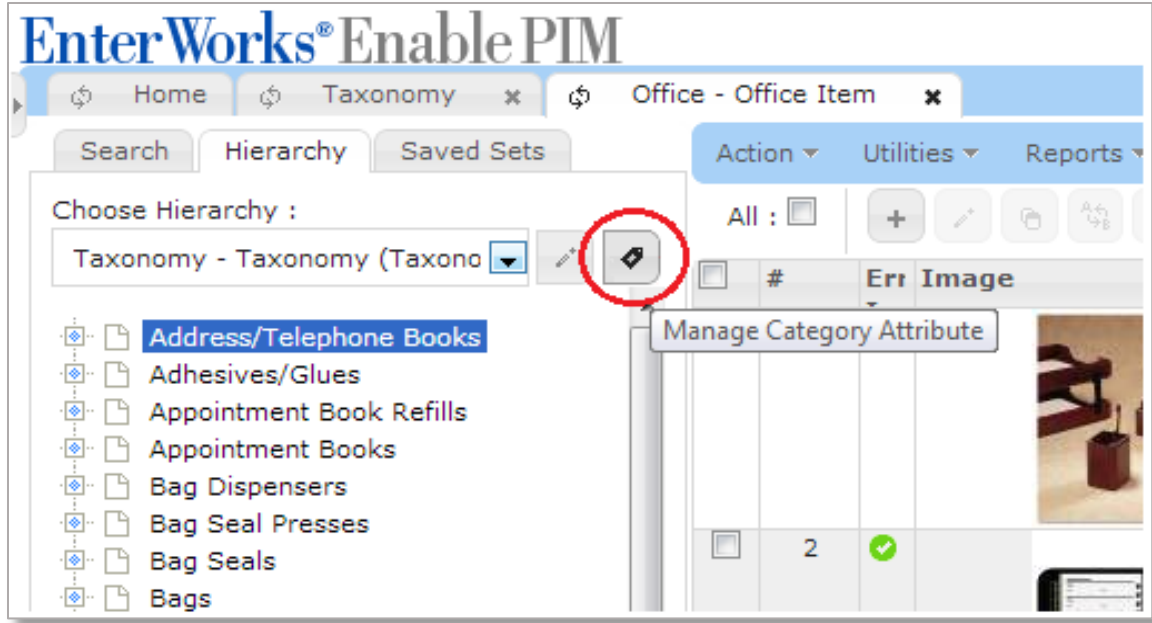
13. Click the **Return** button to return to the **Manage Category/Dynamic Attributes Editor**.
14. Click the **Save** button.

NOTE: If the Save button is not clicked on the **Manage Category Attributes Editor** window, then any changes made to the assignments will be lost.

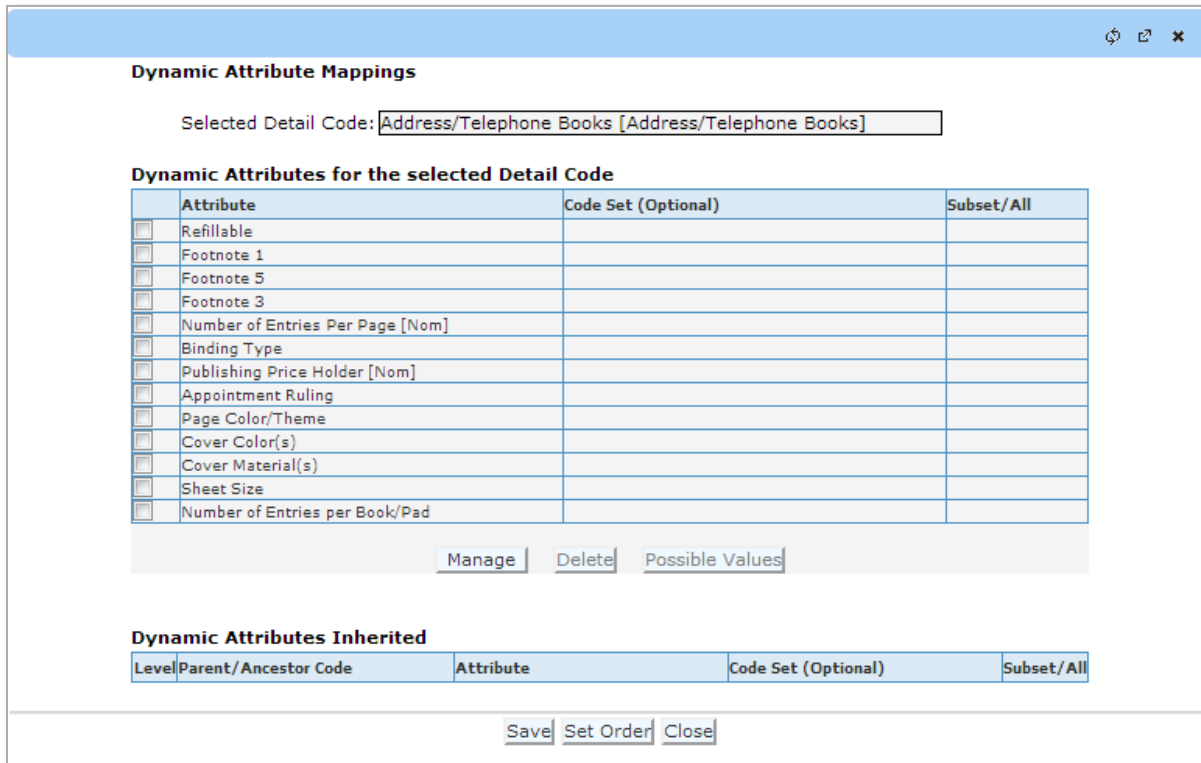
## Manage Category/Dynamic Attributes from Repository View

To manage category or dynamic attributes from the Repository View:

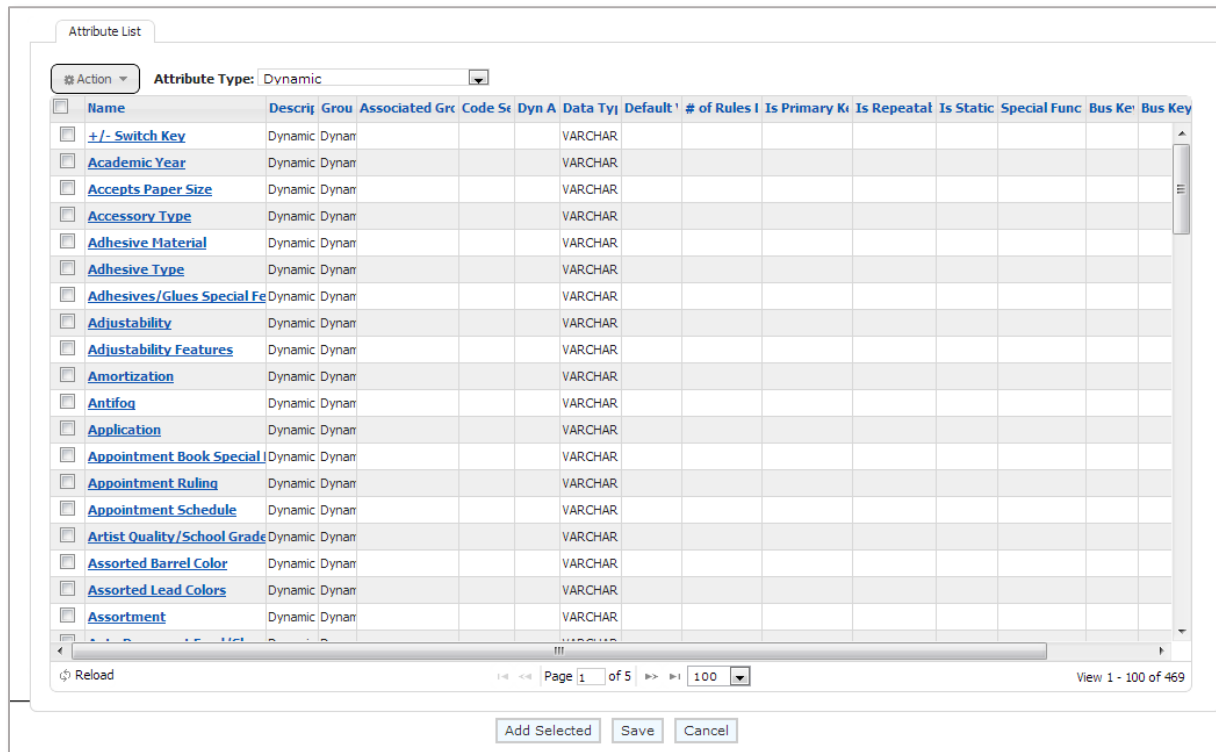
1. Log into EnterWorks.
2. Expand the **Feature** bar, and expand to show the desired repository, open it, and select the **Hierarchy** sub-tab at the top of the tab..
3. Use the **Choose Hierarchy** dropdown to select the Taxonomy. The Taxonomy tree will load below the drop-down list.
4. Select a Taxonomy node and click the Manage Category Attributes icon from the toolbar. If you hover over the icon, it will say **Manage the taxonomy's available category attributes**.



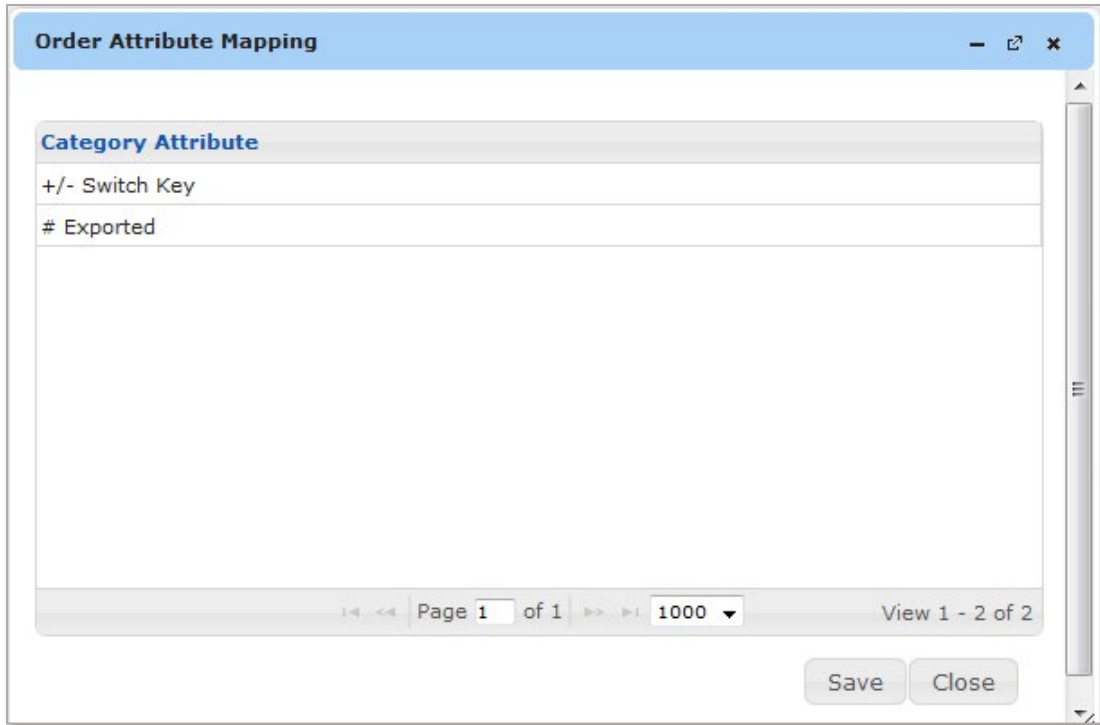
5. The window below is shown with the current list of attributes for the selected node.



- To unassign an attribute from a node, select the attribute and click the **Delete** button. This will not delete the attribute; it will only remove from the node.
- To add a new Category Attributes to this Taxonomy node, click the **Manage** button and the **Select Category/Dynamic Attribute List** will appear.



- Select the checkbox next to the attribute(s) you want to assign to the node.
- Use the **Action** drop-down menu to add new attributes that are not already defined, then click **Add Selected** and you will be returned to the Taxonomy editor.
- In the Taxonomy editor, to change the order of the attributes, click **Set Order**. The **Order Attribute Mapping** editor will appear.



11. Drag and drop the attributes into the desired order.
12. Click **Save** on the **Order Attribute Mapping** editor and you will be returned to the **Taxonomy** editor.
13. Click **Save** to save your changes to the Taxonomy.

**NOTE:** If Save is not clicked, the changes to the Category/Dynamic Attribute assignments for the selected node will be lost.

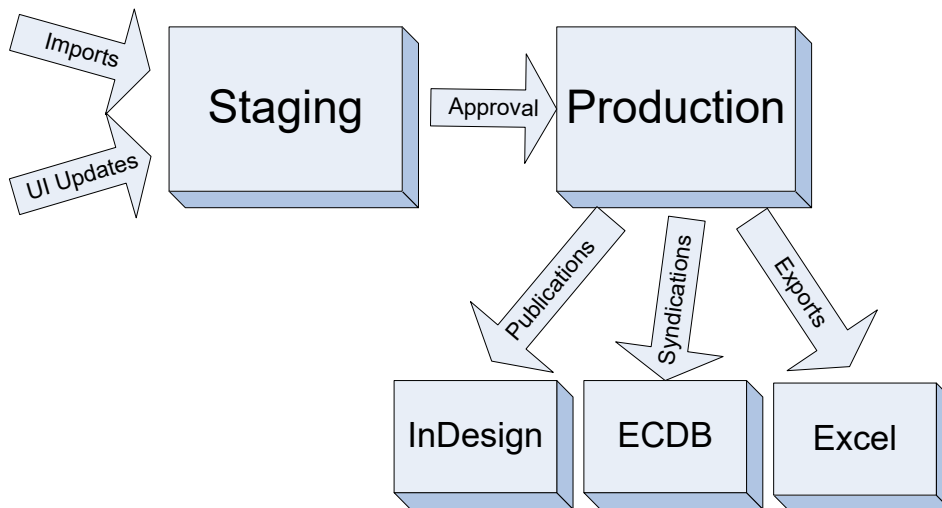
## Using Staging and Production Repositories

### Staging vs. Production Repositories

In order to insulate “production-ready” data from updates that may be incomplete or invalid, EnterWorks supports the creation of separate Staging and Production versions of the data. When this functionality is used, each repository has a staging and a production version.

- Staging – A staging repository is the work area where records are updated by users. A record’s updates will stay in the staging area until there are no severe validation errors on the record. The record(s) can then be promoted to production which takes the changes that were done to staging and applies them to the production record(s). If the record is new, it will be added to production. Promotion of data to production can be done either automatically or manually.
- Production – approved data for generating publications or creating/scheduling exports for syndications.

The following diagram shows the data flow between the EnterWorks Staging and Production repositories as well as to external entities:



Promotion can be controlled by requiring records to be approved in staging before being promoted to Production. Some repositories may be configured to not require an approval and some users may have the authority to bypass an approval. In most cases, records will be automatically promoted to Production if they do not contain severe validation errors.

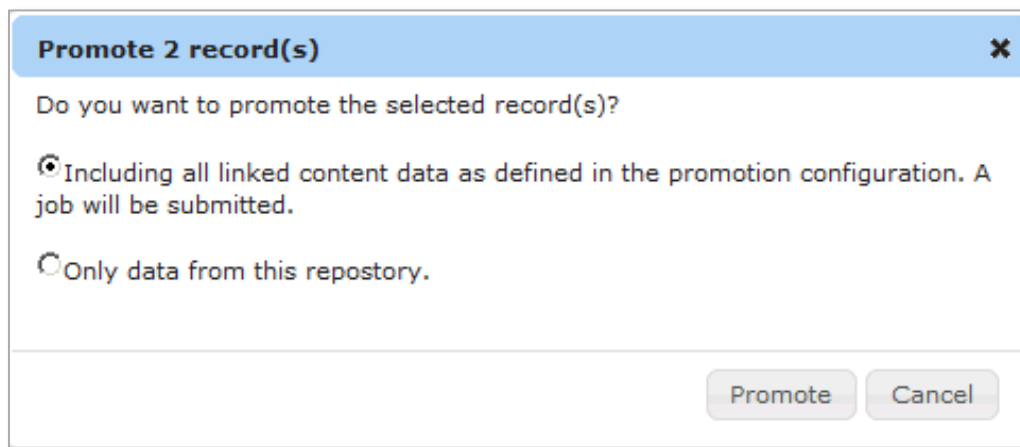
Data that is added or updated in repositories **not** configured for Staging-Production will be “production-ready” when the records are saved.



## Manually Promoting Records

Users can manually promote selected records from a Staging repository to Production using either the EnterWorks New UI or the EnterWorks Classic UI. To manually promote records using the EnterWorks Classic UI:

1. Log into the EnterWorks Classic UI.
2. Open the desired repository and select the record(s) to be promoted, using searches, Saved Sets, or Filters if desired.
3. Open the **Utilities** dropdown list and select **Promote Records**.



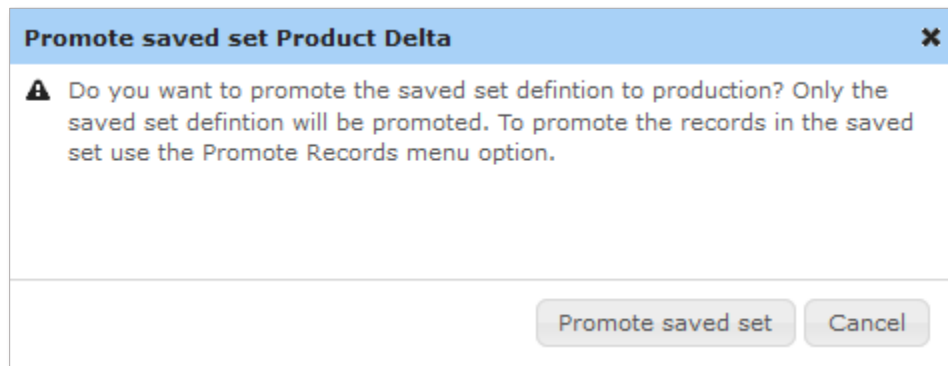
4. To promote the record and all its configured children, select the first option. This will submit as a background job.
5. To promote the selected records only and not any linked children, select the second option. This will immediately copy valid records to production.
6. Click the **Promote** button to submit the promotion request.

**NOTE:** Only the valid selected records will be promoted to Production.

## Manually Promoting a Saved Set Definition

A Saved Sets' definition can be manually promoted from a Staging repository to Production. This action does not promote the records in a Saved Set; it only promotes the definition. To do so:

1. Log into the EnterWorks Classic UI.
2. Open the desired repository, open the SavedSets tab, and select the Saved Set to be promoted.
3. Open the **Utilities** dropdown list and select **Promote Saved Set**. The following dialog will open.

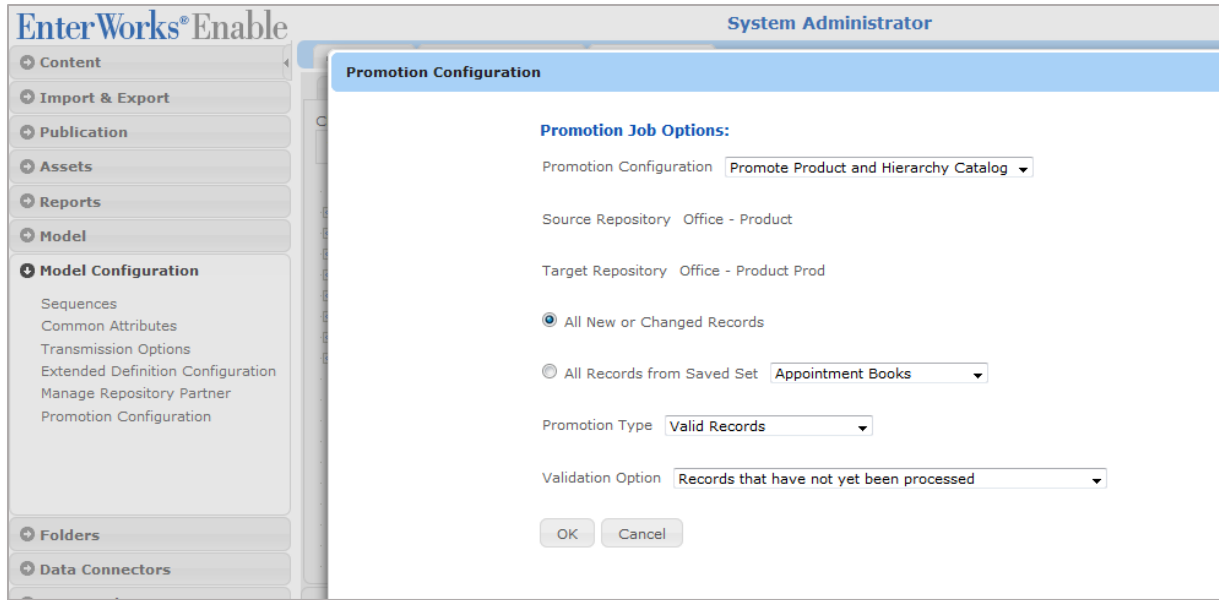


4. Click **Promote saved set**.

## Manually Promoting a Repository

An entire staging repository can be promoted in a single operation by performing the following steps:

1. Expand the **Feature** bar, open **Model Configuration**, and select **Promotion Configuration**.

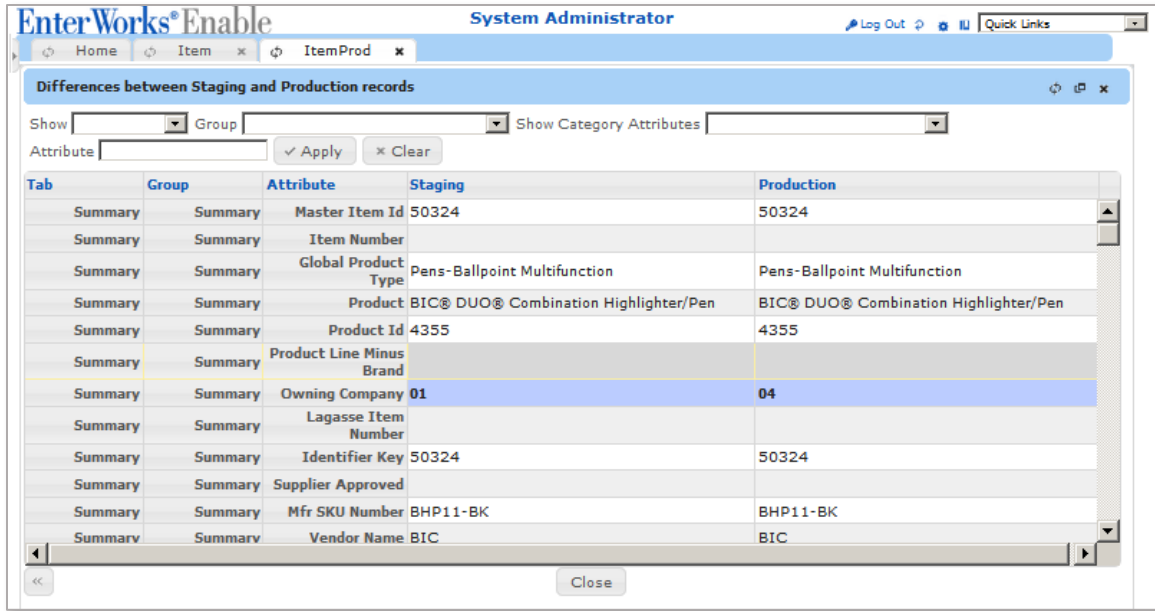


2. Select the **Promotion Configuration** to be used to promote.
3. To promote all new or changed records, select the **All New or Changed Records** radio button.
4. To promote only those records in a Saved Set, select the **All Records from Saved Set** radio button and select the desired Saved Set from the list.
5. Select the desired promotion type.
6. Select the desired validation option.
7. Click **OK**. A background job will be launched that will validate and promote the designated records from the designated repository.

## Review difference from Production

To view differences between the staging record and the production record.

1. Select a single record from listing. From **Utilities** dropdown menu select **Diff Production Record**.
2. The **Differences** viewer will open.



3. The **Differences** viewer will initially show only the attributes that are different between Staging and Production. To show all attributes or duplicate attributes, use the **Show** dropdown list.

## Package Promotions

Records are modified in the staging environment and then moved to production using the promotion process. Promotion can be handled automatically or manually (depending upon the configuration), but records cannot be promoted if there are any validation errors. For a complete description of package promotion, see the Precisely EnterWorks online help at <https://support.precisely.com>.

The package promotion process has several steps:

1. Creation of temporary saved sets for each repository in the package.
2. Validation of the records in the temporary saved sets for each repository.
3. Revision of the temporary saved sets based on package promotion rules (for example, remove any records belonging to packages that have validation errors).
4. Promotion of the records in the temporary saved sets for each repository.
5. Removal of the temporary saved sets (to reduce clutter).

The validation and promotion operations are visible in the **Job Monitor** as individual jobs. The **Scheduled Import Jobs** shows the overall package promotion progress.

## Package Promotion Repository

The Package Promotion repository defines the promotion packages for EnterWorks. It has the following attributes:

Attribute	Description
Display Attributes	List of attributes to display when referencing a record in this repository (instead of primary key)
ID	Unique identifier for each Package element.
Package Dependent	If yes, this indicates that a record in this repository can only be promoted if the entire package containing this record is valid.
Package Level	Level for package. The top level is 1.
Package Name	Name of the promotion package.
Parent Link Relationship Name	Name of the link relationship associating this repository with its parent. This is blank if this is the top level.
Promote Warnings	Records having only Warning validation errors will be promoted for this repository if Yes.
Promotion Attribute	Comma-delimited list of attributes to be updated in each package record from the values of the corresponding work item properties (defined in Promotion Property).
Promotion Configuration Name	Name of promotion configuration if the repository is to be promoted as part of the package promotion processing. This name must match a promotion configuration in the <code>sharedConfig.properties</code> configuration file.
Promotion Property	Comma-delimited list of work item properties to be used to update the corresponding attributes (defined in Promotion Attribute) for each record in the package.
Repository Name	Name of the repository.
Status	Only process if Active. Inactive records will prevent children repositories from being processed.
Update Package Errors	Update the Package Errors attribute in the root repository if Yes. The Package Errors attribute must be defined and have a Validation Rule specifying the attribute must be empty.

## Defining a Data Package

The following steps are used to set up a data package to control the promotion of related records. This functionality is only available to System Administrators.

1. Open the **Feature** bar, open **Content**, open **Scheduled Activities**, and open the **Package Promotions** repository.
2. Click the **Create a New Record (+)** icon. The detail editor for the **Package Promotions** repository will appear:

The screenshot shows the 'Add New Record' interface for 'Package Promotions'. The interface includes a blue header with the title 'Add New Record' and 'Package Promotions'. Below the header, there are two dropdown menus: 'Action' and 'Reports'. The main content area is divided into four tabs: 'Summary', 'Details', 'System', and 'State'. The 'Summary' tab is currently selected and expanded, showing a list of configuration attributes: 'Package Name', 'Repository Name', 'Package Level', 'Status', and 'ID'. The 'Status' field is currently set to 'Active'. On the left side of the form, there is a button labeled 'No Summary Attributes'.

3. On the **Summary** tab, edit the configuration attributes.
  - a. **Package Name:** Enter the name of the package to which this record.

**NOTE:** All package level records for the same package must have the exact same Package Name value.

- b. **Repository Name:** Enter the name of the repository to which this level applies. Each repository can only be listed once for a package.

- c. **Package Level:** The top level repository is level 0. The repositories that link to the top level must be defined as level 1. Repositories that link to the Level 1 repositories must be defined as level 2 and so on. Levels must be consecutive (for example, if there is a level 3, there must be a levels 2, 1, and 0).
  - d. **Status:**
  - e. **ID:**
4. On the **Details** tab, edit the configuration attributes:
- a. If this is not the top level, in the **Parent Link Relationship Name** field, enter the name of the link relationship that connects this repository to its parent. For example, if Level 0 is Product and Level 1 is Item, and the link relationship “Product to Item Staging” defines the relationship between Product and Item, set the field to “Product to Item Staging” (without the quotes).
  - b. If records in this repository should only be promoted if the entire package containing them gets promoted, set **Package Dependent** to “Yes”, otherwise set it to “No”.
  - c. In the **Promotion Configuration Name** field, enter the name of the promotion configuration for this repository. This name should be copied from the `sharedConfig.properties` file.
  - d. If warning validation errors are to be ignored, set the **Promote Warnings** attribute to Yes.
  - e. If attributes other than (or in addition to) the primary key attribute(s) should be used to identify a repository record in the Package Promotion reports, enter the names of the attributes in the **Display Attributes** field (one per line).
  - f. If values from the Package Promotion work item should be used to update attributes in this repository, add the names of the work item properties and the repository attributes to the **Promotion Attributes** table. Whenever a package promotion is initiated, the values from the work item will be set in each package record from this repository before the promotion processing (validation and promotion) begins.
  - g. If a package record from this repository should be flagged with an error when the package promotion fails, set the **Update Package Errors** attribute to Yes.
  - h. **Package Promotion Parent:**
5. On either the Summary or the Details tab, click **Save**.

Repeat the above steps to define all levels for a package.

## Job Scheduling

The EnterWorks Job Scheduling feature allows you to schedule the following types of activities to be run immediately, in the future, or on a recurring basis:

- Imports
- Exports:
  - Repository Exports
  - Database View/DataMart Exports
  - Template Exports
- Promotions

Each type of activity has a pair of repositories that are used to define, trigger, and monitor scheduled instances of the activity, (also called “activity jobs”).

- Scheduled <activity> repository: Each record in this repository defines one scheduled activity, including how often it is to be run, locations of files and templates, pre- or post-processing that is to be performed, and how to respond to any incurred errors.
- Scheduled <activity> Jobs Repository: A job monitor for a scheduled activities. When a scheduled activity is triggered to run, a record in this repository is created that tracks the status of the job.

For instance, Scheduled Imports uses the repositories:

- Scheduled Imports Repository
- Scheduled Imports Jobs Repository

Workflows are used to execute an activity job once it has been triggered.

System configuration settings control management of the activity repositories, for instance, how long job status is to be retained and where activity log files are located.

The following sections describe the repositories and configuration settings for each type of scheduled activity.



## Scheduling Imports

### Scheduled Imports Repository

All scheduled imports are defined in the Scheduled Imports repository. Each record in this repository represents one scheduled import job. Each attribute from this repository is described in the following table.

<b>Attribute</b>	<b>Description</b>
Active Days Only	If this is set to Yes, run the scheduled job on the designated Schedule Days. This only applies if the Schedule Type is Minutes.
Active End Time	If this is a scheduled job of type Minutes, and Active Times Only is set to Yes, jobs will not be run after this time of day. Its format is if HH:MM:SS. Any exports scheduled to be run with a calculated time later than this time will not be run
Active Start Time	If this is a scheduled job of type Minutes, and Active Times Only is set to Yes, jobs will not be begun before this start time. Its format is HH:MM:SS. Jobs with a calculated time earlier than this time will not be run.
Active Times Only	Only run the scheduled job if the calculated Minutes is between the Active Start Time and Active End Time. Applies only when the Schedule Type is Minutes.
Category Attribute Association Name	The name of the Category Attribute Association object to be updated when the Import Type is Cat Attr Assoc.
Code Set Full Replacement	If set to Yes, the Code Set import file fully replaces the existing Code Set. This means that existing nodes not included in the import file will be deleted from the Code Set.
Code Set Name	Name of the Code Set to be updated when the Import Type is Code Set.
Delete Saved Set	If set to Yes, the specified Saved Set should be deleted before adding records for the current import to the saved set.

Attribute	Description
Dependent Keys	<p>List of keys to be passed to each dependent operation. The keys must match attribute names in the dependent repository. The corresponding value in the <code>Dependent Values</code> attribute is then used to set the value of the key attribute when the Job is created. The value can reference attributes from the source job by surrounding the attribute name with double-pipe characters.</p> <p>For example, if an export is dependent on an import and that export should specify the same user, the <code>Dependent Key</code> would be set to “Manager Login” and the value would be “  Manager Login  ”. Conversely, if the same dependent Export needs to be set to the admin user regardless of its initial definition, the <code>Dependent Key</code> would be set to “Manager Login” and the <code>Dependent Values</code> would be set to “admin”.</p>
Dependent Operation	Name of dependent Import, Export, or Promotion to be initiated upon successful completion of this job.
Dependent Operation Immediate	If set to Yes, initiate the dependent operation immediately. Otherwise the operation will be queued and processed by the scheduler, which may take several minutes to start.
Dependent Operation Type	Type of dependent operation: Promotion, Import, or Export.
Dependent Values	<p>Values corresponding to the <code>Dependent Keys</code> that are passed to the dependent operations. Values may reference attributes by surrounding the attribute name with double-pipe characters.</p> <p>For example, to reference the <code>Import Job</code> number, the <code>Dependent Values</code> might contain: “Job  Export Job  ”. If <code>Export Job</code> in the current record is set to 1234, the resulting value will be “Job1234”.</p>
Description	Description of scheduled import. It is strongly recommended that <code>Description</code> is populated with all the details for the import, including any job that launches this job, any pre-processing performed, and what jobs this job launches. If <code>Parameter&lt;x&gt;</code> attributes are used, each should be detailed as to their content and purpose.
Email Body	Override e-mail body for successful jobs. Attributes in the job record can be referenced by surrounding them with double-pipe characters (for example,   Manager Login   is replaced with the value of the <code>Manager Login</code> attribute).
Email Notify Failure	If set to Yes and the import fails to process successfully, send an Import Failed e-mail to the owner.
Email Notify Success	If set to Yes and the import processes successfully, send an Import Successful e-mail to the owner.

Attribute	Description
Email Subject	Subject for success e-mail. May contain references to any other job attribute by surrounding it with double-pipe characters (for example,   Manager Login   is replaced with the value of the attribute Manager Login).
Email To	Email address(es) to which success/failure notifications will be sent. If not defined, the e-mail address(es) associated with the Manager Login user/group will be used.
Error Dependent Keys	Optional keys for arguments to be passed to any Dependent jobs (as properties) when this job fails with an error.
Error Dependent Operation	List of Promotions, Imports, and/or Exports (based on Dependent Operation Type) that should be triggered on successful completion of this one.
Error Dependent Operation Immediate	If set to Yes and this job fails with an error, execute the dependent operation immediately.
Error Dependent Operation Type	List of operation types for each Dependent Operation when this job fails with an error. <ul style="list-style-type: none"> <li>• Export – launch an export job.</li> <li>• Import – launch an import job.</li> <li>• Promotion – launch a promotion job.</li> </ul>
Error Dependent Values	If this job fails with an error, these are the values for Dependent Keys.
Group Lock Action	Group locks allow for multiple concurrent jobs to all attach to the same lock. The lock is not freed until all processes have detached from it. Action to be taken with the lock identified in Group Lock Name: <ul style="list-style-type: none"> <li>• Attach – the import will attach to the specified lock. Processing for the import will begin immediately.</li> <li>• Free – the import will detach from the named lock (presumably set by a previous job that invoked this import) once the import processing has completed. The lock is not completely freed until all attached processes have detached.</li> <li>• Attach And Free – the import will attach to the named lock and begin processing and detach from the named lock once processing has completed.</li> <li>• No Lock – the import will not attach to a group lock.</li> </ul> Group locks will always acquire the designated lock, even if another process has acquired or attached to the same lock. If multiple jobs attach to the same lock via group lock, the lock is not free until all of the processes have freed the lock.
Group Lock Name	Name of the group lock.

Attribute	Description
Ignore Missing File	If an Import is executed and the specified file cannot be found, the Failure e-mail will be suppressed if this flag is set to Yes.
Import Directory Name	The fully qualified path to the directory where the import file is.
Import File Name	Name of the file to be imported.
Import File Sort	Specifies whether the matching files (if Import File Name includes a wildcard character) should be sorted before selecting a file. <ul style="list-style-type: none"> <li>File Ascending – sort file names in ascending order.</li> <li>File Descending – sort file names in descending order.</li> </ul>
Import Language	Specifies the language to be used for the import. If it is empty, the default language is used. If it is set to a non-default language, the multi-language fields for this language are updated.
Import Name	Logical name for the Import.
Import Now	If set to Yes, the import file is to be processed immediately. This flag is immediately set to No once a record for this import has been created in the Scheduled Import Jobs repository. This flag will be set to Yes any time the import is to be processed outside of the scheduled time. The immediate processing of the import file will not alter the next scheduled date and time. For example, if the import is scheduled to be processed nightly at 1:30 a.m. and a user requests an immediate import at 11:00 p.m. by setting this flag to Yes, the file will be processed at 11 p.m. and then again at 1:30 a.m.
Import Option Name	The name of the import option to be applied to this import. The code set Import Options contains a complete list of available options. The description for each code identifies what the option does and what values can be specified.
Import Option Value	The value for the corresponding import option to be applied to this import.
Import Priority	This setting controls the order in which jobs are run. The lower the number, the higher the priority, for example: jobs with priority 1 are run before jobs with priority 5.
Import Type	Identifies the type of import: <ul style="list-style-type: none"> <li>Template – the import uses an Import Template.</li> <li>Repository – the import uses a View Mapping for a specific repository.</li> <li>Custom – the import only performs the assigned pre-processing module.</li> </ul>

Attribute	Description
Inactive Records	<p>Identifies action to be taken with Inactive records when the Import Type is Template and it is configured as a full replacement import (which will flag records not present in the file as being inactive):</p> <ul style="list-style-type: none"> <li>• No Action – do not do anything.</li> <li>• Delete – delete inactive records from Staging and Production after the import has completed.</li> <li>• Reactivate – reactivate inactive records if they are present in the import file.</li> </ul>
Inactive View	Name of the snapshot table view for the target repository. This view is used when processing inactive records in a full replacement template import.
Include Timestamp	If set to Yes, include the date and time in the Import_<job>.log. This can be helpful in troubleshooting pre-processing for an import as it shows how long processing steps took (providing the appropriate messages are being reported).
Last Import Datetime	Date and time of when this import was last initiated (scheduled or manual).
Last Scheduled Import Datetime	Date and time of when scheduled import was last processed.
Lock Action	<p>The action to be taken with the lock identified in Lock Name:</p> <ul style="list-style-type: none"> <li>• Acquire – the import will wait until the named lock is acquired. Processing for the import will not begin until then.</li> <li>• Free – the import will free the named lock (presumably set by a previous job that invoked this import) once the import processing has completed.</li> <li>• Acquire And Free – the import will wait until the named lock is acquired and will free the named lock once processing has completed.</li> <li>• No Lock – the import will not wait for any lock before processing.</li> </ul> <p>Locks will only be acquired if they are currently free. If the designated lock has been attached to by multiple jobs using the Group Lock, the lock does not become free until all of those jobs have freed the lock.</p>

Attribute	Description
Lock Name	Name of the lock to acquire or free (depending upon Lock Action). If more than one import shares the same lock, then they will be processed serially. The name may contain references to other properties by surrounding each property name with double-pipe characters. For example, to ensure only one instance of an import is processed at a time for each user, the Lock Name can be set to: "Import  Manager Login  ". This will produce a lock name of "Importjdoe" for John Doe and "Importjsmith" for John Smith. If the referenced lock is associated with the Group Lock in other jobs, the lock will not be considered free until all the Group Lock jobs have detached from the lock.
Manager Login	Login of the EnterWorks user or EnterWorks role assigned to this import. This user (or group of users if a role is specified) will receive e-mails generated for processing imports.
Move Import File	If set to Yes, the import file is moved to the designated import directory and renamed to include a date/time stamp. If this is set to No, the import file is left in its original location and processed from there.
Next Import Datetime	Date and time of next scheduled import in the format "mm/dd/yyyy hh:mm:ss". EnterWorks will initiate the processing of the import file when this value is in the past. The value is updated to the next scheduled time (using the Import Period) as soon as the processing commences.
Parameter1-5	Five attributes that can serve as parameters when jobs are connected through the Dependent Operation attributes. They are not used directly by the Import processing but can be referenced in other attributes (for example,   Parameter2  ).
Preprocess Class	Fully-qualified path, identifying the Preprocess class to be invoked for this import. It must be a subclass of <code>com.enterworks.services.BaseCustomProcessFile</code>
Preprocess File	If set to Yes, the import file must be preprocessed using the designated class and arguments.
Preprocess Keys	Argument keys identifying the Preprocess Argument Values to be passed to the designated preprocess module.
Preprocess Values	Values for corresponding Preprocess Argument Keys to be passed to the designated preprocess module. Values can contain references to properties from this job by surrounding the property name with double-pipe characters. For example, to pass the Import Job, the value would be set to "  Import Job  ".
Repository Name	Name of target repository.

Attribute	Description
Repository Timeout	The maximum number of minutes allowed for processing a Repository Import. This timeout specifies how long the Scheduled Import processing will wait for the import to complete processing. If a timeout is reached, the job is flagged with an error and the designated user or group is notified. The actual job may still complete successfully, but any dependent jobs will not be triggered.
Repository View Mapping	The name of the mapping view for Repository Import. Mappings are either manually defined in the EnterWorks Classic UI or by manually performing an import using the EnterWorks New UI and selecting the option to Save Mapping.
Run As Manager Login	If set to Yes, run the import using the login designated in Manager Login. When enabled, the Manager Login must be set to a user and not a group. This option ensures the import is subject to the security of the designated login and also identifies the records being created/updated by that login (vs. system).
Saved Set	If this is specified, it is the name of the Saved Set to be created.
Schedule Days	Identifies the days of the week if Schedule Type is Days of Week.
Schedule Exclude Dates Code Set	If set to a code set listing dates as code values, the scheduling logic will skip to the next scheduled date if the calculated date is one of the dates in the code set.
Schedule Period	Number of time units (based on Schedule Type) after Last Scheduled Date before the next scheduled import is initiated. For example, an import that is to be processed daily could be set to type DAYS and have a period of 1 or be set to type MINUTES and have a period of 1440. The scheduled imports are not affected by any manually initiated imports.
Schedule Type	<p>Type of schedule:</p> <ul style="list-style-type: none"> <li>• MINUTES – the Schedule Period identifies the number of minutes before re-invoking the import.</li> <li>• DAYS – the Schedule Period identifies the number of days before re-invoking the import.</li> <li>• DAYS OF WEEK – specific days of the week on which the import is to be executed (which is specified by the attribute Schedule Days).</li> <li>• WEEKS – the Schedule Period identifies the number of weeks before re-invoking the import.</li> <li>• MONTHS – the Schedule Period identifies the number of months before re-invoking the import.</li> </ul>

Attribute	Description
Scheduled Folder	Logical folder in which the scheduled job is assigned for organization purposes. If set and the Scheduled Folder Hierarchy is selected in the repository list view, this import job can be found by clicking a node in the Scheduled Folder Hierarchy.
Scheduled Folder Sequence	Used to control the order of scheduled jobs in the Scheduled Folders.
Scheduled Import	Set to Yes if the import is to be scheduled. Each time the scheduled import is processed, the Next Import Datetime is updated based on the Schedule Type and Scheduled Period along with the previous value for Next Import Datetime. For example, if the Schedule Type is set to DAYS OF WEEK and Scheduled Days is set to Monday, Wednesday, and Friday and the Next Import Datetime is set to 10/3/2014 02:00:00 (2:00 am on Friday, October 3 <sup>rd</sup> ), the Next Import Datetime attribute will be updated to 10/6/2014 02:00:00 (2:00 am on Monday, October 6 <sup>th</sup> ).
Skip File Check	If set to Yes, skip checking for and moving a file. Requires Preprocess File = Yes and Preprocess Class is defined. The assigned class is responsible for ensuring a file is created.
Skip If Busy	Skip this Import if it is scheduled and there is already an active job for the same import. This ensures only one scheduled import job is running at any given time.
Source FTP	FTP Server name if Source Type is FTP
Source FTP Delete File From Server	If set to Yes, delete the file from the FTP server. Its default value is Yes.
Source FTP File Name	Name of source file if Source Type is FTP. The file name may contain wildcards ('*') which may result in multiple files being retrieved. If multiple files are expected, the Import File Name should also have wild cards and the Scheduled Import should be configured to call itself as a dependent operation to ensure each file is processed.
Source FTP List Gaps	Number of gaps in the data returned by the LIST command for the FTP server between the start of each line and the name of the file. This is used by EnterWorks to know how to extract the name of the file.
Source FTP Path	Fully qualified directory path on the FTP server if Source Type is FTP.
Source FTP Port	FTP Server port (21 default, 22 SFTP)
Source FTP User	FTP Server user name if Source Type is FTP



Attribute	Description
Source FTP User Password	FTP Server user password if Source Type is FTP
Source S3 Access Key	Amazon S3 Access Key if Source Type is S3. If not defined, the property named <code>amazon.s3.AWSAccessKey</code> in the <code>&lt;drive&gt;:\Enterworks\EPX\bin\conf\Enterworks.properties</code> file is used.
Source S3 Bucket Name	Name of the Amazon S3 Bucket containing the file to be downloaded if Source Type is S3.
Source S3 Delete File From Server	If set to Yes and Source Type is S3, delete the file from the Amazon S3 Server after it is downloaded.
Source S3 Endpoint	(Optional) The endpoint if Source Type is S3. If set to Use Config, then the Configuration repository property <code>AWS.S3.Endpoint</code> is used (if it is defined).
Source S3 File Name	Name of the file on the Amazon S3 server to be downloaded if Source Type is S3.
Source S3 Is Role Based Auth	If Role Based Auth is false, the Access Key and Secret Key are used.
Source S3 Path	The Amazon S3 directory (under the designated bucket name) containing the file to be imported if Source Type is S3.
Source S3 Region	(Optional) The region if Source Type is S3. If Use Config selected, then the Configuration repository property <code>AWS.S3.Region</code> is used (if defined).
Source S3 Secret Key	Amazon S3 Secret Key if Source Type is S3. If not defined, the property named <code>amazon.s3.AWSSecretKey</code> in the <code>&lt;drive&gt;:\Enterworks\EPX\bin\conf\Enterworks.properties</code> file is used.
Source Type	Type of Source for Import; either File or FTP. The default is File.
Status	Status of the import record: <ul style="list-style-type: none"> <li>• Active – the record is active.</li> <li>• Retired/Inactive – the record is not active.</li> </ul>
Template Name	Name of the Import Template if Import Type is Template.
Template Timeout (Minutes)	Number of minutes before the processing for a Template import is flagged as timing out. When the import times out, an e-mail will be sent to the designated user or group. The import template processing may still finish successfully, but any dependent jobs will not be initiated.

## Scheduled Import Jobs Repository

The Scheduled Import Jobs repository contains records representing the currently active or recently completed scheduled import jobs. When an import is initiated, the details for that import are copied from the Scheduled Imports Repository to a new record in the Scheduled Import Jobs Repository. This record is subsequently updated to reflect the current status of the import.

In addition to the attributes define in the [Scheduled Imports Repository](#), the Scheduled Import Jobs repository also has the attributes listed in the table below.

Attribute	Description
# Created	The total number of records created
# Deleted	The total number of records deleted
# Errors	The total number of errors encountered during the import processing.
# Processed	The total number of records processed
# Updated	The total number of existing records that were updated
Import Errors	Lists the details for the cause of the import to fail (when the Import Status is Error).
Import Job	Unique number for the import job.
Import Status	Status of current or last import. Possible values include: <ul style="list-style-type: none"> <li>Aborted – the import has been aborted by a user.</li> <li>New – the import is new.</li> <li>Processing – the import file is being processed. While in this state, no additional import can be initiated for this import record.</li> <li>Completed – import processing has completed successfully.</li> <li>Error – the import failed. The Import Status must be changed to New or Completed for it to be processed again (if scheduled).</li> </ul>
Job Log File	Name of the Scheduled Import job log file, for example: Import_<job>.log.
Last Submitted By	Login ID of last user who submitted a file for this target.
System Log Directory	Contains the directory for where the EnterWorks import logs are created.
System Log File	Name of the EnterWorks import log file.

## Scheduled Import Configuration Properties

Management of the Scheduled Import feature is defined by configuration settings in the EPX configuration file `Enterworks.properties`, as listed below.

Property	Description
<code>import.baseDirectory</code>	Full path to the directory where the Scheduled Import files will be placed (for example, <code>&lt;drive&gt;:\Enterworks\shared\Imports</code> ). If the EnterWorks is configured in a distributed environment, a network directory path should be specified.
<code>import.expireActiveDays</code>	Number of days before import jobs that still show "Processing" are deleted from the Scheduled Import Jobs repository.
<code>import.expireCompletedDays</code>	Number of days before import jobs that have completed or failed are deleted from the Scheduled Import Jobs repository.
<code>import.logDirectory</code>	Full path to the directory where the EnterWorks import logs are placed (for example, <code>&lt;drive&gt;:\Enterworks\logs\EnableServer\import</code> ).

## Scheduling Exports

Scheduled Exports must be pre-defined and configured using the following high-level steps:

1. Depending upon the type of export being created, create an Export Template or User Preference, or define the SQL Query.
2. Create a new record in the Scheduled Exports repository, specifying the mapping or template created in the first step.

The details for these steps are provided in the following sections.

## Scheduled Export Repository

All scheduled exports are defined in the Scheduled Exports repository. Each record in this repository represents one scheduled export job.

The attributes of the Scheduled Export Repository are listed in the following table, in alphabetical order.

Attribute	Description
Active Days Only	If set to Yes, run the scheduled job on the designated Schedule Days. This only applies if the Schedule Type is Minutes.
Active End Time	The format for this value is HH:MM:SS. If Active Times Only is set to Yes and the job is of type Minutes, no jobs will be run after this time of day. Any jobs with a calculated time later than this time will not be executed.
Active Start Time	The format for this value is HH:MM:SS. If Active Times Only is set to Yes and the job is of type Minutes, this is the daily start time for executing jobs. Jobs with a calculated time earlier than this will not be executed.
Active Times Only	Only run the scheduled job if the calculated Minutes is between the Active Start Time and Active End Time. Applies only when the Schedule Type is Minutes.
Additional Conditions	Additional SQL conditions to filter Delta records. Alias 'v' can be used to reference the view. For example: v.Portal_Status in ('Submitted - Request Pending', 'Reviewed').
Additional Conditions (AND/OR)	This allows the user to select if all or any one of the conditions (if multiple) should be used to filter records.
Compress File	If this is set to Yes, compress the export file into a zip file if.
Correction File Type	Type of correction file to generate when Export Type is Validate. <ul style="list-style-type: none"> <li>• 0 – None – place validation errors in the Validation job log file.</li> <li>• 1 – One file for each attribute with errors.</li> <li>• 2 – Single file for all attributes with errors.</li> </ul>
Created Attribute Name	Name of attribute representing the creation date for the record.
Custom Source File Name	Sets the file name for a custom export for the registered post-processing class (since no file was generated by EnterWorks). It is the responsibility of the post-process class to actually generate the file.

Attribute	Description
Delta Export	<ul style="list-style-type: none"> <li>• &lt;blank&gt; - No filtering of records.</li> <li>• Add – Include only new records since last export.</li> <li>• Change - Include only changed records since last export.</li> <li>• Add or Change - Include only new or changed records since last export.</li> <li>• Full – Include all records subject to Additional Conditions.</li> </ul>
Delta Export Offset	<p>Number of minutes to offset the Delta export end time to give concurrent updates time to complete. The value should be set based on the EnterWorks batch size and database performance. For example, if the batch size is 200 and it may take the SQL Server database 30 seconds to save all 200 records, an offset of 1 or 2 minutes should be ample.</p>
Dependent Keys	<p>List of keys to be passed to each dependent operation. The keys must match attribute names in the dependent repository. The corresponding value in the Dependent Values attribute is then used to set the value of the key attribute when the Job is created. The value can reference attributes from the source job by surrounding the attribute name with double-pipe characters. For example, if an export is dependent on an import and that export should specify the same user, the Dependent Key would be set to Manager Login and the value would be “  Manager Login  ”. Conversely, if the same dependent Export needs to be set to the admin user regardless of its initial definition, the Dependent Key would be set to Manager Login and the Dependent Values set to admin.</p>
Dependent Operation	<p>Name of dependent Import, Export, or Promotion to be initiated upon successful completion of this job.</p>
Dependent Operation Immediate	<p>If this is set to Yes, initiate the dependent operation immediately. Otherwise the operation will be queued and processed by the scheduler.</p>
Dependent Operation Type	<p>Type of dependent operation: Promotion, Import, or Export.</p>
Dependent Values	<p>Values corresponding to the Dependent Keys that are passed to the dependent operations. Values may reference attributes by surround the attribute name with double-pipe characters. For example, to reference the Import Job number, the Dependent Values might contain: “Job  Export Job  ”. If Export Job in the current record is set to 1234, the resulting value will be “Job1234”</p>

Attribute	Description
Description	Optional description for export. It is strongly recommended that the Description be populated with all the details for the export, including any job that launches this job, any pre-processing performed, and what jobs this job launches. If Parameter<x> attributes are used, each should be detailed as to their content and purpose.
Email Attach Logs	If this is set to Yes, attach the log files to the status e-mail.
Email Body	Override e-mail body for successful jobs. Attributes in the job record can be referenced by surrounding them with double-pipe characters (for example,   Manager Login   is replaced with the value of the Manager Login attribute).
Email Notify Failure	If this is set to Yes and the export fails to process successfully, send an Export Failed e-mail to the owner.
Email Notify Success	If this is set to Yes and the export processes successfully, send an Export Successful e-mail to the owner.
Email Subject	Subject for success e-mail. May contain references to any other job attribute by surrounding it with double-pipe characters (for example,   Manager Login   is replaced with the value of the attribute Manager Login).
Email To	Email address(es) to which success/failure notifications will be sent. If not defined, the e-mail address(es) associated with the Manager Login user/group will be used.
Empty File Action	Identifies the action to be taken if the export file is empty. <ul style="list-style-type: none"> <li>• Continue - continue processing.</li> <li>• Error - treat empty file as failed export.</li> <li>• Stop - stop processing (no dependent operations).</li> </ul>
Error Dependent Keys	Optional keys for arguments to be passed to any Dependent jobs (as properties) when this job fails with an error.
Error Dependent Operation	List of Promotions, Imports, and/or Exports (based on Dependent Operation Type) that should be triggered on successful completion of this one.
Error Dependent Operation Immediate	If this is set to Yes and this job fails with an error, execute the dependent operation immediately.
Error Dependent Operation Type	List of operation types for each Dependent Operation when this job fails with an error. <ul style="list-style-type: none"> <li>• Export – launch an export job.</li> <li>• Import – launch an import job.</li> <li>• Promotion – launch a promotion job.</li> </ul>

Attribute	Description
Error Dependent Values	Optional values for dependent keys when this job fails with an error.
Export Encoding	File Encoding for Export (UTF-8, Windows-1251, etc.).
Export Execution Datetime	Date and time at which this export was last executed.
Export File Name	Name of file to be generated by the export. This attribute can include references to other attributes by surrounding the attribute name with double-pipe characters. For example, to include the job number in the export file name, the file name could be “MyFile_  Export Job  .csv”.
Export Include Null Values	If this is set to Yes, include null values in XML Export.
Export Language	Language to generate for export.
Export Name	Name of the export.
Export Now	If this is set to Yes, process the export immediately.
Export Priority	Relative priority for the export determines the order of exports ready to be processed. The lower the number, the higher the priority.
Export Template	If Export Type is Template, this is the name of the Export Template.
Export Type	Type of export: <ul style="list-style-type: none"> <li>• Repository – export of a single repository using the designated User Preference.</li> <li>• View – SQL export against the EPIM database.</li> <li>• DataMart – SQL export against the Data Mart database</li> <li>• Export Template – export using the designated Export Template</li> <li>• Custom – relies on the PostProcessing module to generate a file.</li> <li>• Validate – performs a validation on the designated repository and saved set.</li> </ul>

Attribute	Description
Group Lock Action	<p>Group locks allow for multiple concurrent jobs to all attach to the same lock. The lock is not freed until all processes have detached from it.</p> <p>Action to be taken with the lock identified in Group Lock Name:</p> <ul style="list-style-type: none"> <li>• Attach – the import will attach to the specified lock. Processing for the export will begin immediately.</li> <li>• Free – the import will detach from the named lock (presumably set by a previous job that invoked this import) once the export processing has completed. The lock is not completely freed until all attached processes have detached.</li> <li>• Attach And Free – the export will attach to the named lock and begin processing and detach from the named lock once processing has completed.</li> <li>• No Lock – the export will not attach to a group lock.</li> </ul> <p>Group locks will always acquire the designated lock, even if another process has acquired or attached to the same lock. If multiple jobs attach to the same lock via Group Lock, the lock is not free until all of the processes have freed the lock.</p>
Group Lock Name	Name of the group lock.
Include Timestamp	If this is set to Yes, include date and time in Export_<job>.log file if Yes. This can be helpful in troubleshooting post-processing for an export as it shows how long processing steps took (providing the appropriate messages are being reported).
Last Export Datetime	Date and time of when the export was last executed.
Last Export Update Datetime	Date and time of when last Delta export was executed.
Last Scheduled Export Datetime	Date and time the export was last scheduled.
Linked Repository Name	Optional list of repositories included in the Export Template.
Linked Repository Saved Set	Saved set to be used for each corresponding linked repository to identify which linked records are to be included.
Linked Repository Search Config	List of predefined searches on the linked repositories to be applied to limit the records in the export.
Linked Search Repository Name	For more information on Linked Search Repository Name, contact your EnterWorks account representative.



Attribute	Description
Lock Action	<p>Action to be taken with the lock identified in Lock Name:</p> <ul style="list-style-type: none"> <li>• Acquire – the import will wait until the named lock is acquired. Processing for the import will not begin until then.</li> <li>• Free – the import will free the named lock (presumably set by a previous job that invoked this import) once the import processing has completed.</li> <li>• Acquire And Free – the import will wait until the named lock is acquired and will free the named lock once processing has completed.</li> <li>• No Lock – the import will not wait for any lock before processing.</li> </ul> <p>Locks will only be acquired if they are currently free. If the designated lock has been attached to by multiple jobs using the Group Lock, the lock does not become free until all of those jobs have freed the lock.</p>
Lock Name	<p>Name of the lock to acquire or free (depending upon Lock Action). If more than one import shares the same lock, then they will be processed serially. The name may contain references to other properties by surrounding each property name with double-pipe characters.</p> <p>For example, to ensure only one instance of an import is processed at a time for each user, the Lock Name can be set to: “Import  Manager Login ”. This will produce a lock name of “Importjdoe” for John Doe and “Importjsmith” for John Smith.</p>
Manager Login	<p>Login ID of the user or the name of the EnterWorks role who will receive notification e-mails for the export.</p>
Next Export Datetime	<p>If Scheduled Export is set to Yes, this is the date and time the export will be processed.</p>
Output to File	<p>If this is Yes and Export Type is Export Template, generate an export.</p>
Output to Table	<p>If this is Yes and Export Type is Export Template, save the publication table.</p>
Parameter1-5	<p>These are five attributes that can serve as parameters when jobs are connected through the Dependent Operation attributes. They are not used directly by the export processing but can be referenced in other attributes.</p>
Postprocess Class	<p>Fully-qualified path, identifying the Postprocess class to be invoked for this import. It must be a subclass of <code>com.enterworks.services.BaseCustomProcessFile</code></p>
Postprocess File	<p>If this is Yes, after the export file is generated, it must be postprocessed using the designated class and arguments.</p>

Attribute	Description
Postprocess Keys	Argument keys identifying the Postprocess Argument Values to be passed to the designated postprocess module.
Postprocess Values	Values for the corresponding Postprocess Argument Keys to be passed to the designated postprocess module. Values can contain references to properties from this job by surrounding the property name with double-pipe characters. For example, to pass the Export Job, the value would be set to “  Export Job  ”.
Publication As Of Date	If Export Type is Export Template and the current date and time are not to be used, this is the effective date.
Publication Hierarchy	If Export Type is Export Template this is the Hierarchy used by the Export Template.
Report Validation Level	Specifies what validation level to report.
Repository Format	If Export Type is Repository type, this is the format of the file, for example: CSV, XLS, XML, or DB.
Repository Name	If Export Type is Repository or Export Template, this is the name of the repository.
Repository Preference	If Export Type is Repository type, this is the Preference to use.
Repository Timeout	If Export Type is Repository type, this is the number of minutes before an export has timed out.
Return Correction File	If set to Yes, return the correction file instead of the validation log for the export file.
Root Repository Saved Sets	If Export Type is Export Template this is the Saved Sets for the root repository.
Root Repository Search Configs	List of predefined searches on the root repository to be applied to limit the records in the export.
Run As Manager Login	If this is Yes, run the export using the login designated in Manager Login. When this is enabled, the Manager Login must be set to a user and not a group.
Saved Set	If Export Type is Repository type, this is the Saved Set.
Schedule Days	If Schedule Type is Days of Week, this identifies the days of the week.
Schedule Exclude Dates Code Set	If set to a Code Set that lists dates as code values, the scheduling logic will skip to the next scheduled date if the calculated date is one of the dates in the Code Set.
Schedule Period	If Scheduled Export is Yes, this is the number of minutes between exports.

Attribute	Description
Schedule Type	Type of schedule: <ul style="list-style-type: none"> <li>• MINUTES – the Schedule Period identifies the number of minutes before re-invoking the export.</li> <li>• DAYS – the Schedule Period identifies the number of days before re-invoking the export.</li> <li>• DAYS OF WEEK – specific days of the week on which the export is to be executed.</li> <li>• WEEKS – the Schedule Period identifies the number of weeks before re-invoking the export.</li> <li>• MONTHS – the Schedule Period identifies the number of months before re-invoking the export.</li> </ul>
Scheduled Export	If this is set to Yes, the export is scheduled.
Scheduled Folder	Logical folder in which the scheduled job is assigned for organization purposes. If this is set and the Scheduled Folder Hierarchy is selected in the repository list view, this import job can be found by clicking a node in the Scheduled Folder Hierarchy.
Scheduled Folder Sequence	Used to control the order of scheduled jobs in the Scheduled Folders.
Separate Files for Category Attributes	If this is set to Yes, separate files for will be generated for category attributes.
Skip if Busy	If this is set to Yes, skip the scheduled export if the same named export is already busy processing. This can be used to prevent a backlog of exports if an export has been delayed or takes longer than the schedule period to complete.
Status	Status of export (Active, Inactive, Retired).
Target Email	If Target Type is Email, this is the target email address.
Target Email Attach Logs	If this is set to Yes, attach log files to the target email.
Target FTP	If Target Type is FTP, this is the FTP server name or IP address.
Target FTP User	FTP user name.
Target FTP User Password	FTP user password.
Target Overwrite File	If this is set to Yes, overwrite the output file instead of appending the new output to the existing file.
Target Path	If Export Type is File Directory, this is the location for export file.
Target S3 Access Key	If Target Type is S3, this is the Amazon S3 access key.

Attribute	Description
Target S3 Bucket Name	If Target Type is S3, this is the name of the Amazon S3 bucket containing the directory into which the export file is to be uploaded.
Target S3 Endpoint	(Optional) If Target Type is S3, this is the target endpoint. If Use Config is specified, then the configuration repository property <code>AWS.S3.Endpoint</code> is used (if it is defined).
Target S3 Is Role Based Auth	If Target Type is S3, this indicates if the export is using role based authentication.
Target S3 Region	(Optional) If Target Type is S3, if Use Config is selected, the configuration repository property <code>AWS.S3.Region</code> is used (if it is defined).
Target S3 Secret Key	If Target Type is S3, this is the Amazon S3 secret key.
Target Type	Type of target (Email, FTP, File, S3).
Transmission Option	Identifies the transmission option to be invoked upon completion of the export file generation.
Validate Category/Dynamic Attributes	If set to Yes, validate the Category/Dynamic Attributes.
View Format	<p>If Export Type is View or Data Mart, this is the format:</p> <ul style="list-style-type: none"> <li>• TAB – values separated by tab characters</li> <li>• COMMA – values separated by commas. If a value contains a comma or double-quote, the entire value is surrounded by double quotes and any double-quotes in the value are escaped by another double quote. For example, the value: <code>Monsters, Inc. Size=3" x 4"</code> is converted to: <code>"Monsters, Inc. Size=3"" x 4""</code></li> <li>• ~ - values separated by the tilde character</li> <li>• * - values separated by the asterisk character</li> <li>•   - values separated by the pipe character</li> </ul>
View Name	If Export Type is View or Data Mart, and View SQL is not defined, this is the name of the View.

Attribute	Description
View SQL	<p>If Export Type is View or Data Mart, this is the SQL Query. The value can contain references to other attributes by surrounding them with double-pipe characters.</p> <p>For example, in a repository named “RepositoryView”, to filter data by the user who initiated the import, the SQL would be:  SELECT * FROM RepositoryView WHERE Updated_By = '  Manager Login  '</p> <p>If the View export is a Delta export, then the snapshot table view columns Created and/or Snapshot_Last_Updated can be compared to the '[DELTA_DATETIME]' keyword which is converted to the date and time the export was last executed.</p> <p>For example, to create a Delta (Add) export for the repository named “RepositoryView”, the SQL would be:  SELECT * from RepositoryView WHERE Created &gt; '[DELTA_DATETIME]'</p> <p>To create a Delta (Add or Update) export, the SQL would be:  SELECT * from RepositoryView WHERE Created &gt; '[DELTA_DATETIME]' OR Snapshot_Last_Updated &gt; '[DELTA_DATETIME]'</p>

**Scheduled Export Jobs Repository**

The Scheduled Export Jobs repository contains records representing the currently active or recently completed export jobs. When an export is initiated, the details of that export are copied from the Scheduled Exports Repository to a new record in the Scheduled Export Jobs Repository. This record is subsequently updated to reflect the current status of the export.

Note that when the repository is opened in a Repository View, the default Preference does not show all attributes.

In addition to the attributes define in the Scheduled Exports repository, the Scheduled Export Jobs repository has the attributes listed in the table below.

Attribute	Description
# Errors	Total number of records with errors.
# Exported	Total number of records exported.
# Processed	Total number of records for the export file that were processed.

Attribute	Description
Download Link	URL to download the export file. Requires setting up a virtual directory in IIS.
Export Errors	Error messages encountered during export if there is a failure.
Export Job	Unique number for the Export job.
Export Status	Status of the Export job <ul style="list-style-type: none"> <li>• New – the job has been created but a work item has not been assigned (this is either due to this job being launched as a Dependent job with the Dependent Operation Immediate set to No, or a configuration error preventing a work item from being created for this job).</li> <li>• Queued – the job has been queued and is waiting for a lock to be freed.</li> <li>• Processing – the job is processing.</li> <li>• Completed – the job completed successfully.</li> <li>• Error – the job failed with an error. This could be a timeout, abort by user, or some unexpected processing error.</li> </ul>
Submitted By	Login ID of the user who submitted the Export Job.
System Log Directory	The directory containing the EnterWorks system log file for the export. Do not set this value if Export Type is View, Data Mart, or Custom.
System Log File	The name of the EnterWorks system log file for the export. Do not set this value if Export Type is View, Data Mart, or Custom.
Target FTP Port	FTP Server port (21 default, 22 SFTP).

## Scheduled Export Configuration Properties


Management of the Scheduled Export feature is defined by configuration settings in the EPX configuration file `Enterworks.properties`, as listed below.

Property	Description
<code>export.directoryName</code>	Full path to the directory where the Scheduled Export files will be, such as: D:/Enterworks/shared/Exports
<code>export.epimDirectoryName</code>	EnterWorks directory for export , such as: D:/Enterworks/shared/export

Property	Description
<code>export.expireActiveDays</code>	Number of days before export jobs that still show “Processing” are deleted from the Scheduled Export Jobs repository.
<code>export.expireCompletedDays</code>	Number of days before export jobs that have completed or failed are deleted from the Scheduled Export Jobs repository.
<code>export.logDirectoryName</code>	Full path to the directory containing the EnterWorks export log , such as: D:/Enterworks/logs/EnableServer/export
<code>export.publicationDirectoryName</code>	Full path to the directory containing the EnterWorks publication, such as: D:/Enterworks/shared/publication
<code>export.publicationLogDirectoryName</code>	Full path to the directory containing the EnterWorks publication log, such as: D:/Enterworks/logs/EnableServer/publish

## Creating a Scheduled Repository export

A user with authority creates a scheduled Repository export by performing the following steps:

1. Log into EnterWorks Classic.
2. Open the repository for which a Repository Export is to be generated.
3. Select or create the User Preference view that defines the attributes to be included in the export as columns.
4. In the Feature bar, open the **Content** tab, open the **Scheduled Exports** folder, and open the **Scheduled Exports** repository.
5. Click the  icon to create a new record. The detail editor for a new Scheduled Export record appears.

The screenshot shows a web interface for adding a new record. The title bar reads 'Add New record'. On the left, there is a sidebar with the heading 'Scheduled Exports' and a small square icon below it. The main content area has a top navigation bar with 'Action' and 'Reports' dropdown menus. Below this is a tabbed interface with tabs for 'Summary', 'Export Details', 'Export Target', 'Schedule', 'Export Status', and 'State'. The 'Summary' tab is selected and expanded, showing several input fields: 'Export Name' (a text box), 'Description' (a large text area), 'Manager Login' (a text box), and 'Export Type' (a dropdown menu). In the top right corner of the main content area, there is a dropdown menu with the text 'NO Preference (S)'.

6. Enter a name for the export.
7. Enter the login ID of the user who will receive e-mails for this export in the **Manager Login** field.
8. Select **Repository** in the **Export Type** drop-down selection list.
9. Click the **Export Details** sub-tab
10. Select the repository in the **Repository Name** drop-down selection list.
11. Optionally enter the name of the **Saved Set** to export a subset of the records.
12. Select the output format in the **Repository Format** drop-down selection list.
13. Enter the target details (file/e-mail/FTP) in the appropriate fields:



The screenshot shows the 'Add New record' form for 'Scheduled Exports'. The 'Export Target' sub-tab is active. The form contains the following fields and sections:

- Target Type:** A dropdown menu.
- Target Path:** A text input field.
- Target Email:** A list of email addresses with 'Add', 'Edit', and 'Delete' buttons below it.
- Target FTP:** A text input field.
- Target FTP User:** A text input field.
- Target FTP User Password:** A text input field.

14. Enter the name of the export file to be created.

15. To overwrite an existing file, set **Target Overwrite File** to “Yes”.

The screenshot shows the 'Add New record' form for 'Scheduled Exports'. The 'Schedule' sub-tab is active. The form contains the following fields and sections:

- Target FTP:** A text input field.
- Target FTP User:** A text input field.
- Target FTP User Password:** A text input field.
- Export File Name:** A text input field containing 'exportFile.csv'.
- Target Overwrite File:** A dropdown menu set to 'Yes'.
- Compress File:** A dropdown menu set to 'No'.

16. Click the **Schedule** sub-tab.

17. Set the **Schedule Export** field to “Yes”.

18. Set the **Schedule Type** to the desired type and the **Schedule Period** to the desired interval (based on the selected type). For example, to produce an export nightly, set the Schedule Type to Days and the Period to 1. To produce an export every 2 weeks, set the Schedule Type to Weeks and the Scheduled Period to 2. If the Schedule Type is set to Days Of Week, the Schedule Period is not used and instead the days listed in Schedule Days determine which days of the week the export will be triggered:


The screenshot shows a web interface for adding a new record. At the top, there is a blue header with the text 'Add New record'. Below this, there are two dropdown menus: 'Action' and 'Reports'. A horizontal tab bar contains six tabs: 'Summary', 'Export Details', 'Export Target', 'Schedule', 'Export Status', and 'State'. The 'Schedule' tab is currently selected and expanded. It contains four sections, each with a label and a text input field: 'Scheduled Export' with the value 'Yes', 'Schedule Type' with the value 'Days', 'Schedule Period' with the value '1', and 'Schedule Days' which is currently empty.

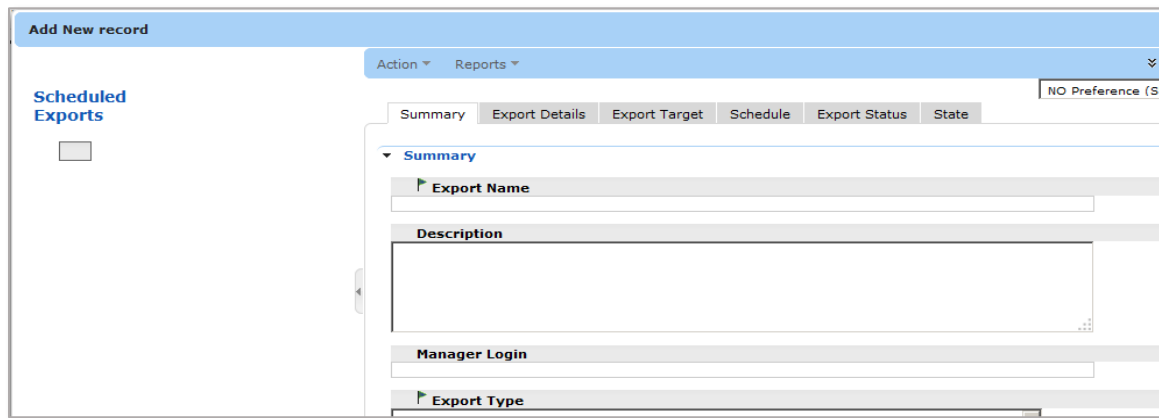
19. Set the **Next Export Datetime** field to the date and time when the export is to run. For Schedule Types other than Minutes, the time portion of this field will determine the time of day the export will be triggered. If the date and time are in the past, the export will be initiated immediately and the Next Export Datetime will be used in conjunction with the Schedule Type and Schedule Period to calculate the next time the export should be initiated.
20. Click **Save**. When the current date and time passes the Next Export Datetime, the Repository export will be initiated. Once triggered, the Next Export Datetime will be updated using the Schedule Type and Schedule Period.

## Creating a Scheduled View/DataMart Export

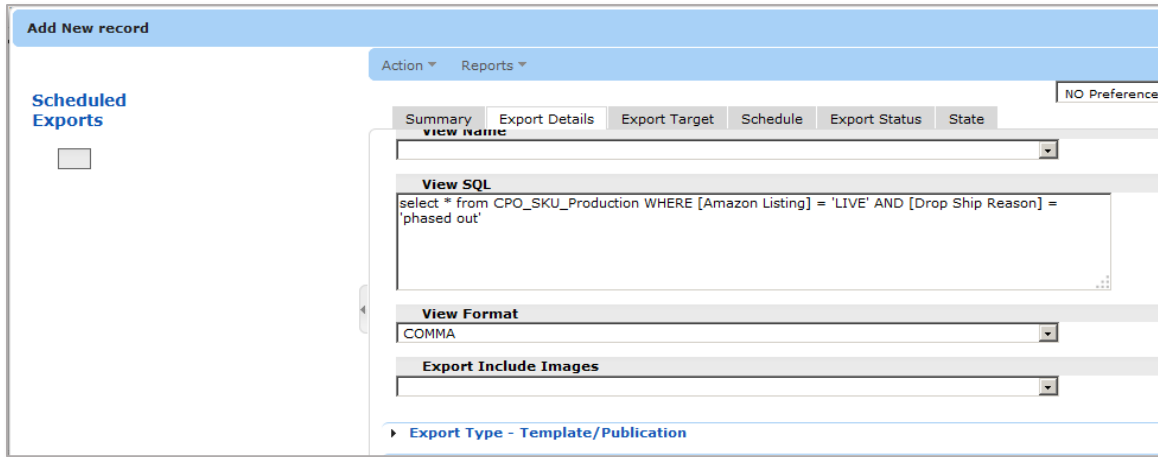
A user with authority creates a scheduled View or DataMart export by performing the following steps:

1. Log into EnterWorks Classic.

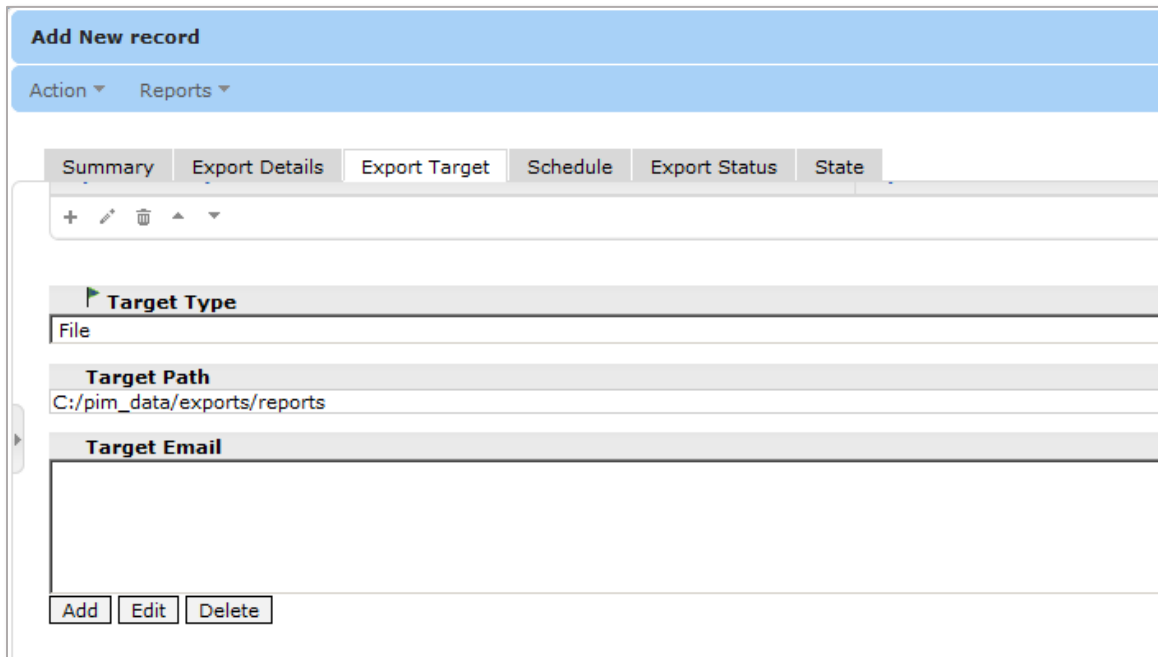
2. In the Feature bar, open the **Content** tab, open the **Scheduled Exports** folder, and open the **Scheduled Exports** repository.
3. Click the  icon to create a new record. The detail editor for a new Scheduled Export record appears.



4. Enter a **Name** for the scheduled export.
5. Enter the login ID of the user who will receive e-mails for this export in the **Manager Login** field.
6. In the Export Type drop-down selection list, select View for a SQL View or DataMart for Data Mart. A View export will be against the EPIM snapshot table views. A DataMart export will be against the Data Mart database.
7. Click the `Export` sub-tab.
8. Enter the SELECT SQL query into the **View SQL** field.
9. Select TAB, COMMA or ~ in the **View Format** field to specify the character to be used as a delimiter.



10. Enter the target details (file/e-mail/FTP/S3) in the appropriate fields:



11. Enter the name of the export file to be created.

12. To overwrite an existing file, set **Target Overwrite** File to “Yes”.

**Add New record**

Action ▾ Reports ▾

Summary
Export Details
Export Target
Schedule
Export Status
State

Add
Edit
Delete

**Target FTP**

**Target FTP User**

**Target FTP User Password**

**Export File Name**

**Target Overwrite File**

**Compress File**

13. Click the **Schedule** sub-tab.

14. Set the **Schedule Export** field to “Yes”.

15. Set the **Schedule Type** to the desired type and the **Schedule Period** to the desired interval (based on the selected type). For example, to produce an export nightly, set the Schedule Type to Days and the Period to 1. To produce an export for every 2 weeks, set the Export Type to Weeks and the Scheduled Period to 2. If the Schedule Type is set to Days Of Week, the Schedule Period is not used and instead the days listed in Schedule Days determine which days of the week the export will be triggered:

**Add New record**

Action ▾ Reports ▾

Summary Export Details Export Target **Schedule** Export Status State

▼ **Schedule**

**Scheduled Export**  
Yes

**Schedule Type**  
Days Of Week

**Schedule Period**

**Schedule Days**  
Monday  
Wednesday  
Friday

Add Edit Delete

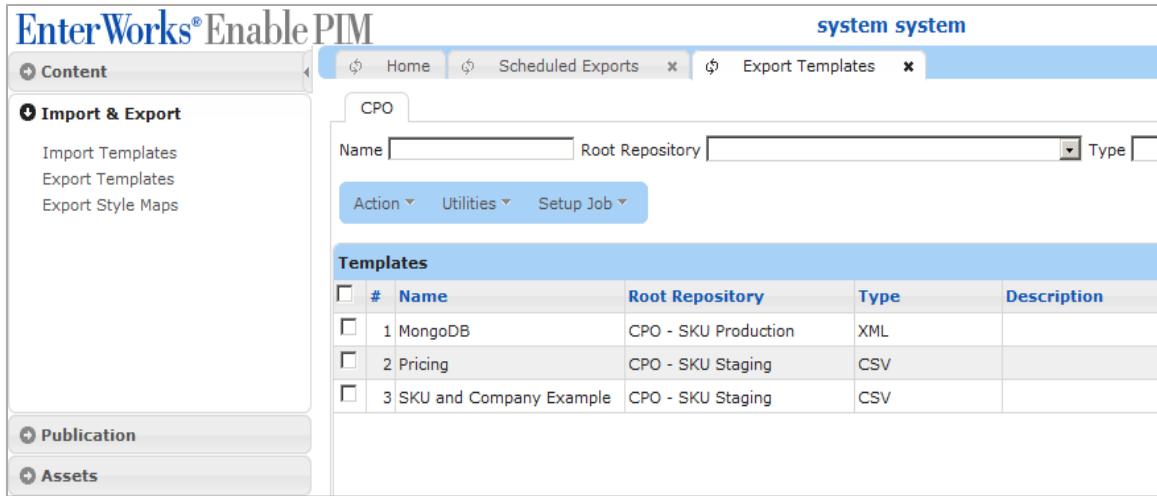
**Schedule Exclude Dates Code Set**

16. Set the **Next Export Datetime** field to the date and time when the export is to run. For Schedule Types other than Minutes, the time portion of this field will determine the time of day the export will be triggered. If the date and time are in the past, the export will be initiated immediately and the Next Export Datetime will be used in conjunction with the Schedule Type and Schedule Period to calculate the next time the export should be initiated.
17. Click **Save**. When the current date and time passes the Next Export Datetime, the Repository export will be initiated. Once triggered, the Next Export Datetime will be updated using the Schedule Type and Schedule Period.

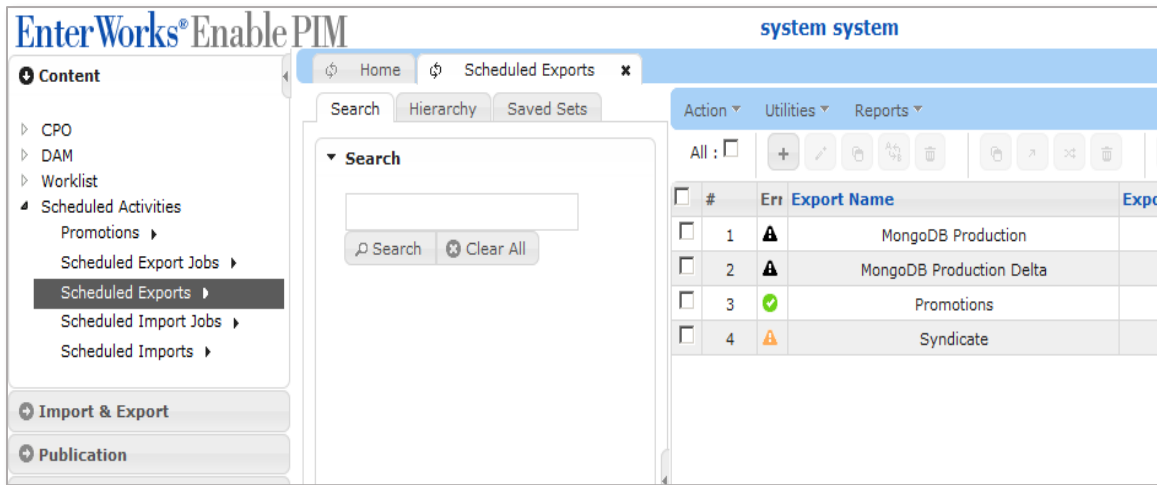
## Creating a Scheduled Template Export


A user with authority creates a Scheduled Export Template by performing the following steps:

1. Log into EnterWorks Classic.
2. If the Export Template does not already exist, in the Feature bar, in the **Import & Export** tab, select **Export Templates** and create a new Export Template, noting its name:



- In the Feature bar, open the **Content** tab, open the **Scheduled Exports** folder, and open the **Scheduled Exports** repository.



- Click the  icon to create a new record. The detail editor for a new Scheduled Export record appears.

The screenshot shows a web application interface for adding a new record. The main header is blue and contains the text "Add New record" on the left and "Action" and "Reports" dropdown menus on the right. Below the header, there are several tabs: "Summary", "Export Details", "Export Target", "Schedule", "Export Status", and "State". The "Summary" tab is currently selected and expanded, showing a form with the following fields: "Export Name" (with a green arrow icon), "Description" (a large text area), "Manager Login", and "Export Type" (with a green arrow icon). On the left side of the form, there is a sidebar with the text "Scheduled Exports" and a small square icon below it.

5. Enter a name for the export.
6. Enter the login ID of the user who will receive e-mails for this export in the **Manager Login** field.
7. Select Template in the **Export Type** drop-down selection list.
8. Click the **Export Details** sub-tab.
9. Select the root repository in the **Repository** drop-down selection list:



**Edit 1 record(s)**

Action ▾ Reports ▾

Summary Export Details Export Target Schedule Export Status State

Export Postprocess Options

Export Details

Repository Name  
CPO\_SKU\_Production

Repository Timeout (Minutes)  
30

ImageType

10. Enter the name of the defined Export Template:

**Edit 1 record(s)**

Action ▾ Reports ▾

Summary Export Details Export Target Schedule Export Status State

Export Type - View/DataMart

Export Type - Template/Publication

Export Template  
MongoDB

Publication As Of Date

Root Repository Saved Sets

Add Edit Delete

11. Set **Output to File** to Yes.

12. Click on the **Export Target** sub-tab.

13. Enter the target details (file/e-mail/FTP/S3) in the appropriate fields:

### Add New record

Action ▾ Reports ▾

Summary | Export Details | **Export Target** | Schedule | Export Status | State

+ ✎ 🗑️ ▲ ▼

**Target Type**  
File

**Target Path**  
C:/pim\_data/exports/reports

**Target Email**

Add Edit Delete

14. Enter the name of the export file to be created.
15. To overwrite an existing file, set **Target Overwrite File** to Yes.
16. Click on the **Schedule** sub-tab.
17. Enter the target details (file/e-mail/FTP/S3) in the appropriate fields.

**Edit 1 record(s)**

Action ▾ Reports ▾

Summary Export Details Export Target Schedule Export Status State

**Target Type**  
File

**Target Path**  
C:/pim\_data/exports/all

**Target Email**  
brian.zupke@enterworks.com

Add Edit Delete

**Target FTP**

**Target FTP User**

18. Enter the name of the export file to be created.

19. To overwrite an existing file, set **Target Overwrite File** to “Yes”.

**Edit 1 record(s)**

Action ▾ Reports ▾

Summary

Export Details

Export Target

Schedule

Export Status

State

**Target FTP**

**Target FTP User**

**Target FTP User Password**

**Export File Name**

MongoDB\_||CURRENT\_DATETIME-yyyyMMdd\_HHmm||.xml

**Target Overwrite File**

Yes

**Compress File**

Yes

20. Click the **Export Schedule** sub-tab.

21. Set the **Schedule Export** field to “Yes”.

22. Set the **Schedule Type** to the desired type and the **Schedule Period** to the desired interval (based on the selected type). For example, to produce an export nightly, set the Schedule Type to Days and the Period to 1. To produce an export for every 2 weeks, set the Export Type to Weeks and the Scheduled Period to 2. If the Schedule Type is set to Days Of Week, the Schedule Period is not used and instead the days listed in Schedule Days determine which days of the week the export will be triggered:

The screenshot shows a web interface for adding a new record. At the top, there is a blue header with the text "Add New record". Below this, there are two dropdown menus: "Action" and "Reports". A navigation bar contains several tabs: "Summary", "Export Details", "Export Target", "Schedule" (which is currently selected), "Export Status", and "State". The "Schedule" section is expanded, showing several fields: "Scheduled Export" with a value of "Yes", "Schedule Type" with a value of "Days Of Week", "Schedule Period" (empty), and "Schedule Days" with a list containing "Monday", "Wednesday", and "Friday". Below the "Schedule Days" list are three buttons: "Add", "Edit", and "Delete". At the bottom of the form, there is a section labeled "Schedule Exclude Dates Code Set".

23. Set the **Next Export Datetime** field to the date and time when the export is to run. For Schedule Types other than Minutes, the time portion of this field will determine the time of day the export will be triggered. If the date and time are in the past, the export will be initiated immediately and the Next Export Datetime will be used in conjunction with the Schedule Type and Schedule Period to calculate the next time the export should be initiated.
24. Click **Save**. When the current date and time passes the Next Export Datetime, the Repository export will be initiated. Once triggered, the Next Export Datetime will be updated using the Schedule Type and Schedule Period.

## Implementing Pre- and Post-Processing of Scheduled Imports and Exports

The Scheduled Imports feature provides the option to pre-process files before they are imported. The Scheduled Exports feature provides the option to post-process files after they have been exported. In both cases, the actual processing is handled by a Java class that is an extension of the `BaseCustomProcessFile` class found in the `Services.jar` file (or in an application-specific JAR file).

When a processing block is being configured within a Scheduled Import or Scheduled Export, details on the function and configurable parameters for the processing block are shown. The

blocks are organized under Exports (`com.enterworks.services.exports.<class>`) or Imports (`com.enterworks.services.imports.<class>`) based on how they are predominately used, but some modules can be used for either pre-processing or post-processing.

The table below lists available pre-defined pre/post-processing blocks.

Classpath	Description
<code>com.enterworks.services.exports.CreateUpdateFile</code>	Generates an update file, setting the desired set of columns to specific values for each primary key in the source file. The resulting file can be submitted to an import. This provides a way to update all records that were included in an export (for example, to indicate the records have been syndicated).

Classpath	Description
<pre>com.enterworks.services.exports.GenerateFixedPositionFile</pre>	<p>Creates a fixed position file using one or more export files as a source and one or more mapping files to define the format of the output file. One format file is defined for each format record appearing in the file. If multiple files are defined, records in each file must be related by a common key and sorted on that same key. This allows the file processing to complete the file merge in a single pass. The order of the records is determined by the order the file mappings are defined. If there is a one-to-one mapping of the different records, then the same file can be used as the source for each format. The mapping files must be comma-delimited files with the following columns:</p> <ul style="list-style-type: none"> <li>• <code>Description</code> - user-description for field (not used in processing)</li> <li>• <code>Type</code> - datatype for field: <ul style="list-style-type: none"> <li>○ <code>N</code> - numeric with leading zeros for padding.</li> <li>○ <code>A</code> - alphanumeric with trailing spaces for padding.</li> </ul> </li> <li>• <code>Length</code> - number of character columns for field</li> <li>• <code>Start</code> - starting column position with first column being 1</li> <li>• <code>End</code> - ending column position with first column being 1</li> <li>• <code>Value</code> - value for field or export file column reference (denoted by double-pipe characters). A single space can be denoted with: <code>[SPACE]</code>.</li> </ul> <p>Each mapping file is validated, ensuring the <code>Start</code> and <code>End</code> positions match the accumulated length of each field.</p>

Classpath	Description
com.enterworks.services.exports.ProcessTaxonomyTemplateExport	Generates a Taxonomy Template in XLSX format using the <code>exportTemplate</code> for global attributes and the category-specific attributes for the designated <code>taxonomyNode</code> . They are shaded if they are mapped in the designated Taxonomy Node in the designated Publication Template.
com.enterworks.services.exports.ProcessTaxonomyTemplateNodeList	<p>Reads the taxonomy template entries in the file and kicks off a Template Taxonomy export for each one, setting:</p> <ul style="list-style-type: none"> <li>• Parameter1 to the publication name</li> <li>• Parameter2 to the taxonomy node</li> <li>• Parameter3 to the name of the saved set for each job in the form: <code>'TaxonomyTemplate_&lt;taxonomyNode&gt;_&lt;datetime&gt;'</code></li> <li>• Parameter4 to the batch number.</li> </ul> <p>Each launched job should use the <code>ProcessTaxonomyExportTemplate</code> post-processing block to generate the corresponding template. The collection of template jobs can be consolidated into a single file using the <code>TaxonomyTemplateExportZip</code> post-processing block.</p>
com.enterworks.services.exports.RemoveHeaderRow	Removes the first line of the CSV file.
com.enterworks.services.exports.SplitCsvFile	Splits a CSV file into multiple parts, each no larger than the specified maximum number of records. Each part will be named <code>&lt;baseFileName&gt;_&lt;partNumber&gt;.csv</code> . The collection of files will be placed in a ZIP file which is returned.



Classpath	Description
<pre>com.enterworks.services.exports.SplitDeltaExportIntoMultipleParts</pre>	<p>Splits a delta export into multiple export jobs, each including up to the maximum records per job. This can be used in situations where the target system cannot process large files. The delta date and time is specified and optionally additional conditions. The processing uses this information to generate separate saved sets for each batch of the specified size and then launches Scheduled Export jobs using the designated Scheduled Export as a template for each job, updating only the Parameter1 attribute to the name of the saved set for the part and Parameter2 to the part number, and optionally Parameter3 - Parameter5 with any additional data. This allows the target Scheduled Export to have full control over the file naming convention and how the saved set is used (for example, in 'Saved Set' or 'Root Repository Saved Sets' attributes. This processing can be used in conjunction with any export type and format that can operate on a saved set.</p>
<pre>com.enterworks.services.exports.TaxonomyTemplateExportZip</pre>	<p>Packages the TaxonomyTemplateExport files for the same batch into a single .zip file. The ProcessTaxonomyTemplateNodeList processing block launches separate TaxonomyTemplateExport jobs for each taxonomy node listed in the seed file, each being identified as being part of the same batch in Parameter4. This post-processing block collects the files from each job for the same batch and packages them in a single .zip file.</p>

Classpath	Description
com.enterworks.services.imports.ConcatenateCSVFiles	Concatenate a set of files in the designated source directory that match the designated file name pattern, using the header from the designated import template for all files. If the sources files are not identical in structure and the import template contains a superset of attributes, some columns may be padded in each appended file. To prevent the attributes from being cleared, the <code>keepRepoValues</code> import option should be set to true.
com.enterworks.services.imports.CopyImportFile	Copies the import file using the designated file name, then processes the original file so that it can be processed by a second import job (for example, for another repository or different pre-processing).
com.enterworks.services.imports.EncodeFile	Converts the import file from one encoding to another.
com.enterworks.services.imports.EnterworksFileDiff	Generates a delta file using the current file and the previous one that was processed. The current and previous files must be CSV format. Requires the EnterworksDiff utility be installed and configured on the Enable server. The generated delta file will include the column <code>il_modification_status</code> , indicating whether the record is new, updated, or removed. If there is no previous file, the current file will be processed in full without the <code>il_modification_status</code> column added. If new records need a specific status then the corresponding status attribute should have that default value.

Classpath	Description
<pre>com.enterworks.services.imports.HorizontalToVerticalAttValUomFileFormat</pre>	<p>Converts a csv file containing multiple attributes as key-value-uoms into (several) vertical files containing a separate line for each triplet. No more than 500k lines will be saved in each target file, using the naming convention: <i>vertical_&lt;fileNumber&gt;_&lt;sourceFilename&gt;</i>. For example, consider an input file containing the following columns:  ITEM_ID, MFR_PART, STATUS, GROUP_1, ATTRIBUTE_NAME_1, VALUE_1, UOM_1, DIFFERENTIATOR_1, GROUP_2, ATTRIBUTE_NAME_2, VALUE_2, UOM_2, DIFFERENTIATOR_2, ...</p> <p>This will be converted into the multiple rows, with one row per attribute, with the following headers:  ITEM_ID, ATTRIBUTE_NAME, VALUE, UOM  Any global attributes (MFG_PART, STATUS) and extra columns (GROUP_*, DIFFERENTIATOR_*) are ignored.</p> <p>Note: This class returns the source filename. It does not return the vertical files. Separate jobs must be run to process the generated files.</p>
<pre>com.enterworks.services.imports.ImportCustomCodeSets</pre>	<p>Imports updates to existing single or multiple code sets from a file. If a single code set is imported, the expected columns are the same that are required when importing a code set through the UI. If multiple code sets are imported, the first column must be the code set name and all codes for that code set must be consecutive. For multiple code sets, all options apply to each code set and the file type must be csv. Each code set must already be defined in EnterWorks - the import will fail if the code set does not exist.</p>

Classpath	Description
com.enterworks.services.imports.InitiateSaveAndSendForSavedSet	<p>Initiates a 'Save and Send' work item on the designated workflow and starting point for the designated saved set and the specified properties. Several reserved words can be specified for the property values:</p> <ul style="list-style-type: none"> <li>• %savedSetId% - indicates to use the ID for the saved set identified by the savedSetName property.</li> <li>• %userId% - use the ID for the user identified by the userName property.</li> <li>• %repositoryId% - use the ID for the repository identified by the repositoryName property</li> </ul>
com.enterworks.services.imports.PreProcessAddFields	Adds columns and values to the import file before loading.
com.enterworks.services.imports.PreProcessAddHeader	Adds a header line to the CSV import file before loading.
com.enterworks.services.imports.PreProcessConcatenateColumns	Performs concatenations of data to specific columns within an import template. A formula expression of other columns within the template can be used. Assumes that all columns already exist and does not create/remove any columns.
com.enterworks.services.imports.PreProcessXLSXAddFields	Preprocesses a .xlsx file containing with possible multiple header rows. Adds the designated columns and their values to the file to facilitate batch processing of the file. Generates a new .xlsx file for import into an EnterWorks repository.
com.enterworks.services.imports.ProcessImagePackage	Processes a single file or a zip file containing one or more image files. If the submitted file has the .csv extension, it is passed on for import processing by EPIM. If the submitted file has the zip extension, the contents of the zip file are processed. Any valid image files are copied to the designated image directory. If the submitted file is a valid image file, it is copied to the designated image directory.

Classpath	Description
<code>com.enterworks.services.imports.ProcessMultiRepositoryFile</code>	<p>Splits a multi-repository comma-delimited CSV export file into separate import files based on the Import Template definitions referenced by the designated Scheduled Imports. Duplicate rows (providing they are consecutive in the file) are removed as well as rows containing no values. Jobs for each separate scheduled import are launched by this module. The main file contains only those columns in the Import Template assigned to the Scheduled Import launching this pre-processing.</p>
<code>com.enterworks.services.imports.SplitImportFile</code>	<p>Splits the import file into two parts: The first part is processed and the second part is staged in the designated target directory.</p>

Classpath	Description
<pre>com.enterworks.services.imports.SplitKeyValueUomTriplexFile</pre>	<p>Preprocess import file containing dynamic attributes in key/value pairs or optionally in key/value/UOM triplets. Files may contain explicit attribute names or pairs/triplets of columns that are numbered consecutively for each pair/triplet.</p> <p>When a file is processed, the contents of each record are split into pre-defined parts as defined in the designated import templates and each file is loaded separately. The first part is loaded by this import and subsequent parts are loaded by dependent imports that do not require pre-processing.</p> <p>If consecutive files contain the same primary key, the values from those lines are combined into a single update (split amongst the defined parts). This allows for vertical files where each row contains the primary key and a single key/value pair or key/value/UOM triplet and multiple rows are for the same repository record.</p> <p>Except for the last part, any empty rows for a part are filtered out since they will not make any changes to the target record, reducing overall import processing time. All records are included in the last part as it should be the only one that is validated but this requires the parts to be daisy-chained together to ensure it is truly the final part that is loaded.</p> <p>Each part import template can have up to 1022 attributes, including the primary key.</p>
<pre>com.enterworks.services.imports.TransferFiles</pre>	<p>This class will move all files from source to target directory passed to the pre-processing module that match the specified file patterns (allow up to 20 to be specified as separate arguments for the module) using the asterisk (*) as the wildcard indicator.</p>

Classpath	Description
com.enterworks.services.imports.TransformFile	Transforms a .csv or .xlsx file into an .xlsx or .csv file containing either: <ul style="list-style-type: none"> <li>• The columns that match the designated import template.</li> <li>• Only the valid and transformed columns from the import file.</li> </ul> Optionally, it validates designated columns for required or specific values, and rejects a row if the values are empty or do not match.
com.enterworks.services.imports.UncompressZipFile	Decompresses zip file before processing.

To configure a Scheduled Import or Export with a pre/post-processing block:

1. Open the Scheduled Import or Scheduled Export repository.
2. Open the record for the Import or Export.
3. Open the **Import Details** or **Export Details** tab, and open the **Import Preprocess Options** or **Export PostProcess Options** sub-tab.
4. Set **Preprocess File** or **Postprocess File** to “Yes”.
5. Enter the full class path for the processing block class and click the **calculate** button on the **Pre-process Class** or **Post-process Class** field.

Action ▾ Reports ▾

NO Preferen

Summary Import Details Schedule Import Status State

▼ **Import Type - Code Set**

**Code Set Name**

**Code Set Full Replacement**

No

▼ **Import Preprocess Options**

**Preprocess File**

Yes

**Preprocess Class**

com.enterworks.services.imports.PreProcessImageLinksFile

**Preprocess Keys**

delimiter,sourceEncoding,targetEncoding,businessKey0,businessKey1,businessKey2,businessKey3,businessKey4,idAttribute,repositoryName

- The define arguments window will open showing a description for what the block does along with what arguments can be set and the current values.



**Define Pre-Processing Arguments**

<b>Processing class:</b>	com.enterworks.services.imports.PreProcessImageLinksFile	
<b>Description</b>	Preprocesses a generic DAM Link file that contains one or more business keys, the file name, and the image context, priority, and caption. The business keys are replaced with a single ID value.	
<b>Argument</b>	<b>Value</b>	<b>Description</b>
delimiter	<input type="text"/>	Delimiter for file ( TAB COMMA) (default is COMMA)
sourceEncoding	Windows-1252	source encoding for files (UTF-8 default)
targetEncoding	UTF-8	target encoding for files (UTF-8 default)
businessKey0	Manufacturer Name	Name of the 1st business key for the repository.
businessKey1	<input type="text"/>	Name of the 2nd business key for the repository.
businessKey2	<input type="text"/>	Name of the 3rd business key for the repository.
businessKey3	<input type="text"/>	Name of the 4th business key for the repository.
businessKey4	<input type="text"/>	Name of the 5th business key for the repository.
idAttribute	Manufacturer Id	Name of the ID attribute that uniquely identifies a single record
repositoryName	PIM_Manufacturer_Staging	Name of repository for which the assets are being linked

7. The argument values can be changed and saved by clicking **Update Attributes**.

## Pre/Post-Processing Block

Each processing block class must implement the `processFile` method. This method is called when there is an import or export file to be processed:

```
String processFile (String <directory_name>, String <file_name>,
HashMap <args>, HashMap <inactive_records>, TreeMap
<primary_key>, StringBuffer <msgs>)
```

Argument	Data Type	Description
<directory_name>	String	Fully-qualified path to the directory containing the file to be processed. The file to be returned must also be placed in this same directory.
<file_name>	String	Name of the file to be processed.
<args>	HashMap	Map of any pre/post processing arguments defined in the Scheduled Import/Export.

Argument	Data Type	Description
<i>&lt;inactive_records&gt;</i>	HashMap	Map containing the primary keys of any records in the repository having a Status of Inactive. This is only set for imports and only if the Inactive Records flag is set to Reactivate.
<i>&lt;primary_key&gt;</i>	TreeMap	The primary key for the repository.
<i>&lt;msgs&gt;</i>	StringBuffer	Medium for returning error messages to be displayed with the job.

The method must return either the name of the processed file or null if the processing block failed.

If the processing block class has configurable arguments, there are three methods that must be implemented:

- `String getDescription()`
  - Returns a detailed description of what the processing block does.
- `void defineArguments()`
  - Builds the list of arguments that can be configured. Each argument is defined by calling the method: `addArgument`.
- `void addArgument(String <arg>, String <description>)`
  - Adds an argument to the list of arguments/properties that can be set for the class in the Scheduled Import or Scheduled Export record:

Argument	Data Type	Description
<i>&lt;arg&gt;</i>	String	Name of argument. This name will be used to retrieve the actual value for the argument. Each defined argument must be uniquely named
<i>&lt;description&gt;</i>	String	Detailed description of the argument. It should include a list of possible values or a range of values, any default values, and any other information.

The `BaseCustomProcessFile` class has a set of methods that help minimize the amount of coding required in a processing block class:

- `void clearBadDate(HashMap <parsed_line>, String <column>)`

- Clears the date value if it is not 10 characters (in mm/dd/yyyy format) or is an invalid date (for example, 00/00/0000).
- `void closeInput(BufferedReader <buffered_reader>)`
  - Closes the opened CSV or TXT file.
- `void closeOutput(PrintWriter <output>)`
  - Closes the opened CSV or TXT file.
- `void convertToBoolean(HashMap <parsed_line>, String <column>)`
  - Converts the values “Y” or “Yes” to 1 and everything else to 0 for the designated column.
- `boolean doesFileExist(String <directory_name>, String <file_name>)`
  - Returns true if the specified file in the specified directory exists.
- `void dropLeadingZeros(HashMap <parsed_line>, String <column>)`
  - Removes leading zeros from each value containing them.
- `ArrayList extractFiles(String <directory_name>, String <file_name>, String <file_encoding>, StringBuffer <msgs>)`
  - Extracts the contents of a zip file and returns a list of unzipped files.
- `void freeQuery(DBQuery <database_query>)`
  - Frees the query connection that was previously obtained with `getQuery()`
- `String[] getHeaderForImportTemplate(String <import_template_name>)`
  - Returns a list of columns based on the mappings in the designated import template.
- `String getHeaderForImportTemplateAsCsvString(String <import_template_name>, String <delimiter>)`
  - Returns a delimited list of columns based on the mappings in the designated import template.
- `String getJobNumber()`

- Retrieves the identification number of the job being processed. `select`
- `HashMap<String, String> getMapForHeader(String[] <header>)`
  - Returns a map of columns based on the list of columns for the header.
- `DBQuery getQuery()`
  - Retrieves a query connection that can subsequently be used to query the EPIM database.
- `PrintWriter getReport()`
  - Retrieves the `PrintWriter` object that is configured to generate the report for the job. Any calls on this object will update that report.
- `void insertDecimal(HashMap <parsed_line>, String <column>, int <decimal_position>)`
  - Inserts a decimal point character in a value at the designated number of digits from the right.
- `void logDebug(String <message>)`
  - Generates a message in the log file if debug logging is enabled (`debugEnabled=true`) in the `Enterworks.properties` file.
- `void logReport(String <message>)`
  - Adds a line to the import or export report file.
- `void logError(String <message>)`
  - Adds a line to the EPX BIC log file.
- `void logError(StringBuffer msgs, String <message>)`
  - Adds a line to the EPX BIC log file and to the `Errors` attribute for the Scheduled Import Job or Scheduled Export Job record.
- `BufferedReader newInput(String <directory_name>, String <file_name>, String <character_set>)`
  - Opens a CSV or TXT file for reading.
- `PrintWriter newOutput(String <directory_name>, String <file_name>, String <encoding>)`

- Opens a CSV or TXT file for writing.
- `void outputHeaderLine(PrintWriter <output>, String[] <columns>, String <delimiter>)`
  - Outputs the header line with each column separated using the specified delimiter.
- `void outputHeaderLine(PrintWriter <output>, String[] <columns>, char <delimiter>, char <text_qualifier>)`
  - Outputs the header line with each column separated using the specified delimiter and text qualifier (for when column names include the delimiter or text qualifier character).
- `void outputParsedLine(PrintWriter <output>, HashMap <parsed_line>, String[] <columns>, char <delimiter>, char <text_qualifier>)`
  - Outputs a line using the parsed values and the designated delimiter and text qualifier.
- `void outputParsedLine(PrintWriter <output>, HashMap <parsed_line>, String[] <columns>, String <delimiter>)`
  - Outputs a line using the parsed values and the designated delimiter.
- `String[] parseHeader(String <header_line>, String <delimiter>)`
  - Parses the header line using the designated delimiter. If the delimiter is a comma, then special processing is done for commas and quotes embedded in the header names.
- `String[] parseHeader(String <header_line>, char <delimiter>, char <text_qualifier>)`
  - Parses the header line using the designated delimiter. Uses the designated textQualifier to handle values that contain the delimiter or the text qualifier. Assumes the embedded text qualifier is escaped with the same character. For example, if the delimiter is a comma and the text qualifier is a double quote, then the value: `,"3" x 4"`, Rough Cut" would be stored as: `3" x 4"`, Rough Cut.
- `HashMap parseLine(String <line>, String[] <header>, String <delimiter>)`

- Parses a line from the file using the defined header and delimiter. Returns a HashMap where each key matches a column name and its value is the corresponding value from the file.
- HashMap `parseLine(String line, String[] <header>, String <delimiter>, boolean <trim_white_space>)`
  - Parses a line from the file using the defined header and delimiter. Returns a HashMap where each key matches a column name and its value is the corresponding value from the file. Trims white space from values if `<trim_white_space>` is true.
- HashMap `parseMultiLine(BufferedReader <buffered_reader>, String[] <header>, String <delimiter>)`
  - Parses a multi-line (where one or more values contains linefeed/carriage return characters and is properly quoted using the header for the map returned. Returns null if end of file or empty line is encountered.
- HashMap `parseMultiLine(BufferedReader <buffered_reader>, String[] <header>, char <delimiter>, char <text_qualifier>)`
  - Parses a multi-line (where one or more values contains linefeed/carriage return characters and is properly escaped with the designated `<text_qualifier>` using the header for the map returned. Returns null if end of file or empty line is encountered.
- void `reactivateRecord(HashMap <parsed_line>, HashMap <inactive_records>, TreeMap <primary_key>, String <reactivate_column_name>)`
  - Reactivates a record that was previously inactivated but is now in the import file.
- void `removeCharacter(HashMap <parsed_line>, String <character>)`
  - Removes the designated character from each parsed value.
- void `removeSpaces(HashMap <parsed_line>)`
  - Removes leading and trailing white space from each parsed value.
- String[] `simpleParseHeader(String <header_line>, String <delimiter>)`

- Parses the header line using the designated delimiter. Delimiter is passed to the `String.split()` method.
- `void updateExport(HashMap <update>)`
  - Updates the specified attributes in the export job with the specified values.
- `void updateExportStatus(String <records_processed>, String <records_with_errors>, String <status>, String <export_errors>)`
  - Updates the Scheduled Export Job record with the specified details. This call should be made if the post-processing is going to take a considerable amount of time to complete. The call should be made no more than once every several minutes.
- `void updateExportStatus(String <records_processed>, String <records_with_errors>, String <status>, String <download_link>, String <export_errors>)`
  - Updates the Scheduled Export Job record with the specified details, including a URL for downloading the processed file. This call should be made after processing of the file has completed.
- `void updateImportJob(HashMap <update>)`
  - Updates the specified attributes in the import job record with the specified values.
- `void updateImportStatus(String <records_processed>, String <records_updated>, String <records_created>, String <records_deleted>, String <records_with_errors>, String <status>, String <import_errors>)`
  - Updates the Scheduled Import Job record with the specified details. This call should be made if the pre-processing is going to take a considerable amount of time to complete. The call should be made no more than once every several minutes.

## Scheduling Promotions

Staging records can be promoted manually from the EnterWorks UI via the Package Promotion workflow. They can also be automatically promoted via the Scheduled Promotions feature.

## Scheduled Promotions Repository

When a scheduled promotion is activated (either manually or on a schedule), the records in the designated repository are validated. If the repository’s validation requirements are met, the records are then promoted to Production.

All scheduled promotions are defined in the Promotions Repository. The attributes for this repository are listed alphabetically in the following table.

Attribute	Description
Active Days Only	If this is set to Yes, run the scheduled job on the designated Schedule Days. Only applies if the Schedule Type is Minutes.
Active End Time	If Active Times Only is set to Yes and Schedule Type is set to Minutes, jobs will not be executed after this time of day. Any jobs with a calculated time later than this time will be not be executed. The format of this field is HH:MM:SS.
Active Start Time	If Active Times Only is set to Yes jobs will not be executed before this time of day. Any jobs with a calculated time earlier than this time will be not be executed. The format of this field is HH:MM:SS.
Active Times Only	If Schedule Type is set to Minutes, only run the scheduled job if the calculated Minutes is between the Active Start Time and Active End Time.
Dependent Keys	The list of keys to be passed to each dependent operation. The keys must match attribute names in the dependent repository. The corresponding value in the Dependent Values attribute is then used to set the value of the key attribute when the job is created. The value can reference attributes from the source job by surrounding the attribute name with double-pipe characters. For example, if an export is dependent on a promotion and the export should specify the same user, the Dependent Key would be set to “Manager Login” and the value would be “  Manager Login  ”. Conversely, if the same dependent Export needs to be set to the admin user regardless of its initial definition, the Dependent Key would be set to “Manager Login” and the Dependent Values set to “admin”.
Dependent Operation	Name of the dependent Import, Export, or Promotion to be initiated upon successful completion of this job.
Dependent Operation Immediate	If this is set to Yes, initiate the dependent operation immediately. Otherwise the operation will be queued and processed by the scheduler.



Attribute	Description
Dependent Operation Type	The type of dependent operation: Promotion, Import, or Export.
Dependent Values	<p>Values corresponding to the <code>Dependent Keys</code> that are passed to the dependent operations. Values may reference attributes by surround the attribute name with double-pipe characters. For example, to reference the <code>Import Job</code> number, the <code>Dependent Values</code> might contain: “<code>Job  Export Job  </code>”. If <code>Export Job</code> in the current record is set to 1234, the resulting value will be “<code>Job1234</code>”</p>
Description	Description for promotion.
Group Lock Action	<p>Group locks allow for multiple concurrent jobs to all attach to the same lock. The lock is not freed until all processes have detached from it.</p> <p>Action to be taken with the lock identified in <code>Group Lock Name</code>:</p> <ul style="list-style-type: none"> <li>• <code>Attach</code> – the promotion will attach to the specified lock. Processing for the import will begin immediately.</li> <li>• <code>Free</code> – the promotion will detach from the named lock (presumably set by a previous job that invoked this promotion) once the import processing has completed. The lock is not completely freed until all attached processes have detached.</li> <li>• <code>Attach And Free</code> – the promotion will attach to the named lock and begin processing and detach from the named lock once processing has completed.</li> <li>• <code>No Lock</code> – the promotion will not attach to a group lock.</li> </ul> <p>Group locks will always acquire the designated lock, even if another process has acquired or attached to the same lock. If multiple jobs attach to the same lock via group lock, the lock is not free until all of the processes have freed the lock.</p>
Group Lock Name	Name of the group lock.
Last Promotion Datetime	Date and time of when this promotion was last initiated (scheduled or manual).

Attribute	Description
Lock Action	<p>Action to be taken with the lock identified in Lock Name:</p> <ul style="list-style-type: none"> <li>• Acquire – the promotion will wait until the named lock is acquired. Processing for the promotion will not begin until then.</li> <li>• Free – the promotion will free the named lock (presumably set by a previous job that invoked this promotion) once the promotion processing has completed.</li> <li>• Acquire And Free – the promotion will wait until the named lock is acquired and will free the named lock once processing has completed.</li> <li>• No Lock – the promotion will not wait for any lock before processing.</li> </ul> <p>Locks will only be acquired if they are currently free. If the designated lock has been attached to by multiple jobs using the group lock, the lock does not become free until all of those jobs have freed the lock.</p>
Lock Name	<p>Name of the lock to acquire or free (depending upon Lock Action). If more than one promotion shares the same lock, then they will be processed serially. The name may contain references to other properties by surrounding each property name with double-pipe characters.</p> <p>If the referenced lock is associated with the group lock in other jobs, the lock will not be considered free until all the group lock jobs have detached from the lock.</p>
Next Promotion Datetime	<p>If a promotion is scheduled, this is the date and time it will be launched.</p>
Parameter1-5	<p>Five attributes that can serve as parameters when jobs are connected through the Dependent Operation attributes. They are not used directly by the export processing but can be referenced in other attributes.</p>
Promote Now	<p>If this is set to Yes, perform the promotion now. When the promotion is processed, this setting will be reset to No.</p>
Promote Warnings	<p>If this is set to Yes, promote records that are valid and records that have validation warnings.</p>
Promotion Configuration Name	<p>The configuration name for the promotion. It must match a named configuration in the EPIM sharedConfig.properties file.</p>
Promotion Errors	<p>Detailed error messages if promotion processing failed.</p>
Promotion Priority	<p>Relative priority for each promotion (the lower the number, the higher the priority).</p>
Promotion Production Repository	<p>Name of Production repository into which records are to be promoted.</p>

Attribute	Description
Promotion Skip Validation	If this is set to Yes, validation is skipped (not performed) and the current validation status determines if records are promoted.
Promotion Staging Repository	Name of Staging repository from which records are to be promoted.
Promotion Timeout (Minutes)	Number of minutes the Promotion job will wait for the promotion to complete. The default is 600, (10 hours).
Saved Set	Name of the saved set containing items to be promoted. If this setting is empty, all items are promoted.
Schedule Days	If Schedule Type is Days of Week, this setting identifies the days of the week.
Schedule Exclude Dates Code Set	If this is set to a code set that lists dates as code values and if the calculated date is one of the dates in the code set, the scheduling logic will skip to the next scheduled date.
Schedule Period	Number of minutes after Last Scheduled Date that the next scheduled promotion is initiated. For example, a promotion that is to be processed daily would have a schedule period of 1440. The scheduled promotions are not affected by any manually initiated promotions
Schedule Type	Type of schedule: <ul style="list-style-type: none"> <li>• MINUTES – the Schedule Period identifies the number of minutes before re-invoking the promotion.</li> <li>• DAYS – the Schedule Period identifies the number of days before re-invoking the promotion.</li> <li>• DAYS OF WEEK – specific days of the week on which the promotion is to be executed.</li> <li>• WEEKS – the Schedule Period identifies the number of weeks before re-invoking the promotion.</li> <li>• MONTHS – the Schedule Period identifies the number of months before re-invoking the promotion.</li> </ul>
Scheduled Folder	Logical folder in which the scheduled job is assigned for organization purposes. If this is set and the Scheduled Folder hierarchy is selected in the repository list view, this import job can be found by clicking a node in the Scheduled Folder hierarchy.
Scheduled Folder Sequence	Used to control the order of scheduled jobs in the Scheduled Folders.
Scheduled Promotion	Set to Yes if the promotion is to be scheduled. Each time the scheduled promotion is processed, the Next Promotion Datetime is updated based on the Schedule Period.

Attribute	Description
Status	Status of the Promotion: <ul style="list-style-type: none"> <li>• <b>Active</b> – the promotion is active and will be processed when conditions are met</li> <li>• <b>Inactive</b> – the promotion is inactive and will not be processed.</li> </ul>

## Promotion Jobs Repository

The Promotion Jobs Repository contains records representing the currently active or recently completed promotion jobs. When a promotion is initiated, the details for that promotion are copied from the Promotions Repository to a new record in the Promotion Jobs Repository. This record is subsequently updated to reflect the current status of the promotion.

In addition to the attributes define in the Promotions Repository, the Promotion Jobs Repository has the following attributes.

Attribute	Description
# Errors	Total number of records with errors.
# Processed	Total number of records processed.
# Updated	Total number of records updated in Production.
Promotion Execution Datetime	Date and time of when this promotion was executed.
Promotion Job	Unique number for this job.
Promotion Status	Status of promotion jobs: <ul style="list-style-type: none"> <li>• <b>New</b> – a new job that has not started yet because either it is being launched as a Dependent job with <code>Dependent Operation Immediate</code> set to <code>No</code>, or because an error occurs when launching the work item.</li> <li>• <b>Processing</b> – the Promotion is currently processing.</li> <li>• <b>Completed</b> – the Promotion has completed successfully.</li> <li>• <b>Aborted</b> – the Promotion was aborted by a user.</li> <li>• <b>Error</b> – the Promotion failed with an error.</li> </ul>

## Scheduled Promotion Configuration Properties

Management of the Scheduled Promotions feature is defined by configuration settings in the EPX configuration file `Enterworks.properties`, as listed below.

Property	Description
<code>promote.expireActiveDays</code>	Number of days before import jobs that still show “Processing” are deleted from the Scheduled Import Jobs repository.
<code>promote.expireCompletedDays</code>	Number of days before import jobs that have completed or failed are deleted from the Scheduled Import Jobs repository.

## Channel Readiness

Channel readiness is a measure of how ready a Channel is to be syndicated to its target, that is to say, how many of the required record attribute values have reached a validation level sufficient to be syndicated to their target.

The interface to the Channel Readiness function is comprised of the **Channel Readiness** tab and the **Channel Readiness** widget. The configuration and use of both using the EnterWorks New UI are described in the Precisely EnterWorks online help at <https://support.precisely.com>.

## Data Model Management

### Managing Folders

Data model objects are organized into folders, depending on their type. In order to create a data model object, you must have a folder to store it in. There are folder types for the following data model objects:

- Repository
- Code Set
- Taxonomy
- Hierarchy
- Export Style Map
- Export Template
- Publication Style Map

- Publication Template
- Syndication Template
- Exchange Template

To view existing folders, in the **Feature** bar, open the **Folders** tab. You will see a list of the different types of folders. Click the type of folder you wish to manage. A tab will open that displays the existing folders of that type.

For instance, if you click **Feature > Folders > Code Set**, the tab will show a list of the Code Set folders.

The **Action** dropdown shows available actions you can perform on folders:

- **New:** Create a new folder of that type.
- **Edit:** Edit a folder.
- **Delete:** Delete a folder.
- **Sequence:**
  - **By Name:** Lists the folders alphabetically by name.
  - **Manually:** You can drag and drop folders to arrange the list of folders.

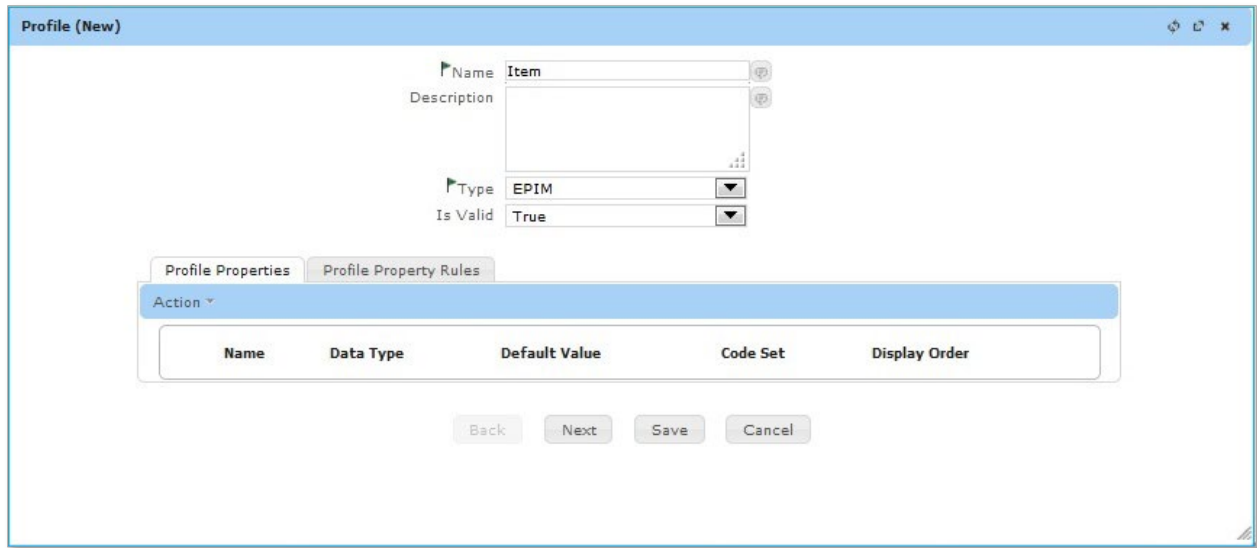
## Managing Profiles

The Profiles function is used to manage the structure applicable to a given repository and the collection of attributes contained within. A repository can only have one active profile.

### Creating and Editing Profiles

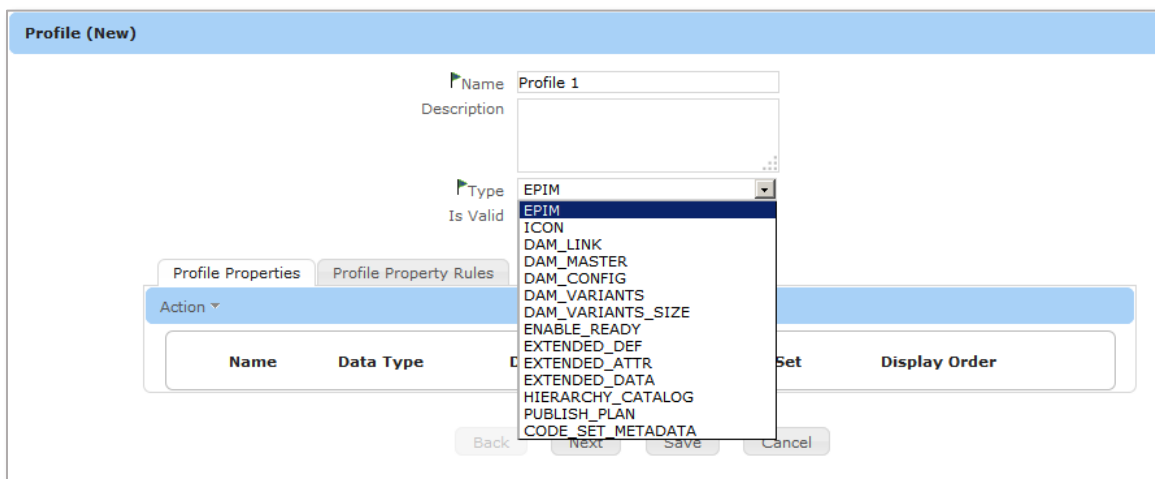
To create or edit a profile:

1. In the Feature bar, open the **Model** tab and select **Profiles**. A list of the existing profiles will appear.
2. To create a new profile, in the **Action** dropdown, select **New**.
3. To edit an existing profile, select the profile, then in the **Action** dropdown, select **Edit**. Select the **New** option under the **Action** drop-down menu.
4. The **Profile** editor will appear.



5. Edit the page's configuration options:

- **Name:** Enter a name for the profile. Do not include special characters in the name, such as & < > ".
- **Description** (optional): Enter a description for the profile.
- Select the profile **Type**. (See [Profile Types](#).)



- After you have selected the profile type, click the **Next** button to edit the profile's properties.

## Profile Types

Type	Description
EPIM	Used for generic repositories. Most Profiles are of this type.
ICON	Used to define an Icon repository for use in the publication process where it holds definitions and rules for translating specific attribute values into icon filenames.
DAM_LINK	Used to define the DAMLink repository. The DAMLink repository contains the link relationship attributes that link digital assets to data records, (for example, linking an image to a product record).
DAM_MASTER	Used to define the repository to hold the DAM Master attributes for the Digital Asset metadata.
DAM_CONFIG	Used to define the repository to hold the configuration attributes for the DAM process.
DAM_VARIANTS	Used to hold the configuration definitions for the variants created in the DAM.
DAM_VARIANTS_SIZE	Used to hold the configuration definitions for the variants created in the DAM.
ENABLE_READY	Deprecated
EXTENDED_DEF	Deprecated
EXTENDED_ATTR	Deprecated
EXTENDED_DATA	Deprecated
HIERARCHY_CATALOG	Used to define the repository for use in linking data to the Hierarchy.
PUBLISH_PLAN	Deprecated



Type	Description
CODE_SET_METADATA	Used to define the repository that extends code sets with custom attributes.

### Profile Properties

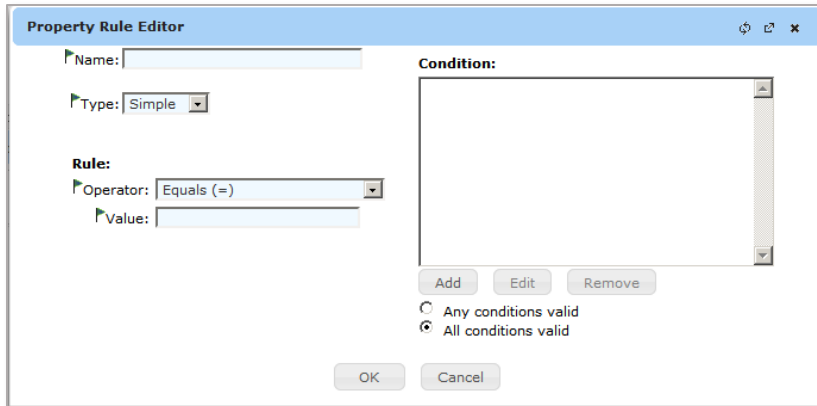
7. Create any needed **Profile Properties** to store attribute metadata. For example, a profile property called **Portal\_Use\_Flag** could be used to identify any attribute that needs to be sent to an external portal.
  - a. Select the **New** option under the **Action** Drop-down list.
  - b. Select a **Property Template** if one exists that satisfies the requirement (optional).

- Enter a Property **Name**. Do not include special characters in the name, such as & < > ".
  - c. Select the **Data Type**.
  - d. Fill in the additional fields, which are different depending upon the **Data Type**.
  - e. Click the **OK** button.

### Profile Property Rules

8. Create any needed **Profile Property Rules** (also called **Validation Rules**) to define checks made against one or more fields when a record is modified. These Profile Property Rules can be assigned to one or more attributes.
  - a. Click on the Profile Property Rules tab.

- b. Click the **New** option under the **Action** drop-down menu.
- c. The **Property Rule Editor** window is shown.




- d. Enter the **Property Rule** conditions.
- e. Click the **OK** button to be returned to the Profile editor.
- f. Click the **Next** button to move to the **Define Attributes** page.

### *Profile Attributes*

9. The **Define Attributes** window will appear. Select the **New** or **Edit** option under the **Action** drop-down menu.
10. The attribute will be opened in a **Manage Attribute** window.
11. Edit the attribute configuration fields as desired. (See [Profile Attribute Configuration Fields.](#))
12. When you are finished editing the Attribute, click the **OK** button.
13. In order for your changes to the Profile to be saved, you must click the **Save** button on the **Profile** editor before exiting.

### *Profile Attribute Configuration Fields*

The following describes the configuration fields for Profile attributes.

- **Name** (Required) and **Description** (Optional): Do not include special characters in the name, such as & < > ". The name and description fields for attributes are translatable. To add translations, click the multi-language icon  corresponding to either the name or description to open the multi-language editor (shown below). After the user is done

adding translations, click the **Save** button on the multi-language editor page and click **OK** on the Define Attribute Details page to save your translations.

- **Restricted Name:** (Required) The relational database snapshot column name. Typically this is comprised of only alphanumeric and underscore characters. The auto-conversion function will change any blank characters to underscores. Do not include special characters in the name, such as & < > " .
- **Language:** For attributes of data type CURRENCY and DECIMAL, the language you choose determines how their values are stored and how EnterWorks editors display their values and interpret entered values.

**Warning:** Once you have data stored in a CURRENCY or DECIMAL type attribute, if you change the attribute's language, the data will not be updated to reflect the new data type. EnterWorks will change the way it interprets the data, but the data itself will not be changed, therefore, the data will be invalid. Do not change the language of CURRENCY and DECIMAL type attributes once you have entered data values.

- **Data Type:** (Required) By defining the Data Type, you are defining what values can be stored in the attribute and what functionality can be performed on it. CURRENCY and DECIMAL attributes are affected by the **Language** selected, as described above.
- **Data Type specific:** Depending on which data type you choose, the following options will appear:
  - **Data Precision:** Appears for **Data Types: BIGINT, CURRENCY, DECIMAL, INTEGER.** Indicates the number of digits the value can have. While you can set the precision for BIGINT and INTEGER values, it is not used during validation and has no effect.
  - **Data Scale:** Appears for **Data Type: Currency, Decimal.** Indicates how many digits are to the right of the decimal point.
  - **Data Size:** Appears for **Data Type: VARCHAR.** Indicates the maximum number of characters in the data field.
- **Default Value:** (Optional) Specifies a default value for the Profile's attribute property. This will not affect any existing repositories' attribute property.
- **Group:** (Required) A group organizes a repository's attributes in a meaningful manner inside a repository's **Detail Editor Tab.**

The names of the groups are displayed in a “xx – yy” format or “xx | yy” format. The first name is the name of the **Tab** and the second name after the dash is the **Group** (or section) on that tab that the attribute will be shown.

- **Associated Group:** (Optional) An association group is a set of attributes that coordinate with each other, in that each attribute field consists of a list of values, where the first value in the list of one attribute relates to the first value in the list of the other attributes in the association group. Any attribute that is going to be in an Association Group must be a repeating, or multi-value attribute. Association groups cannot contain multi-language attributes.

If this attribute will be part of an association group, select the association group or click **Manage** to create an association group. Association Groups are not specific to a Profile so they must have a unique Name across all other Association Groups in the system.

- **Code Set:** (Optional) Code sets are comprised of predefined values. The predefined values are used to give a selection to the user so that an attribute can be defined. For more information about code sets, see the Precisely EnterWorks online help at <https://support.precisely.com>.
- **Control Type:** (Optional) Control types specify how data can be entered into an attribute. The way an attribute is displayed depends on the **Control Type**. The **Is Repeatable** also affects how the attribute is displayed. For instance, the use of a check box or radio button is dependent upon the **Is Repeatable** option.

The following table describes each control type within the EnterWorks system.

Control Types	Description
<b>Check Box or Radio</b>	Check Boxes are a series of small boxes which allow a user to select one or more specified values. Radio buttons also offer a set of values relative to an attribute however, only one can be selected at a time.
<b>Code Set List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Code Sets within the EnterWorks system.
<b>Code Set – Taxonomy and Hierarchy List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of a combined list of all Taxonomies and Hierarchies within the EnterWorks system.
<b>Custom Dropdown List</b>	Allows a user to specify values via a specified SQL query. The values are displayed via a dropdown list or multi-select list.

Control Types	Description
<b>Custom Format Pattern</b>	<p>Allows you to define your own format for user input. Enter a string representing the pattern of the desired input. Sections of the pattern that represent user input begin with {{ and end with }}. Use the following tokens to specify the characters the user is allowed to enter:</p> <ul style="list-style-type: none"> <li>• 9: The character must be a single digit.</li> <li>• a: The character must be an uppercase or lowercase letter in the English alphabet, A through Z.</li> <li>• *: The character can be a single digit or any uppercase or lowercase letter in the English alphabet, A through Z.</li> </ul> <p>For example, a phone number may be represented with the following string:        ({{999}}) {{999}}-{{999}}</p> <p>This control type does not support the multi-language capability.</p>
<b>Export, Publication and Syndication Template List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Export, Publication, and Syndication Templates within the EnterWorks system.</p>
<b>Export Style Map List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Export Style Maps within the EnterWorks system.</p>
<b>Export Template List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Export Templates within the EnterWorks system.</p>
<b>Group List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Groups within the EnterWorks system.</p>
<b>Hierarchy List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Hierarchies within the EnterWorks system.</p>
<b>HTML Editor</b>	<p>Allows the user to format data via a WYSIWYG editor. This control type supports multi-language.</p>
<b>Import Template List</b>	<p>Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Import Templates within the EnterWorks system.</p>

Control Types	Description
<b>Numeric</b>	Attribute field that only permits numeric characters. <ul style="list-style-type: none"> <li>In the Classic UI, non-numeric characters can be entered into attributes with a control type of numeric. The value will be accepted, validated when the value is saved, then marked as invalid.</li> <li>The New UI prevents values containing non-numeric characters to be saved.</li> </ul>
<b>Profile List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Profiles within the EnterWorks system.
<b>Publication Style Map List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Publication Style Maps within the EnterWorks system.
<b>Publication Template List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Publication Templates within the EnterWorks system.
<b>Repository List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all repositories within the EnterWorks system.
<b>Syndication Template List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Syndication Templates within the EnterWorks system.
<b>Taxonomy List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all Taxonomies within the EnterWorks system.
<b>User List</b>	Allows the user to choose a single value via a dropdown or multiple values via a multi-select box of all users within the EnterWorks system.

- Special Function Indicator:** (Optional) Allows the user to specify that the attribute serves a special function, such as a **Taxonomy Node Link** or that it is a status indicator. The available selections for this characteristic are based on the type of Profile being defined. All special function indicators whose name begins with **Extended** have been deprecated.
- Category Attribute Association:** (Optional) This is only used for Taxonomy Attributes and Dynamic Attributes. In the case of Taxonomy Attributes, it identifies the Category Attribute Association Object to be used. If an attribute is a Dynamic Attribute, this

identifies the Association Object to be used. For more information about Association Objects, see the Precisely EnterWorks online help at <https://support.precisely.com>.

- **Is Variant:** A configuration dependent variable. For more information, contact your EnterWorks account representative.
- **Copywriter:** A configuration dependent variable. For more information, contact your EnterWorks account representative.
- **Is Multi-Language:** Indicates if this attribute is to use the Multi-Language capability. For more information on the Multi-Language capability, see [Multiple Language Support](#).
- **Is Global/Category/Dynamic:** (Required) : Indicates if this attribute is a Global Attribute, Category Attribute or Dynamic Attribute. Note that the attribute will only be displayed in the Category Attribute tree if it is a Category Attribute.
- **Is Required:** (Optional) Allows the attribute to be defined as a required element for each record within a repository.
- **Is Primary Key:** (Optional) Indicates this attribute contains the record's unique identifier.
- **Seq Gen Ind:** (Optional) This indicates that the attribute is the repository's sequence attribute and its value will be automatically generated by EnterWorks unless it is supplied by the user or in an import when the record is created. There can be only one Auto-sequenced Attribute per repository.
- **Is Repeatable:** Indicates if a record can have multiple values for this attribute. When checked, a field appears to the right to specify the delimiter between values. The default delimiter is a comma.
- **Is Business Key:** Deprecated.

Note that other attribute definition fields may be present, depending on system configuration.

## Adding an Attribute to a Repository

To add an attribute to a repository, you must edit the repository's profile:

1. On the **Feature** bar, in the **Content** tab, open the repository's folder.
2. Click the arrow to the right of the repository you want to add a new attribute to, click the **Manage Model** submenu, and select **Profile**. The Profile Editor will open.
3. Click the **Next** button to move to the list of attributes..



4. In the Action dropdown menu, select **New**.
5. A new attribute will appear. Edit the configuration settings as desired. For information about the attribute settings, see [Profile Attribute Configuration Fields](#).
6. When you are finished editing the Attribute, click the **OK** button.
7. In order for your changes to the Profile to be saved, you must click the **Save** button on the **Profile** editor before exiting.

## Managing Repositories

There are two ways to manage a repository:

- From the **Model** Feature: to create or delete a repository, or to perform some editing of properties and source mapping.
- From the **Content** Feature: to manage all repository properties, manage the data model aspects of the repository, and to manage the repository's security filters.

### Manage a Repository from the Model Feature

#### *Create a Repository*

To create a repository:

1. Log into the EnterWorks Classic UI.
2. Open the **Repositories** tab by opening the **Feature Bar**, opening the **Model** tab, and selecting **Repositories**. The **Repositories** tab will open.
3. From the list of folders displayed at the top of the pane, select the folder where you would like the repository to be stored.
4. On the **Repositories** tab **Activity Bar**, open the **Action** drop-down and select **New**. A **New Repository** window will appear.

**New Repository**

Name:

Description:

Profile:

PreStaging/Staging/Production:

Repository Type:

Show Tab:

Tab Name:

Sequence:

Auto Generate Sequence:

Derive Indicator Digit:

Transmission Option:

*New Repository Tab*

5. Edit the new repository's configuration options as desired.

- **Name:** (Required) Name of the new repository. Do not include special characters in the name, such as &lt; > ".
- **Description:** A description of the new repository.
- **Profile:** (Required) The Profile to be used to define the new repository.
- **Pre-Staging/Staging/Production:** If this repository will be used in a Pre-Staging(optional)/Staging/Production sequence, indicate which one of these the repository will be. Otherwise, leave the field empty.
- **Repository Type:**
  - **Hierarchy Catalog Repository View:** Shows all the records from a Hierarchy properties repository.
  - **Web Product Root:** Pertains only to the Split/Clone functionality. Enables the **Split/Clone** menu option from the **Action** menu on repository.
  - **Web Product:** Pertains only to the Split/Clone functionality. Enables the **Clone** menu option on the **Action** menu.
  - **Option Type:** Pertains only to the Split/Clone functionality. Allows a user to select multiple variant attributes to split or to enter a count for cloning a product repository record.
  - **Option:** Pertains only to the Split/Clone functionality.
  - <empty>: If the repository is not one of the types listed above, leave the field empty.

- **Show Tab:** Whether or not the repository's name will appear in the **Feature Bar**.
  - **Tab Name:** (Required) The name of the repository's tab when it is opened in a **Repository View** and the name that will appear in the **Feature Bar** if **Show Tab** is selected.
  - **Sequence:** (Required) The sequence object to be used to track record sequence numbers. Repositories can share sequence objects, which means that each record in the repositories has a unique sequence number across all the repositories using that object. Either select an already defined sequence or click the **Manage** button to define a new sequence:
    - **Name:** (Required) The name of the sequence.
    - **GTIN Prefix:** If you want your sequence number to be an auto-generated GTIN then put your GTIN prefix here.
    - **Starting Sequence:** (Required) The first number in the sequence.
    - **Current Max Sequence:** This is a displayed value only; not a configuration option. It displays what is currently the highest number in the sequence.
  - **Auto-Generate Sequence:** If selected, when a record is created, its sequence number will be automatically generated. If not selected, when a record is created, its sequence number must be manually entered.
  - **Derive Indicator Digit:** Deprecated – no longer in use. Depending on the release version of your EnterWorks system, this configuration option may not appear.
  - **Transmission Option:** Defines the method of transmitting data exported from the repository, such as the protocol used, modifications to the resulting file name, destination, and any necessary destination-imposed user authentication. Transmission options can be used by more than one repository. Either select an already defined transmission option, or click the **Manage** button to define a new option.
6. When you have finished editing the configuration options, select **Next**. The **New Repository Attribute Configuration** window will appear. Edit the repository attribute properties as desired.

New Repository  
Profile Name: CZ Vehicle Profile Show All Hide All

**Summary - Titles**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Label
Product ID	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Product Name	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Brand	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Seriously 2	BOOLEAN	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No	<input type="checkbox"/>		

**Summary - Bookkeeping**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Label
LinkRelationshipPath	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
ParentID	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Taxonomy	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
thisSequenceNum	BIGINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Linked Outlets	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Autosequenced Number	BIGINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		

**Test Values - Numbers**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Label
aDecimalValue	DECIMAL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		

**Test Values - Checkboxes**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Label
CheckBoxTesting	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		

**New Taxonomy Tab - New Taxonomy Group**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Label
-----------	------	------------	--------------	------------------	---------------	----------------------	-----------------	-------------------

New Repository Profile Configuration Options

- When you are finished editing the repository's attribute properties, select **Next**. The final configuration options appear.

**Automatically Publish Ready Records**

Auto Publish Interval (in hours):

Auto Publish Start:  Now

Specific Date:

**Track Record Changes**

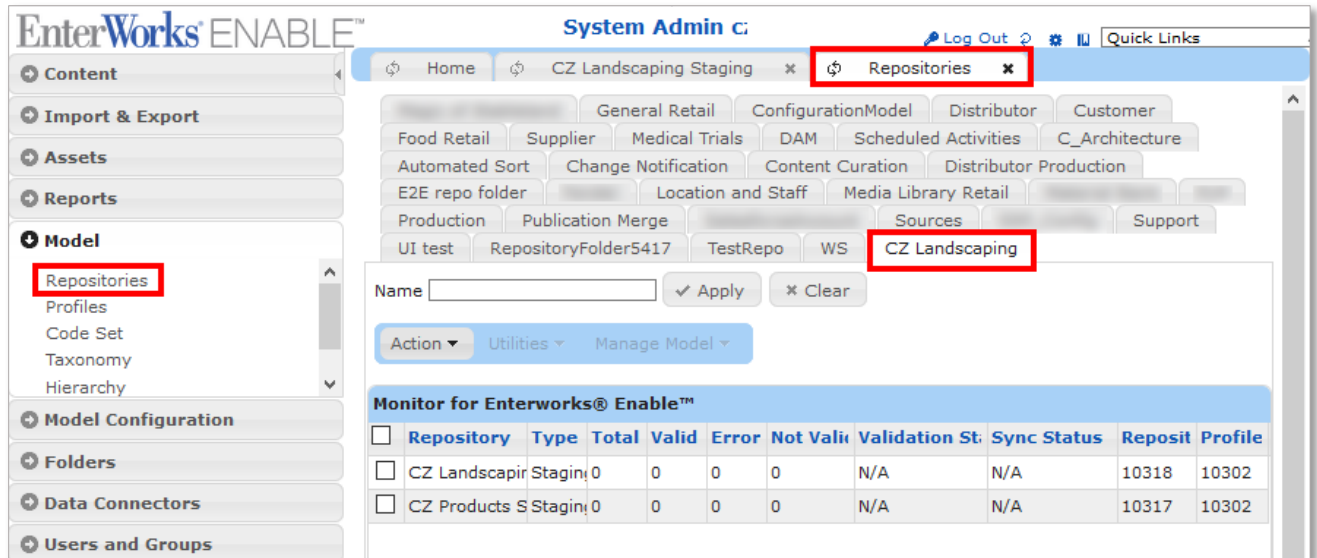
Final Repository Configuration Options

8. The following settings have been deprecated. Depending on the release version of your EnterWorks system, they may or may not appear. If they do, leave them empty.
  - **Automatically Publish Ready Records:** This configuration option is deprecated. Ignore this field and all subfields.
    - **Auto Publish Interval (in hours):** Deprecated.
    - **Auto Publish Start:** Deprecated.
      - **Now:** Deprecated.
      - **Specified Date:** Deprecated.
  - **Track Record Changes:** This setting is deprecated. All record changes are now tracked automatically.
9. Select **Finish** to save the repository and exit the **New Repository** tab or select **Cancel** to exit the tab without creating the repository.
10. Clear the data cache.

### *Delete a Repository*

Use the EnterWorks Classic UI to delete a repository:

1. Log into the EnterWorks Classic UI.
2. Open the **Repositories** tab by opening the **Feature Bar**, opening the **Administration** tab, and selecting **Repositories**. The **Repositories** tab will open.



Open the Repositories Tab

3. Select the repository to be deleted.
4. On the **Repositories** tab's **Activity Bar**, open the **Action** drop-down and select **Delete**. The repository will be deleted.
5. Clear the data cache.

## Manage a Repository from the Content Feature

To manage a repository from the **Content** Feature:

1. Open the **Feature** bar, open **Content**, and right-click the desired folder, then click the arrow to the right of the desired repository's name to see your repository management options:
  - **Open:** Open the repository in a Repository View.
  - **Edit:**
    - **Properties**
    - **Summary Attributes**
    - **Attribute Properties**
    - **Rule Properties**
    - **Source Mapping**

- **Validation Model**
  - **Workflow Properties**
  - **Promotion Properties**
  - **Trigger Properties**
  - **Manage Model:**
    - **Manage Links**
    - **Profile**
    - **Attribute Tabs**
    - **Attribute Groups**
    - **Attribute Security Filters**
    - **Record Security Filters**
  - **Security:** Allows you to set a repository's access permissions, record attribute filter and record filter, according to user and user groups.
2. After you are done creating or editing a repository, clear the data cache so it will appear in the Feature bar.

## Manage a Repository's Properties

To modify a repository's properties through the EnterWorks Classic UI:

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. In the **Feature** bar, open the **Content** Feature, select the desired repository, select **Edit**, and select **Properties**.
3. The **Properties** editor will appear. Edit the properties as desired.
  - **Name:** (Required) Name of the repository.
  - **Description:** A description of the repository.
  - **Profile:** (Required) The repository's profile is selected when the repository is created. It cannot be changed later.
  - **Pre-Staging/Staging/Production:** If this repository will be used in a Pre-Staging(optional)/Staging/Production sequence, indicate which one of these the repository will be. Otherwise, leave the field empty.
  - **Repository Type:**

- **Hierarchy Catalog Repository View:** Shows all the records from a Hierarchy properties repository.
  - **Web Product Root:** Pertains only to the Split/Clone functionality. Enables the **Split/Clone** menu option from the **Action** menu on repository.
  - **Web Product:** Pertains only to the Split/Clone functionality. Enables the **Clone** menu option on the **Action** menu.
  - **Option Type:** Pertains only to the Split/Clone functionality. Allows a user to select multiple variant attributes to split or to enter a count for cloning a product repository record.
  - **Option:** Pertains only to the Split/Clone functionality.
  - **<empty>:** If the repository is not one of the types listed above, leave the field empty.
- **Repository Folder:** (Required) The folder the repository is stored in.
  - **Show Tab:** Whether or not the repository's name will appear in the **Feature Bar**.
  - **Tab Name:** (Required) The name of the repository's tab when it is opened in a **Repository View** and the name that will appear in the **Feature Bar** if **Show Tab** is selected.
  - **Sequence:** (Required) The sequence object to be used to track record sequence numbers. Repositories can share sequence objects, which means that each record in the repositories has a unique sequence number across all the repositories using that object. Either select an already defined sequence or click the **Manage** button to define a new sequence:
    - **Name:** (Required) The name of the sequence.
    - **GTIN Prefix:** If you want your sequence number to be an auto-generated GTIN then put your GTIN prefix here.
    - **Starting Sequence:** (Required) The first number in the sequence.
    - **Current Max Sequence:** This is a displayed value only; not a configuration option. It displays what is currently the highest number in the sequence.
  - **Auto-Generate Sequence:** If selected, when a record is created, its sequence number will be automatically generated. If not selected, when a record is created, its sequence number must be manually entered.
  - **Clone Action:**
  - **Derive Indicator Digit:** Deprecated – no longer in use. Depending on the release version of your EnterWorks system, this configuration option may not appear.
  - **Transmission Option:** Defines the method of transmitting data exported from the repository, such as the protocol used, modifications to the resulting file



name, destination, and any necessary destination-imposed user authentication. Transmission options can be used by more than one repository. Either select an already defined transmission option, or click the **Manage** button to define a new option.

- **Default Mapping:** Deprecated. Use Scheduled Imports instead.
- **Validation Level Post Action:** This specifies how to handle default validation levels for a record during the validation process. The validation level for a record can be manually set when the record is listed in a Repository View by using the **Set Validation Level** menu option. The **Validation Level Post Action** setting indicates that the record's default validation level will be automatically increased after the record is validated.
  - **Auto Elevate After Validate:** If the record is successfully validated with no errors and no warnings, increase the default validation level.
  - **Auto Elevate After Validate - Ignore Warnings:** If the record is successfully validated with no errors (but it may have warnings), increase the default validation level.
- **Snapshot Language Support:** If selected, there will be additional snapshot tables created for this repository. There will be one additional snapshot table for each language enabled in the Language List other than the default language. The name of the table be appended with `<language-code>`. These additional snapshot tables will only include the repository's multilanguage attributes. Using this functionality is expensive in terms of processing time. For more information regarding multi-language snapshot tables, please see the EnterWorks online help at <https://support.precisely.com/products/precisely-enterworks>.
- **Validate Items after Multi-Edit:** If you perform a multi-edit and this option is enabled, EnterWorks will validate the affected records after the edit. If this option is not enabled, after the multi-edit the validation state of the records will be unknown and their validation icon will be the black triangle.
- **Create Category Attribute Snapshot:** Deprecated. Not recommended.
- **Links to DAM objects:** Indicates that this repository will link to digital assets. If this option is selected, the DAM Link Setup options will appear.
- **DAM Link Setup:** These options will appear if **Links to DAM objects** is selected.
  - **Enable DAM Link Trigger:** If this option is enabled, the rest of the **DAM Link Setup** options will appear. If it is enabled, when digital assets are uploaded, EnterWorks will attempt to automatically link each asset to any existing record that has an attribute value that matches the criteria specified in the **Attribute Linked to DAM File Name Column** setting. For this functionality to work, a link relationship must be defined between this repository and the DamLink repository.

A key limitation to this feature is that if more than one repository is configured to automatically link digital assets, when an asset is uploaded, it will be linked to any records whose attribute matches the specification, in any of those repositories. You may be linking the same digital asset to multiple records in multiple repositories.

- **Attribute Linked to DAM File Name Column:** This specifies which attribute's value will be used to match the pattern specified in the **Parsed Token Position or Custom Class** setting.
- **Parsed Token Position or Custom Class:** This field can hold one of three types of values:

- `"token=<delimiter>;position=<n>"`: EnterWorks will parse the digital asset's filename, using the character specified by `<delimiter>` to divide the name into portions. If the portion in the position specified by `<n>` matches the value in the attribute specified by **Attribute Linked to DAM File Name Column**, the record is linked to the asset.

For example, if this setting's value is:

```
"token=_;position=2"
```

And an asset has the filename:

```
Product_clothing_hat_3456.png
```

Any record that has the value `clothing` in the specified attribute will be linked to the asset.

- A regex pattern.
- A custom callout path.
- **Dam Hierarchy Node Value: Dunno – DAM hierarchy is a restricted hierarchy in DAMlink – it is a way you can virtually arrange your assets in a virtual folder structure. Ask engineering. Maybe every asset that gets uploaded gets assigned to this node. This may be deprecated.**
- **DAM Link Include Children: Ask.**
- **Generate Variants: To get this to appear, set up the group as type category attribute. It must be set up on both the parent and child. Find Brian's note.**
  - **Enable Variants:**

4. When you have finished editing the configuration options, select **Save**.

## Manage a Repository's Trigger Properties

The `sharedConfig.properties` file used to be the only location of a repository's Trigger property settings, but now many of them are also stored in the EPIM database and can be accessed through the EnterWorks Classic UI. In order to maintain backward compatibility, if a property is not available in the database, EnterWorks will fall back to reading the property from the `sharedConfig.properties` file.

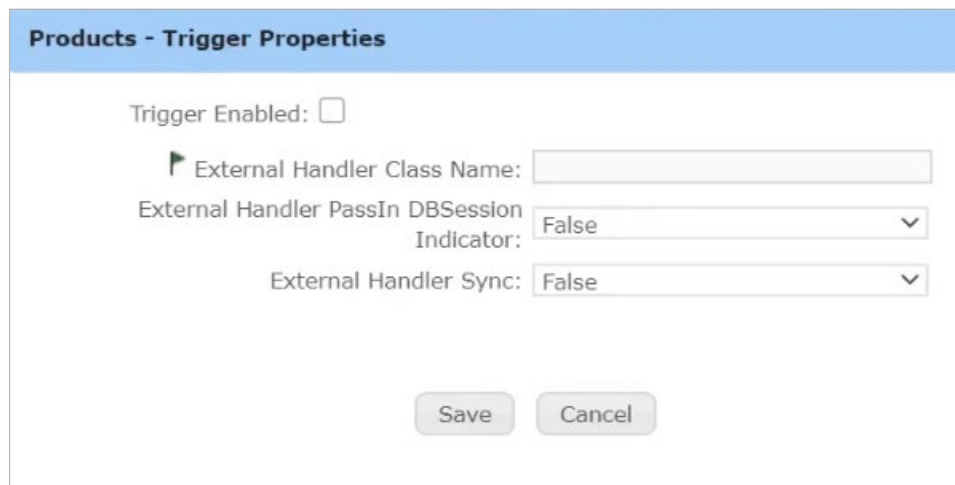
Anytime you modify trigger properties, whether they are in the EPIM database or the `sharedConfig.properties` file you must restart services and clear the data cache for the new property values to take effect.

If you migrate a repository, its Trigger properties will be migrated as well.

### *Repository Trigger Properties in EPIM Database*

To modify a repository's Trigger properties through the EnterWorks Classic UI:

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. In the **Feature** bar, open the **Content** Feature, select the desired repository, select **Edit**, and select **Trigger Properties**.
3. The **Trigger Properties** editor will appear.



4. Edit the settings as desired:
  - **Trigger Enabled:**

- Check this checkbox to enable the trigger.
- Uncheck this box to disable the trigger. After a trigger has been disabled and the Trigger Properties are saved, all settings in the database regarding this Trigger will be cleared.
- **External Handler Class Name:** The java class name. The class must be accessible from the EnableServer Tomcat or EnableServer Jboss services.
  - To be accessible for the EnableServer Tomcat service, the class must be a JAR file and be placed in the folder:

```
<drive>:\Enterworks\EnableServer\tomcat\webapps
\webcm\WEB-INF\lib
```

- To be accessible for EnableServer Jboss service, the class must be injected into the `epimServer.ear` file. Execute the following for each EnableServer Jboss service:

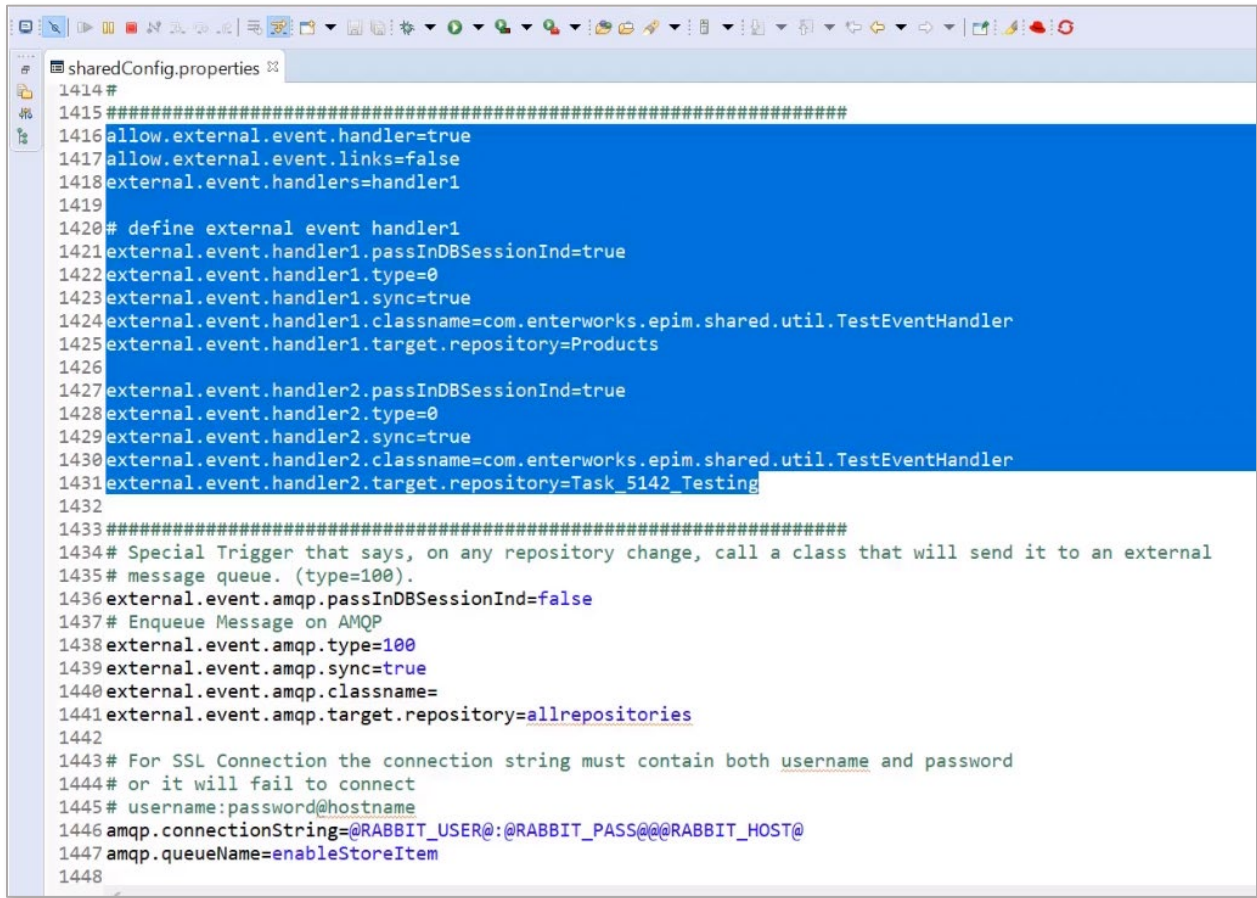
```
<install_drive>:\Enterworks\bin\DeployServicesJ
ar.bat
```

- **External Handler PassIn DBSession Indicator:** If this is set, the database session is passed to the trigger's `doWork()` method as an argument. This session provides access to the EPIM database without having to open another connection.
- **External Handler Sync:** This is the event synchronous/asynchronous indicator:
  - If it is set to true, the event is a synchronous event, which means that if a user saves a record in the UI, the UI will not refresh until after the trigger has completed processing. Any changes the trigger made to the record being edited will be shown in the editor (if the editor is kept open).
  - Otherwise it is an asynchronous event, which means that if a user saves a record in the UI, the UI will refresh immediately, potentially before the trigger has completed processing. If the trigger changes the record being edited, those changes will not be reflected in the editor (if the editor is kept open).

5. Save and exit the **Trigger Properties** editor.
6. Clear the data cache and restart EnterWorks services.

## Repository Trigger Properties in the sharedConfig.properties File

Anytime you modify sharedConfig.properties file, restart services and clear the data cache afterward.



```
1414 #
1415 #####
1416 allow.external.event.handler=true
1417 allow.external.event.links=false
1418 external.event.handlers=handler1
1419
1420 # define external event handler1
1421 external.event.handler1.passInDBSessionInd=true
1422 external.event.handler1.type=0
1423 external.event.handler1.sync=true
1424 external.event.handler1.classname=com.enterworks.epim.shared.util.TestEventHandler
1425 external.event.handler1.target.repository=Products
1426
1427 external.event.handler2.passInDBSessionInd=true
1428 external.event.handler2.type=0
1429 external.event.handler2.sync=true
1430 external.event.handler2.classname=com.enterworks.epim.shared.util.TestEventHandler
1431 external.event.handler2.target.repository=Task_5142_Testing
1432
1433 #####
1434 # Special Trigger that says, on any repository change, call a class that will send it to an external
1435 # message queue. (type=100).
1436 external.event.amqp.passInDBSessionInd=false
1437 # Enqueue Message on AMQP
1438 external.event.amqp.type=100
1439 external.event.amqp.sync=true
1440 external.event.amqp.classname=
1441 external.event.amqp.target.repository=allrepositories
1442
1443 # For SSL Connection the connection string must contain both username and password
1444 # or it will fail to connect
1445 # username:password@hostname
1446 amqp.connectionString=@RABBIT_USER@:@RABBIT_PASS@@@RABBIT_HOST@
1447 amqp.queueName=enableStoreItem
1448
```

The settings in the sharedConfig.properties file that affect Triggers are:

- **allow.external.event.handler:** This enables or disables external event handling. If it is false, all the event handler settings will be ignored.
- **allow.external.event.links:**
- **external.event.handlers:** Lists the names of the defined events. To disable a particular event, remove it from this list. The names must match the **handlerName** specified in:  
**external.event.<handlerName>.<property>**
- Settings for individual Triggers:

- **external.event.<handlerName>.passInDBSessionInd:** If this is set, the database session is passed to the trigger's `doWork()` method as an argument. This session provides access to the EPIM database without having to open another connection.
- **external.event.<handlerName>.type:** This is always set to 0 (zero).
- **external.event.<handlerName>.sync:** This is the event synchronous/asynchronous indicator:
  - If it is set to true, the event is a synchronous event, which means that if a user saves a record in the UI, the UI will not refresh until after the trigger has completed processing. Any changes the trigger made to the record being edited will be shown in the editor (if the editor is kept open).
  - Otherwise it is an asynchronous event, which means that if a user saves a record in the UI, the UI will refresh immediately, potentially before the trigger has completed processing. If the trigger changes the record being edited, those changes will not be reflected in the editor (if the editor is kept open).
- **external.event.<handlerName>.classname:** The java class name. The class must be accessible from the EnableServer Tomcat or EnableServer Jboss services.
  - To be accessible for the EnableServer Tomcat service, the class must be a JAR file and be placed in the folder:
 

```
<drive>:\Enterworks\EnableServer\tomcat\webapps
          \webcm\WEB-INF\lib
```
  - To be accessible for EnableServer Jboss service, the class must be injected into the `epimServer.ear` file. Execute the following for each EnableServer Jboss service:
 

```
<install_drive>:\Enterworks\bin\DeployServicesJar.bat
```
- **external.event.<handlerName>.target.repository:** The name of the repository the trigger is registered to.

Any time you modify `sharedConfig.properties` file, restart services afterward.

## Manage a Repository's Promotion Properties

The `sharedConfig.properties` file used to be the only location of a repository's Promotion property settings, but now many of them are also stored in the EPIM database and can be accessed through the EnterWorks Classic UI. In order to maintain backward compatibility, if a property is not available in the database, EnterWorks will fall back to reading the property from the `sharedConfig.properties` file.

If you are modifying Promotion properties in the EW classic UI, you do not have to restart EnterWorks services. However, if you are modifying the `sharedConfig.properties` file you will have to restart services for the new property values to take effect.

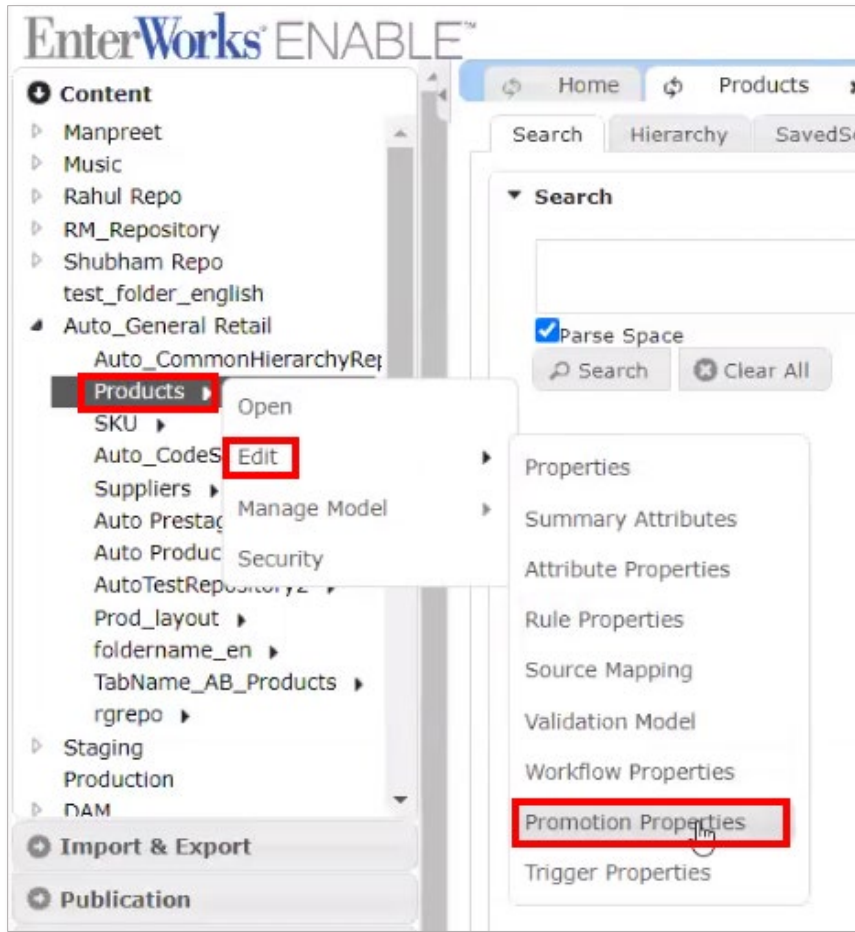
If you migrate a repository, its Promotion properties will be migrated as well.

### *Repository Promotion Properties in the Classic UI*

**Note:** You can modify a repository's Promotion properties through the EnterWorks Classic UI, however, the name of the source repository must be defined in the `sharedConfig.properties` file.

To modify a repository's Promotion properties through the EnterWorks Classic UI:

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. In the **Feature** bar, open the **Content** Feature, select the desired repository, select **Edit**, and select **Promotion Properties**.



3. The **Promotion Properties** editor will appear.



**Products - Promotion Properties**

IsPromotionEnabled

ValidationOptions 0

**LinkRelationships**

Auto\_Dam\_Link

Auto\_Hierarchy\_Link

Auto\_product\_to\_SupplierProducts\_Staging

Auto\_SKU\_Link

AutoProd2Link

newLink

NEWPRODTodl

**Source Attribute List**

Attribute	Value
<input type="checkbox"/> Full Description	promoted to production
<input type="checkbox"/> Age Codeset	313

Add Delete

**Source Status List**

Attribute	Value
-----------	-------

Add Delete

**Target Attribute List**

Attribute	Value
<input type="checkbox"/> Full Description	promoted from products
<input type="checkbox"/> Age Codeset	323

Add Delete

**Target Status List**

4. The **Promotion Properties** editor will allow you to modify the following values:
- **IsPromotionEnabled:** Check this checkbox to enable promotions for this repository.
  - **ValidationOptions:**
    - 0 = Promote all records.
    - 1 = Validate and promote all records.

- 2 = Validate and promote valid records only -- no records with errors or warnings.
- **LinkRelationships:** These checkboxes identify the repositories (via link relationships) that are linked to the one being edited. If checked, when a record in the parent repository is promoted, any linked record in the designated linked repository will also be promoted (subject to the same validation options).
- **Attribute and Status Lists:** To add attributes to the list, click the **Add** button. Each time you click **Add**, a new attribute or status dropdown selector will be added. Use the dropdown's down arrow to select the desired attribute or status. Enter the desired value in the **Value** box.

To enable an attribute or status value assignment, check the checkbox to the left of the attribute's or status' name.

Attribute Lists:

- **Source Attribute List:** Contains a list of attributes in the source repository that you want to set to a particular value after the records are promoted. The most common use of this feature is when needing to distinguish between add/new vs. update for a record in a syndication. For example, if the attribute "Add or Update" represents this status, it would default to the value "Add" for new records. When that record is promoted to Production and then syndicate, the downstream system will be notified that it is a new record. The promotion operation itself causes the attribute to be updated to "Update" in the Staging repository. The next time the staging record is modified, promoted, and syndicated, the syndication will indicate the record is existing and is being updated. This assumes the Production record is syndicated after the initial promotion before the same record can be modified and promoted a second time.

**Warning:** Attributes set through this mechanism are not recorded in the History information.

- **Target Attribute List:** Contains a list of attributes in the target repository that you want to set to a particular value when the records are promoted.

Status Lists:

- **Source Status List:** Contains a list of attribute statuses in the source repository that you want to set to a particular value after the records are promoted. The common state fields set are:

- **record\_state = 0:** This indicates the record is in sync.
  - **external\_session\_info:** This indicates the update was due to promotion.
  - **Target Status List:** Contains a list of attribute statuses in the target repository that you want to set to a particular value after the records are promoted. The common state fields set are:
    - **record\_state = 0:** This indicates the record is in sync.
    - **external\_session\_info:** This indicates the update was due to promotion.
5. Edit the properties as desired, then save the properties and exit from the Promotion Properties editor.
  6. Restart EnterWorks services and clear the data cache.

### *Repository Promotion Properties in sharedConfig.properties*

The `sharedConfig.properties` file identifies which repositories can be promoted and how each of those promotions will behave.

**Warning:** All `sharedConfig.properties` files should have the exact same property definitions for promotions. As a best practice, one file should be updated and then the entire `repo.promote` section be copied from that file to all of the other `sharedConfig.properties` files in the same environment.

```
sharedConfig.properties x
1272 #####
1273 # Repository Promotion Configuration
1274 #
1275 #####
1276
1277 repo.promote.config.list=Brand_STG,Product_STG,
1278
1279 #default validation check for promotion - 0 promote All, 1 promote valid only, 2 promote val
1280 repo.promote.config.Brand_STG.default.validation.option=2
1281
1282 repo.promote.config.Brand_STG.desc=Promote Brand Items
1283 repo.promote.config.Brand_STG.srcRepo.name=Brand_STG
1284 repo.promote.config.Brand_STG.srcRepo.linkList=
1285 repo.promote.config.Brand_STG.srcRepo.linkList.link1.name=
1286
1287 repo.promote.config.Product_STG.desc=Promote Product
1288 repo.promote.config.Product_STG.srcRepo.name=Product
1289 repo.promote.config.Product_STG.srcRepo.linkList=
1290 repo.promote.config.Product_STG.srcRepo.linkList.link1.name=
1291
1292 # Optional
1293 repo.promote.config.Brand_STG.srcRepo.updateStatus.list=status1,status2,
1294 repo.promote.config.Brand_STG.srcRepo.updateStatus.status1.name=RECORD_STATE
1295 repo.promote.config.Brand_STG.srcRepo.updateStatus.status1.value=0
1296 repo.promote.config.Brand_STG.srcRepo.updateStatus.status2.name=EXTERNAL_SESSION_INFO
1297 repo.promote.config.Brand_STG.srcRepo.updateStatus.status2.value=promote
1298
1299 repo.promote.config.Brand_STG.srcRepo.updateAttr.list=attr1
1300 repo.promote.config.Brand_STG.srcRepo.updateAttr.attr1.name=Status
1301 repo.promote.config.Brand_STG.srcRepo.updateAttr.attr1.value=Active
1302
1303
1304 repo.promote.config.Brand_STG.targetRepo.updateStatus.list=status1,status2,
1305 repo.promote.config.Brand_STG.targetRepo.updateStatus.status1.name=RECORD_STATE
1306 repo.promote.config.Brand_STG.targetRepo.updateStatus.status1.value=0
1307 repo.promote.config.Brand_STG.targetRepo.updateStatus.status2.name=EXTERNAL_SESSION_INFO
1308 repo.promote.config.Brand_STG.targetRepo.updateStatus.status2.value=promote
1309
1310 repo.promote.config.Brand_STG.targetRepo.updateAttr.list=attr1
1311 repo.promote.config.Brand_STG.targetRepo.updateAttr.attr1.name=Status
1312 repo.promote.config.Brand_STG.targetRepo.updateAttr.attr1.value=Active
1313
```

1. The Promotion Properties editor will allow you to modify the following values:
  - **repo.promote.config.list:** A list of the names of the repositories that can be promoted. This list must specify the repository name of each source repository to be promoted. The repository name must exactly match what is defined in the Name field on the Edit -> Properties page for the repository. This name must be referenced as the <promotion\_name> in the properties listed below.
  - **repo.promote.config.<promotion\_name>.default.validation.option:**
    - 0 = Promote all records.
    - 1 = Validate and promote all records.

- 2 = Validate and promote valid records only -- no records with errors or warnings.
- **repo.promote.config.<promotion\_name>.desc:** The display name of the promotion. In the **Model Configuration** feature, if you select **Promotion Configuration**, the **Promotion Configuration** drop-down list will display this description as the name of the promotion.
- **repo.promote.config.<promotion\_name>.srcRepo.name:** The name of the source repository.
- Linked Repositories settings:
  - **repo.promote.config.<promotion\_name>.srcRepo.linkList= <linkName1>, <linkName2>,...,<linkName<n>>:** a list of the linked repositories to be promoted at the same time a record is promoted from the source repository.  
  
For example, if the promotion is for the `Product_Staging` repository, which is linked to the `Item_Staging` repository, when a `Product` record is promoted, all of the `Item` records linked to that `Product` will also be promoted (subject to the promotion settings). The `<linkName>` values are used in subsequent properties
  - **repo.promote.config.<promotion\_name>.srcRepo.linkList.<linkName<n>>.name=** The name of the linked repository.
- Attribute settings: (Optional)
  - Source Repository Attribute settings:
    - **repo.promote.config.<promotion\_name>.srcRepo.updateAttr.list=<attrName1>,<attrName2>,...,<attrName<n>>:** The values in this list will be used in the subsequent properties to identify attributes and their values.  
  
The attributes listed here will be set to their specified values in the source repository after the record is promoted.
    - **repo.promote.config.<promotion\_name>.srcRepo.updateAttr.<attrName<n>>.name:** The name of the attribute.
    - **repo.promote.config.<promotion\_name>.srcRepo.updateAttr.<attrName<n>>.value:** The value the attribute will be set to after the record is promoted.
  - Target Repository Attribute settings:

- **repo.promote.config.<promotion\_name>.targetRepo.updateAttr.list=<attrName1>,<attrName2>,...,<attrName<n>>**: The values in this list will be used in the subsequent properties to identify attributes and their values.

The attributes listed here will be set to their specified values in the target repository when the record is promoted.

- **repo.promote.config.<promotion\_name>.targetRepo.updateAttr.<attrName<n>>.name**: The name of the attribute.
- **repo.promote.config.<promotion\_name>.targetRepo.updateAttr.<attrName<n>>.value**: The value the attribute will be set to when the record is promoted.

- Status Settings:

- Source Repository Status settings:

- **repo.promote.config.<promotion\_name>.srcRepo.updateStatus.list=<statusName1>, <statusName2>,..., <statusName<n>>**: Contains a list of attribute statuses in the source repository that you want to set to a particular value after the records are promoted. The common state fields to set are:

- **record\_state = 0**: This indicates the record is in sync.
- **external\_session\_info**: This indicates the update was due to promotion.

- **repo.promote.config.<promotion\_name>.srcRepo.updateStatus.<statusName<n>>.name**: The name of the status.

- **repo.promote.config.<promotion\_name>.srcRepo.updateStatus.<statusName<n>>.value**: The value of the status.

- Target Repository Status settings:

- **repo.promote.config.<promotion\_name>.targetRepo.updateStatus.list=<statusName1>, <statusName2>,..., <statusName<n>>**: Contains a list of attribute statuses in the target repository that you want to set to a particular value when the records are promoted.

- **record\_state = 0**: This indicates the record is in sync.
- **external\_session\_info**: This indicates the update was due to promotion.

- `repo.promote.config.<promotion_name>.targetRepo.updateStatus.<statusName<n>>.name`: the name of the status.
  - `repo.promote.config.<promotion_name>.targetRepo.updateStatus.<statusName<n>>.value`: the value of the status.
2. Edit the properties as desired, then save the file and exit.
  3. Restart services for the new property values to take effect.

## Repository Link Management

This section describes how to create and edit link relationships. For information regarding the nature of link relationships, linked records, linked repositories, and link tables and how they are used, see the Precisely EnterWorks online help at <https://support.precisely.com>.

Be careful when modifying the definition of existing link relationships. They are fundamental to many EnterWorks functions. Modifications may affect the stability and functional behaviors of the EnterWorks system in ways you do not intend.

If a repository has staging and production versions of repositories, any link relationships must be duplicated in both environments.

### Create or Edit a Link Relationship

To create or edit a link relationship:

1. Log into the EnterWorks Classic UI as a user with Administrative privileges.
2. Open the parent repository's **Manage Links** editor: open the **Feature** bar, open the **Content** folder, open the desired repository folder, click the arrow to the right of the repository's name, click the arrow to the right of **Manage Model**, and select **Manage Links**. The list of all link relationships for the repository will be displayed.

**Product Staging - Manage Links** 🔍 🔄 ✕

Action ▾

<input type="checkbox"/> Link Relationship	Child Repository	Sub Type	Id
<input type="checkbox"/> Pricing	Distributor Price	One to Many (Parent-Child)	10439
<input type="checkbox"/> Product to Assets	DAMLink	Digital Assets	10308
<input type="checkbox"/> Product to Exterior	DAMLink	Digital Assets	10445
<input type="checkbox"/> Product to Facades	DAMLink	Digital Assets	10443
<input type="checkbox"/> Product to Floor Plans	DAMLink	Digital Assets	10446
<input type="checkbox"/> Product to Inclusions	DAMLink	Digital Assets	10448
<input type="checkbox"/> Product to Interior	DAMLink	Digital Assets	10444
<input type="checkbox"/> Product to Video	DAMLink	Digital Assets	10447
<input type="checkbox"/> Related Products	Related Distributor Products	Many to Many (Parent-Child-Parent)	10437
<input type="checkbox"/> Related Products Relationship	Related Distributor Products	Many to Many (Parent-Child-Parent)	10438
<input type="checkbox"/> RS_Catalog	RS_Catalog	Hierarchy View	10309
<input type="checkbox"/> RS_Item	RS_Item	One to Many (Parent-Child)	10307

Total Records: 12

Close

3. Do one of the following:

- To create a new link relationship, open the **Action** dropdown list and select **New**. An empty link relationship will be opened in the **Manage Links** editor.
- To edit an existing link relationship, select the link relationship, open the **Action** dropdown list, and select **Edit**. The link relationship will be opened in the **Manage Links** editor.

**WARNING:** Do not change the names of existing links, otherwise the EnterWorks views, Export Templates, and other EnterWorks features may break.



**Product Staging - Manage Links**

Name:

Description:

Child Repository:

Link Type:  Use Join Condition  
 Manually choose record(s)

Sub Type:

Max Link Count:

Show in Simplified Display List:  User Preferences  
 Search  
 Import Templates  
 Export Templates  
 Reports

Link Relationship Properties | Link Relationship Property Rules

Action ▾

Name	Data Type	Default Value	Code Set	Display Order
------	-----------	---------------	----------	---------------

Back Next Save Cancel

4. Edit the configuration values:

- **Name:** The name of the link relationship. Link relationships must have names that are unique across the EnterWorks system. Do not include special characters in the name, such as & < > ".
- **Description:** A description of the purpose of the link relationship.

- **Child Repository:** Choose the child repository.
- **Link Type:**
  - **Use Join Condition:** One or more pairs of attributes are used to link the repositories. For more information, see [Join Conditions](#).
  - **Manually choose record(s):** Deprecated. **Do not use this option.** If you choose this option, link tables may not load, the Hierarchy/Taxonomy Manager may not load, users may not be able to create linked preferences, and other errors may occur. For more information, see [Link Management Troubleshooting](#).
- **Sub Type:**
  - **One to Many (Parent-Child):** One parent repository is linked to one child repository.
  - **Many to Many (Parent-Child):** Many parent repositories are linked to many child repositories.
  - **Accessory:** Deprecated.
  - **Hierarchy View:** To use a non-restricted hierarchy in a repository, you must create a link of sub-type **Hierarchy View** between the repository and the Hierarchy Catalog repository.
  - **Extended:** Deprecated.
  - **Digital Assets:** In order to link a repository to DamLink, you must create a linked relation of sub-type **Digital Assets**.
  - **Retail Product:** Deprecated.
  - **Retail Product SKU:** Deprecated.
- **Show in Simplified Display List:** These settings determine where this link relationship will be available to be used.
  - **User Preferences:** This link relationship will be available for use in user preferences.
  - **Search:** The link relationship will be available for use in searches.
  - **Import Templates:** The link relationship will be available for use when you create an import template.

- **Export Templates:** The link relationship will be available for use when you create an export template.
  - **Reports:** The link relationship will be available for use when you create a report.
5. When you are finished editing the configuration options, click **Next**.
  6. If you selected **Use Join Condition** for the **Link Type**, the **Join Condition** list will be displayed. Edit the configuration options:

**Products Staging - Manage Links**

**Choose Attribute to Sort Child Records**  
 Child Repository Sort Attribute:

**Choose Attribute Pair**  
 Parent Repository Attribute:   
 Child Repository Attribute:

**Join Conditions**

	Parent Repository Attribute	Child Repository Attribute
<input type="button" value="🗑️"/>	SKU Group Auto-Id	SKU Group Auto-Id

- a. **Child Repository Sort Attribute:** This attribute will be used to determine the order in which child records are displayed in the parent record. This attribute must be in the snapshot table.

If you want to be able to reorder the records, this attribute must be a dedicated attribute, meaning it must only be used for sorting. If a child record may be linked to more than one parent record, do not reorder the child records, because when you reorder the records in one parent record, it will disrupt the sort order in the other parent records.

- b. **Choose the Attribute Pair:** Join attribute pairs are used to identify linked records. They consist of an attribute in the parent repository and an attribute in the child repository. Both attributes must be in the snapshot table. For more details on join attribute pairs, see [Join Conditions](#).

Define one or more join attribute pairs:

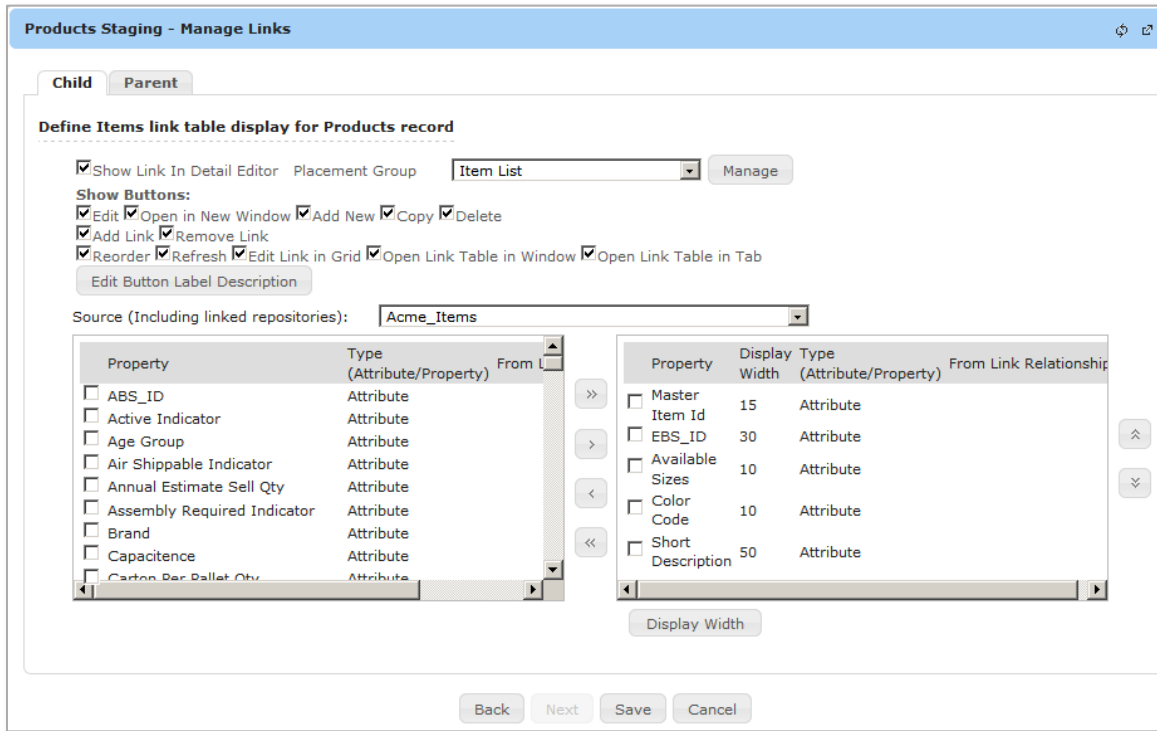
- To add a join attribute pair that does not use a literal value, (for information on literal values, see [Using Literal Values to Sort Child Records](#)):
    - (1) **Parent Repository Attribute:** Select the join attribute for the parent repository. The parent repository's join attribute's values must be unique, meaning that no two records in the repository will have the same value in their join attribute.
    - (2) **Child Repository Attribute:** Select the join attribute for the child repository. The child repository's join attribute must be dedicated – it must only be used to identify a child record's parent record.
    - (3) Click the **Add** button.
  - To add a join attribute pair that uses a literal value:
    - (1) **Parent Repository Attribute:** Scroll to the bottom of the attribute dropdown list and select **Enter Literal Value**.
    - (2) A field will appear to the right of the **Parent Repository Attribute** field. Enter the character string you will use to identify the subset of child records, for instance "Documents", (without the quotes).
    - (3) **Child Repository Attribute:** Select a dedicated attribute that will only be used to record which subset of child attributes it will be displayed in, for instance "Image Context", (without the quotes).
    - (4) Click the **Add** button.
  - To delete a join attribute pair, in the **Join Conditions** list, click the trashcan to the left of the attribute pair you wish to delete.
- c. When you have finished editing the join attributes, click **Next** to move to the **Display Options** screen.

7. The Display Option screen has two tabs:

- **Child:** These settings define which attributes from the child repository will be displayed when the parent record is opened in the Detail Editor, and how those attributes will be displayed.

- **Parent:** These settings define which attributes from the parent repository will be displayed when the child record is opened in the Detail Editor, and how those attributes will be displayed.

Edit the configuration options on the **Child** tab, then do the same on the **Parent** tab. The configuration options are the same on both tabs.

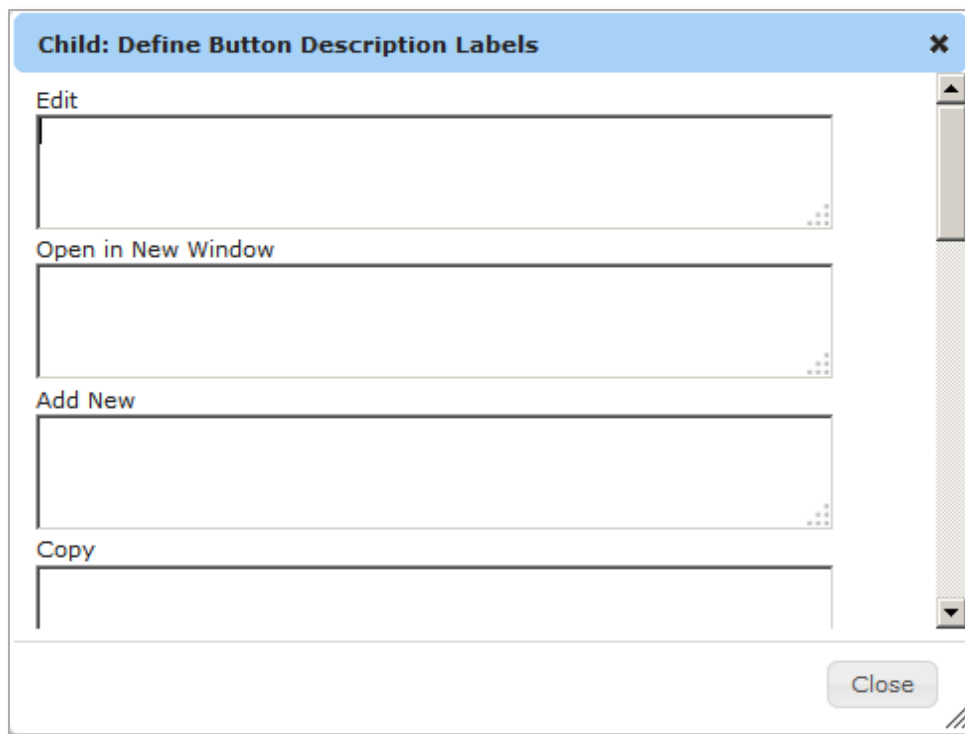


- **Number of Records Per Page:** How many records will be displayed on a page.
- **Window Display Height:** The height of the display, in pixels.
- **Show Link in Detail Editor:** If this is checked, the link table will be shown when the parent repository record is opened in the Detail Editor.
- **Placement Group:** When a record is opened in the Detail Editor, this is the Attribute Group and Attribute Tab in which the linked record's attributes will appear. Either select an existing Attribute Group and Tab or click the **Manage** button to add or change the Attribute Group or Tab.
- **Show Buttons:** These checkboxes control which action buttons will be available to users.

- **Open Link Table in Tab with Preference:** The linked records will be displayed in a tab, as a table. The only attributes that will be displayed are those that are included in the specified preference.

**NOTE:** The **Edit Link in Grid** option will allow the link record attributes to be modified from the link relationship table in an Excel-like format.

- **Edit Button Label Description:** Click this button to modify the default labels for the selected buttons:



- There are two lists of attributes. The attribute list on the left holds attributes that are available to be shown in the parent's record when it is in the Detail Editor. The attributes in the list on the right are the attributes that will be shown.
  - To move an attribute from one list to the other, select it and use the arrow buttons to move it to the left or the right.

- To change the display order of attributes in the Detail Editor, select one or more attributes and use the up and down arrows to move the attributes to the desired location.
- **Source:** To display attributes from another linked repository, select the repository in the **Source** dropdown list. That repository's attributes will be displayed in the list of available attributes.
- **Display Width:** To change the width of the boxes used to display the attributes, click the **Display Width** button, and edit the settings as desired.

Name	Display Width
Supplier Catalog Number	15
Catalog Product Description	100
Country Of Origin	20
Diversity	20

Save Cancel

- **Define Parent JSON Properties:** On the **Parent** tab, if you want to configure the Unified Parent Child View for the New UI, define the JSON properties that control it.

When the Unified Parent Child View capability is configured, when a child record is displayed in the Detail Editor in the New UI, selected attributes from the parent record are displayed in the group indicated by the setting **Placement Group**.

Note that the Unified Parent Child View capability does not support multi-edit (editing more than one record at once).

Build the JSON properties string to define how the parent record's attributes will be displayed.

**NOTE:** The quotes that surround the properties are conforming JSON (straight) double quotes, not beginning and ending double quotes. For example:

`"mergeView":true`

not:

`“mergeView”:true`

JSON Property	Purpose
<code>"mergeView":true</code>	This property is mandatory. It enables the Unified Parent Child View.
<code>"hideGrid":&lt;true-or-false&gt;</code>	<p>If this property is <code>false</code>, the parent record's attributes will be displayed in a table rather than in a list.</p> <p>The default value for this is <code>true</code>.</p>
<code>"filterByColocatePrefix":&lt;true-or-false&gt;</code>	<p>If this property is <code>true</code>, the only parent attributes that will be displayed are those whose name begins with the text string specified by the <code>colocatePrefix</code> property.</p> <p>The default value for this is <code>false</code>.</p> <p>If <code>"filterByPreference"</code> is set to <code>true</code>, it will override the <code>"filterByColocatePrefix"</code> setting. The attributes will be filtered by the specified preference instead of filtered by the prefix specified by <code>colocatePrefix</code>.</p>



JSON Property	Purpose
<pre>"colocatePrefix": "&lt;put-your-prefix-string-here&gt;"</pre>	<p>If this property is <code>true</code>, display the parent's attributes before the child attributes that begin with this string.</p> <p>If the <code>filterByColocate</code> property is <code>true</code>, the only parent attributes that will be displayed are those whose name begins with the text string specified by this property.</p>
<pre>"filterByPreference": &lt;true-or-false&gt;</pre>	<p>If this property is <code>true</code>, the only parent record's attributes that will be displayed are those that are included in the preference specified by the <b>Open Link Table in Tab with Preference</b> setting.</p> <p>The default value for this is <code>false</code>.</p> <p>If <code>"filterByPreference"</code> is set to <code>true</code>, it will override the <code>"filterByColocatePrefix"</code> setting. The attributes will be filtered by the specified preference instead of filtered by the prefix specified by <code>colocatePrefix</code>.</p>

Examples of Unified Parent Child View Configurations
<p>To display the parent record's attributes that begin with the text string "member":</p> <pre>{"mergeView":true, "filterByColocatePrefix":true, "colocatePrefix":"member"}</pre>
<p>To display the parent record's attributes in a table:</p> <pre>{"mergeView":true, "hideGrid":false}</pre>

### Examples of Unified Parent Child View Configurations

To display only the parent record's attributes included in the preference specified in the **Open Link Table in Tab with Preference** setting:

```
{"mergeView":true, "filterByPreference":true}
```

To display in a table the parent record's attributes that begin with the text string "member":

```
{"mergeView":true, "hideGrid":false,  
"filterByColocatePrefix":true, "colocatePrefix":"member"}
```

8. When you are finished editing the settings on the **Child** tab, move to the **Parent** tab. This tab defines which of the parent record's attributes will be displayed when a child repository's record is viewed in the Detail Editor. The configuration options for the **Parent** tab are the same as those on the **Child** tab. Configure the **Parent** tab.
9. When you are finished editing the **Parent** tab, click the **Save** button to save your changes to the link relationship.
10. Clear the cache in both the Classic UI and the New UI.

### Join Conditions

Join attribute pairs are used to identify linked records. They consist of an attribute in each repository. Both attributes must be in the snapshot table. Multi-language attributes cannot be used as a join attribute.

- Parent Repository's join attribute: The join attribute values in the parent's repository records must be unique, meaning that no two records in the repository will have the same value in their join attribute. Since each parent record has a different value in their join attribute, the join attribute value uniquely identifies each record.
- Child repository's join attribute: The child repository's join attribute must be dedicated – it is only used to identify a child record's parent record. When a child record is created, the value of the parent record's join attribute will be stored in the child record's join attribute. That means that from the child record, you can tell which is the parent record. This creates the link from the child record to the parent record.

For example, you could have a parent repository with an attribute called "ProductID" and a child repository that also has an attribute called "ProductID". (The attributes used to link records do not need to have the same names, but if they do, it is easier to see at a glance which attributes are being used as the join attributes.) When a record in the child repository is linked

to a record in the parent repository, the parent record's ProductID attribute's value is copied into the child record's ProductID attribute.

It is possible to define more than one join attribute pair. This may be useful, for instance, if you are using a set of primary keys to uniquely identify the parent record.

### *Using Literal Values to Sort Child Records*

You can define a linked relation such that when a parent record is displayed in the Detail Editor, a subset of the linked child records will be displayed. To do this, in addition to defining one or more attribute join pairs that identify a child record as belonging to a particular parent record, you define an attribute pair that identifies the child record's type.

To do this, the child records must contain an attribute whose value indicates what type of record it is, for instance, "Document", or "Image", or whatever other character string you want to use, (without the quotes). In the attribute pair, you would select this attribute as the child attribute.

For the parent attribute selection, you would scroll to the bottom of the attribute selection dropdown list and select **Enter Literal Value**. A field would appear to the right of the dropdown list that allows you to enter the character string for the type of child record you wish to display. You would enter one of the values stored in the children records' type attribute. For instance, if the set of values used in your children records' type attributes was "Document", "Image", and "Certificate", you would enter one of those strings, (without the quotes).

Literal value attribute join pairs are often used to display subsets of a record's digital assets. For instance, the linked relations defined may be:

- Product to Digital Assets
- Product to Digital Assets – Documents
- Product to Digital Assets – Images

In this case, when a product record is opened in the Detail Editor, it would display three lists of linked records:

- A list of all linked digital assets.
- A list of all linked digital assets that are documents.
- A list of all linked digital assets that are images.

An advantage of using literal value attribute pairs to display subsets of children records is that if when you have a parent record open in the Detail Editor, you add or upload a child record

through one of the lists of records of a particular type, EnterWorks will assume that the record you are adding must be of the same type and it will enter the type into the new child record's type attribute.

To continue our example, if I upload a digital asset through the list of "Image" child records, EnterWorks will set the type of the new record to "Image". If I enter it through the list of all digital assets (whose link relationship does not contain a literal value attribute pair), EnterWorks will not set the new child record's type.

## Link Management Troubleshooting

When a link relationship is created, there are two options for the **Link Type**:

- **Use Join Condition**: One or more pairs of attributes are used to link the repositories.
- **Manually choose record(s)**: Deprecated. **Do not use this**. If you choose this option, link tables may not load.

If **Manually choose record(s)** is selected, a variety of errors can occur, including:

- Linked Tables may not load or expand.
- In a Repository View, if the **Hierarchy** button is clicked, the Hierarchy/Taxonomy Manager will appear, but it may get stuck in the loading state and fail to list the existing hierarchies and taxonomy.
- Users may be unable to create the linked preference.
- In the log:

```
<drive>:\Enterworks\logs\EnableServer\enable_webMaster  
_<service-id>.log
```

the below error may appear, or the error may appear as a server error:

```
EPIMLogger.severe - getRepositoryLinkRelationshipList  
EXCEPTION: user=<user-name> userId=<user-id> link=<name-  
of-link> linkId=<link-id> repoId=<repository-id>  
org.hibernate.LazyInitializationException: failed to  
lazily initialize a collection of role:  
com.enterworks.epim.shared.model.BlinkRelationship.blinkJ  
oinConditions, could not initialize proxy - no Session
```

- Other issues.

To fix this situation, the link must be deleted and recreated to use a join condition.

## Configuring a Repository's Edit Screens

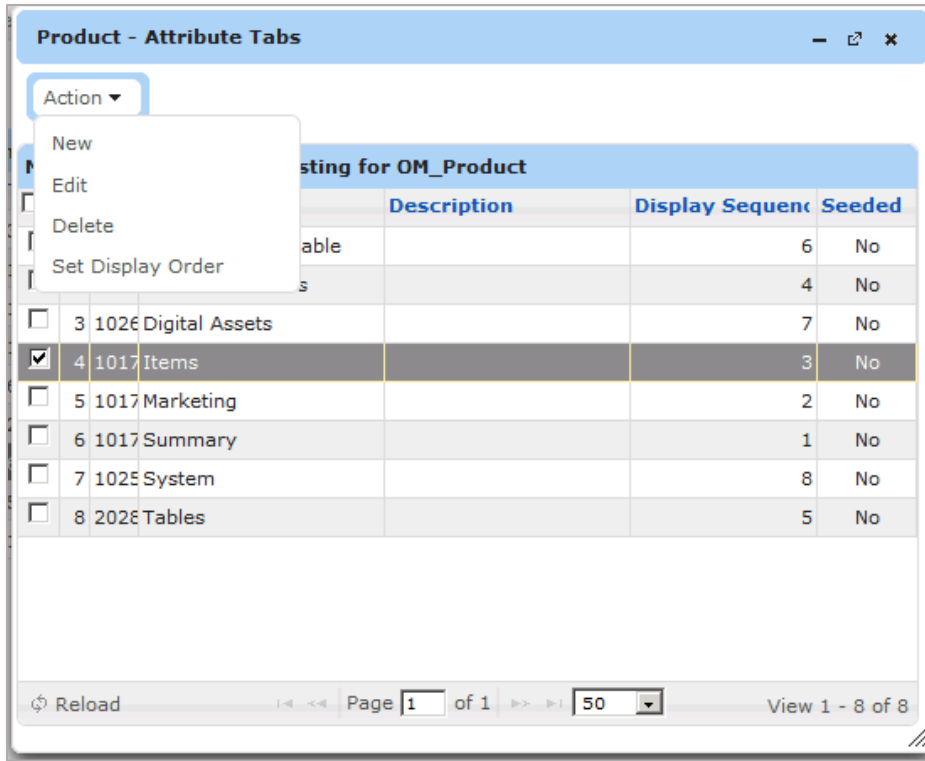
A repository's edit screens can be customized in the following ways by a Systems Administrator:

- **Tabs** – attributes can be arranged on one or more tabs and the order of the tabs can be specified.
- **Groups** – attributes can be grouped in one or more sections and the order of the groups can be specified.
- **Group Assignment** – attributes can be moved from one group to another.
- **Attribute order** – order of attributes within each group can be specified.
- **Preference Views** – views can be defined for each repository, controlling which attributes are visible. See the “**Using Repository Preference Views**” section in the User Manual for more information. (Note that if you are using the EnterWorks New UI, in order to see category attributes, the **Show Category Attributes** checkbox needs to be selected.)
- **Security** – security can be utilized for each repository to control which attributes can be seen and accessed (for example, read-only vs. read and edit). See the “**Managing Profiles**” section in the Administration Manual for more information.

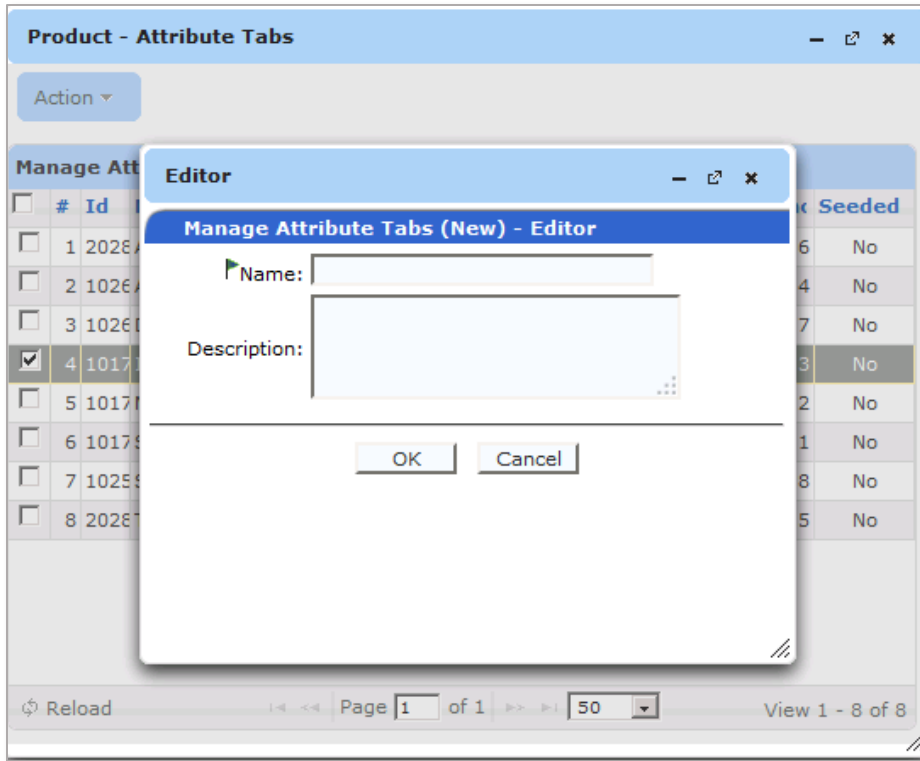
### *Tabs*

The following procedure describes how to configure/manage the editor Tabs. Use tabs to organize attributes into logical groupings.

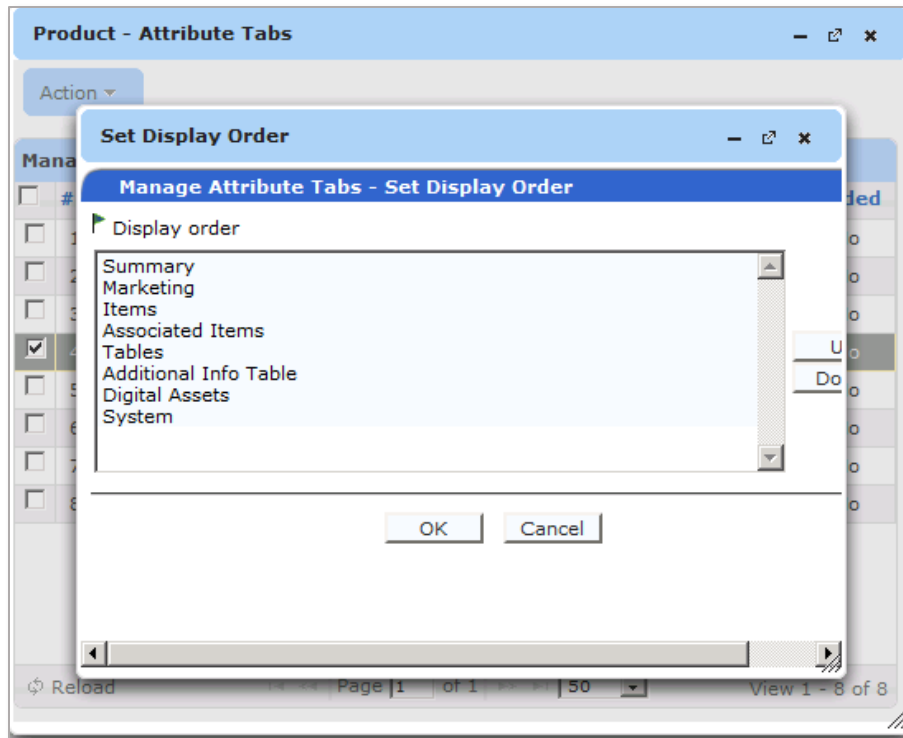
1. Log into EnterWorks Classic as a user with Administrative privileges.
2. From **Quick Links** dropdown, and select **Repositories**. The list of repositories will appear. Open the appropriate folder and select the desired repository.
3. Select the **Attribute Tabs** from the **Manage Model** drop-down menu.
4. The **Attribute Tab** listing will appear, showing the currently defined tabs and display order.



5. To create a tab, open the **Action** dropdown and select **New**.
6. To edit an existing tab, open the **Action** dropdown and select **Edit**.
7. The **Attribute Tab – Details** editor will appear.



8. Enter or edit the name of the tab and a description (optional). Do not include special characters in the name, such as & < > ".
9. Click **OK**. The **Attribute Tab – Details** editor will close. If you have created a new tab, it will be at the end of the list of tabs.
10. To reorder the tabs, open the **Action** dropdown and select **Set Display Order**.



11. Select a tab and use the **Up** and **Down** buttons to move it to the desired position. When you are finished, click the **OK** button to close the editor.
12. To delete a tab, open the **Action** dropdown and select **Delete**.

## Groups

The following procedure describes how to configure/manage the Groups editor. Use Groups to organize attributes into logical groupings shown as sections in a tab.

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. Open the **Repositories** tab, either from selecting it from the **Quick Links** dropdown or by opening the **Feature** bar, opening the **Model** folder, and selecting **Repositories**.
3. Open the appropriate sub-folder and select the desired repository.
4. Open the **Manage Model** dropdown and select **Attribute Groups**. The list of all attribute groups for the repository will be displayed, in the order they are displayed.



**Product - Attribute Groups**

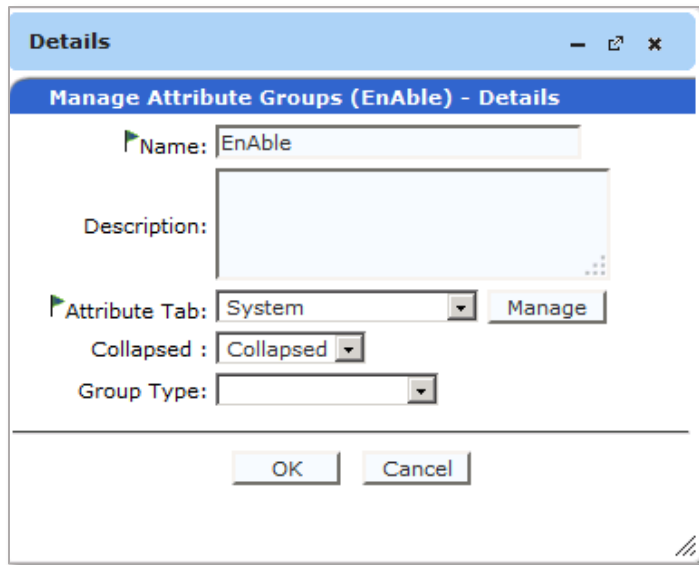
Action ▾

**Manage Attribute Groups -- Listing for Distributor Product**

<input type="checkbox"/>	#	Id	Name	Description	Tab	Group Type	Collaps	Display Sequ	Seeded
<input type="checkbox"/>	1	1117	Agency Status	(12,4)	System	Attribute	No	38	No
<input type="checkbox"/>	2	1161	BWC Assets	BWC Assets	Assets	Attribute	No	24	No
<input type="checkbox"/>	3	1117	Category Rules	Category Rules	System	Attribute	No	36	No
<input type="checkbox"/>	4	1161	Certifications	(12,4)	Certifications	Attribute	No	31	No
<input type="checkbox"/>	5	1155	Classification	(12,4)	Brownells	Variant	Yes	22	No
<input type="checkbox"/>	6	1119	DAM		Assets	LinkRelationship	No	23	No
<input type="checkbox"/>	7	1119	ETIM Specific Attributes	(12,3)	Variant Attributes	Variant	No	17	No
<input type="checkbox"/>	8	1120	Event Log		Project Mgt	LinkRelationship	No	34	No
<input type="checkbox"/>	9	1163	Exterior   Assignments	Exterior	Assets	LinkRelationship	No	28	No
<input type="checkbox"/>	10	1163	Facades   Assignments		Assets	LinkRelationship	No	26	No
<input type="checkbox"/>	11	1163	Floor Plans   Assignments		Assets	LinkRelationship	No	29	No
<input type="checkbox"/>	12	1120	Hierarchy   Assignments	(9,0)	Marketing	LinkRelationship	No	12	No
<input type="checkbox"/>	13	1162	Home Builder Variants		Product Detail	Attribute	No	5	No
<input type="checkbox"/>	14	1163	Home   Details		Summary	Attribute	No	2	No
<input type="checkbox"/>	15	1163	Home   Dimensions	(12,4)	Product Detail	Attribute	No	6	No
<input type="checkbox"/>	16	1117	Identifier	(12,4)	Summary	Attribute	No	1	No
<input type="checkbox"/>	17	1164	Inclusion   Assignments		Marketing	LinkRelationship	No	10	No
<input type="checkbox"/>	18	1163	Interior   Assignments		Assets	LinkRelationship	No	27	No
<input type="checkbox"/>	19	1119	Items		Variant Attributes	LinkRelationship	No	18	No

Page 1 of 1 | 50 | View 1 - 41 of 41

- To add a new group, open the **Action** dropdown and select **New**.
- To edit an existing group, select the group, then open the **Action** dropdown and select **Edit**.
- The **Attribute Group – Details** window will appear.



8. Edit the configuration options as desired:

- **Name:** The name of the new group. Do not include special characters in the name, such as & < > ".
- **Description:** There are two ways you can use the description field:
  - Enter a description of the group.
  - Enter text that dictates how many columns will be used to display the group's attributes in the EnterWorks New UI. The format for the text is:

*(<number-of-columns-the-group-uses>, <number-of-columns-for-each-attribute>)*

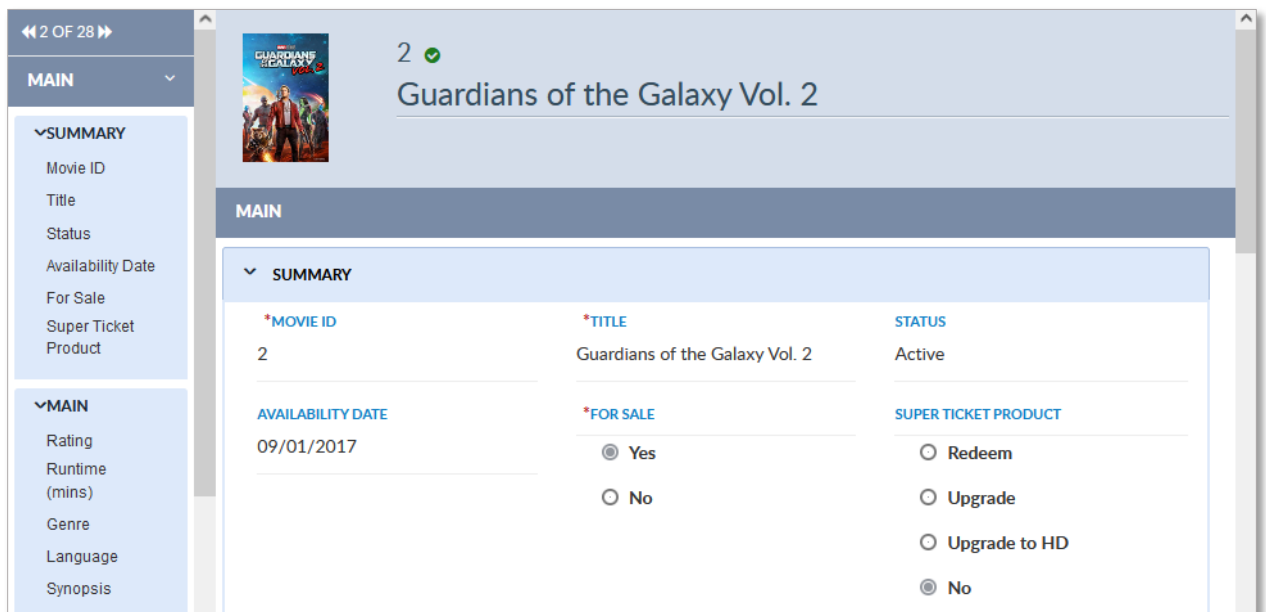
Note that there is no space between the comma and the second value.

- *<number-of-columns-the-group-uses>*: The width of the display area is 12 columns. This setting indicates the number of columns the group will span.
- *<number-of-columns-for-each-attribute>*: The area the group will use is also divided into 12 columns. This setting indicates how many columns in the group's area each attribute will use. If this number is set to 0 (zero), all 12 columns will be used, just as if it was set to 12.

For example:

- (12, 3): The group would use all twelve columns of the display area, (the entire width). The attributes would be displayed in four columns.
- (12, 6): The group would use all twelve columns of the display area. The attributes would be displayed in two columns.
- (6, 3): The group would use six of the twelve display columns. The attributes would be displayed in 4 columns. Each column would use three of the group's display columns.
- (3, 0): The group would use three of the twelve display columns. The attributes would be displayed in one column, (the full width of the group's display area).

The image below shows the Summary group configured in a (12, 4) format, which results in the attributes being displayed in three columns.



If an attribute group does not use the full width of the display area, if the next attribute group can fit in the remaining columns, EnterWorks will place the next attribute group alongside the first attribute group.

- **Attribute Tab:** Which attribute tab the group will be shown in.
- **Collapsed:** Whether the group will be initially displayed collapsed or expanded.
- **Group Type:**

- <not specified>: If this field is left empty, it is assumed that the group will be used to display attributes.
- Link Relationship: The group will be used to display link relationships.
- Variants: The group will be used to display variants. When a user tells EnterWorks to generate the variants, EnterWorks will use the attributes specified in the group to generate the variants. It will generate a record for each combination of existing values of the attributes.

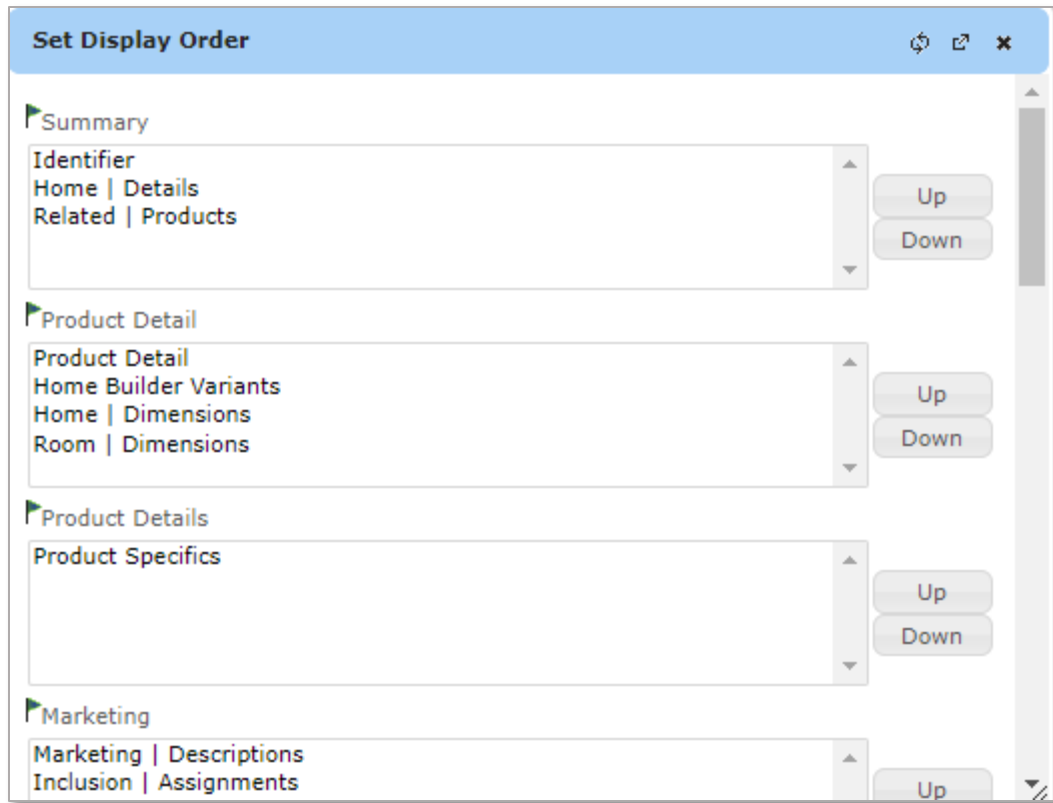
For instance, if the record's variant attributes have the following values:

- Color: Red, Green, Blue
- Size: Small, Medium, Large, Extra-large
- Sleeve Length: Long, Short

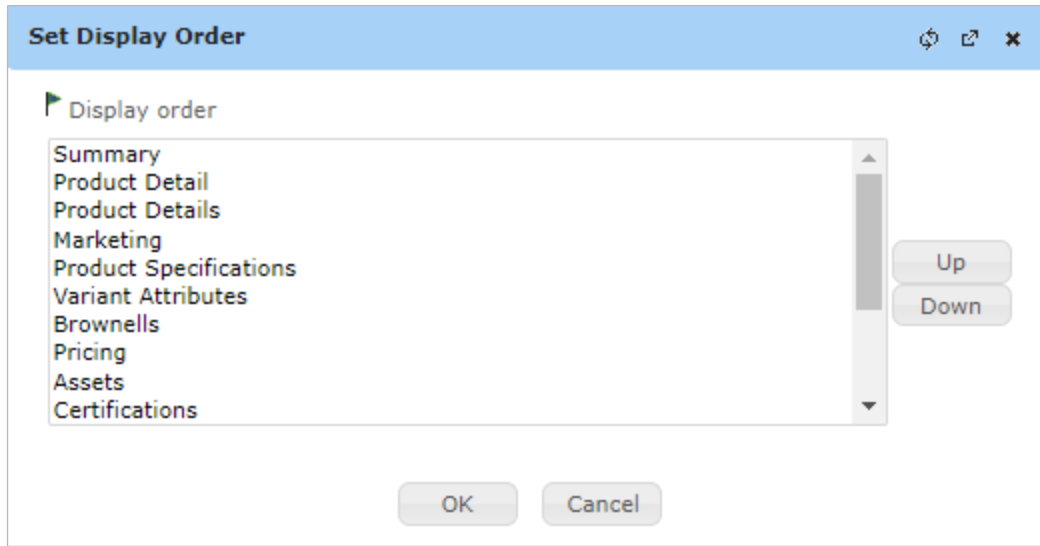
There would be 24 variant records generated, (3 x 4 x 2):

- Variant record 1: Red, Small, Long
- Variant record 2: Red, Medium, Long
- .... Etc. ....
- Variant record 24: Blue, Extra-large, Short

9. When you are finished, click **OK** to save your changes and exit back to the **Attribute Groups** list, or click **Cancel** to exit without saving your changes.
10. If you created a new group, it will be added to the end of the **Attribute Groups** list.
11. To change the order in which the groups are displayed, open the **Action** dropdown and select **Set Group Display Order**.



12. Each tab is listed and the groups are arranged in the tab as they will be displayed. To change the display order of a group, select it and use the **Up** and **Down** buttons to move it to the desired position.
13. When you are finished arranging the groups, click **OK** to save your changes and exit back to the **Attribute Groups** list, or click **Cancel** to exit without saving your changes.
14. From the **Attribute Groups** list, you can also define the display order of the existing tabs. To do so, open the **Action** dropdown and select **Set Tab Display Order**. The Set Display Order screen will open.



15. To change the display order of a tab, select it and use the Up and Down buttons to change its position in the list.
16. When you are finished arranging the tabs, click **OK** to save your changes and exit back to the **Attribute Groups** list, or click **Cancel** to exit without saving your changes.
17. To delete a group, on the **Attribute Groups** list, select the group, open the Action dropdown menu, and select **Delete**.

### *Group Assignment*

Attributes can be moved from one tab/group to another by editing the repository's Profile, (see [Profile Attributes](#)).

### *Attribute Order*

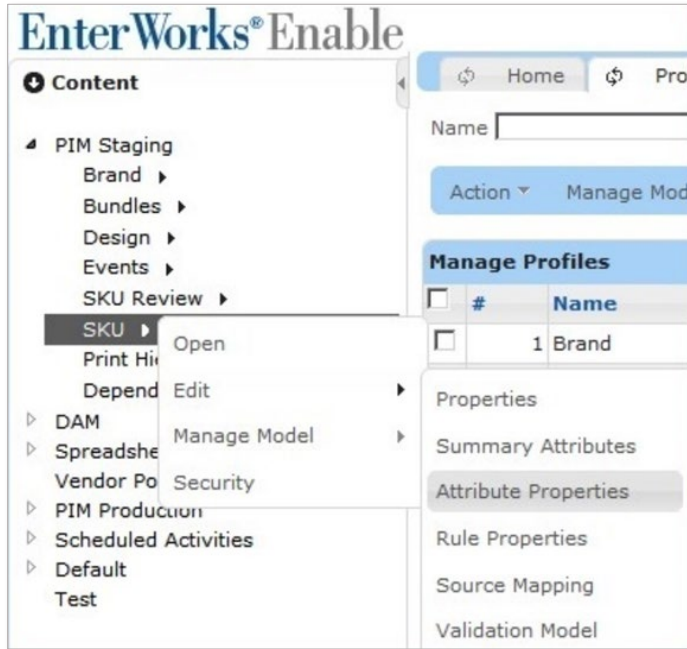
The display order of attributes can be changed by editing the repository's Profile, (see [Profile Attributes](#)).

## Managing Attributes

### Making an Attribute a Search Filter

To make an attribute a search filter:

18. On the **Feature** bar, in the **Content** tab, open the repository's folder.
19. Click the arrow to the right of the repository you want to make an Attribute Search Filter for.
20. Click on the **Edit** menu option.
21. Select the **Attribute Properties** menu option.



22. The **Attribute Properties** window will appear. It lists the properties for each attribute defined in the repository's profile.

SKU - Attribute Properties

Profile Name: SKU

Show All Hide All

**System**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Url
Design Code	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Brand Code	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Event Code	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Enable Status	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Active [Active]	<input type="checkbox"/>		
Legacy SKU	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No [No]	<input type="checkbox"/>		
Item Number	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Availability Catalog	INTEGER	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Page Number	BIGINT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Workflow Status	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Workflow State	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		

**Core**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Url
Item Number	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
UPC	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Created Date	DATE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Item Status Code	VARCHAR	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Quoted Item [22]	<input type="checkbox"/>		
Campaign Name	VARCHAR	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		
Workamajig Project Number	VARCHAR	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>		

**Planning**

Attribute	Type	Relational	Create Index	Drill Down Index	Default Value	Default Override Ind	Calculation Ind	Calculation Url
-----------	------	------------	--------------	------------------	---------------	----------------------	-----------------	-----------------

23. The attributes are organized by the tab they are assigned to, so find the attribute under the tab section and click the **Drill down Index** checkbox.

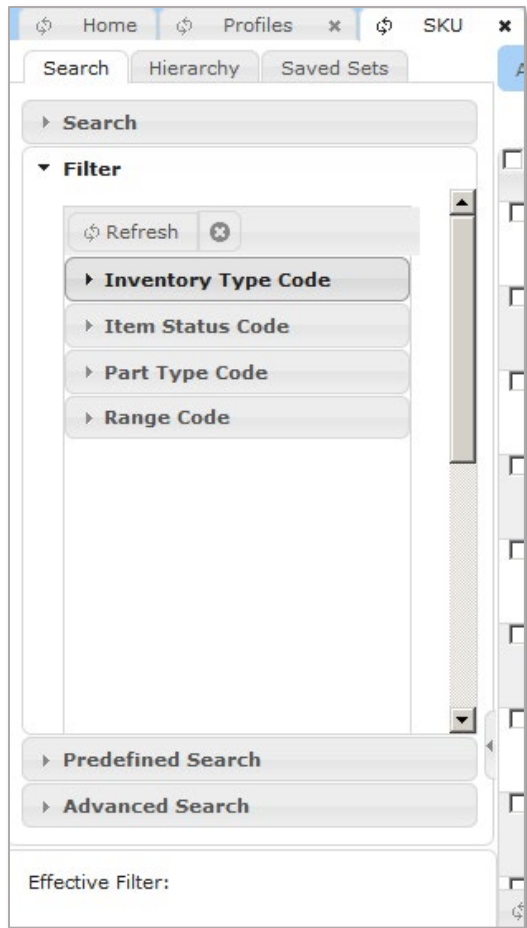
24. Click the **Save** button on the bottom of the list.

NOTE: To define the attribute as a Filter search field, the attribute Relational field must be checked also.

25. Open the repository or click the refresh  icon on the repository tab if it's already open.

26. The attribute will now appear as an option under the **Filter** area in the **Search** tab of the repository.





## Managing Snapshot Tables

External applications have the option to access EnterWorks data via ODBC accessible database views (snapshot tables) for each repository. The contents of these views are dictated by which attributes are identified as being relational (on the Repository Attribute Properties page). Any time the list of relational attributes changes, the corresponding views will be automatically updated.

Keep performance in mind: while you are able to make all of your repository attributes Relational (SQL Server maximum is ~1,000) it will affect your system performance. When a repository record is created or edited, the Snapshot table must be updated for all relational attributes. In addition, before insertion into or an update of the Snapshot table, each relational attribute value must be validated for data type and size restrictions.

For each repository attribute that is declared as Relational, there is the option to create a database index on the Snapshot table column. A database table index allows for significant

performance improvement when users will use the attribute frequently for searches in the repository. Keep in mind: It is important to keep the number of indexed columns as minimal as possible to keep the database workload to a minimum.

## Enable or Disable Multi-language Snapshot Table Support

EnterWorks supports multi-language attributes, which may have a value for each language EnterWorks has been configured to support. If a repository has multi-language attributes, and those attributes are defined as relational, and the repository has been configured to create multi-language snapshot tables, a multi-language snapshot table will be created for each language other than the default language.

Enabling or disabling multi-language support for a repository's snapshot table requires rebuilding the snapshot table. Rebuilding snapshot tables should be done after work hours to prevent users from accessing the tables while they are being rebuilt.

**Warning:** Configuring an attribute to be multi-language or not multi-language does not cause snapshot tables to be rebuilt. Enabling or disabling multi-language support for a snapshot table does not cause the snapshot tables to be rebuilt. When you enable or disable multi-language properties for an attribute, profile, or snapshot table, you must rebuild the snapshot tables to ensure the data in the snapshot table is current.

To enable or disable multi-language support for a snapshot table:

1. Log into EnterWorks Classic as a user assigned to the Administrator group.
2. Select **Repositories** from the **Quick Links** dropdown list on the top right of the screen, or from the Feature bar, open the **Model** tab, and select **Repositories**. The Repository list will open.
3. From the Repository list, single-click to select the repository that contains the attribute(s) to add to the snapshot table.
4. From the **Action** drop-down menu, select **Edit**, then select **Attribute Properties**. The profile Properties editor will open.
5. Set the **Snapshot Language Support** setting:
  - Checked: multi-language snapshot tables will be used.
  - Unchecked: multi-language snapshot tables will not be used.
6. Save and exit the Properties editor.

7. When you are finished making all the changes you want to make to the profile, rebuild the snapshot table.

## Add or Remove Attributes from a Snapshot Tables

To add (or remove) one or more attributes to a repository's snapshot table:

1. Log into EnterWorks Classic as a user assigned to the Administrator group.
2. Select **Repositories** from the **Quick Links** dropdown list on the top right of the screen, or from the Feature bar, open the **Model** tab, and select **Repositories**. The Repository list will open.
3. From the Repository list, single-click to select the repository that contains the attribute(s) to add to the snapshot table.
4. From the **Action** drop-down menu, select **Edit**, then select **Attribute Properties**. The profile Properties editor will open.
5. Check or uncheck the **Relational** checkbox for each attribute to be added to or removed from the Snapshot table.
6. Click the **Save** button at the bottom of the screen.

## Rebuilding, Dropping, or Repopulating a Snapshot Table

**Warning:** Depending on the number of records being processed, snapshot table jobs can take a significant amount of time to run and they can impact system performance. It is recommended that you run snapshot table jobs during non-business hours.

You can perform the following actions on a repository's snapshot table:

- **Full snapshot population:** This is also called "rebuilding" a snapshot table. Rebuilding a repository's snapshot table repopulates the entire table. EnterWorks will clear any existing records from the snapshot table, then for each record in the repository, write the values of the attributes included in the snapshot table to the snapshot table.
- **Full snapshot population and drop the snapshot table:** This is also called "dropping" the snapshot table. If you drop the snapshot table, the table will be entirely recreated, then EnterWorks will retrieve the values of all the attributes in the snapshot table for all the records in the repository, and store them in the new snapshot table.
- **Populate missing records:** If there are any records in the repository that have not been added to the snapshot table, EnterWorks will add those records to the snapshot table.

This is particularly useful if a snapshot job has been interrupted or aborted and not all records were added to the snapshot table.

To trigger a job to perform one of these actions:

1. Log into EnterWorks Classic as a user assigned to an Administrator group.
2. Select **Repositories** from the Quick Links dropdown list on the top right of the screen; or from the Feature bar, select **Model**, then select **Repositories**.
3. A window will open and all the repository folders will be displayed. Open the folder containing the desired repository, and select the repository.
4. Open the **Utilities** dropdown list, open **Job**, and select **Snapshot**.
5. The Snapshot Job window will appear. Select the desired options.
  - **Full Snapshot Population:** This will rebuild the repository's snapshot table.
    - **Drop Snapshot Table:** You can only select this if you have already selected **Full Snapshot Population**. This will cause the snapshot table to be recreated before it is populated.
  - **Populate Missing Records:** This will retrieve the attribute values for any records missing from the snapshot table and store those values in the snapshot table.
6. When you have finished making your selections, select **Populate Snapshot**.

## Auto-Generated Sequence Maintenance

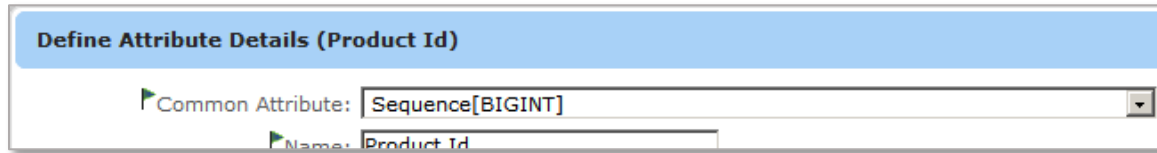
The following sections cover defining and/or maintaining an auto-generated sequence attribute for a repository.

### Defining the Sequence Attribute

To define the attribute that will use a generated sequence number definition.

1. In the Feature bar, right-click the desired repository, select **Manage Model**, then select **Profile**.
2. Click the **Next** button.
3. Select **New** from the **Action** drop-down list (or select an existing field and select **Edit**).

4. Select the Common Attribute of “Sequence” and that field will use the repository’s sequence definition to get the next sequential value if the field is blank when the record is saved.



The screenshot shows a dialog box titled "Define Attribute Details (Product Id)". It has two input fields: "Common Attribute" which is a dropdown menu currently showing "Sequence[BIGINT]", and "Name" which is a text box containing "Product Id".

## Adding a Sequence Definition

An auto-generated sequence value for a repository is controlled by the corresponding sequence definition associated to the repository. When a new record is created in the repository and no value for the auto generated id field is provided, the next sequence number is selected from the sequence definition. Even though a sequence definition can be shared by multiple repositories, it is recommended as best practice to create a separate sequence definition for each repository. A repository sequence definition is defined as follows:

1. Log into EnterWorks Classic as a user assigned to the Administrator group.
2. Select **Repositories** from the **Quick Links** dropdown list on the top right of the screen, or you can open the Feature bar, open the **Model** tab, and select **Repositories**.
3. From the **Repository** list, single-click to select the repository record that contains the auto-generated sequence attribute.

Name

Action ▾ Utilities ▾ Manage Model ▾

**Monitor for Enterworks® Enable™**

Repository	Type	Total	Valid	Errors	Not Va
Customer Products List		9	0	0	9
CustomerProd	Production	0	0	0	0
Facility		3	0	3	0
Icon		0	0	0	0
Item	Staging	16137	3759	6752	5626
Item Catalog		33465	0	0	33465
Item Prod	Production	13024	4820	8204	0
List		1	0	0	1
Manufacturer	Staging	226	214	10	2
Manufacturer Prod	Production	0	0	0	0
Product	Staging	6531	6225	306	0
Product Catalog Prod	Production	3031	940	0	2081

4. Select the **Edit/Properties** option from the **Action** drop-down menu. The Properties editor will appear.
5. Select the **Auto Generate Sequence** checkbox.

**PIM\_Product\_Staging - Properties**

Name:

Description:

Profile:

PreStaging/Staging/Production:

Repository Type:

Repository Folder:

Use Security Context:

Show Tab:

Tab Name:

Sequence:

Auto Generate Sequence:

Derive Indicator Digit:

6. Click the **Manage** button next to the **Sequence** drop-down list (above the Auto Generate Sequence checkbox) to create a new sequence definition.

**NOTE:** Sequence definitions can also be accessed via the **Model Configurations / Sequences** option on the Feature bar.

The screenshot shows a dialog box titled "PIM\_Product\_Staging - Properties" with a sub-section "Sequence Editor (New)". It contains four input fields: "Name", "GTIN Prefix", "Starting Sequence Number", and "Current Max Sequence". Below the fields are three buttons: "Save", "Cancel", and "Clear".

7. Enter a **Name** and **Starting Sequence Number** for the sequence number definition. Do not include special characters in the name, such as & < > ".  
8. Click the **Save** button.

### Adjusting the Starting Sequence Number

If the auto-generated sequence attribute is the primary key, a validation error will be shown if the value is not unique. This could indicate a need to reset the starting sequence value for the repository. Perform the following steps to adjust the next sequence for a repository.

1. Log into EnterWorks Classic as a user assigned to the Administrator group.
2. Find the highest currently used sequence value:
  - a. Open the repository.
  - b. If the sequence attribute is not listed in default preference view, select a user preference that contains the auto-generated attribute (or the "No Preference" option that contains all fields).

#	Error Ind	Over Err	Image	Product Id	Product Line	Product Name
1	✓	✓	NO IMAGE	49662		
2	⚠	⚠	NO IMAGE	49660		

- c. **Single-click** on the table header column for the sequence attribute. First time will sort ascending, second time will sort descending. Note the highest value for this attribute will appear as the first or last item.
3. Select **Repositories** from the Quick Links dropdown list on the top right of the screen, or you can open the Feature bar, open the **Model** folder, and select **Repositories**.
4. From the **Repository** list, select the repository record that contains the auto-generated sequence attribute.
5. In the **Action** dropdown menu, select **Edit**, then select **Properties**.
6. If the repository has an auto-generated sequence attribute the **Auto Generate Sequence** checkbox will be selected.

**PIM\_Product\_Staging - Properties**

Name: PIM\_Product\_Staging

Description:

Profile: Product

PreStaging/Staging/Production: Staging Repository

Repository Type:

Repository Folder: Office Manage

Use Security Context:

Show Tab:

Tab Name: Product

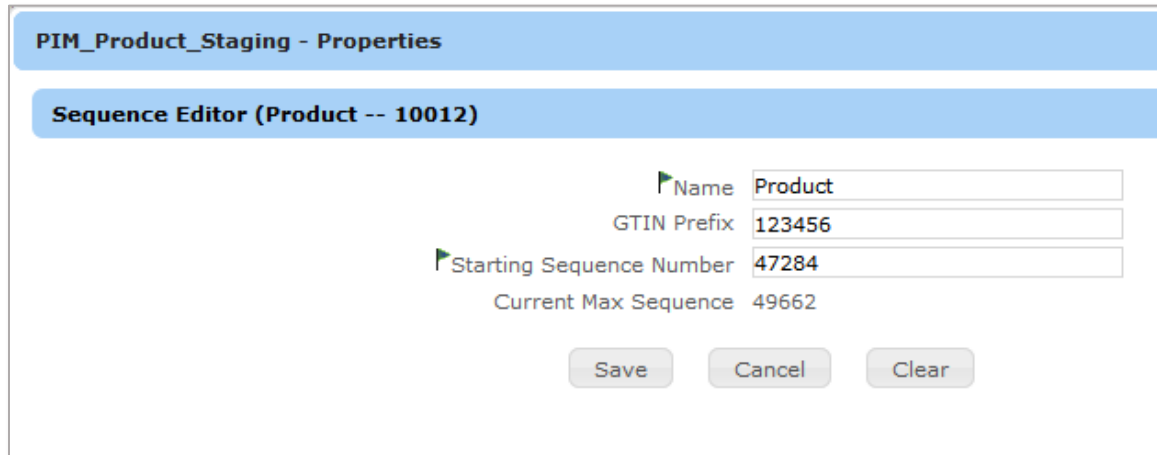
Sequence: Product Manage

Auto Generate Sequence:

Derive Indicator Digit:



7. Click the **Manage** button next to **Sequence** drop-down list to change the next sequence number to use, which is the field above the checkbox.



The screenshot shows a dialog box titled "PIM\_Product\_Staging - Properties" with a sub-section "Sequence Editor (Product -- 10012)". It contains the following fields and values:

Name	Product
GTIN Prefix	123456
Starting Sequence Number	47284
Current Max Sequence	49662

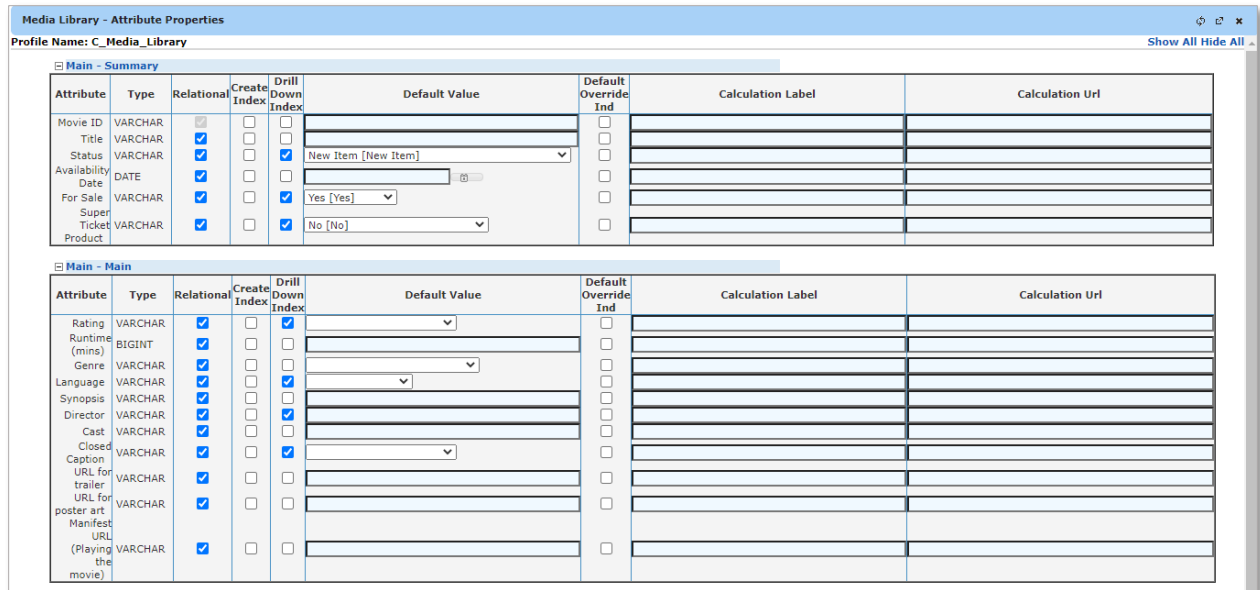
At the bottom of the dialog are three buttons: "Save", "Cancel", and "Clear".

8. Change the **Starting Sequence Number** to the next number you want the system to use for the auto-generated field. (Set the **Starting Sequence Number** to number greater than the value retrieved in step 2.)
9. Click the **Save** button and a confirmation window will be shown.

## Managing the Custom Controls to select linked data

Perform the following steps to configure the custom JSP UI control that is to be displayed next to a repository attribute.

1. Log into EnterWorks Classic as a user in the Administrator group.
2. In the **Feature** bar, select the desired repository, click **Edit**, and select **Attribute Properties**.
3. The **Attribute Properties** screen for the selected repository will appear.



4. In the attribute's **Calculation Label** field, enter the text you want to appear when the user hovers over the button to the right of the field.
5. Click the attribute's **Calculation URL** field. The **Attribute Calculation URL Setup** editor will appear. Set the configuration values as desired:
  - a. **Calculation Ind.:** The Calculation Indicator controls how the control will behave when the control is clicked.
    - i. <none>: No calculation indicator will be displayed. This is the default for any attribute for which no calculations are to be performed.
    - ii. **Open as Dialog:** This opens the selected JSP page as a dialog prompt.
    - iii. **No User Interface:** This opens the selected JSP page in the background. There will not to be any user input fields on the form.
    - iv. **Open as Window:** This opens the selected JSP page in a new window.
    - v. **Hidden Window:** Similar to No User Interface. No pop-up is shown.
    - vi. **Control with Window:** A pop-up window will appear that has a button for the attribute. The attribute will not be updated.
    - vii. **Control with Hidden Window:** A hidden pop-up window will appear that has a button for the attribute. The attribute will not be updated.

viii. **Calculated Attribute:** The attribute value may be updated by the custom control. This is used for an attribute that is going to be updated by a pop-up page but there is not a button to open the page.

b. **Calculation URL:** This is the name of the JSP page you want to invoke.

**Note:** The EnterWorks Classic UI calls a JSP page. The New UI looks for an HTML page with the same name as the JSP page specified here and uses it. If this control will only be used in the New UI, the contents of the JSP page are not used, however, if the control needs to operate in both UIs, both the JSP page and the HTML page need to be operational.

6. When you are finished editing the custom controls, click the **OK** button to save your changes and exit. Default mapping messages will indicate updates or snapshot update.

## Migrating EnterWorks

The EnterWorks migration feature allows for the transfer of the following data model components from one EnterWorks server to another:

- Users and user groups.
- Data sources.
- Security filters.
- Transmission options.
- File definitions.
- Home configuration.
- Server Properties.
- Data model objects: such as Repositories, Code Sets, Profiles, etc.

For example, a common installation setup consists of three separate environments: DEV, QA, and PROD. The data model is developed in the DEV environment, tested in the QA environment, and put into production in the PROD environment. The migration feature is designed to facilitate the transfer of initial data model components, plus all future changes and improvements, from DEV to QA, and ultimately to PROD. Each data model component has an underlying unique identifier assigned to it, which ensures the correct migration of components even if their names have been changed.

For example, if we have an attribute called “User Desc” in the DEV environment that is migrated to QA and PROD as “User Desc”. Later, the attribute name is changed to “User Description” but all other qualities of the attribute remain the same. A second migration of the profile containing this attribute would successfully locate the “User Desc” attribute in the QA and PROD environments and update its name to be “User Description”.

The migration flow should always be from DEV to QA and then to PROD. Development changes and enhancements should always be performed in DEV, migrated to QA, and then migrated to PROD. Conforming to this protocol allows migration to properly locate the data model components to change.

Examples of reasons you may want to migrate data model components:

- EnterWorks Data Model objects:
  - Code set definition changes.
  - Addition, modification (for example, character length, data type, etc.) or removal of attributes to any profiles.
  - Hierarchy or Taxonomy changes (for example, addition of new nodes).
  - Change to attribute order, group, or tab assignments.
- Repository Migration:
  - Addition of attributes to the Snapshot Table.
  - Change of default values.
  - Change of assignment of Pop-Up controls and pages.
  - Change in Trigger or Promotion properties.

For a detailed description of when you should migrate data model components, which components should be migrated together, how migration of the data model fits in the EnterWorks deployment process, and how to migrate EPX, see *Best Practices for EnterWorks Deployments and Migrations*.

## Migration Process Overview

The below is an overview of the process for migrating EnterWorks data model components:

1. Migrate out the desired data model components from the source environment.
2. Migrate in the desired data model components to the target environment.
3. On the target environment:

- a. Perform any necessary data model component integration as specified in *Best Practices for EnterWorks Deployments and Migrations*.
- b. [Update the Snapshot Table](#).
- c. [Restart EnterWorks services](#).

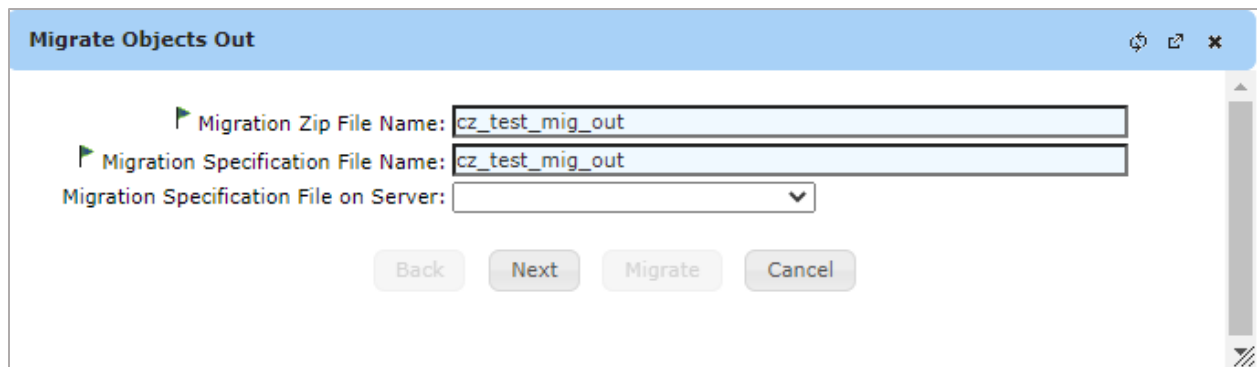
The following sections provide detailed directions for migrating data model components.

## Migrate Out Data Model Components

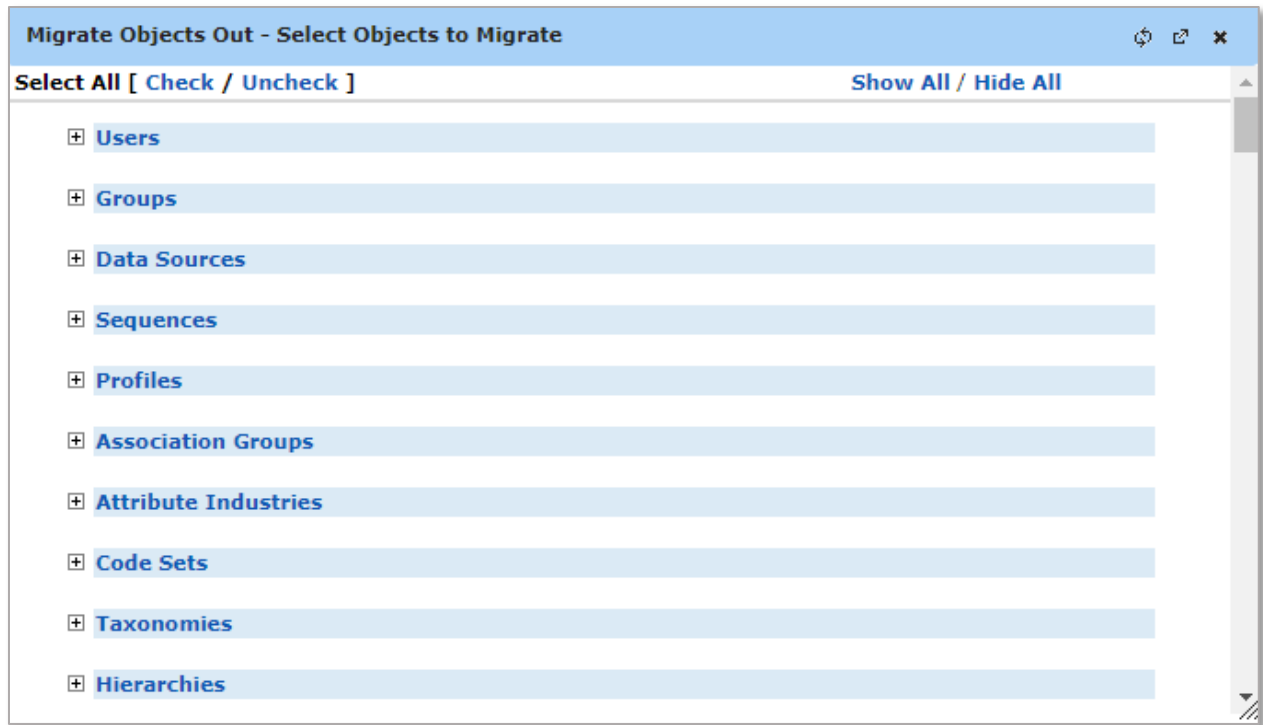
Perform the following steps to migrate the EnterWorks data model components from one EnterWorks server to another.

**NOTE:** Before performing these steps, be sure there is a backup of the target EnterWorks database in case migration failures necessitate restoring the database from the backup.

1. Log in to the source EnterWorks server (for example, QA) as a system administrator.
2. Click on the **Migration** option on the feature bar and select **Migrate Out** function. The Migrate Objects Out page appears.



3. Enter the name for the **Migration file** in the Zip and **Specification file** name fields. The actual file name will include a time-based number to ensure uniqueness.
4. Optionally select an existing **Migration Specification** file. This will pre-select the objects to be migrated.
5. Click **Next**. The Select Objects to Migrate page appears.




*Migrate Objects Out page with all dropdowns collapsed.*

6. To collapse all the dropdown menus, click **Hide All**. To expand all dropdown menus, click **Show All**. To expand or collapse a dropdown menu, click the + or – icons to the left of the dropdown name.
7. Scroll down the list of data model components and select the items you would like to migrate by checking the checkbox to the right of the item's name. Click **Check** or **Uncheck** to select or unselect all the items in a dropdown.
8. When you have finished selecting the components to migrate, scroll to the bottom of the list and click **Migrate**.
9. Check the **Job Monitor** to see the progress of the migration. You may need to refresh the **Job Monitor** tab.
10. Repeatedly click **Refresh** to monitor the progress of the migration, until the Status shows **Completed**.
11. Two migration files will have been created in

`<install_drive>:\Enterworks\shared\migration`

They are:

- <your-file-name>.zip
- <your-file-name>.msf: This is the migration specification file.

12. To download the log file for the migrate out procedure, in the **Job Monitor**, click the job's Download Log File icon .

## Migrate In Data Model Components


To migrate in data model components:

1. Copy your migration .zip and .msf files from the folder:

```
<install_drive>:\Enterworks\shared\migration
```

folder on the source system (for example, QA) to the same location on the target system (for example, PROD).

2. Log into the target EnterWorks server (for instance, PROD) as a user who belongs to the System group.
3. Click on the **Migration** option on the feature bar and select **Migrate In** function. The **Migrate Objects In** page appears.
4. Select the .zip and .msf files and click **Next**. The **Select Objects to Migrate** page will appear.
5. The **Select Objects to Migrate In** page contains a list of dropdowns, one for each type of component that can be migrated. All dropdowns will be expanded and all objects that can be migrated in will be pre-selected.
6. To select all available components, click the **Overwrite All Check link**.
7. Scroll to the bottom of the list and click **Migrate**.
8. Check the **Job Monitor** to see the progress of the migration. You may need to refresh the **Job Monitor** tab.
9. If you have imported Groups, select the **Users and Groups** Feature and select the **Groups** function. The list of groups will appear.
10. For each group, perform the following:
  - a. Click the checkbox next to the group record, open the **Action** dropdown, and select **Security**. The security page for the group will appear.
  - b. Click the **Hide All** link on the top right and open the Code Sets table.

- c. Click the **Check** link under the **Read** column to ensure all code sets are readable by the group.
  - d. Click **Save**. The changes to security are saved.
11. To download the log file for the migrate out procedure, in the Job Monitor, click the job's Download Log File icon . Open the migration log file and check for errors. If the errors are substantial and cannot be quickly remedied, abort the migration attempt by restoring the target EnterWorks database from the backup.

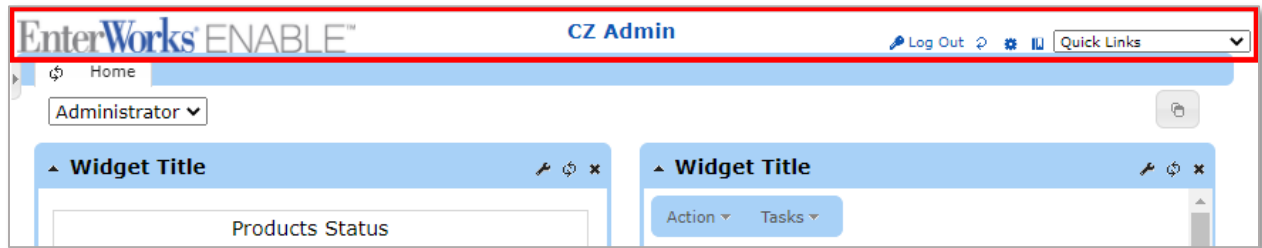
## System Administration

### UI Customization

#### Customize the EnterWorks Classic UI Title Bar

Customizing the EnterWorks Classic UI's title bar is a good way to provide a visual aid to administrators who are working on multiple EnterWorks environments. If you use different colors for the title bars, it is easy for administrators to see at a glance which environment they are working on.

Upon installation, the EnterWorks title bar looks like the below image.



To customize the EnterWorks title bar:

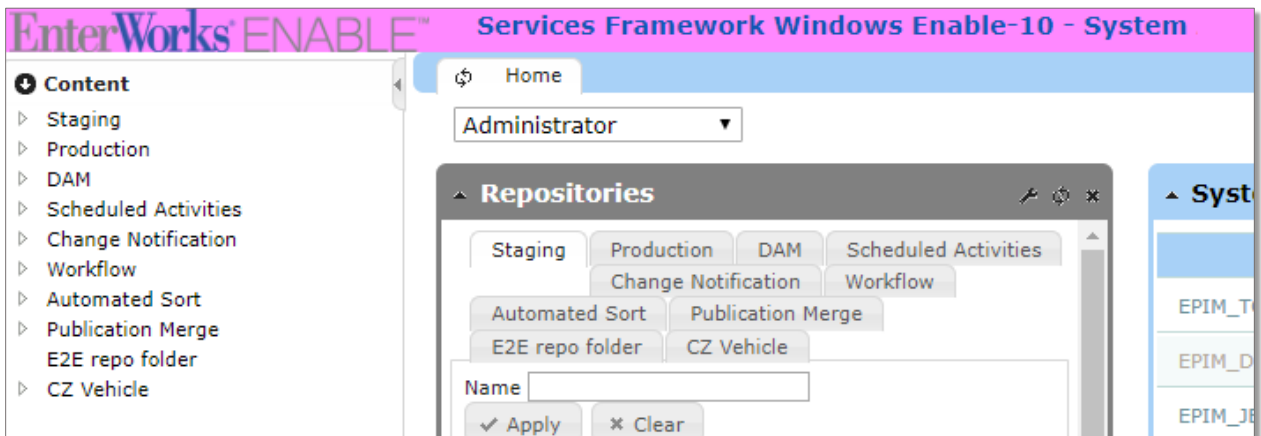
1. Log into the EnterWorks Classic UI.
2. In the `sharedConfig.properties` file, edit the following fields.
  - `enviroment.color`: This hex number is an RGB value for the title bar's color. This will affect only the Classic UI.



- `enviroment.type`: This value will appear as the text in the title bar in both the Classic and EnterWorks 2020 UIs.

```
70 *****
71 ##
72 ##
73 ##
74 webapps.baseUrl=http://IP-AC1F50BD/webcm/
75 webservices.baseUrl=http://IP-AC1F50BD:8090/webcm/
76
77 enviroment.type=Services Framework Windows Enable-10
78 enviroment.color=#ff88ff
79
80 ##
81 ##
82 ## The directory under /webcm to place the custom JSP pages which
83 ## perform attribute values calculations
84 format_attribute_calculation_custom_jsp_dir=/custom
```

3. Restart all services.
4. After you restart services, you may need to log in again to see the title change take effect.



## System Architecture

### EnterWorks Environments

EnterWorks recommends at least two separate environments for the EnterWorks implementation: DEV and PROD. The DEV server should be used to test system upgrades, import templates, and system changes before applying to production.

If possible, a QA test environment should also be set up and configured to mimic PROD. This is especially useful for system upgrade testing.

### Controller and Workers Services

Some user tasks such as Importing, Exporting, Validation, and Mass updates are submitted as background jobs. These background jobs are distributed to an available free Worker service to execute. The Controller service monitors for new jobs. When it receives one, it determines if it can be run simultaneously with other jobs or if it must be run sequentially, then sends the job to the next free Worker service. If the job is run on a separate server, log files associated with the job will be located with that server. In general, the greater the number of Workers configured, the greater the number of jobs that can run concurrently. However, some jobs will be serialized due to resource constraints. For example, import jobs to the same target repository will be processed serially, regardless of the number of configured workers.

**NOTE:** In distributed environment, for the EnterWorks system to access files, the services must be run using a network account that has permission to access shared resources for all locations where the EnterWorks application is installed.

### Configuration Files

The following configuration files have settings that are unique to an organization or an organization's environment (DEV, QA, and Production). They include configuration settings, custom property files, and settings within standard property files.

The folder/directory tree structure is the same for all installations; however, the location of the folder/directory tree varies depending on system configuration.

## EPIM Configuration Files

### Shared Properties

EPIM has shared configuration settings, such as: the location of EPIM folders; the server host name; port usage; database connection information; and controls for triggers, promotion configuration, the UI, and publication options.

The `sharedConfig.properties` file used to be the only location of these settings, but now many of them are stored in the Shared Configuration List in the EPIM database and can be accessed through the EnterWorks Classic UI. In order to maintain backward compatibility, if a property is not available in the database, EnterWorks will fall back to reading the property from the `sharedConfig.properties` file.

### Installing and Patching EnterWorks

Fresh installs of EnterWorks populate the Shared Configuration List by running the seed script:

```
EPIM_SQLServer_DB_SharedConfig_102.sql
```

If you are patching EnterWorks, edit the seed script to supply any custom configurations, then run the seed script. Properties that are already defined in the Shared Configuration List will NOT be updated by the script. Their values can only be updated through the UI. It is possible that a patch may have additional properties defined in the script. Only these new properties will be added.

**CAUTION:** If you are patching EnterWorks and the SQL script is not updated to contain the values in the current `sharedConfig.properties` file, it is possible that when you run it you will be changing the configuration of your system, which may lead to unexpected results. Do not run the seed script until you compare it with the current `shared.Config.properties` file.

### Shared Configuration List

Anytime a change is made to this file the EnterWorks services must be restarted.

To modify the Shared Configuration List:

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. In the Feature bar, open the **System** tab and select **Shared Configuration**.
3. The **Shared Configuration List** will appear.

**Tip:** If a field value is too long to be seen comfortably, double-click it and the property will be opened in the editor where the value can be seen in its entirety.

**Shared Configuration List** 🔍 ↻ ✕

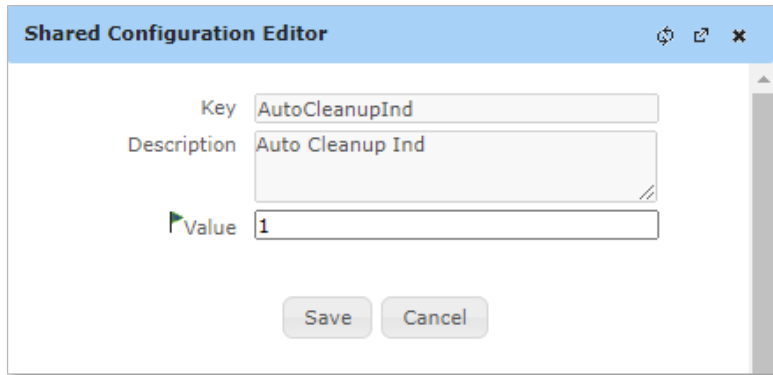
Name  ✓ Apply ✕ Clear

#	Key	Value	Type	Description	Proper
1	AutoCleanupInd	1	ServerProperties	Auto Cleanup Ind	1
2	AutoCleanupInterval	24	ServerProperties	Auto Cleanup Interval	2
3	TimeoutSessionsInd	1	ServerProperties	Timeout Sessions Ind	3
4	TimeoutSessionsInterval	24	ServerProperties	Timeout Sessions Interval	4
5	CleanJobHistoryInd	1	ServerProperties	Clean Job History Ind	5
6	CleanJobHistoryInterval	30	ServerProperties	Clean Job History Interval	6
7	CleanItemHistoryInd	0	ServerProperties	Clean Item History Ind	7
8	CleanItemHistoryInterval		ServerProperties	Clean Item History Interval	8
9	CleanSystemLogsInd	1	ServerProperties	Clean System Logs Ind	9
10	CleanSystemLogsInterval	7	ServerProperties	Clean System Logs Interval	10
11	S_SSO_CERTIFICATE	14560B1F3BBFA	ServerProperties	Single Sign On certificate	11
12	JOB_BLOCKING_LEVEL_1	Repository	JobTypePropertie	Import Job Blocking Level	1000
13	JOB_BLOCKING_LEVEL_3	None	JobTypePropertie	Validation Job Blocking Level	1000
14	JOB_BLOCKING_LEVEL_4	...	JobTypePropertie	...	1000

Page 1 of 15 50 View 1 - 50 of 741

Close

4. Descriptions of properties are provided in the **Description** column. For more information on a particular setting, please contact your EnterWorks account representative.
5. Double-click the setting you wish to change. A **Shared Configuration Editor** will appear.



6. Set the Value as desired, then click the **Save** button.
7. The setting will be changed to reflect your changes. Clear the data cache and restart EnterWorks services.

### *sharedConfig.properties File*

Anytime a change is made to this file the EnterWorks services must be restarted.

There are multiple `sharedConfig.properties` files: one for Tomcat; one for Jboss Controller; and one for each Jboss Worker. With the exception of a small number of properties, the multiple `sharedConfig.properties` files must be the same. If you update a property in one of the `sharedConfig.properties` files, make the same change to each of the other copies of the file, unless the property is specific to the service to which it applies.

The following properties may be different in each `sharedConfig.properties` file:

Property	Description
<b>jboss.context.url</b>	Defines the server on which the process resides and the port number on which it listens: <ul style="list-style-type: none"> <li>• 8034 – Tomcat and Controller</li> <li>• 8x34 – Worker (where x is the worker number)</li> </ul>
<b>epim.connection.url</b>	Potentially different on each server if the <b>appName</b> property is set so that each connection to the EPIM database can be traced to the specific process.
<b>debug.file.prefix</b>	Prefix for the debug log file for the process. <ul style="list-style-type: none"> <li>• enable_webController</li> <li>• enable_appController</li> <li>• enable_appWorker&lt;n&gt;</li> </ul>

Property	Description
<b>job.controller.type</b>	Type of process: <ul style="list-style-type: none"> <li>• 0 – independent (Tomcat and JBoss Controller when there are no workers)</li> <li>• 1 – Controller (Jboss Controller)</li> <li>• 2 – Worker (Jboss Worker)</li> </ul>

### Configure Validation Levels

Validation levels are defined in the Shared Properties, (see [Shared Properties](#)).

- **validation.error.stop:** This value will only affect validation jobs and sync out jobs.
  - 1: Stop validating all rules for an item after encountering first error.
  - 0: Continue validating all rules for an item even after encountering error.
- **validation.level.num.active:** Holds the number of active validation levels.
- **validation.job.level.<x>:** These hold the names (labels) of the validation levels. Level e is the lowest quality bar to pass.
- **item.default.validationLevelInd:** Default validation level indicator for new repository items. Possible values are: 1,2,3,4,5; where 1 is level a.
- **validation.write.to.log**

### Configure Google Translate API

In order to use the Google Translate API, the customer must have created an account with Google and obtained the following:

- URL for the Google IP site.
- Google Translate API key

Configuration of the Google Translate API is performed by modifying Shared Properties values, (see [Shared Properties](#)).

If you are configuring Google Translate API by modifying the `sharedConfig.properties` file and the following settings do not appear in the file, stop the services using the `STOP ENTERWORKS.cmd` script, add the property settings to the bottom of the `sharedConfig.properties` file, and start the services again using the `START ENTERWORKS.cmd` script.

- **google.translate.enabled:** Specified whether the translation is enabled.
  - 1: Google Translate is enabled.
  - 0: Google Translate is disabled.
- **google.translate.refer.site:** URL for the Google IP site.
- **google.translate.key:** The Google Translate API key.

### *Regional Language Login Access*

Configuration of Regional Language functionality is performed by modifying Shared Properties values, (see [Shared Properties](#)).

If you are configuring Regional Language functionality by modifying the `sharedConfig.properties` file and the following setting does not appear in the file, stop the services using the `STOP ENTERWORKS.cmd` script, add the property to the bottom of the `sharedConfig.properties` file, and start the services again using the `START ENTERWORKS.cmd` script.

- **allowRegionalLogin:**
  - true: the Regional Login link is to be available on the Login Page.

### *EPIM .bat Files*

The EPIM .bat files are:

- `service.bat`
- `standalone.bat`
- `standalone.conf.bat`

These files contain Tomcat, JBoss Controller, and JBoss Worker settings, including:

- Class paths to custom library files and log level and log file settings.
- Memory setting information for the JBoss JVM.
- EPIM product Master JBoss service settings.

Generally, changes should only be made to these property files based on direction from a patch deployment procedure. Any time a change is made to these files, the EnterWorks services must be restarted.

service.bat	
Component	File Path
Tomcat	<install_drive>:\Enterworks\EnableServer\tomcat\bin\
JBoss Controller	<install_drive>:\Enterworks\EnableServer\jbossMaster\bin\service\
JBoss Worker<n>	<install_drive>:\Enterworks\EnableServer\jbossSlave<n>\bin\service\

standalone.conf.bat	
Component	File Path
JBoss Controller	<install_drive>:\Enterworks\EnableServer\jbossMaster\bin\standalone\
JBoss Worker <n>	<install_drive>:\Enterworks\EnableServer\jbossSlave<n>\bin\standalone\

standalone.bat	
Component	File Path
JBoss Controller	<install_drive>:\Enterworks\EnableServer\jbossMaster\bin\standalone\
JBoss Worker <n>	<install_drive>:\Enterworks\EnableServer\jbossSlave<n>\bin\standalone\

### workFlowHandlerConfig.properties

**workFlowHandlerConfig.properties** – This properties file has been deprecated. For more information, contact your EnterWorks account representative.

### EPX Property Files

#### config.properties

The config.properties file contains configuration settings for the Web Server's EPX component, including:

- The location of EPX folders.
- The server host name.
- Port usage.



Generally, changes should only be made to this property file based on direction from a patch. Any time a change is made to this file the EPX services must be restarted.

This file is located in:

```
<install_drive>:\Enterworks\EPX\bin\
```

### esjdbc.properties

The `esjdbc.properties` file contains configuration settings for EPX, including information for connecting to the SQL Server database.

The password to the database is encrypted. Use the script in:

```
<install_drive>:\Enterworks\EPX\DesignConsole\getEncryptedString.bat
```

to get the encrypted password.

Generally, changes should only be made to this property file based on direction from a patch. Any time a change is made to this file the EPX services must be restarted.

This file is located in:

```
<install_drive>:\Enterworks\EPX\bin\
```

### Enterworks.properties

The `Enterworks.properties` file contains custom settings, including:

- Scheduled Exports, Imports, Promotions, Package Promotions Workflow.
- Email notification.
- Database connectivity for EPIM and EPX databases.
- Server connectivity for EnableServer (Tomcat/Jboss) and EPX (Tomcat/EPXJboss).
- Amazon S3 connectivity.
- Validation Levels.
- Miscellaneous defaults.

Any time a change is made to this file the EPX services must be restarted.

This file is located in the following directories:

- `<install_drive>:\Enterworks\EnableServer\jbossController\standalone\configuration\conf`
- `<install_drive>:\Enterworks\EnableServer\jbossWorker<n>\standalone\configuration\conf`
- `<install_drive>:\Enterworks\EnableServer\tomcat\webapps\wcbcm\WEB-INF\classes`
- `<install_drive>:\Enterworks\EPX\bin\conf\`

### service.bat

**service.bat** – Contains the EPX product Tomcat and JBoss service settings, including:

- Class paths to custom library files.
- Memory setting information for the Tomcat JVM.

Generally, changes should only be made to this property file based on direction from a patch. Any time a change is made to this file the EPX services must be restarted.

This file is located in:

```
<install_drive>:\Enterworks\EPX\tomcat\bin\
```

### EnterWorks DAM & PUB

The EnterWorks DAM settings are controlled through the EnterWorks `DAMConfig` repository. Variant settings are set in the `DAMVariant` repository and the file:

```
<install_drive>:\Enterworks\EnableWeb\web.config
```

Any changes to image location and file paths must be updated in these repositories and the system must be restarted for changes to take effect.

The Enable DAM will load images from the defined `DAMRoot` folder path. The DAM will automatically create all defined variants in separate folders under `DAMRoot`. The `DAMFilesPerNumberedFolder` property in the `DAMConfig` repository dictates how many images are to be placed into each physical file folder. When the number of images reaches the configured maximum, the system will automatically create a separate subfolder for the next set of images.

For uploading large image files or multiple image files, a bulk upload utility service will run in background. This service monitors the `DAMDrop` folder for new images. The drop folder must

be shared and have the correct security setting so that users can drop new images into the folder. The background bulk upload service will periodically monitor the DAMDrop folder and upload the new images.

## Elasticsearch Configuration

The **Elasticsearch Config** settings available in the Feature bar, **System** tab is deprecated.

## System Management

Additional system management tasks are documented in the EnterWorks Online Help at <https://help.winshuttle.com/>.

## Stopping and Starting EnterWorks Services

The scripts for stopping and starting EnterWorks services on Windows servers are:

- `Stop Enterworks.cmd`
- `Start Enterworks.cmd`

They are installed on each server containing non-3<sup>rd</sup> party components, (EnterWorks Server, EnterWorks New UI, Utilities, and EPX). These scripts will call the appropriate component Stop and Start scripts to stop and start the services running on that server. This may include one or more of the following commands:

- `EPIM Stop.cmd` and `EPIM Start.cmd`: Stop and start the EnterWorks Server services.
- `EPX Stop.cmd` and `EPX Start.cmd`: Stop and start the EPX services.
- `Microservices Stop.cmd` and `Microservices Start.cmd`: Stop and start the EnterWorks microservices.

The component scripts will only stop and start the corresponding services that have been configured on the server. Some components may be installed on more than one server. For example, the `EPIM Stop.cmd` and `EPIM Start.cmd` scripts on one server might stop and start the EnterWorks Tomcat and JBoss Controller processes, while the same scripts on another server might stop and start the JBoss Worker processes.

The `Start Enterworks.cmd` and `Stop Enterworks.cmd` command scripts on each server are located in:

```
<install directory>\Enterworks\bin
```

When you run any start or stop scripts, run them as Administrator.

It is important to stop and start services in the correct order, depending on your system's configuration. See the configuration-specific documentation provided to you by your EnterWorks account representative for more information regarding the stop and start orders for services.

In general, the proper order for stopping the EnterWorks components is:

1. EPX
2. EnterWorks microservices
3. EPIM

The proper order for starting the components is:

1. EPIM
2. EnterWorks microservices
3. EPX

Typically, when EnterWorks is running on Windows, 3<sup>rd</sup> party applications (such as Elasticsearch, RabbitMQ, and MongoDB) are not restarted. If they are restarted, all EnterWorks components on all servers must also be restarted.

Every time the EnterWorks services are started using the `<XXX> Start.cmd` scripts, system log files are moved to an archive folder whose name is comprised of the current year, date, and time. The archive folders are stored in:

```
<install_drive>:\Enterworks\logs\archive
```

For instance:

```
<install_drive>:\Enterworks\logs\archive\2019-03-14@15-23-43-84
```

might hold the following folders:

- DAMReportUtility
- EnableServer
- EPX
- MonitorDAMDrop
- RegenerateDAMVariant

which would contain logfiles for those components of EnterWorks.

Periodically, older archive directories should be removed to prevent the archives from becoming large enough to impact EnterWorks system's efficiency.

### *To Stop all EnterWorks Services*

To stop all EnterWorks services on all servers in an environment, perform the following steps on each server:

1. Open **File Explorer** and navigate to:

```
<install_drive>:\Enterworks\bin
```

2. Right-click **Stop Enterworks** and select **Run as Administrator**.
3. A command prompt will appear and services will start shutting down.
4. Once the script is finished, verify that no EnterWorks related services are still running.
  - a. Open **Task Manager** and select the **Processes** tab.
  - b. Right-click either **CPU** or **Memory** on the header to bring up the list of available columns.
  - c. Select **Command line**.
  - d. Expand the width of the **Command line** column until you can see the path and name of any running processes.
  - e. Verify that the following processes do not appear in the list:
    - o Commons Daemon Service Runner: Any that have EnableServer in the file path.
    - o enable2020-<serviceName>-service
    - o Java(TM) Platform SE Binary: Any that have EnableServer or EPX in the file path.
  - f. If any of the listed processes appears, right-click the process and select **End Task**.

EnterWorks services are now shut down.

### *To Start all EnterWorks Services*

To start all EnterWorks services on all servers in an environment, repeat the following steps on each server:

1. Open **File Explorer** and navigate to:

```
<install_drive>:\Enterworks\bin
```

2. Right-click **Start Enterworks** and select **Run as Administrator**.
3. A command prompt will appear indicating that the services are starting. This prompt will close once all services have been started.
4. Verify that the services have been started by opening the **Services** control panel and confirming that all `Enable*` and `enable*` services (that have not been disabled) have a status of **Running**.

The EnterWorks services have been started.

## Changing the EPIM Database Password

For the most current instructions for changing the EPIM or EPX Database Passwords, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Users, Groups, and Security Management

Users are entered into EnterWorks and then assigned to Groups. The groups are defined based on types of system responsibilities, such as Administrator, Product Manager, Publications Manager, or Syndication Manager. These groups are designed around each organization's specific business processes. To efficiently manage the EnterWorks users' security, EnterWorks recommends that system security is managed at the group level.

Some user and user group management must be performed in both the EnterWorks Classic UI and the EnterWorks New UI. For more information regarding users and user groups see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Groups

EnterWorks Groups control both the functional areas of the application a user is allowed to view or perform, and what level of access a user has access to objects within EnterWorks, (for example code sets, users, groups, repositories etc.). Each type of object has permissions that can be set at the group level. These permissions give users in the group the ability to create an object and to read, edit or delete an existing object.

Repositories have additional permissions that allow a user to add, edit, sync-in (import), and delete records inside of repository. Be careful to assign correct permissions to a repository and

its underlying objects. Anyone given access to the repository must have read privilege on the underlying profile and the code sets used by the repository. If you are allowing a user to import data into a repository, they must have permission to create File Definition and Data Source objects.

Repositories security assignments must include an attribute security filter and optionally a record security filter. The filter attribute controls which attributes in a repository the user is allowed to read and edit. If no specific filter is defined for profile, the default filter must be specified. If no filter is defined, the user will not see any data. The Record security filter applies a search condition on records returned from a repository to limited access to only those records that match the Record filter criteria.

### *Adding or Removing a User to or from an User Group*


To add or remove a user from a Group:

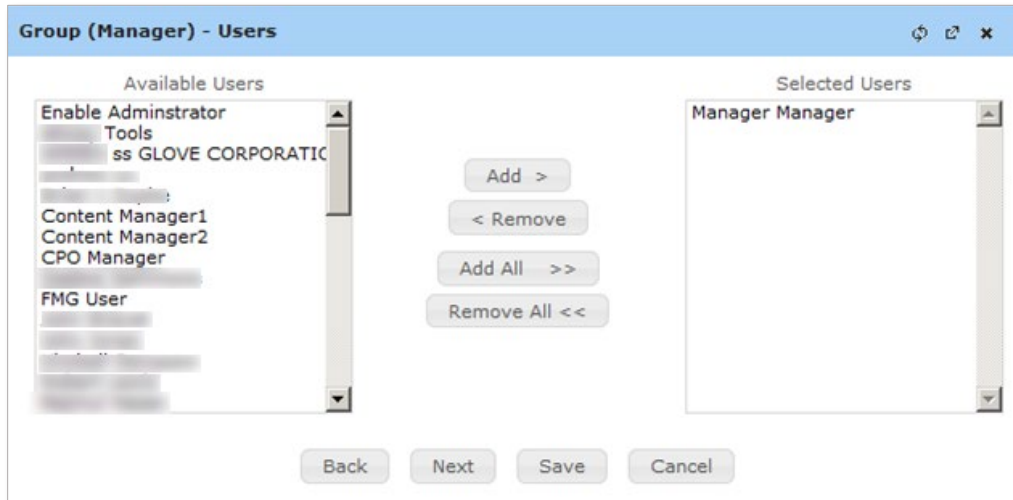
1. Log into EnterWorks Classic as an Administrator.
2. Open the Feature bar, open the **Users and Groups** folder, and select **Groups**. A list of the existing groups will appear.
3. Open the Group editor:
  - If you are creating a new group, open the **Action** dropdown menu, and select **New**.
  - Or, if you want to edit an existing group, select the group, open the **Action** dropdown menu, and select **Edit**. Alternatively, double-click the group and it will open in the Group editor.

The screenshot shows a window titled "Group (Administrator)". Inside the window, there is a form with the following fields:

- Name:** Administrator
- Description:** System Administration group
- Overwrite User Security Context:** No

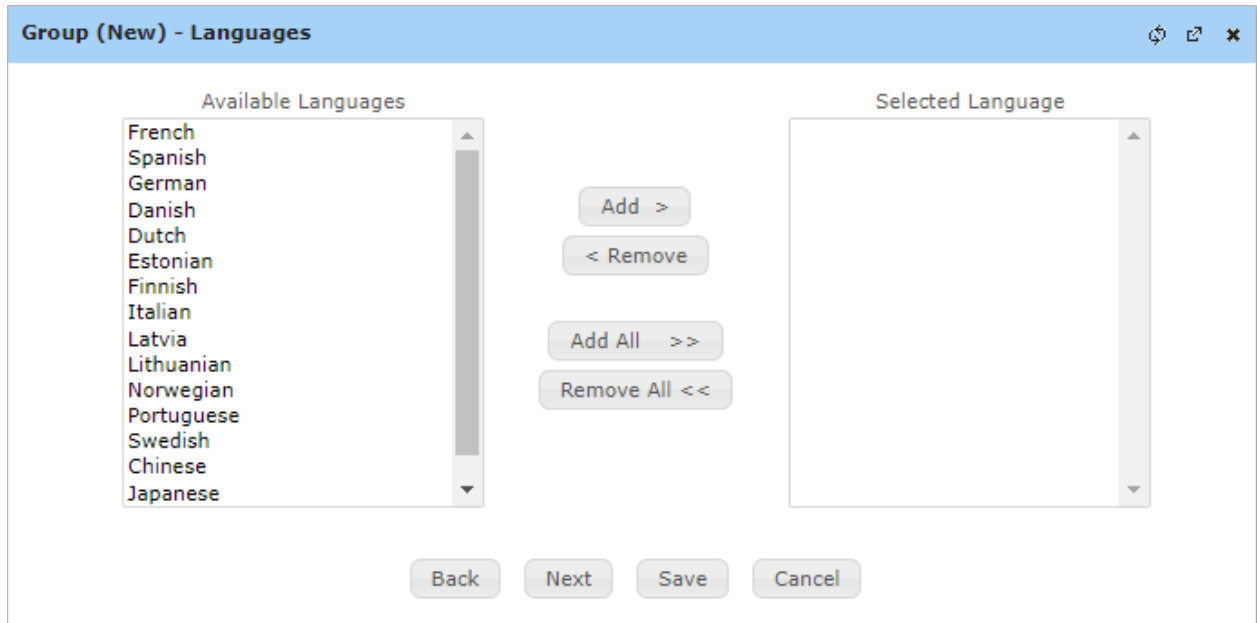
At the bottom of the form, there are four buttons: **Back**, **Next**, **Save**, and **Cancel**.

- The group's name and description fields are multilanguage. In order to add translations simply click the multi-language icon  corresponding to name or description and add the translations through the multilanguage editor.
- Click **Next**. The **Users** page appears.



- Select the users to be added from the **Available Users** list.
- Click the **Add >** button. The users will be added to the **Selected Users** list.
- Select the users to be removed from the **Selected Users** list and click the **< Remove** button. The users will be removed from the **Selected Users** list.
- Click **Next** to continue to the **Languages** page.





10. Select the Languages to be added from the **Available Languages** list.
11. Click the **Add >** button. The languages will be added to the **Selected Languages** list.
12. Select the languages to be removed from the **Selected Languages** list and click the **< Remove** button. The languages will be removed from the **Selected Languages** list.
13. Click **Next** to continue to the **Group Capability** page.



14. These capabilities represent aspects of the user interface that members of the Group will have access to. By expanding each Node in the list, individual aspects of each section can be granted to the Group. A checkmark in the checkbox indicates that users in the Group will have access to the capability.

15. Click the **Next** button to advance to the **Set Display Attribute Tabs** page.



16. Here all of the Attribute Tabs (collections of attributes within a repository) within each defined Profile are displayed. Check the checkbox next to each Attribute Tab that members of this Group should be able to view with Detail Item Editor. This feature allows certain collections of Attributes to be hidden from group members.

17. Click **Save** if no more changes are desired. The changes to the Group are saved and an Operation Successful dialog prompt appears.

### *Assigning Object Security to an EnterWorks Group*

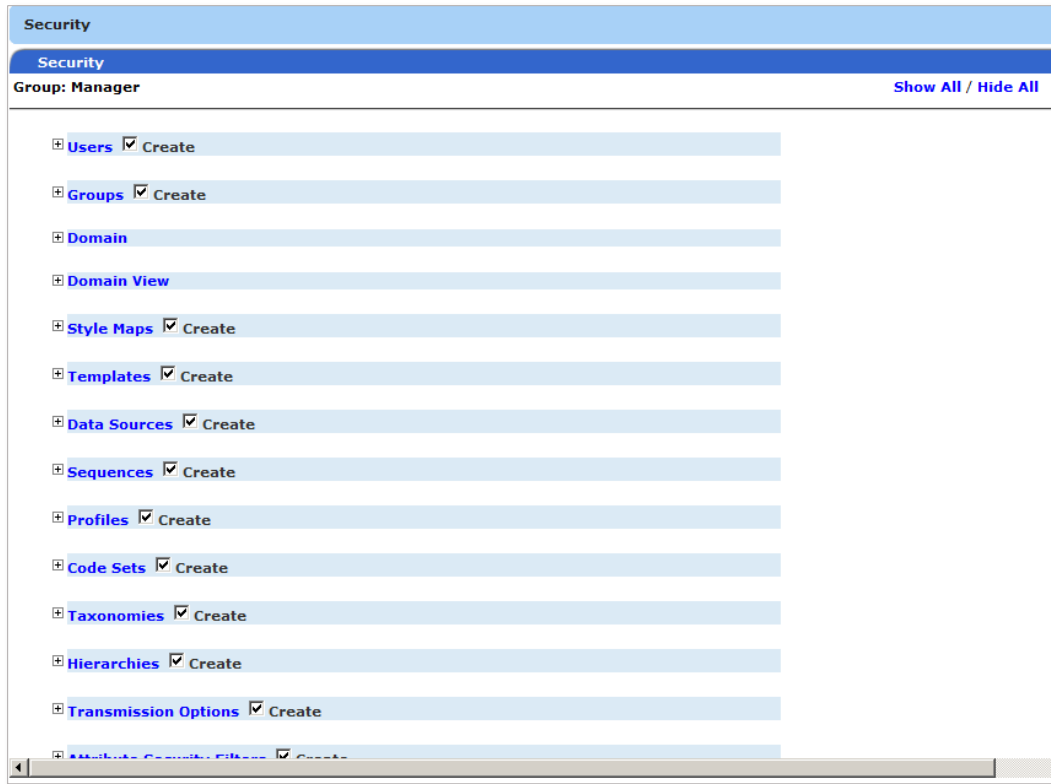
To assign object security to a Group:

1. Log into EnterWorks Classic as an Administrator.
2. In the Feature bar, open the **Users and Groups** folder, and select **Groups**. The **Groups** list will appear.
3. Select the Group to which privilege(s) will be added or removed.
4. Open the **Action** dropdown menu and select **Security**.
5. The **Security** page will be shown for the selected group.

The screenshot shows a web application window titled "Security". Below the title bar, there is a navigation bar with "Security" and a "Help" link. The main content area shows "Group: Manager" and a "Show All / Hide All" link. A "Users" section is expanded, showing a table with columns for "Users", "Read", "Edit", and "Delete". Each permission column contains a "Check / Uncheck" link. The "Read" column has checked boxes for all users, while "Edit" and "Delete" have checked boxes for most users, except for "bz1" and "manager" which have unchecked boxes.

Users	Read Check / Uncheck	Edit Check / Uncheck	Delete Check / Uncheck
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
axu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
bz1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
contentMgr1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
contentMgr2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
manager	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
fmguser1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

6. Click the **Hide All** link on the upper right of the page to view all categories of security.



7. Each category of permissions pertains to a specific EnterWorks security-enabled object type.
  - a. Click the **Create** check box in each subgroup to allow a user in the group to create that object.
  - b. Click the **Read**, **Edit** or **Delete** checkbox for each object to set permissions for that object.
8. Additional permissions are available on repositories. The Attribute filter and optionally the item filter need to be defined.
  - a. **Sync-in** allows importing of data into a repository.
  - b. **Add Records** allows adding new records to repository.
  - c. **Delete Records** allows deleting records from repository.
  - d. **Edit MetaData** allows changing repository properties.
  - e. **Edit Record Attribute Filter**: You must select a filter. The default filter allows access to all attributes.

- f. **Record Filter:** You must select a defined available item filter. If none is specified, this will give access to all rows of data in the repository.

The security categories available and permissions for each are described in this table:

<b>Security Category</b>	<b>Permissions Available</b>	<b>Description</b>
Users	Read/Edit/Delete	User has Read/Edit/Delete permission by default to themselves; must be granted access to others.
Groups	Read/Edit/Delete	User has Read permission by default to Groups they belong to; must be granted access to others.
Style Maps	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Style Maps (used in publication) created themselves; must be granted access to others.
Templates	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Templates (used in publication) created themselves; must be granted access to others.
Data Sources	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Data Sources (used for import) created themselves; must be granted access to others.
Sequences	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Sequences (used in repositories with auto-generated sequence attributes) created themselves; must be granted access to others.
Profiles	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Profiles (attributes defined for a repository) created themselves; must be granted access to others.
Code Sets	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Code Sets created themselves; must be granted access to others.
Taxonomies	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Taxonomies created themselves; must be granted access to others.
Hierarchies	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Hierarchies created themselves; must be granted access to others.
Transmission Options	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Transmission Options (used in definition of a repository) created themselves; must be granted access to others.

Security Category	Permissions Available	Description
Attribute Security Filters	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Attribute Security Filters (used to grant access to repository attributes) created themselves; must be granted access to others.
Record Security Filters	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Record Security Filters (used to grant select privilege to certain rows in repositories) created themselves; must be granted access to others.
File Definitions	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any File Definitions created themselves; must be granted access to others.
Repository Folders	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Repository Folders created themselves; must be granted access to others.
Media Groups	Read/Edit/Delete	User has Read/Edit/Delete permission by default to any Media Groups created themselves; must be granted access to others.
Repositories	Read/Edit/Delete/ View/SyncIn/ Add Records/ Delete Records/ Edit Meta Data/ Edit Record Attribute Filter/Record Filter	Read: permission to read repository definition Edit: permission to Edit items within repository; combine with Edit Record Attribute Filter to determine which attributes have Read/Edit Delete: permission to delete repository and contents View: permission to view the repository in context of EnterWorks Content feature listing SyncIn: permission to use Import to add data to repository Add Records: permission to add new rows to repository Delete Records: permission to delete rows from repository Edit Metadata: permission to change repository properties

- Click **Save**. The changes to the Group Security are saved and an Operation Successful dialog prompt appears.

## Users

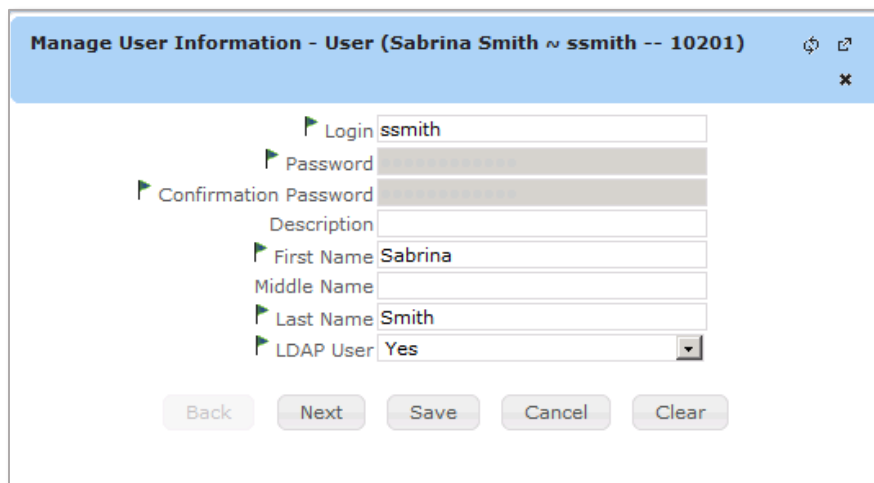
Users must be defined locally and assigned to one or more Groups for maintaining security and access. User's authentication can be from SSO or from the Local EnterWorks Database.

## Configuring the SSO, LDAP(S), or Active Directory

For information on configuring EnterWorks to use SSO, LDAP(S), or Active Directory, see the Precisely EnterWorks Installation Guide or the Precisely EnterWorks online help at <https://support.precisely.com>.

## Change a Local User to SSO, LDAP(S) or Active Directory User

1. Log into EnterWorks Classic as a user with Administrative privileges.
2. Click on the **Users and Group** feature bar.
3. Open the Users, select a user and click on menu **Action** → **Edit** or double-click on a user row in listing
4. Change the **LDAP User** dropdown to **Yes**.



The screenshot shows a window titled "Manage User Information - User (Sabrina Smith ~ ssmith -- 10201)". The window contains several input fields for user details:

- Login: ssmith
- Password: [Redacted]
- Confirmation Password: [Redacted]
- Description: [Empty]
- First Name: Sabrina
- Middle Name: [Empty]
- Last Name: Smith
- LDAP User: Yes (selected in a dropdown menu)

At the bottom of the window, there are five buttons: Back, Next, Save, Cancel, and Clear.

## Managing Users with Active Directory

When SSO is configured and the user is identified as an “**Is LDAP**” user, the EnterWorks system will validate the user password using SSO.

## Automatically Add a User from Active Directory

In order for EnterWorks to automatically add users from Active Directory, it must be configured to do so. For information on performing this configuration, see the *EnterWorks Installation Guide*.



## Managing Users Locally

### Adding a Local User

1. Log into EnterWorks Classic as an Administrator.
2. Open the Feature bar, open the **Users and Groups** folder, and select Users. The User List will appear.

#	Login	First Name	Last Name	LDAP U	Createc	Creation D	Last Upd	Last Updat	Id
1	admin	Enable	Administrator	No	system	11-12-2009	admin		10002
2			Tools	No	system	07-25-2012	system		10178
3		ss	CORPORAT	No		07-25-2012		04-07-2013	10177
4	axu	andrew	xu	No	axu	11-30-2012	axu	03-27-2013	10188
5	bz1	Brian		No	bz1	04-14-2013	bz1	05-31-2013	10202
6	contentMgr1	Content	Manager1	No	contentM	02-26-2013	contentMg	05-09-2013	10194
7	contentMgr2	Content	Manager2	No		02-26-2013		02-26-2013	10195
8	cpomanager	CPO	Manager	No		05-09-2013		05-09-2013	10204
9	debbie			No		11-09-2012		11-09-2012	10182
10	fmguser1	FMG	User	No	fmguser1	12-03-2012	fmguser1	12-03-2012	10189
11		John		Yes		12-17-2012		12-17-2012	10191
12		John		Yes		10-25-2012		02-11-2013	10180

3. Select the **New** menu option from the **Action** menu. The New User editor will appear.

**Manage User Information - User (New)**

Login

Password

Confirmation Password

Description

First Name

Middle Name

Last Name

LDAP User

Back Next Save Cancel Clear

- Fill in all fields prefaced with a green flag (required fields) and click **Next** to continue. The Detailed user editor appears. These fields are all optional; click **Next** to continue.

**Manage User Information - Detail (New)**

**Company Information**

Company

Employee Number

Department

Title

Manager

**Contact Information**

Work Number

Mobile Number

Home Number

Fax Number

Email

Address 1

Address 2

City

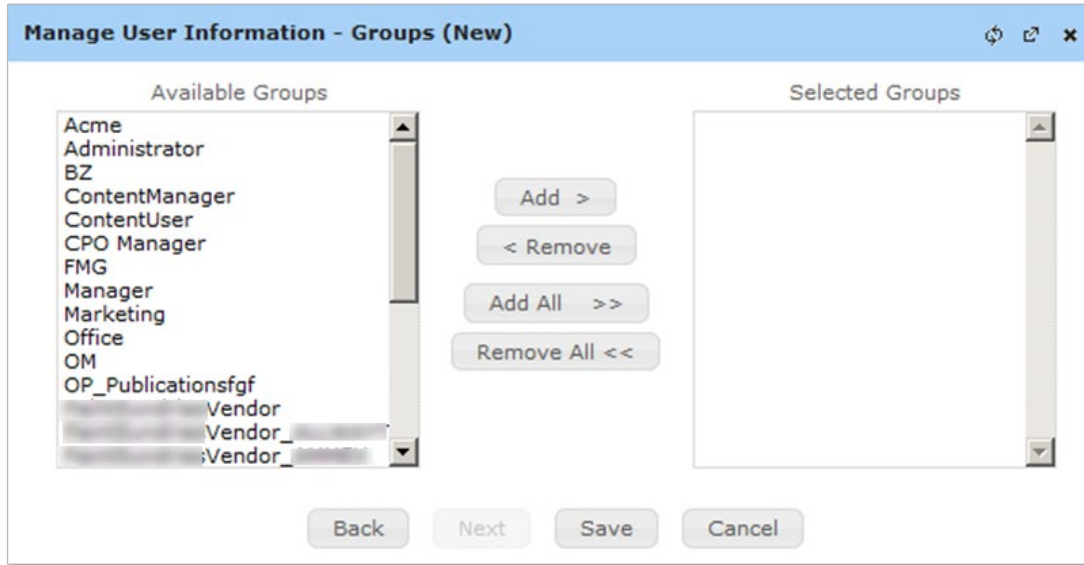
State

Postal Code

Country

Back Next Save Cancel Clear

- The **Manage User Groups** editor appears.
- Move entries from the **Available Groups** list to the **Selected Groups** list. The entries in the **Selected Groups** list will be the groups to which your user will belong.



7. Click **Save** to save the user.

### Deleting a Local User

To delete a local user:

1. Log into EnterWorks Classic as an Administrator.
2. Open the Feature bar, open the **Users and Groups** folder, and select Users. The User List will appear.

Login

Action ▾ Utilities ▾

User Listing										
<input type="checkbox"/>	#	Login	First Name	Last Name	LDAP U:	Created	Creation D	Last Upd	Last Updat	Id
<input type="checkbox"/>	1	admin	Enable	Adminstrator	No	system	11-12-2009	admin		10002
<input type="checkbox"/>	2			Tools	No	system	07-25-2012	system		10178
<input type="checkbox"/>	3		ss	CORPORAT	No		07-25-2012		04-07-2013	10177
<input type="checkbox"/>	4	axu	andrew	xu	No	axu	11-30-2012	axu	03-27-2013	10188
<input type="checkbox"/>	5	bz1	Brian		No	bz1	04-14-2013	bz1	05-31-2013	10202
<input type="checkbox"/>	6	contentMgr1	Content	Manager1	No	contentM	02-26-2013	contentMg	05-09-2013	10194
<input type="checkbox"/>	7	contentMgr2	Content	Manager2	No		02-26-2013		02-26-2013	10195
<input type="checkbox"/>	8	cpomanager	CPO	Manager	No		05-09-2013		05-09-2013	10204
<input type="checkbox"/>	9	debbie			No		11-09-2012		11-09-2012	10182
<input type="checkbox"/>	10	fmguser1	FMG	User	No	fmguser1	12-03-2012	fmguser1	12-03-2012	10189
<input type="checkbox"/>	11		John		Yes		12-17-2012		12-17-2012	10191
<input type="checkbox"/>	12		John		Yes		10-25-2012		02-11-2013	10180

3. Select the desired user, open the **Action** dropdown menu, and select **Delete**.
4. Respond to the confirmation request.

### Change a Local User's Password

While other attributes of a user account can be edited in the New UI, changing a user's password must be done in the Classic UI.

To change a user's password:

1. Log into the Classic UI as an Administrator.
2. Open the Users tab: in the Features bar, open the **Users and Groups** folder and select **Users**.
3. Open the Manage User Information pop-up: either double-click the desired user; or select the desired user, open the **Action** dropdown, and click **Edit**.
4. Edit the password as desired.
5. Click the **Save** button.

## Configure User Impersonation

The Impersonate User capability allows a user with proper credentials to impersonate another user to carry out EnterWorks tasks using the user's security credentials. The actions of the impersonator are logged and auditable.

Note that the impersonator must have read access for the users they impersonate.

Also, for security reasons, users who belong to the system administration group are prevented from being able to impersonate the EnterWorks system user.

The Classic UI is used to give the impersonator the ability to impersonate a user. To enable this feature:

Log into the Classic UI.

1. Open the **Users** tab: from the Feature Bar, open **Users and Groups**, and select **Users**.
2. Select the user who will be able to impersonate, open the **Action** drop-down, and select **Impersonate Config**.
3. An Impersonate Config window will appear. Check the **Impersonate Enabled** checkbox.
4. After the **Impersonate Enabled** checkbox is checked, the **Available Users by Group** and **Selected Users can be Impersonated** lists will appear. Select a user group the member belongs to.

If the user belongs to the System Administration group, the Impersonate Config window will not include the **Available Users by Group** and **Selected Users can be Impersonated** lists because system administrators can impersonate any user.

5. To add a user who can be impersonated, select their name from the **Available Users** list and click the right-arrow icon to move the user to the **Selected Users can be Impersonated** list.

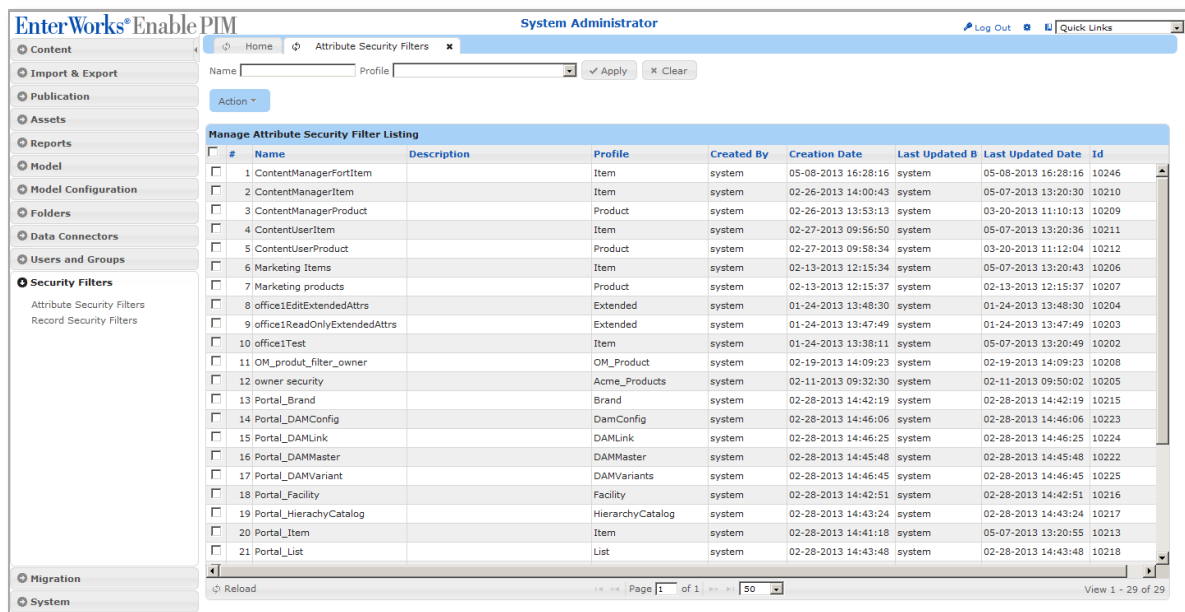
To remove a user who can be impersonated, select their name from **Selected Users can be Impersonated** list and click the left-arrow icon to move the user to the **Available Users** list.

6. Select **OK** to save your changes and exit, or select **Cancel** to exit the dialog without saving your changes.

## Attribute Security Filter Management

An Attribute Security filter controls access to column data (attributes) in a repository. The filters are created based on the profile definition and assigned to repository security to users and groups.

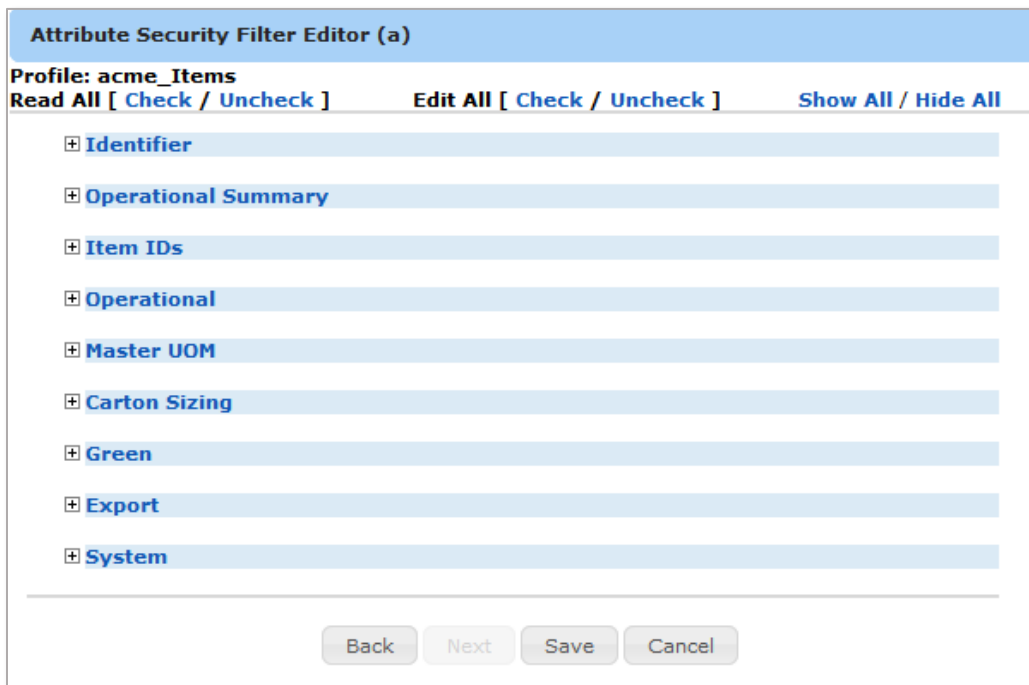
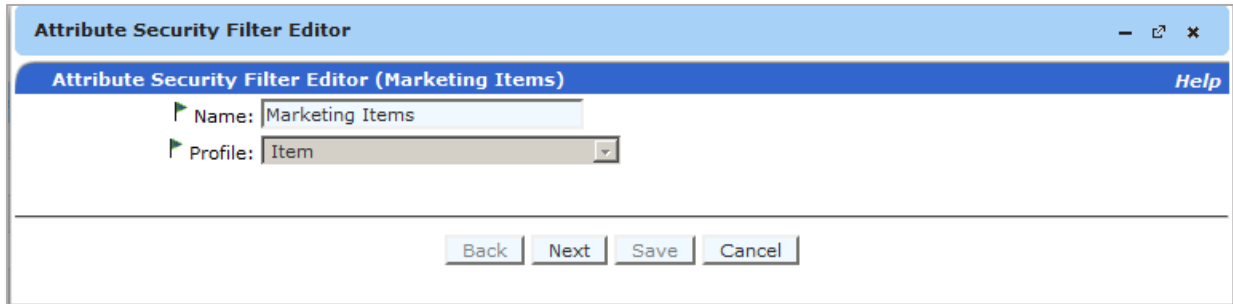
1. Log into EnterWorks Classic as an Administrator.
2. Open the Feature bar, open the Security **Filters** folder, and select **Attribute Security Filters** function. The list of existing attribute security filters will appear.



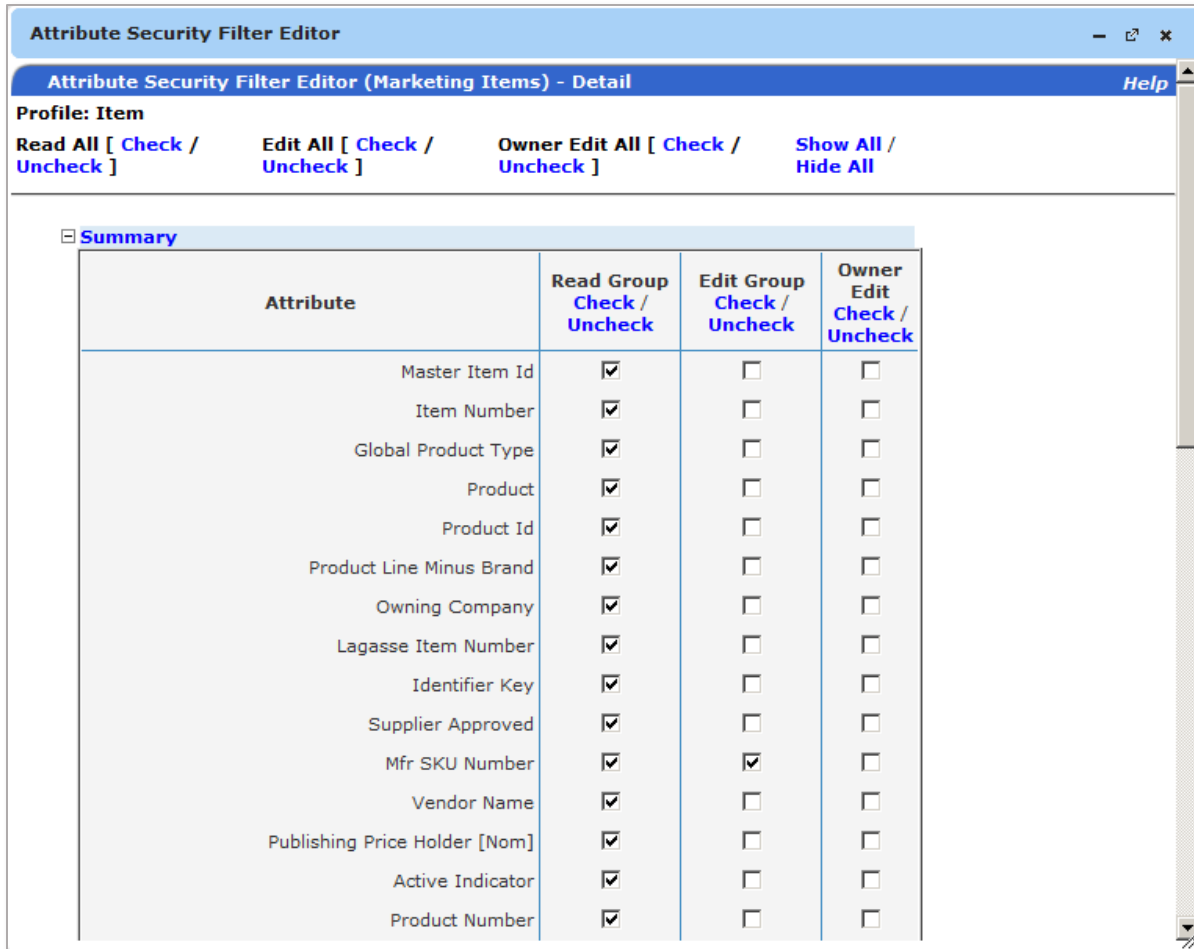
The screenshot displays the 'Manage Attribute Security Filter Listing' table in the EnterWorks Classic System Administrator interface. The table lists various attribute security filters with columns for Name, Description, Profile, Created By, Creation Date, Last Updated By, Last Updated Date, and Id. The filters are numbered 1 through 21. The 'Security Filters' folder is selected in the left-hand navigation menu.

#	Name	Description	Profile	Created By	Creation Date	Last Updated By	Last Updated Date	Id
1	ContentManagerFortItem		Item	system	05-08-2013 16:28:16	system	05-08-2013 16:28:16	10246
2	ContentManagerItem		Item	system	02-26-2013 14:00:43	system	05-07-2013 13:20:30	10210
3	ContentManagerProduct		Product	system	02-26-2013 13:53:13	system	03-20-2013 11:10:13	10209
4	ContentUserItem		Item	system	02-27-2013 09:56:50	system	05-07-2013 13:20:36	10211
5	ContentUserProduct		Product	system	02-27-2013 09:56:34	system	03-20-2013 11:12:04	10212
6	Marketing Items		Item	system	02-13-2013 12:15:34	system	05-07-2013 13:20:43	10206
7	Marketing products		Product	system	02-13-2013 12:15:37	system	02-13-2013 12:15:37	10207
8	office1EditExtendedAttr		Extended	system	01-24-2013 13:48:30	system	01-24-2013 13:48:30	10204
9	office1ReadOnlyExtendedAttr		Extended	system	01-24-2013 13:47:49	system	01-24-2013 13:47:49	10203
10	office1Test		Item	system	01-24-2013 13:38:11	system	05-07-2013 13:20:49	10202
11	OM_produt_filter_owner		OM_Product	system	02-19-2013 14:09:23	system	02-19-2013 14:09:23	10208
12	owner security		Acme_Products	system	02-11-2013 09:32:30	system	02-11-2013 09:50:02	10205
13	Portal_Brand		Brand	system	02-28-2013 14:42:19	system	02-28-2013 14:42:19	10215
14	Portal_DAMConfig		DamConfig	system	02-28-2013 14:46:06	system	02-28-2013 14:46:06	10223
15	Portal_DAMLink		DAMLink	system	02-28-2013 14:46:25	system	02-28-2013 14:46:25	10224
16	Portal_DAMMaster		DAMMaster	system	02-28-2013 14:45:48	system	02-28-2013 14:45:48	10222
17	Portal_DAMVariant		DAMVariants	system	02-28-2013 14:46:45	system	02-28-2013 14:46:45	10225
18	Portal_Facility		Facility	system	02-28-2013 14:42:51	system	02-28-2013 14:42:51	10216
19	Portal_HierarchyCatalog		HierarchyCatalog	system	02-28-2013 14:43:24	system	02-28-2013 14:43:24	10217
20	Portal_Item		Item	system	02-28-2013 14:41:18	system	05-07-2013 13:20:55	10213
21	Portal_List		List	system	02-28-2013 14:43:48	system	02-28-2013 14:43:48	10218

3. Select the desired attribute security filter record.
4. Select **Edit** from the **Action** menu. The **Attribute Security Filter** page will appear.
5. The attribute security filter is defined for a profile. For new filter, a profile must be chosen. Click **Next**.

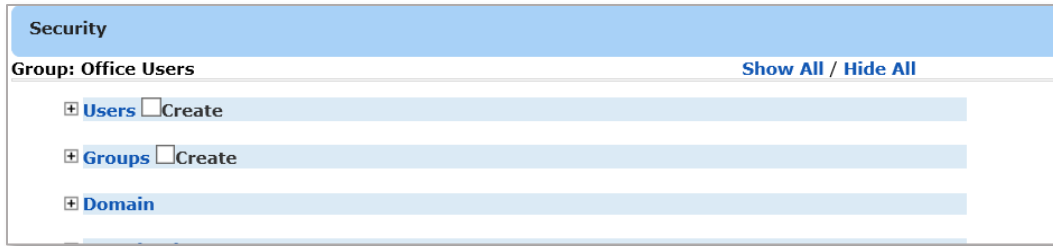


6. The Attributes for the profile are listed in a collapsible sub grouping list. Set read and edit permission for each attribute in all sub groups. If you have selected a profile where a **Taxonomy with Owner** configuration is defined for one of the attributes, then an additional column titled **Owner Edit** will appear for each attribute section. A check in this column means that only the user defined as the owner of the Taxonomy code value will have edit permission to item data containing that value.
7. Click **Save**.

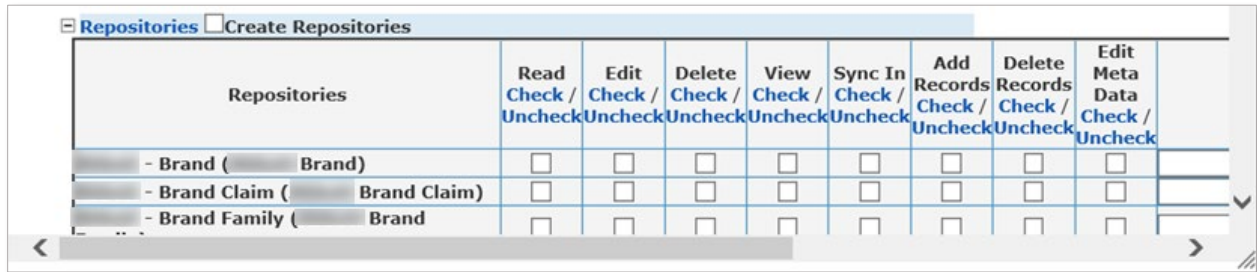


8. To assign the Attribute Security Filter to a Group:
  - a. Open the **Users and Group** feature
  - b. Open the **Group** function
  - c. Select the Group record the Attribute Security Filter applies to.
  - d. Select the **Security** option from the **Action** drop-down menu.
  - e. The **Security** window will appear.





- f. Click the **Hide All** link on the top right of the screen to close all the category details.
- g. Navigate to the **Repositories** section and open it by clicking the + plus sign.



- h. Navigate to the repository needing the attribute security filter applied for the group.
- i. Select the Attribute Security Filter in the **Record Attribute Filter** column.
- j. Navigate to the bottom of the list and click the **Save** button.



Office - Customer Locations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Office - Customer Products List	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Office - Facility	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Office - Icon	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Office - List	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Office - Brand Prod (PIM_Brand_Production)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Brand_10012__DEFAULT
Office - Brand Staging (PIM_Brand_Staging)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Brand_10012__DEFAULT
Office - CustomerProd (PIM_Customer_Production)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Customer_10064__DEFAULT
Office - Customer (PIM_Customer_Staging)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Customer_10064__DEFAULT
Office - Product Line Catalog Stage (PIM_HierarchyCatalogPL_Staging)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068__DEFAULT
Office - Item Catalog (PIM_HierarchyCatalog_Item)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068__DEFAULT
Office - Product Catalog Prod (PIM_HierarchyCatalog_Production)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068__DEFAULT
Office - Product Catalog Stage (PIM_HierarchyCatalog_Staging)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068__DEFAULT
Office - Item Prod (PIM_Item_Production)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Item_10001__DEFAULT
Office - Item (PIM_Item_Staging)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Marketing Items

- The same method can be used to assign an Attribute Security Filter to a User (starting with a User instead of a Group), but it is recommended as best practice to use a group (even if there is only one person in the group).

## Record Security Filter Management

Record Security filter controls access to row data in a repository. The filters are created based on the profile definition and assigned to repository security to users and groups.

- Log into EnterWorks Classic as an Administrator.
- Open the Feature bar, open the **Security Filters** folder and select **Record Security Filters**. The list of existing record security filters will appear.

Home Record Security Filters x

Name  Profile

Action ▾

**Record Security Filter Listing**

#	Name	Description	Profile	Created By	Creation Date	Last Updated B	Last Updated Date	Id
<input type="checkbox"/>	1 aaaf		acme_Items	system	11-30-2012 15:46:19	system	04-07-2013 22:10:07	10020
<input type="checkbox"/>	2 debCodeSet		Brand	system	02-27-2013 12:21:14	system	02-27-2013 13:29:41	10021
<input type="checkbox"/>	3 kkjlijio		Acme Collection	system	11-30-2012 15:43:45	system	11-30-2012 15:43:45	10018
<input type="checkbox"/>	4 Vendor_ Items		Items	system	07-25-2012 14:17:15	system	07-25-2012 14:17:15	10016
<input type="checkbox"/>	5 Vendor_ Items		Items	system	07-25-2012 14:18:05	system	07-25-2012 14:18:05	10017
<input type="checkbox"/>	6 Acme Collection		Acme Collection	system	11-30-2012 15:44:42	system	11-30-2012 15:44:42	10019
<input type="checkbox"/>	7 Item		Item	system	07-21-2012 14:05:02	system	07-21-2012 14:05:02	10014
<input type="checkbox"/>	8 Purchase Pri		Purchase Pri	system	07-21-2012 14:19:19	system	07-21-2012 14:19:19	10015
<input type="checkbox"/>	9 User Item		Item	system	04-26-2012 15:01:25	system	04-26-2012 15:01:25	10013
<input type="checkbox"/>	10 Vendor Item		Item	system	04-25-2012 18:01:24	system	04-26-2012 10:50:14	10011
<input type="checkbox"/>	11 Vendor product		Product	system	04-26-2012 11:02:26	system	04-26-2012 11:02:26	10012

3. Select the item security filter record to edit and double-click the record or select **Edit** from the **Action** drop-down menu.
4. The **Record Security Filter** page will appear.

**Record Security Filter Details** - [x] [Help]

**Record Security Filter Details (Supplier American Item)**

Name:

Description:

Profile: American Item

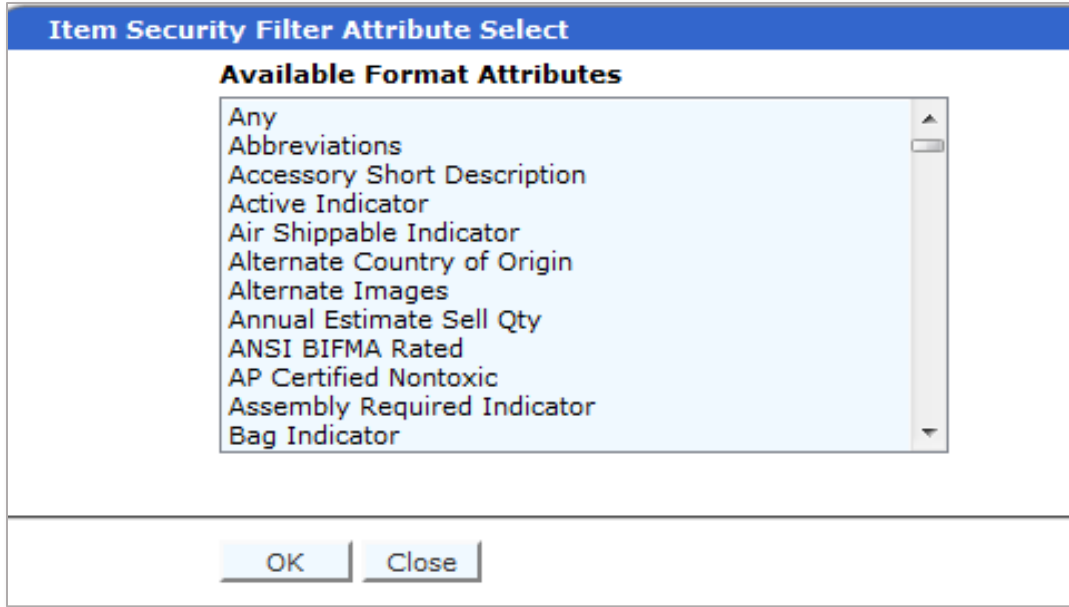
---

**Record Security Filter Conditions**

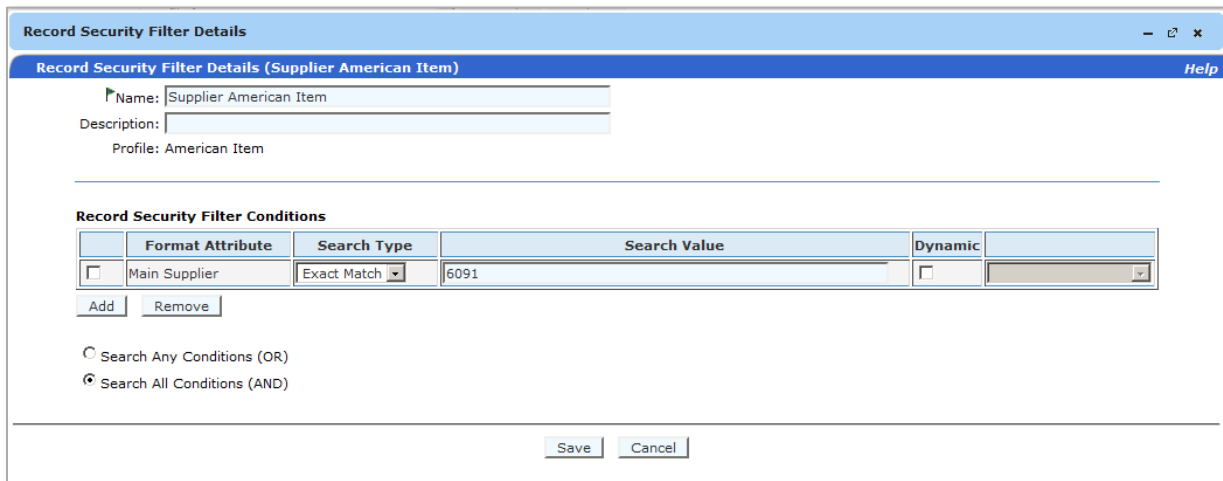
	Format Attribute	Search Type	Search Value	Dynamic
<input type="checkbox"/>	Main Supplier	Exact Match ▾	6091	<input type="checkbox"/>

Search Any Conditions (OR)  
 Search All Conditions (AND)

5. Add filter conditions by pressing the add button. A list of available attributes will be displayed.



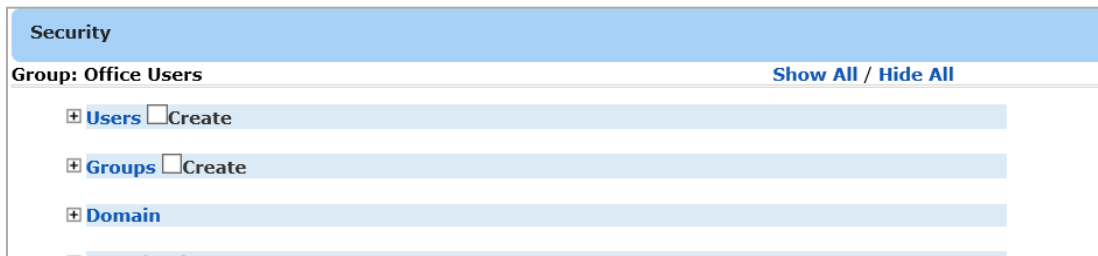
6. Select the desired attribute, then click **OK** to exit the list and return to the **Record Security Feature Details** screen.



7. Select a **Search Type** (Contains or Exact Match) and enter the **Search Value**.
8. If more than one search condition is defined, select the "**OR**" or "**AND**" operation radio button.
9. Click the **Save** button.

10. To assign the Record Filter to a Group:

- a. Open the **Users and Group** feature.
- b. Open the **Group** function.
- c. Select the **Group** record the Attribute Security Filter applies to.
- d. Select the **Security** option from the **Action** drop-down menu. The Security editor will appear.
- e. Click the **Hide All** link on the top right of the screen to close all the category details.



- f. Navigate to the **Repositories** section and open it by clicking the **+ plus** sign.
- g. Navigate to the repository needing the record security filter applied for the group.
- h. Select the Record Security Filter in the Record Attribute Filter column.

Repositories	Read Check / Uncheck	Edit Check / Uncheck	Delete Check / Uncheck	View Check / Uncheck	Sync In Check / Uncheck	Add Records Check / Uncheck	Delete Records Check / Uncheck	Edit Meta Data Check / Uncheck	Edit Record Attribute Filter	Record Filter
--------------	----------------------------	----------------------------	------------------------------	----------------------------	-------------------------------	--------------------------------------	---	--	------------------------------	---------------

Office - Item Catalog (PIM_HierarchyCatalog_Item)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068_DEFAULT	
Office - Product Catalog Prod (PIM_HierarchyCatalog_Production)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068_DEFAULT	
Office - Product Catalog Stage (PIM_HierarchyCatalog_Staging)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ProductTaxonomy_10068_DEFAULT	
Office - Item Prod (PIM_Item_Production)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Item_10001_DEFAULT	
Office - Item (PIM_Item_Staging)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Marketing Items	User Item
Office - Manufacturer Prod	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Manufacturer_10016_DEFAULT	

- i. Navigate to the bottom of the list and click the **Save** button.
11. The same method can be used to assign a Record Security Filter to a User (starting with a User instead of a Group), but it is recommended as best practice to use a group (even if there is only one person in the group).

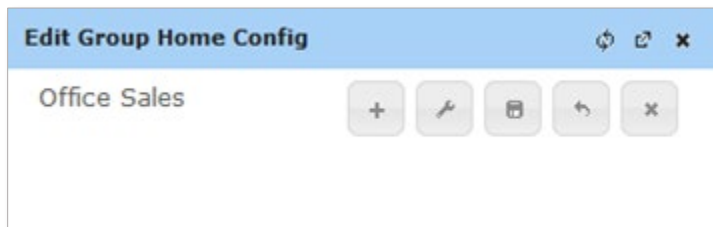
## Configuring a Group's Home Page Dashboard


The terms in “dashboard” and “home page” are used interchangeably.

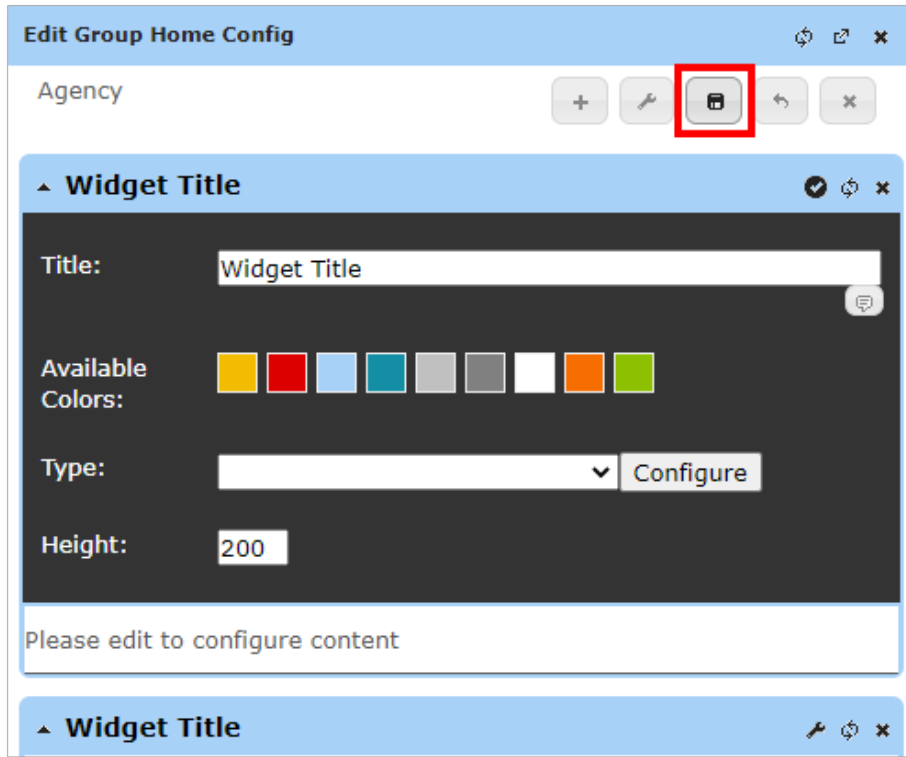
### Define a Home Page in the Classic UI

To define a user group home page in the EnterWorks Classic UI:

1. In the **Feature** bar, open the **Users and Groups** folder and select **Groups**.
2. Select the **User Group** that needs a default home page defined.
3. Open the **Action** dropdown and select **Edit Home Page**. (The **Edit Home Page** action edits the EnterWorks Classic homepage; the **Edit User Portal Home Page** action edits the EnterWorks New UI dashboard.)
4. The **Edit Group Home Config** editor will open.



5. Set up (or edit) the report widgets for the user group.
6. Before exiting, make sure to click the Save icon .

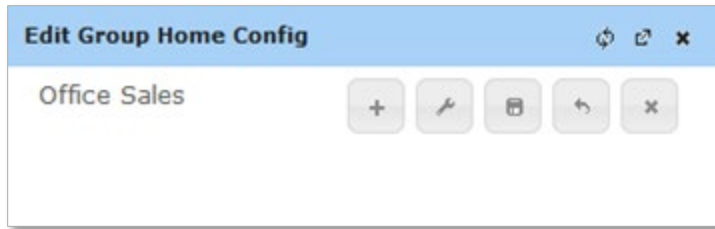



## Define a Dashboard in the New UI

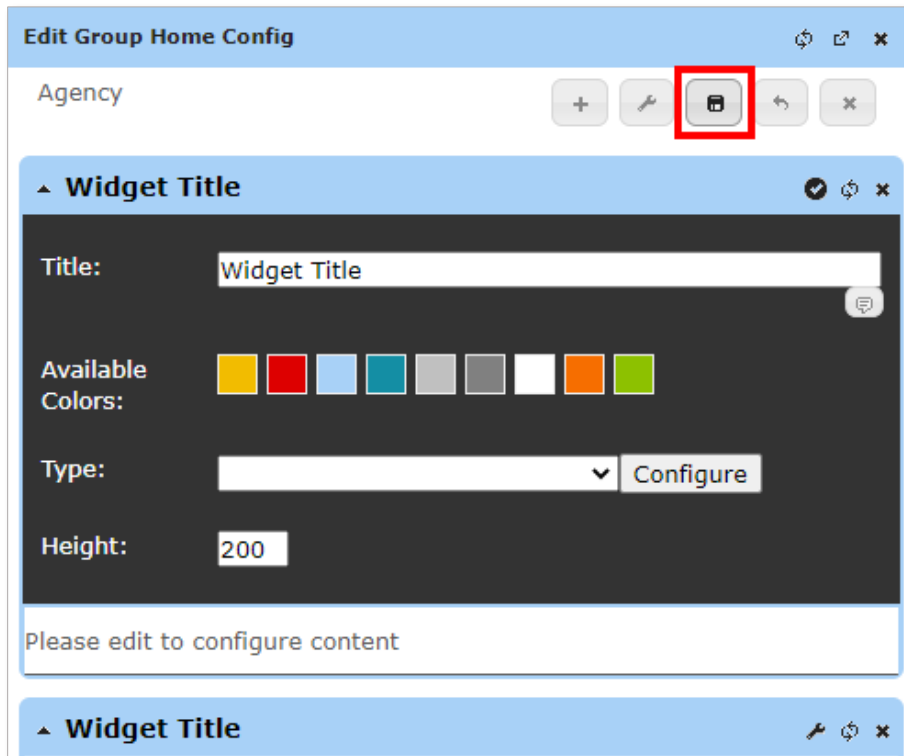
To define a user group's dashboard in the EnterWorks New UI, you must first use the EnterWorks Classic UI to create the user group's EnterWorks New UI dashboard and save it. You do not have to configure it completely, but you do need to create it and save it. Then you need to log into the EnterWorks New UI and configure the dashboard there.

To do so:

1. Log into the EnterWorks Classic UI as a user with administrative privileges.
2. In the **Feature** bar, open the **Users and Groups** folder and select **Groups**.
3. Select the **User Group** that needs a default home page defined.
4. Open the **Action** dropdown and select **Edit User Portal Home Page**. (The **Edit Home Page** action edits the EnterWorks Classic home page; the **Edit User Portal Home Page** action edits the EnterWorks New UI home page.)
5. The **Edit Group Home Config** editor will open.



6. If you'd like, set up (or edit) the report widgets for the user group. This can also be done later, in the EnterWorks New UI.
7. Before exiting, make sure to click the Save icon , even if you did not set up any widgets.



## Workflow

EPX is a standalone workflow application used to drive automation. The EnterWorks Workflow feature consists of EPX, custom workflow processes, and scheduling activities. EPX is comprised



of a JAVA thick client application used to build and manage workflows, and a browser web based portal for users to interact with the workflow. EPX is fully integrated with EnterWorks.

## Workflow Properties

Workflow Properties are used to tailor the EnterWorks Workflow functionality according to enterprise requirements. The following screen shows the Workflow Properties screen which is accessed via the **Repository** → **Edit** menu.

**Product - Workflow Properties**

Workflow Enabled

Synchronous Submission

Process Name

Starting Point

Work Item Name

Default Submitting EPX User

Submit Multiple Selected Records as Single Work Item

Lock Record From Edit

Custom Condition Class

Validate after any updates

Attributes to Update

Attribute	Value
<input type="checkbox"/> Status	Inactive

Add Delete

Workflow Properties to Submit

Property	Value
<input type="checkbox"/> taxonomy	[Taxonomy]
<input type="checkbox"/> name	[Item Number]
<input type="checkbox"/> description	[productDisplayName]

Add Delete

Save Cancel

To access the Workflow Properties window:

1. Log into the EnterWorks Classic UI.
2. Open the **Content** tab, the repository's folder, select the repository's name, select **Edit**, and select **Workflow Properties**.
3. The **Workflow Properties** window will appear.

The configuration options on the **Workflow Properties** window are:

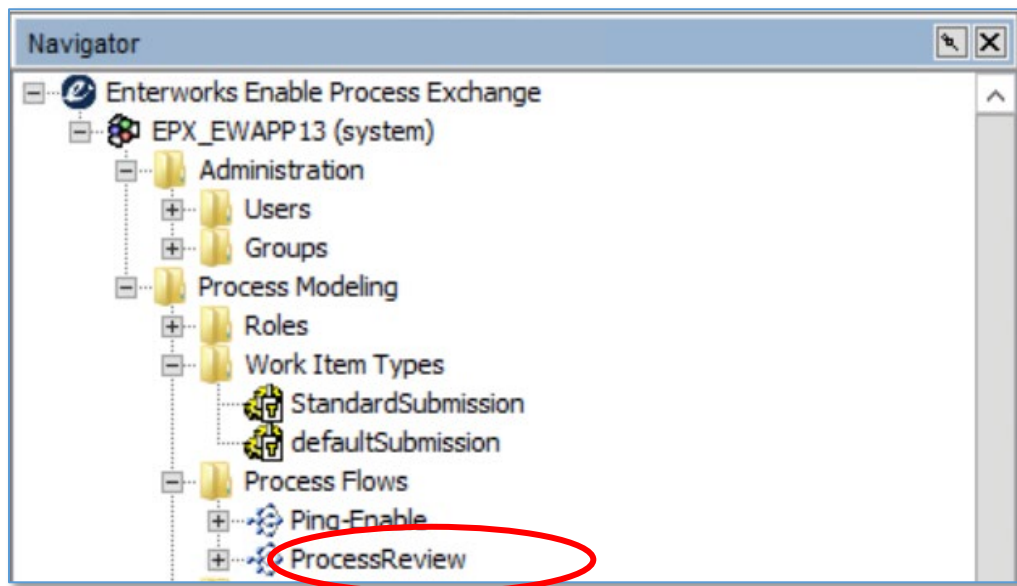
- **Workflow Enabled** – When checked, the repository has the ability to link to an EPX Process Flow.
- **Synchronous Submission** – **Synchronous Submission** will be checked by default. When checked, the user interface will wait for the submission to process before rendering. Normally, when a submission is large or requires time to interface with other systems, this option will not be checked so that the submission is asynchronous, meaning the user interface will not have to wait for a submission to complete in order to render.
- **Process Name, Starting Point, and Work Item Name** are the default work item configurations when a work item is automated.
  - **Process Name** – EnterWorks PIM uses the **Process Name** value to locate the EPX Process Name so that it knows where to send a work item.
  - **Starting Point** – EnterWorks PIM uses the **Starting Point** value to locate the starting point for an EPX process flow.
  - **Work Item Name** – EnterWorks PIM uses the **Work Item Name** to give a default name to an automated work item.
- **Default Submitting EPX User** – EnterWorks PIM will login to EPX as the specified user when an automated work item is submitted.
- **Submit Multiple Selected Records as a Single Work Item** – When this option is checked, multiple records can be submitted as a single work item/multi-record work item to an EPX process flow. When this option is unchecked, multiple records will be submitted as individual work items.
- **Lock Record from Edit** – When this option is checked, any record that is part of a work item is locked for all users, except for certain user roles when the work item is at an activity for that role.
- **Custom Condition Class** – Allows EnterWorks PIM Workflow to callout to customized JAVA class(es) that define customer specific processes.
- **Validate After Any Update** – Works in conjunction with **Attributes to Update** and only executes for a new submission. Upon submission of a work item, all attributes within **Attributes to Update** will be updated with the specified value.
- **Attributes to Update** – Provides a list of attributes within a repository's Profile and provides the ability to specify a value that an attribute will utilize upon an update.

- **Workflow Properties to Submit** – This option only applies to a single record work item. It allows an administrator to associate properties with a work item and gives the administrator the ability to specify either literal values or dynamic values retrieved from a repository’s profile.

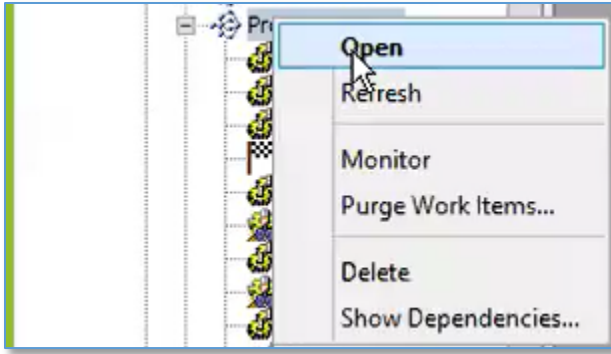
## EnterWorks PIM Options

The EnterWorks PIM Options provide the ability to link an EPX activity to an EnterWorks PIM repository. Values either reference work item properties or are literal words. Typically, an administrator will keep **Repository Name**, **Repository Friendly Name**, and **Repository Id** dynamic. However, in rare scenarios, they can have literal values. For instance, the activity of processing a specific image of the DAMMaster repository.

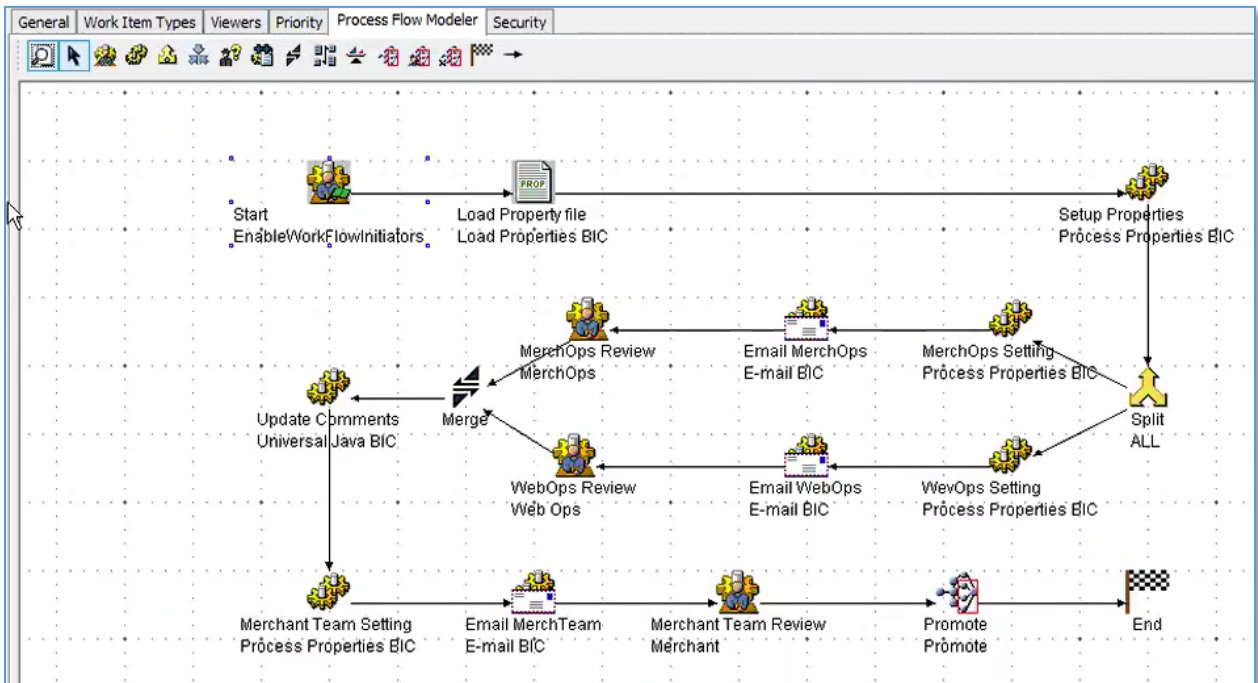
1. Log into EPX as an administrator.
2. From the **Navigator**, go to **Process Modeling** → **Process Flows** → *<select a flow>*.



3. Right-click the flow.
4. Click **Open**.



5. Click on the **Process Flow Modeler** tab.



6. Right click on a manual activity and select **Properties**.

7. Click on the **Details** tab.

**Manual Activity - Start Product**

**Manual Activity**  
Configure the basic properties of this activity

General | Details | Viewers | Expiration | Custom Action

Actor: EnableWorkflowInitiators Select...

Activity Options

- Work Item Name Editable
- Starting Point
  - Default Work Item Name:
  - Work Item Priority:
  - Persist Properties:
- Delayed Send
- Work Item Key:
- Date Format:
- Format Test:

Enable PIM Options

- Linked to Enable PIM
  - Repository Name:
  - Repository Friendly Name:
  - Repository Id:
  - Preference Name:
  - Preference Id:
  - Saved Set Name:

OK Cancel Apply Help

Ready EPX\_EWAPP13 (system)

*EnterWorks PIM Options*

Enable PIM Options	
<input checked="" type="checkbox"/> Linked to Enable PIM	
Repository Name	%repositoryName%
Repository Friendly Name	%repositoryFriendlyName%
Repository Id	%repositoryId%
Preference Name	
Preference Id	
Saved Set Name	%savedSetName%
Saved Set Id	%savedSetId%
Search Name	
Search Id	
Additional Search Attribute	
Item Id(s)	%itemIds%
Task Name	View Work Item
Task Instructions	Send work item on once viewed.
Task Role	
Task Status	
Task Icon	
Task Notification	%workflowCommentHistory%
Task Object	

*En PIM Options Details*

Anytime you submit an item to a workflow there are properties that are attached to the work item. For instance the following properties are always attached:

- **Repository Name**
- **Repository Friendly Name**
- **Repository Id**
- If you submit a preference: (Deprecated)
  - **Preference Name** (Deprecated)
  - **Preference Id** (Deprecated)
- If you submit a saved set:
  - **Saved Set Id**
  - **Saved Set Name**
- If you submit a search: (Deprecated)
  - **Search Name** (Deprecated)

- **Search Id** (Deprecated)
- **Additional Search Attribute** (Deprecated)
- **Item Id(s)**: If more than one item is selected, values will be in a comma separated list format.

The following options allow you to add additional information to submit to the workflow. Anything between two percent signs is a dynamic reference to an attribute inside an associated repository's Profile.

- **Task Name** – Details the process flow activity.
- **Task Instructions** – Informs the user what actions need to be done.
- **Task Role** – Informs the user the role assigned to a task.
- **Task Status** – Allows the user to manage the status of a task. The value specified will be the default value.
- **Task Icon** – Requires a URL of a 16x16 icon.
- **Task Notification** – option to specify notifications to the user when viewing the work item in the EnterWorks UI. This is typically set to the `%workflowCommentHistory%` property reference.
- **Task Object** – References the object the activity will act upon. If you know it will be a saved set, then specify the saved set name. If you know it will be a record, then specify the name of the record. If you leave **Task Object** blank, it will not appear.
- **Allow Reassign** – Future feature that allows users to reassign their own tasks to someone else.
- **Allow Listing Send** – this option controls whether or not the work item can be sent directly from the Active Work Items list on the **Workflow Task Manager** widget.
- **Listing Viewer Type** – When a user is in the **Workflow Task Manager** viewer, the user will be able to be redirected to:
  - **Editor** – the **Detail Editor Tab**.
  - **Listing** – a record listing of repository records associated with the work item (for example, in the referenced saved set).
  - **Listing Preference**
- **Work Item Type** – References an EPX Work Item Type and allows EnterWorks PIM to render the workflow submission screen based on the work item type selected

during the initial workflow submission or upon completion of an activity. If a Work Item Type is specified (recommended), the Task Attribute JSON field is disabled.

- **Task Attribute JSON** – Specifies the content of the submission form when creating or advancing a work item for a repository record. This is only required if a **Work Item Type** is not selected. The data inside the square bracket is a JSON array object. Everything inside each curly bracket is one attribute. For each attribute, there is:
  - **name** – name of attribute.
  - **default** – default value.
  - **type** – determines an HTML control type.
    - **Type 0** – Text Box
    - **Type 1** – Dropdown Select Box
    - **Type 2** – Select Area Box
    - **Type 3** – Check Box
    - **Type 4** – Radio Selection
    - **Type 5** – Hidden
  - **list** – values for the control type.
  - In the second attribute there is **“name” : “workflowComment”**. **workflowComment** is a special feature that causes a viewer of an activity to append whatever the user enters into a comment history, so when the next user comes up then they can look at the comment history.

```
Task Attributes JSON [{"default": "Approved", "displayname": "Submission Type", "name": "workflowApproval", "type": "1", "list": ["Approve", "Rejected", "Review"]}, {"default": "", "displayname": "Submission Notes", "name": "workflowComment", "type": "2", "list": []}]
```

## Work Item Types

When a record, multiple records, or a saved set are sent to a workflow, the EnterWorks New UI will display a **Workflow Submission** screen with multiple input fields, such as the following screen.



WORKFLOW SUBMISSION X

> PRODUCT SETUP AND REVIEW - A\_PRODUCT\_PROCESS S\_START - A\_PRODUCT\_PROCESS

▼ PRODUCT SETUP AND REVIEW - Z REVIEW PRODUCT START - Z REVIEW PRODUCT

Start - Z Review Product START

Submission Name  
Product Review

Submission Approval ▼

Submission Notes

CANCEL

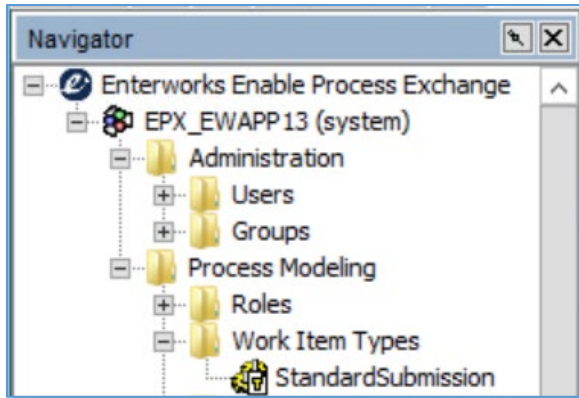
*Workflow Submission Screen*

The **Workflow Submission** screen is rendered based on the specified EPX Work Item Type. A Work Item type allows admins to specify how input should be specified by a user and what input should be gathered from the user.

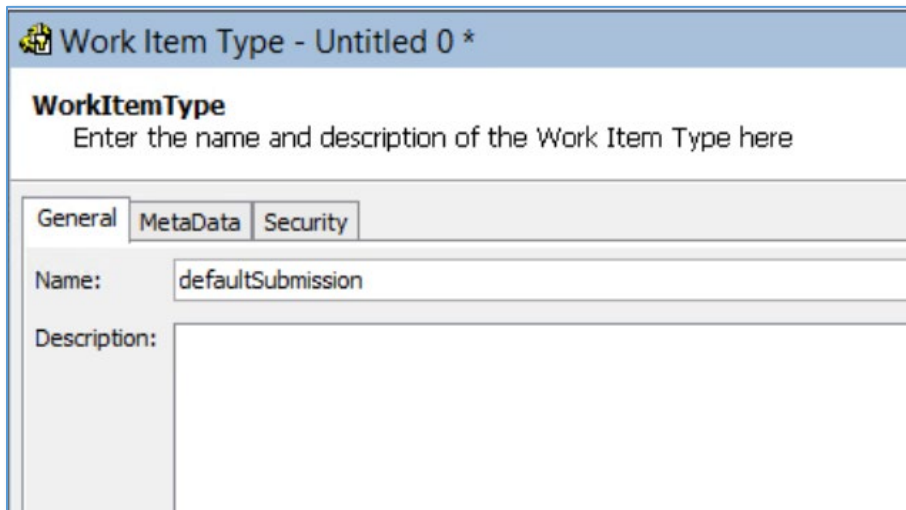
### *Create a Work Item Type*

Work Item Types are created in EPX.

1. Log into EPX as an admin.
2. From the **Navigator**, go to **Process Modeling** → **Work Item Types**.



3. Right click on the directory **Work Item Types**.
4. Click **New...**
5. On the **General** tab, in the **Name** field, enter a name for your Work Item Type.



6. Click on the **MetaData** tab.
7. Right click inside the table.
8. Select **Add**.
9. At this point there are many different combinations of html input fields you can create. For this example, we are going to create a dropdown list.

10. In the **Name** field, enter a name for the property that is to be defined in the Work Item.

Name	workflowApproval
------	------------------

11. Select String for the **Data Type**.

Data Type	String
-----------	--------

12. In the **Label** field, enter the text you want to appear on the Workflow Submission screen, such as "Submission Type".

Label:	Submission Type
--------	-----------------

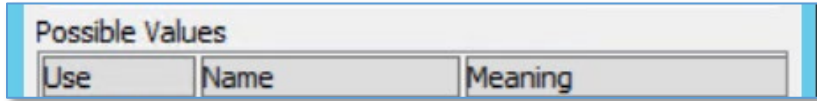
13. Select **List** for the HTML Input Type.

HTML Input Type:	List
------------------	------

14. In the **Value** field, enter a default value for the dropdown list, for example, "approval". This must be one of the values you later define. If you do not want to have a default value, leave this field blank.

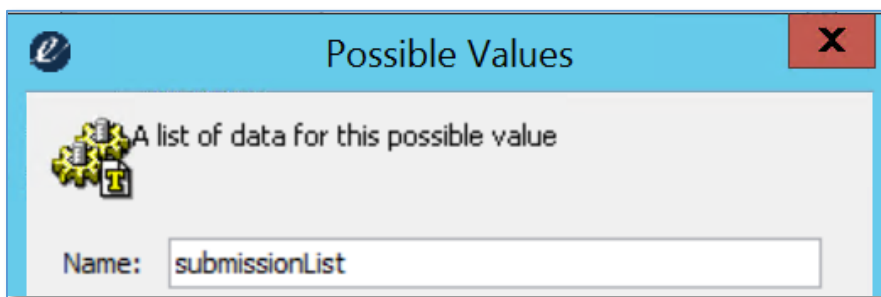
Value:	approval
--------	----------

15. Right click in the **Possible Values** table.

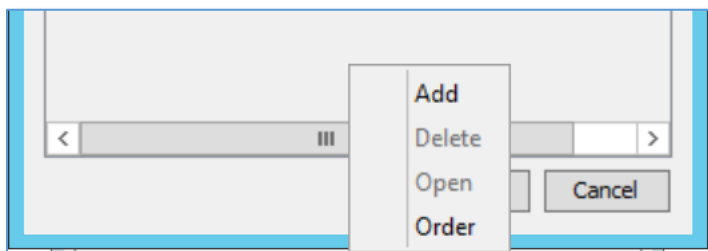


16. Select **Add...**

17. In the **Name** field, enter the name for the dropdown list.

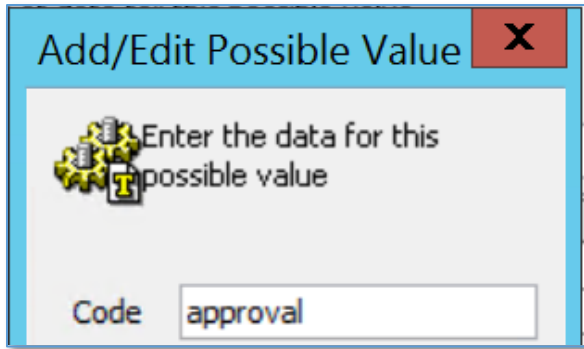


18. Right click in the table below the **Name** field.

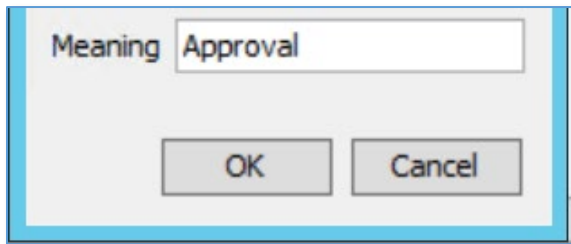


19. Select **Add**.

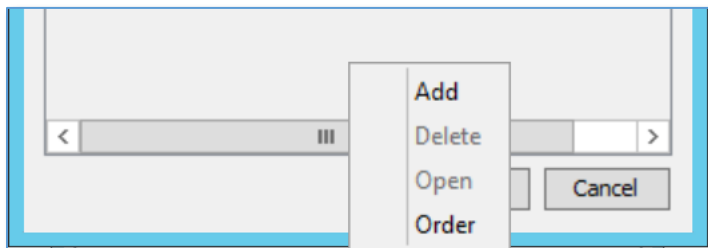
20. In the **Code** field, enter the value you want associated with that item, for instance, "approval".



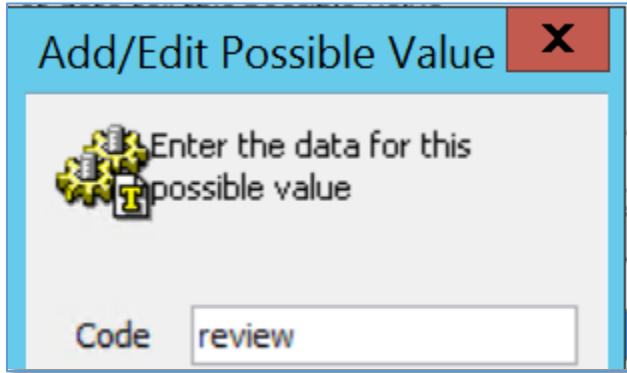
21. In the **Meaning** field, enter the label the user will see and select when they look at the dropdown list, such as “Approval”.



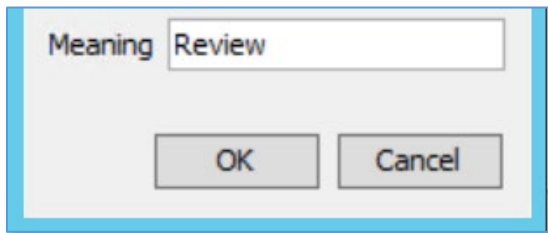
22. Click **OK**.
23. Right click in the **Possible Values** table again.
24. Select **Add**.



25. In the **Code** field, enter the value you want associated with that item, for instance “review”.



26. In the **Meaning** field, enter the label the user will see and select when they look at the dropdown list, such as “Review”.



27. Click **OK** to save the new submission value.

28. Click **OK** again to return to the Define the Work Item Type’s metadata window.

29. Within the **Possible Values** table under the **Use** column, click on the checkbox for the newly created possible value.

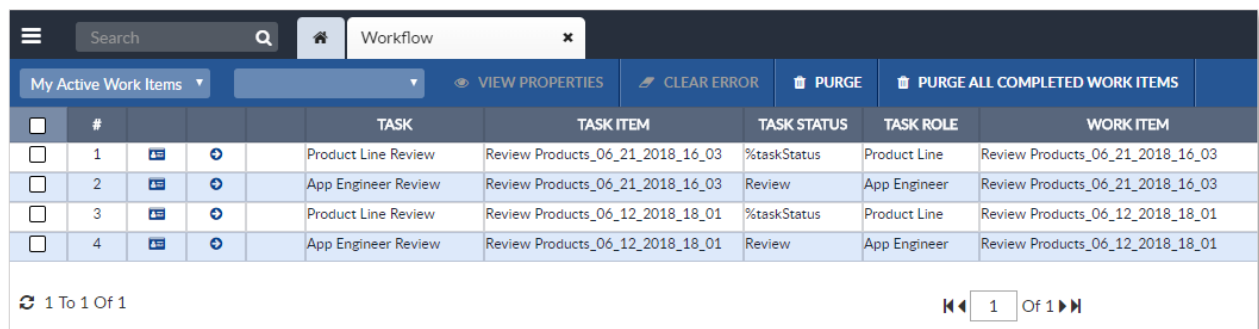
Possible Values		
Use	Name	Meaning
<input checked="" type="checkbox"/>	submissionList	Approval,Review

30. Click **OK** to return to the **WorkitemType** window.
31. Click **Apply**.
32. Click **OK**.

## Open Workflow Task Manager

To open the **Workflow Task Manager** in a tab:

1. Open the **Feature Sidebar**, select the **Administration** tab, and select **Workflow Task Manager**.
2. The **Workflow Task Manager** will open.



	#			TASK	TASK ITEM	TASK STATUS	TASK ROLE	WORK ITEM
<input type="checkbox"/>	1			Product Line Review	Review Products_06_21_2018_16_03	%taskStatus	Product Line	Review Products_06_21_2018_16_03
<input type="checkbox"/>	2			App Engineer Review	Review Products_06_21_2018_16_03	Review	App Engineer	Review Products_06_21_2018_16_03
<input type="checkbox"/>	3			Product Line Review	Review Products_06_12_2018_18_01	%taskStatus	Product Line	Review Products_06_12_2018_18_01
<input type="checkbox"/>	4			App Engineer Review	Review Products_06_12_2018_18_01	Review	App Engineer	Review Products_06_12_2018_18_01

1 To 1 Of 1

1 Of 1

*Workflow Task Manager Detail*

## Monitor Workflow Process

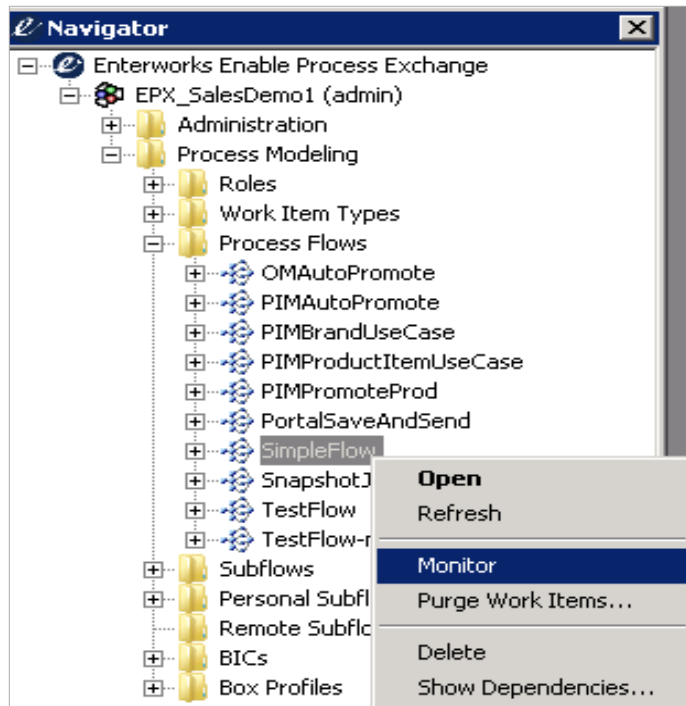
EnterWorks administrator can monitor and manually initiate workflow process from the EnterWorks application. For information on monitoring Workflow, see the Precisely EnterWorks online help at <https://support.precisely.com>.

## Manage Workflow Process




To manage workflow you must install and use the JAVA thick client “EPX Design Console”. From the **Design** console, you can monitor workflow activities, add or modify work flow activities.

1. Log onto APP server or a client box that has **EPX Design Console** installed.
2. Open **EPX Design Console**.

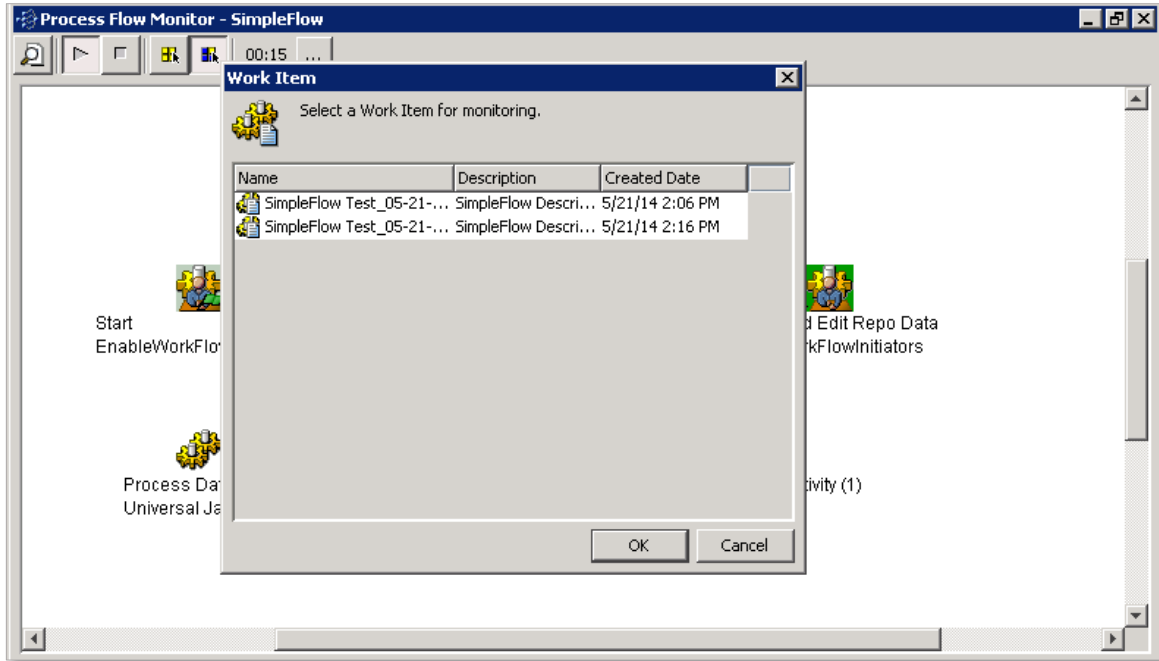
- Expand the tree and log on using an administrative account.
- Open the Process Flow Monitor by opening the **Process Modeling** folder, opening the **Process Flows** folder, right-clicking on a workflow, and selecting **Monitor**.



Open the Process Flow Monitor

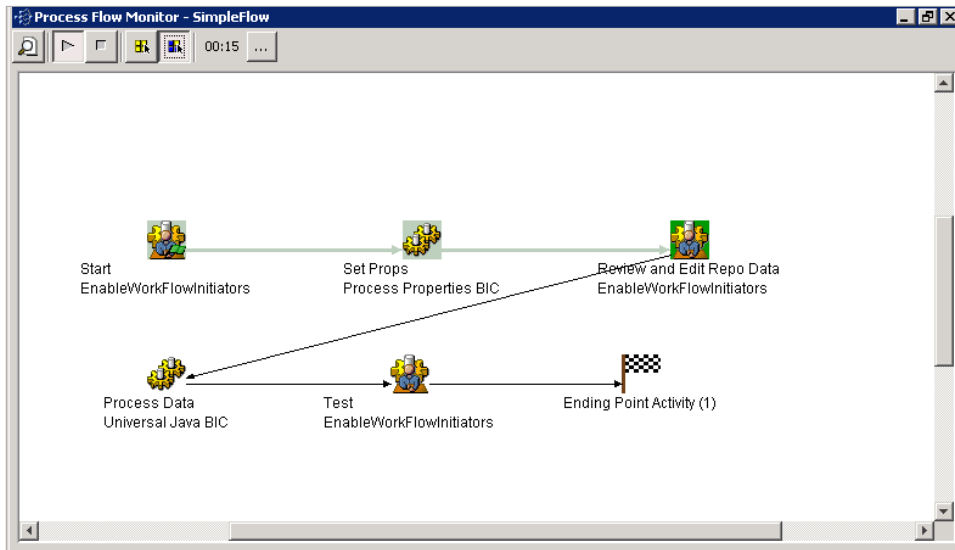
- Click the **Play**  icon to view a count of all work items at each activity.
- Click the Monitor One Work Item  icon to select an individual work item to monitor. After selecting, click the **Play**  icon to track the flow of the work item.





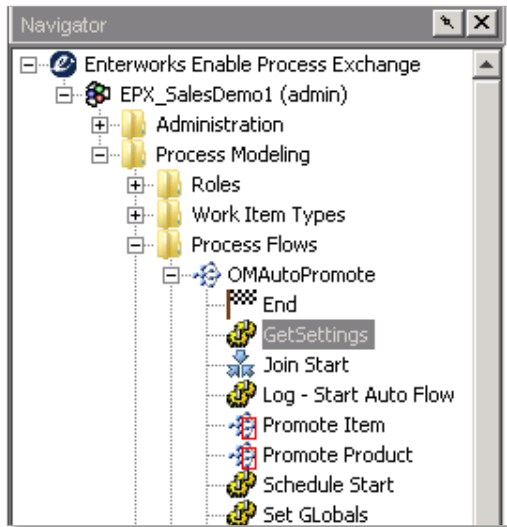
Select a Work Item for Monitoring

- The work item flow will highlight the path the work item has taken. A green icon indicates the current location of the work item. A red icon indicates an error at that activity.



The Path and Status of a Work Item

8. If an activity has error, expand the navigation menu to find the activity under the corresponding flow or sub-flow.
9. Right-click on the activity and select **Work Items**.



*View Work Items at an Activity*


10. Double-click the work item from the list.
11. Click on the **Error** tab. Review the error. Once resolved, click on the **Clear Error** button.

## Purge a Workflow – DO NOT USE

**WARNING:** Purging a workflow will delete all work items (active and completed) from the workflow, even those not visible to the user. **DO NOT USE THIS FUNCTION.**

Purging a workflow also clears the **Locked in Workflow** state for all records associated with the work items.

To purge a workflow, first open the **Workflow Task Manager**. This can be done either by opening a **Workflow Task Manager Widget** or by opening the **Workflow Task Manager** in a new tab.

To open it in a tab, open the **Feature Sidebar** by selecting the **Feature Sidebar** icon  on the **Tab Bar**, select the **Administration** tab, then select **Workflow Tasks**. The **Workflow Task Manager** tab will appear.

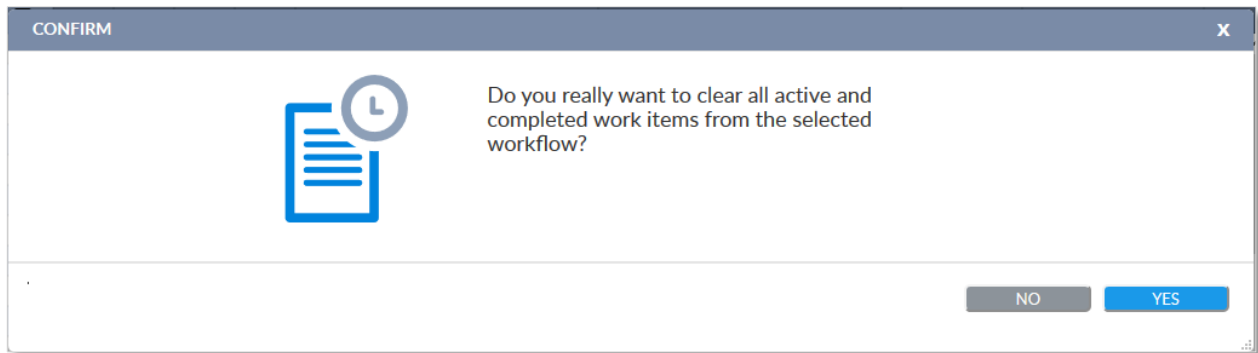
Workflow									
My Active Work Items		VIEW PROPERTIES		CLEAR ERROR		PURGE		PURGE ALL COMPLETED WORK ITEMS	
	#			TASK	TASK ITEM	TASK STATUS	TASK ROLE	WORK ITEM	
<input type="checkbox"/>	1			Product Line Review	Review Products_06_21_2018_16_03	%taskStatus	Product Line	Review Products_06_21_2018_16_03	
<input type="checkbox"/>	2			App Engineer Review	Review Products_06_21_2018_16_03	Review	App Engineer	Review Products_06_21_2018_16_03	
<input type="checkbox"/>	3			Product Line Review	Review Products_06_12_2018_18_01	%taskStatus	Product Line	Review Products_06_12_2018_18_01	
<input type="checkbox"/>	4			App Engineer Review	Review Products_06_12_2018_18_01	Review	App Engineer	Review Products_06_12_2018_18_01	

1 To 1 Of 1 1 Of 1

Workflow Task Manager Detail

To purge all work items (active and completed) from a workflow:

1. Select one item in the workflow and click the **Purge** button on the **Workflow Task Manager's Activity Bar**. A confirmation window will appear:

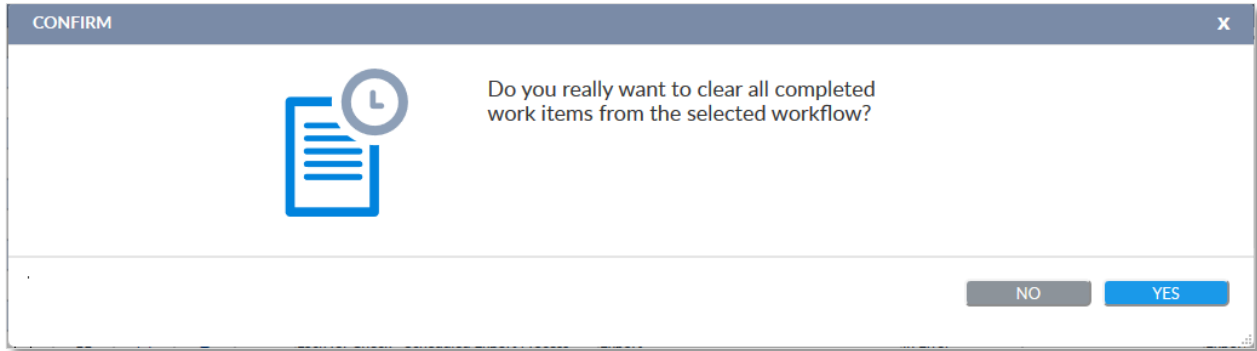


Purge Workflow: Confirm the Purge of All Work Items from a Workflow

2. Click **Yes** to purge all work items.

To purge all completed work items in a workflow:

1. Select one item in the workflow and click the **Purge All Completed Work Items** button on the **Workflow Task Manager's Activity Bar**. A confirmation window will appear.



Purge Workflow: Confirm the Purge of All Completed Work Items from a Workflow

2. To purge the workflows, click the **Yes** button.

## Add a Task to the Worklist

To add an additional task that can be initiated from the EnterWorks work list, the workflow process first must be defined. The `sharedConfig.properties` files must be updated to add the task to the configuration. Search the `<drive>:\Enterworks` folder to find all instances of the `sharedconfig.properties` file and update them.

1. Add the new task's name to key `epx.tasklist.tasks`.
2. Using the new task name, add the following keys, replacing `<task>` with the new task name and the appropriate value for the key.

Key	Value examples	Description
<code>epx.tasklist.&lt;task&gt;.name</code>	Start PMFImport	Name of task as it will appear in UI
<code>epx.tasklist.&lt;task&gt;.flowname</code>	PMFImport	Name of workflow in EPX
<code>epx.tasklist.&lt;task&gt;.startingpoint</code>	Manual Start	Starting activity for workflow
<code>epx.tasklist.&lt;task&gt;.workitemname</code>	Manual Initiated PMFImport	Name for work item
<code>epx.tasklist.&lt;task&gt;.workitemdescription</code>	Manual Initiated PMFImport	Description for work item.

Key	Value examples	Description
<code>epx.tasklist.&lt;task&gt;.propList</code>	<code>prop1, prop2</code>	Optional list of property values to pass to work flow.
<code>epx.tasklist.&lt;task&gt;.props.prop1.name</code>	<code>test1</code>	Name of property, substitute "prop1" for value in prop list, repeat for each value.
<code>epx.tasklist.&lt;task&gt;.props.prop1.value</code>	<code>test1val</code>	Value of property, substitute "prop1" for value in prop list, repeat for each value.
<code>epx.tasklist.&lt;task&gt;.groupList</code>	<code>Administrator</code>	Optional comma separated list of groups that have access to execute this task.
<code>epx.tasklist.&lt;task&gt;.userList</code>	<code>admin</code>	Optional comma separated list of users that have access to execute this task.

3. All `sharedConfig.properties` files must be updated with the same settings.
4. EnterWorks services must be restarted for the changes to take effect.

## Logs and Debugging

For more information about debugging an EnterWorks system, see the Precisely EnterWorks online help at <https://support.precisely.com>.

### Logging Levels

To track errors in the system, it can be useful to turn on the EnterWorks detailed debugging logs. However, these logs can produce a large amount of log information, so it is recommended to turn on debugging logs only after requested to do so by EnterWorks support staff. It is recommended that you turn off detailed debugging for daily use.

To set the log levels:

1. Log into EnterWorks Classic as a user with the Administrator group.
2. Open the **Feature** bar and open the **System** folder.

3. Click on **Log Level Setting**.
4. Set the log levels as desired:
  - Severe: Only severe messages are logged.
  - Warning: Warning and severe messages are logged.
  - Info: Informational, warning, and severe messages are logged.
  - All: All messages are logged.

## Log Format

Log messages can be configured to output as a string or as JSON key value pairs. To change the log format, set the `EnableServer.sharedConfiguration.properties` setting `debug.file.msg.format.type`.

If `debug.file.msg.format.type=old`, log messages will use the legacy string format. This is the default setting.

If `debug.file.msg.format.type=json`, log messages will use the JSON format:

```
{
  "time": "mandatory",
  "level": "mandatory",
  "component": "mandatory",
  "message": "mandatory",
  "exception": "optional"
}
```

## Using the Logs for Debugging

To track errors in the system, it can be useful to turn on the EnterWorks detailed debugging logs. However, these logs can produce a large amount of log information, so it is recommended to turn on debugging logs only after requested to do so by EnterWorks support staff. It is recommended that you turn off detailed debugging for daily use.

To use the logs for debugging EnterWorks:

1. Log into EnterWorks Classic as a user with the Administrator group.
2. Open the **Feature** bar and open the **System** folder.

3. Click on **Log Level Setting**.
4. Set all the components to the level **All** and click **Save**.
5. Perform the action(s) that caused the errors.
6. Repeat this process, but this time set the log levels to **Severe**.
7. When you are finished generating the logs, it is recommended that you disable the Debug Logs feature, as the debugging logs create a large number of entries. To disable the debug logs, set all the log levels to **Off**.

## Debugging Logs for EPX Workflows

There are configuration options in two files that control debug logging for workflow processes:

- `config.properties`
- `Enterworks.properties`

The debugging logs configured in the `config.properties` file provide a large amount of detailed debugging log messages. It is recommended that you disable this debug log function for daily use.

To configure this function:

1. In the configuration file:

```
<install_drive>:\Enterworks\EPX\bin\config.properties
```

set the flag:

```
debug.out=true: logging is enabled.
```

```
debug.out=false: logging is disabled.
```

2. Restart the EPX services for the configuration change to take effect.

The debugging logs configured in the `Enterworks.properties` file provide debugging log messages, but they have a smaller footprint. It is not necessary to disable this debug log function for daily use; however, you may wish to do so once the EnterWorks system has been installed and live for a few months, and this debug log is not necessary. To configure this function:

1. In the configuration file:

```
<install_drive>:\Enterworks\EPX\bin\conf\Enterworks.pr  
operties
```

set the flag:

`debug.enable=true`: logging is enabled.

`debug.enable=false`: logging is disabled.

2. Restart the EPX services for the configuration change to take effect.

## Configure Cached Repository Permissions Tracing Log Messages

The shared configuration property `trace.cachedRepo.permission.enabled` allows you to enable or disable detailed logging for tracing user permission errors.

To configure the log output format:

1. Stop all EnterWorks services.
2. Update the shared configuration properties with the value given below. If the setting does not appear in the `sharedConfig.properties` files, add it to the files.

`trace.cachedRepo.permission.enabled`:

- o `debug.file.msg.format.type=false`: (Default) Log messages will not be produced.
- o `debug.file.msg.format.type=true`: Log messages will be produced.

3. Restart EnterWorks Services.

## System Maintenance

For the most recent recommendations for configuring and performing EnterWorks maintenance and monitoring tasks, see the Precisely EnterWorks online help at <https://support.precisely.com>.

### Clear Data Cache

Any time there are changes to the data model (for example, Profiles, Code Sets, Taxonomies, Hierarchies, etc.), the Data Cache must be cleared to ensure the changes go into effect.

1. Log into EnterWorks Classic as a user with administrative privileges.
2. Open the **Feature** bar, open the **System** folder, and select **Data Cache**.
3. Click the **Clear All** button to clear all caches.



4. In order for open tabs and windows to reflect the new state of the cleared cache, they must be refreshed.

## Configure System Maintenance Tasks

Server property settings allow you to specify how often specific system maintenance tasks are to be performed.

To set the server property settings:

1. Log into EnterWorks Classic as a user with administrative privileges.
2. Open the **Feature** bar, open the **System** folder, and select **Server Properties**.
3. Edit the properties as desired, then click the **Save** button.
4. Restart all EnterWorks services.

The server properties are as follows:

- **Schedule Cleanup Job**
- **Timeout User Sessions**
- **Clean Job History**
- **Clean Record History**
- **Clean System Log Files**
- **Clean Saved Sets**

The following settings are deprecated:

- **Configure Retail Product Generation:**
  - **Retail Product Generation Type**
  - **Retail Product Clone Max Count**