WebSphere Product Center: User Guide

Version 5.1

Note! Before using this information and the product it supports, read the information in "Notices" at the end of this document.

27 September2004

This edition of this document applies to WebSphere Product Center (5724-I68), version 5.1, and to all subsequent releases and modifications until otherwise indicated in new editions.

Table of Contents

Part I - Introduction

The WebSphere Product Center User Guide introduces and explains the functional components of WebSphere Product Center's core modules. Users unfamiliar with the WebSphere Product Center environment are recommended to read the "WebSphere Product Center: Overview" document, which provides the basic concepts that are used by the WebSphere Product Center.

Part I – Navigation
Learn how to navigate WebSphere Product Center, customize the home page, and configure user settings.

Part II - Product Manager
Learn about the primary PIM (Product Information Management) module. This section should be read by users who manage product information on a daily basis, i.e. Content Managers, Product Managers, Pricing Specialists, etc.

Part III – Collaboration Manager
Learn how information is imported into and exported from WebSphere Product Center. View the list of files that are stored in the Document Store. Create and manage workflows.

Part IV Data Model Manager
Learn how to manage scheduled activities, alerts and external communication. Create and edit specifications and mappings. Administer users, roles and privileges. View and create scripts.

Part V - System Administrator
Learn how to use the functional tools available to troubleshoot and maintain the WebSphere Product Center Application. Review configuration settings and application performance measurements. View various log files and system status information, as needed.

Note: Information in this document is subject to change. Graphics that appear in this document may not appear identical to what is displayed on the user's screen due to the version of WebSphere Product Center installed.

Ch 1 User interface

WebSphere Product Center is a web-based application that is accessible by a web browser with access to the web/application server where WebSphere Product Center has been installed. The graphical elements create an easy and pleasant experience for users.

Each of WebSphere Product Center’s modules is aligned with a user’s role and responsibility. By providing a role-based modular design, users can focus on information relevant to their position, which decreases the learning curve and accelerates product training.

Three primary user roles have been aligned with each of WebSphere Product Center module.

Module	**Role Name**	**Role Description**
Product Manager	Content Managers	Users that extensively use WebSphere Product Center for data entry and data maintenance
Collaboration Manager and Data Model Manager	Business Process Analysts	Users responsible for setting up the data module and business rules for Content Managers
System Administrator	System Administrator	Users responsible for the uptime of the WebSphere Product Center application and related services

User interface requirements

Browser software

Internet Explorer 6.0 is REQUIRED to be used with WebSphere Product Center. If WebSphere Product Center is accessed using a browser version prior to IE 6.0, an error message appears and notifies the user of a required upgrade to IE 6.0.

Minimum recommended resolution

The minimum recommended resolution is a "full" 1024 x 768 ratio. If the resolution is less than 1024 x 768 and the Windows taskbar is located on the left or right sides of the screen, the application will not fully maximize.

IE settings

The following IE setting are required when running WebSphere Product Center.

1. Automatically check for new version of stored page

2. Use the menu path Tools > Internet Options and click on the General tab. From the "Temporary Internet Files section, click "Settings".

3. Set cookies, files and history to be cleared after WebSphere Product Center has been upgraded

4. If a new version of WebSphere Product Center has been installed, all users need to clear their cookies, files, and history from Internet Explorer. This clears any temporary information that was used for the previous version of WebSphere Product Center.

- From the Internet Explorer browser window, use the menu path **Tools > Internet Options** and click on the General tab. Click Delete Cookies, Delete Files, and Clear History.

Browser toolbars

It is not recommended to install non-IE toolbars as it may cause issues when using WebSphere Product Center. For example, some toolbars may include ad pop-up blockers set as a default, which would not allow new pop-up window to open in WebSphere Product Center.

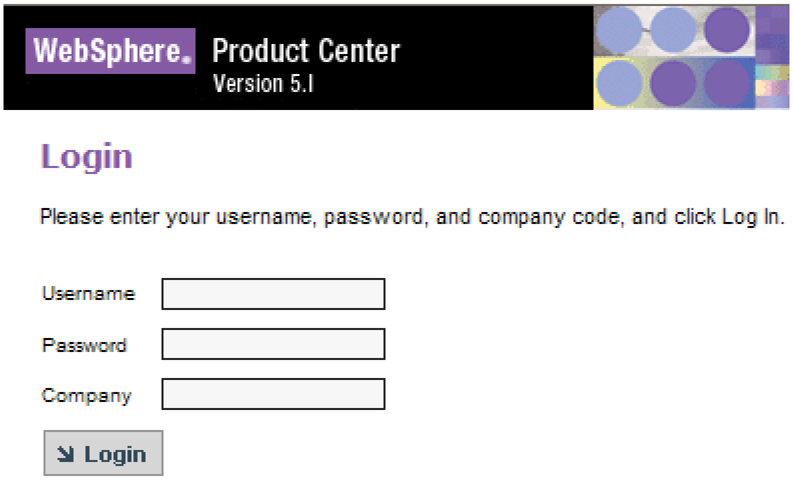
WebSphere Product Center Login

The WebSphere Product Center login screen is accessed through a web browser with intranet access.

Note: To login to WebSphere Product Center, a user account must be created, which is handled by the WebSphere Product Center Application Administrator.

Logging in to WebSphere Product Center

1. Type the URL provided by the WebSphere Product Center administrator. The following screen appears:



Please enter your username, password, and company code, and click Log In.

Figure 1.1 - WebSphere Product Center Login Screen

2. Type your "Username", "Password", and "Company", then click the Login button. To enter a company name, click "Options >>" and the "Company" field appears.

Once the user logs into WebSphere Product Center, the user’s Collaboration Area Console appears with access to all of WebSphere Product Center’s PIM modules in the menu bar and the left pane components, which can be customized to catalogs, alerts, selections, etc.

User information

The User’s name and associated company code appear in the top of the quick access bar of the WebSphere Product Center application.

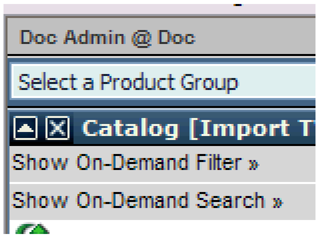


Figure 1.2 - User Information

Troubleshooting login

The following are some issues that may occur during the login process:

- Error message "Invalid username/password/company code"

  If this message appears in a popup dialog box, click OK and re-enter the information in the Login screen.

  If the WebSphere Product Center application continues to deny access, contact the local WebSphere Product Center Administrator to check the status of the user account.

- Browser cannot open the WebSphere Product Center web application.

  Click the Refresh button. If the problem persists, contact the WebSphere Product Center Administrator as the application server may be out of service or undergoing maintenance.

Left Pane access

Navigating WebSphere Product Center is made simple with the use of the user-configured Left Pane Access toolbar on the left side of the GUI. Adding a catalog allows quick access to its items and attributes. If desired, the Left Pane Access bar can be hidden to expand WebSphere Product Center's workspace.

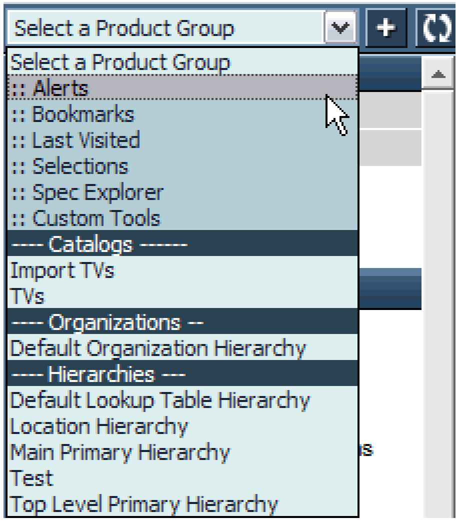


Figure 1.3 - Object list

Alerts – Identifies the status of any current alerts.

Bookmarks – Links to user-defined locations, including searches. Great for components most commonly use.

Last visited – Stores the last five pages that have been visited.

Selections – All saved catalog selections are displayed.

Spec Explorer –Displays the search result of specifications and attributes.

Custom Tools – Displays the available customs tools available to the user

Catalogs – Displays a list of available catalogs to the user

Organizations –All Organizations available are displayed in the drop-down selection.

Hierarchies –All Hierarchies available are displayed in the

	drop-down selection.

Note: Components and catalogs in the Left Pane appear in the order they are added.

The following icons are available in the left pane and are referred to in the user guide by the names shown in the table below:

↻ **REFRESH** - refresh the left pane	▲ **COLLAPSE** - collapse a product group
+ **ADD** - add a Product Group	▼ **EXPAND** - expand a product group
☒ **EXIT** - Remove the product group from Left Pane	

Hide Left Pane

Hiding the Left Pane can increase the amount of usable WebSphere Product Center workspace.

Hiding the left pane access bar

1. To hide the Left Pane Bar, select **Window > Hide left pane** from the menu bar. The Hide left pane is marked with a check.

2. To unhide the Left Pane Bar, unselect **Window > Hide left pane** from the menu bar. The Hide left pane is unchecked.

Alerts table

The alerts table presents the status and number of alerts that are active. The types of alerts include:

- Error alerts
- Info alerts
- Actionable alerts

Adding the Alerts table to the Left Pane

1. To add the Alerts Component to the Left Pane, select **Alerts** from the Left Pane drop-down selection and click the **ADD** button. The Alerts component appears in a table, which displays any active alerts.

2. Click on any alert number hyperlink to view the details for the alert. Click **REFRESH** to refresh the table information.

Bookmarks

Different areas of WebSphere Product Center can be bookmarked, which appear in the Bookmarks list.

Adding Bookmarks list to the Left Pane

- To add Bookmarks to the Left Pane, select **Bookmarks** from the Left Pane drop-down field and click the **ADD** button.

Adding or removing a bookmark from the Bookmarks list

1. Navigate to the page in WebSphere Product Center that will be bookmarked.

2. Used the Add Link button from the Bookmarks list in the Left Pane to add the page as a bookmark.

3. Use the Remove Link button from the Bookmarks list in the Left Pane to remove bookmark links from the Bookmarks list.

⊞	Add link
☒	Remove link

Last Visited List

When a page is visited, it is automatically added to the Last Visited component and captures the last ten pages that have been visited. The list is cleared when the user logs out of WebSphere Product Center.

Adding Last Visited list to the Left Pane

1. To add the last visited list to the Left Pane, select **Last Visited** from the Left Pane drop-down field and click the **ADD** button.

2. Click the page name to display the page.

Note: The last visited list resets when the user logs out.

Selections

Adding the Selections component to the Left Pane will display all of the Basic Selections that have been created. When an catalog item selection is created, they appear in the Item Selection Console and are automatically added to the list of Selections in the Left Pane drop-down field.

Adding Selections to the Left Pane

1. To add selections to the Left Pane, select **Selections** from the Left Pane drop-down field and click the ADD button.

2. Once a selection has been added to the Click on any selection name from the Left Pane drop-down field to view the results/details for the selection. Click **REFRESH** to update the list of selections.

Spec Explorer

Add the Spec Explorer to the Left Pane to search and view all specs available to the user.

Adding Spec Explorer to the Left Pane

1. To add the last visited list to the Left Pane, select **Last Visited** from the Left Pane drop-down field and click the **ADD** button.

2. From the Spec Explorer, click Show Search and search for a spec.

3. From the results list, click on a spec name and it is added to the Spec Explorer.

Custom Tools

All custom tools (customized application within WebSphere Product Center) can be added to the Left Pane.

Adding Custom Tools to the Left Pane

1. To add custom tools to the Left Pane, select **Custom Tools** from the Left Pane drop-down field and click the **ADD** button.

2. From the Custom Tools section, select a custom tool to add to the Left Pane.

Catalogs

All catalogs available to the user including the catalog's category hierarchy is listed in the drop-down field and can be added individually.

Adding a catalog

- To add a catalog to the Left Pane, select a catalog from the drop-down field and click **ADD**. The Catalog appears in a collapsed format.

Organizations

All organizations that have been created are included in the drop-down field and can be added to the Left Pane.

Adding an Organization Hierarchy

- To add an organization hierarchy to the Left Pane, select an organization hierarchy from the drop-down field and click **ADD**. The hierarchy appears in a collapsed format.

Hierarchies

All hierarchies that have been created are included in the drop-down filed and can be added to the Left Pane.

Adding a hierarchy

1. To add a hierarchy to the Left Pane, select a hierarchy from the drop-down field and click **ADD**. The hierarchy appears in a collapsed format.

2. To remove the catalog from the Left Pane, click **Exit**. This does not delete the catalog, but removes it form the Left Pane view.

Short menus

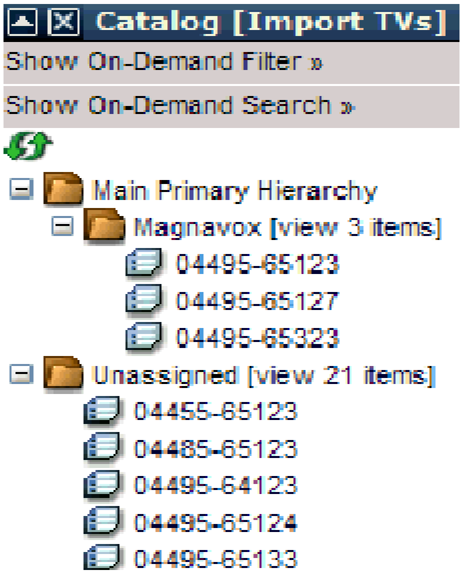
There are a series of short menus that can be accessed in the Left Pane for additional functional options. To access a short menu in the Left Pane, right click on an object in the Left Pane and a short menu appears with different functional options. The available menu options differ between objects.

For example, right click on an item and the short menu displays the following functions (Cut, Copy, Insert Before, Remove, Delete). If a menu selection text is colored gray, then the function is not available.

Catalog/Category Explorer

Add a Catalog to the Left Pane to view its contents. The catalog explorer is displayed in a Windows explorer folder format.

- Folder structure represents the catalog’s hierarchy tree
- Items are ordered within category folders
- Unassigned items are placed in a separate "Unassigned" folder
- Click on a category to view its attributes
- Click to view an individual item or list of items
- Right click in the Left Pane to access the short menus



Catalog [Import TVs]

Show On-Demand Filter »

Show On-Demand Search »

- Main Primary Hierarchy
  - Magnavox [view 3 items]
    - 04495-65123
    - 04495-65127
    - 04495-65323
- Unassigned [view 21 items]
  - 04455-65123
  - 04485-65123
  - 04495-64123
  - 04495-65124
  - 04495-65133

Figure 1.4 - Catalog/Category explorer

Changing catalog version, hierarchy, or selection used

1. To change the current version, hierarchy, selection that is being used for a catalog, click **Show On-Demand Search**.

2. The current values for the catalog are displayed. Change the Version, Hierarchy, and/or Selection as desired and the catalog is updated in the Left Pane.

3. Click **Hide On-Demand Filter** to collapse this feature.

Show Populated Categories

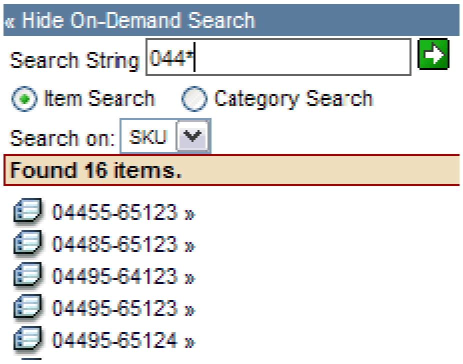
There is an option to hide the list of categories that are not populated with items. From the Show On-Demand Filter features in the Left Pane, click on **Show Populated Categories** checkbox.

Searching a catalog in the Left Pane

A search for an Item or Category can be performed in the Left Pane. Each catalog that is added to the Left Pane has a Show On-Demand Search function, which allows an item or category search by a user-defined string or primary key.

1. Click **Show On-Demand Search** to expand this feature and enter a search criterion.

2. Select to perform an Item Search or Category Search and press Enter or click ➡ to view the results.



« Hide On-Demand Search

Search String 044*

● Item Search ○ Category Search

Search on: SKU

Found 16 items.

04455-65123 »
04485-65123 »
04495-64123 »
04495-65123 »
04495-65124 »

Figure 1.5 - Search in left pane

3. Click **Hide On-Demand Search** to collapse this feature.

Menu bar

At the top of the application is a menu bar that navigates to all the major functional areas of WebSphere Product Center. Each of the top-level menu items identifies a separate major functional area of WebSphere Product Center and all sub-level menu items include related functions to the major functional area.

- Click on a top-level menu item to view all of its associated components

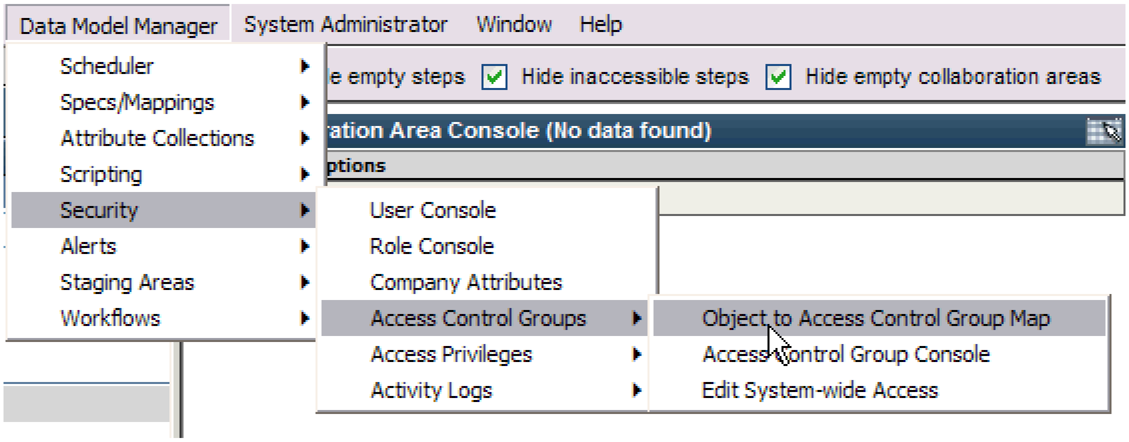
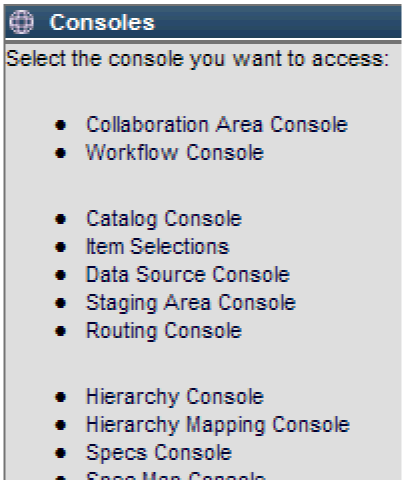


Figure 1. 6 – Menu bar

Consoles

WebSphere Product Center uses the concept of consoles throughout the application (i.e. Catalog Console, Import Console, etc.). A console allows a user to configure and save actions that are often repeated, thus complex operations can be performed with a single click of the mouse.

Note: The list of consoles can be accessed from Bookmarks in the Left Pane. "Consoles" and "My Home" are default bookmarks and cannot be deleted.



Consoles

Select the console you want to access:

- Collaboration Area Console
- Workflow Console

- Catalog Console
- Item Selections
- Data Source Console
- Staging Area Console
- Routing Console

- Hierarchy Console
- Hierarchy Mapping Console
- Specs Console
- Spec Map Console

Figure 1.7 - WebSphere Product Center console list

Accessing the console list

To access the list of available consoles in WebSphere Product Center, add the bookmarks component to the Left Pane.

1. Select Bookmarks from the drop-down menu and click the **ADD** icon.

2. The default bookmark that appears, which cannot be deleted, is "My Home", "Consoles", and any saved Rich Search templates. Click the Consoles bookmark to view the list of available consoles.

Navigation map

WebSphere Product Center’s navigation map displays a detailed level of every WebSphere Product Center component, which is mostly organized as they appear in the dynamic menu bar.

Accessing the navigation map

1. To access the navigation map, use the following menu path:

Window > Navigation Map

2. Several items in the navigation map are hyperlinked. Click on the hyperlink to display the corresponding page.

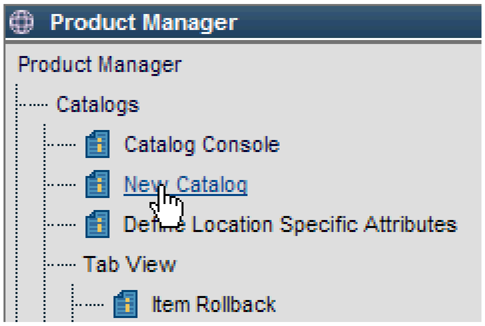


Figure 1.8 - WebSphere Product Center 's navigation map

Ch 2 Home

Each WebSphere Product Center user has a customizable home page with settings that are saved in the WebSphere Product Center system and appear each time the user logs in. This chapter covers the Home menu items and describes the following topics:

- View My Home
- Configure My Task List
- Edit My Settings
- Configure My Profile
- Create Task List

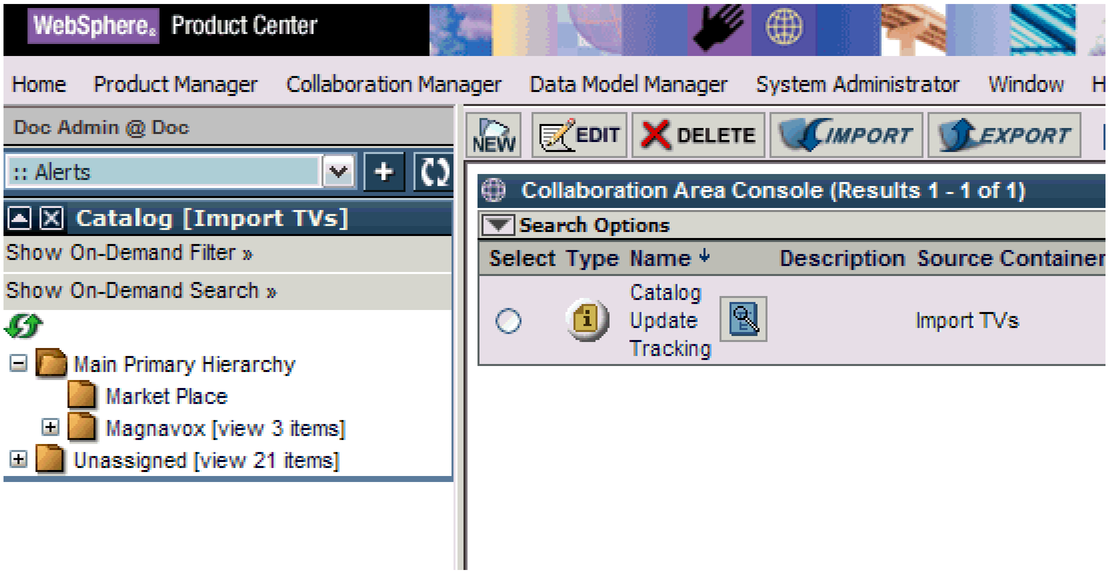
Home

Select the Home module from the menu bar to view the associated components that make up the Home module.

- My Home
- My Task List
- My Settings
- My Profile
- Signout
- Exit

My Home

A typical home page includes the left pane bar and the Collaboration Area Console in the right frame. The WebSphere Product Center user interface can be customized to a user’s role. Allow or restrict selected components of WebSphere Product Center for all roles and it is applied to all users assigned to the role. In the case of a user being assigned to multiple roles, the highest degree of access is allowed.



Select	Type	Name	Description	Source Container
○		Catalog Update Tracking		Import TVs

Figure 2.1 - User home page

My Task List

WebSphere Product Center's user interface allows further customization that allows a user to configure a Task List screen that is specific to the user. The task list settings are saved on the application server; therefore the user can login from any web browser location within the network and access their customized settings.

The task list can be customized to display the following components:

- **Alerts** – a display of alerts associated to the user
- **Approval** – a display of requests awaiting approval
- **Company Activity History**– a display of events that have been performed or scheduled

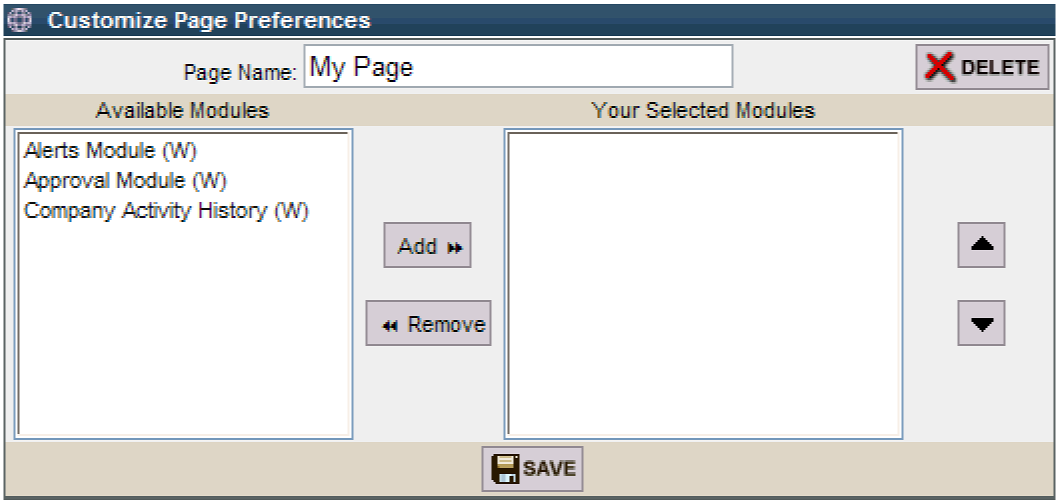
Creating my task list page

Before a user can use the components of My Task List, the page must first be created.

1. Use the menu path **Home > My Task List**. Click **NEW** on the My Page List table heading.

2. Enter a name for the Task List screen in the **Page Name** field.

3. On the left is a list of available modules and on the right is the list of selected modules. Select a module from the Available Modules and click the **ADD**.



Page Name: My Page

Alerts Module (W)
Approval Module (W)
Company Activity History (W)

Figure 2.2 - Customized page preferences

4. To remove modules from the customized page, select a module from "Your Selected Modules" and click the remove icon. Arrange the order in which the modules are displayed using the up and down arrow buttons.

5. When all the modules have been selected for the customized view, click the **Save** button. The selected modules appear in the user’s My Task List view.

6. Click on the My Task List page to edit the available modules.

Note: Remember to click **Save** to commit the settings.

My Settings

User settings can be reviewed and edited in the User Settings interface. The settings are specific to the user and are saved on the application server; therefore the user can login from any web browser location within the network and access their customized settings.

Updating user settings

1. Use the menu path **Home > My Settings** to update the user settings.

2. Modify the user settings and click the Save button to store the new configuration.

Note: Refer to the User Settings table on the next page for a description of each setting.

Figure 2.3 - User settings interface

User settings description

General Settings	
Locale for User Interface Display	User interface translation.
Locale for Item and Category Data Display	Select a locale to view item and category data in the GUI. When NONE is selected, the base name is used.
Restrict the displayed attributes in item and category screens to the selected locale	Restricts the display of attributes in the item and category screens to selected locales. Note: No locales will be available if no locales have been configured for use. Set locales in "Company Attributes".
Timezone	Select the proper value based on the user’s region
Select Datetime Input Format	Select the Date/time format used for inputs
Base font size used application wide	Select the size of the font displayed in the GUI
Toolbar position	Select to position the toolbar at the top or bottom of the application screen
Email Upon Alert	Email sent to the address set in the user profile, informing of an alert

Table Display Settings	
Rows per Page in Specs Console	The number of rows that appear on a single page in the Specs console
Rows per Page in Scripts Console	The number of rows that appear on a single page in the Scripts console
Rows per Page in Item Set	The number of rows that appear on a single page in screens displaying an set of items
Rows per Page in Multi Edit	The number of rows that appear on a single page in the multiple edit screen
Rows per Page in Alerts Display	The number of rows that appear on a single page in the Alerts display
Rows per Page in Lookup Table	The number of rows that appear on a single page in the Lookup table
Rows in Textarea	The number of rows to display in a text area
Columns in a Textarea	Set the number of columns in a text area
Number of characters in a string beyond which a multiline display is used to conserve horizontal space	Set the number of characters display in a string beyond a mulit-line display
Specific Screen Settings	
Display Type of Node in Specs screen	Display the node type icon in the spec tree
Use detailed node icons in the Spec screen	Select this option o provide detailed node information
Remember Last Saved Category Tree Used For Browsing a Catalog in the Left Pane	When a user logs out and logs back in, the last saved category tree used for browsing a catalog in the left pane is remembered
Display Type of Node in Item Edit	Display the node type icon in the Item

Screen	Edit Screen
In Multi Edit, sort only visible records (if unchecked the entire workset is sorted)	Select to sort records on Multi Edit page, otherwise it will sort the whole work set
In Multi Edit, show all category specific attributes from user defined view when category is not known	Display all category specific attributes in a user-defined view. Only when the category is not known.
Display Spec Attribute as	Select popup window or drop-down menu
For editing and entering data, use:	Select Stand Alone or Advanced content authoring screens
Maximum number of values for which a dropdown is available in Multi Edit	Select a number of values that can appear in a dropdown list in the multiple edit screen
Sort all item lists by primary key by default:	Sorts the primary key by default for item lists
Enable persisting of lookup table sort columns	Set to "yes", the sorting column of a lookup table (chosen by the user) will persist. Set to "no", lookup tables will always be sorted by the primary key column.
Go to Item List screen for displaying 1 item	Select whether or not to use the Item List screen to display 1 item
Number of entries in Data Entry beyond which [Save] and [Macro] are done in the background	Set the number of entries to be saved in the background
Number of entries in Data Entry beyond which [Macro] or [Set Value in Column] automatically save the entries to the database	Set number of entries that are automatically saved to the database
For initial display of multivalued groupings, show	Show first, last or all multivalued groupings
For the initial display of grouped attributes	Set to expand or collapse groupings
Always display help text	Set to always display help text

Display the locked icon on the catalog and hierarchy consoles	Set to display locked icon on catalog and hierarchy consoles
Display the UOM Browser in item edit screen	Select to display UOM browser
Hide left navigation pane	Hide the Left pane at login
Use a Custom Tool page as the start page	Select a custom tool page to use as the start page

Before exiting/signing out of WebSphere Product Center, make sure to save any work.

Signout

When Signout is used, the user is returned to the Login screen. Before signing out of WebSphere Product Center, make sure to save any work.

Signing out of WebSphere Product Center

- To sign out of WebSphere Product Center and return to the Login page, use the following menu path: **Home > Signout**

Exit

When Exit is used, the browser is closed. Before exiting out of WebSphere Product Center, make sure to save any work.

Exiting WebSphere Product Center

- To exit WebSphere Product Center web application and close the browser, use the following menu path: **Home > Exit**

Inactivity dialog box

After 15 minutes of non-activity, a dialog box appears warning the user that the session will timeout. Click **OK** to reset the session or **Cancel** to logout.

Part II - Product Manager

The Product Manager module, designed for business users, reduces the complexity in managing content from multiple sources and audiences. It allows users the ability to provide customer-focused content.

The Product Manager offers the following features:

- Catalog creation and management
- Category Hierarchy/Taxonomy and management
- Lookup Table creation and management
- Importing and exporting of data to the catalogs
- Item selections

Part II section covers each of the components and sub-components that are available within the Product Manager module.

Product Manager Component	**Description**
Catalog	The catalog console is where most of the data management activities start
Hierarchies	Update hierarchies (category and organization)
Selections	Selections are subsets of catalogs which can be used for edit, updates and exports
Reports	Custom Reports can be written according to specific business requirements on data statistics, audit trail, etc.
Lookup Tables	Lookup tables are used to store data that items, categories, validation rules and other business rules

Ch 3 Catalogs

Catalogs are repositories or containers for items with the following characteristics:

- Catalogs are defined by several components that are required to build a catalog (Name, Primary spec, Primary category hierarchy, (ACG)Access control group)
- Additional optional attributes can be specified for any catalog (Secondary category hierarchy, Linked catalogs, catalogs scripts (pre and post processing)
- Catalogs can be linked, allows attributes to be managed for a single item across multiple catalogs
- Catalog views can be customized to personalized how item attributes are presented to a user

Catalog console

The Catalog Console is where all the centralized product information is managed and manipulated before it is exported to a defined destination.

Through the Catalog Console, the following tasks can be performed:

- Browse and modify a catalog
- View items within a category
- View and modify catalog's associated specs
- Edit the contents of a catalog using the Single or Multiple edit screens
- View catalog attributes
- View catalog differences
- Rollback a catalog version
- Search a catalog's content
- Delete a catalog
- Customize catalog views

The Catalog Console displays all of the current catalogs that have been created. The console view can be customized to show specific catalog attributes, if desired. The default view shows the catalog name, catalog spec, primary/secondary hierarchies, access control group, and the catalog view that is curretnly applied to the catalog.

Customizing the console view

The console can be customized to sort or hide columns. The settings are saved to the user’s profile. To customize the view of the Catalog Console, do the following.

1. In the top right corner of the Catalog console, click the configure button . The Configure table appears in a popup window.

2. To sort a column, make a selection in the Sort Column. Only one selection can be made.

3. To hide a field, make a selection in the Hide column. Multiple selections can be made.

4. In the "Other Options" table, set the sorting order to display in ascending or descending order and set the number of rows to display per page.

5. To save the customized settings, click **Save**. The Catalog Console appears with the new customized settings.

Catalog console columns

Name	The name of the catalog.
Spec	The name of the catalog spec used to build the catalog. When clicked, the specification can be viewed an edited.
Primary Hierarchy	The name of the hierarchy used to organize the catalog items. When clicked, the hierarchy details screen appears.

Secondary Hierarchies	The name of the secondary hierarchy used to create sub-categories. When clicked, the hierarchy details can be viewed.
Access Control Group (ACG)	The Access Control Group used to control the access levels for each role.
Views	The catalog view that is applied to the catalog. New catalogs are set to use the system default. The edit icon allows the user to edit the customized view; the system default view cannot be edited.

Catalog console toolbar

The Catalog console toolbar contains buttons that are used to manage the catalogs in the console.

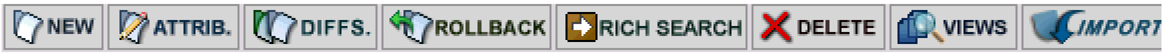


Fig 3.1 - Catalog console buttons.

NEW	Create a new catalog
ATTRIBS.	Edit/modify the attributes of a catalog. Create different versions of a catalog, view a different version of a catalog, rename the version of a catalog, create a new hierarchy, add a second hierarchy to a catalog, review the summary statistic of a particular catalog
DIFFS.	Provides the ability for the user to run a comparison between two different versions of a catalog
ROLLBACK	Provides the ability for the user to

	rollback a catalog
RICH SEARCH	Provides the ability for the user to run a more comprehensive search within a catalog and view the results in an item list table
DELETE	Delete a catalog
VIEWS	Create different views of a catalog
IMPORT	Select a catalog and click the Import button. All imports associated to the catalog appears
EXPORT	Select a catalog and click the Export button. All exports associated to the catalog appears

Catalogs

The Catalogs component of the Product Manager is used to design, structure, and maintain product information within catalogs. This section assumes that all components required to create a catalog have been completed. Refer to the appropriate sections in this user's guide for information on creating the following catalog components:

- Primary/secondary specs
- Access Control Groups (optional)
- Hierarchies (optional)

New Catalog

Catalogs can be designed and structured to meet the requirements of business units. Once a data model has been built, catalogs can be created to maintain product information within catalogs.

This section assumes that all components required to create a catalog have been completed. Refer to the appropriate sections in this user’s guide for information on creating the following catalog components:

- Primary/secondary specs
- Access Control Groups (optional)
- Hierarchies (optional)

Creating a new catalog

Use the menu path **Product Manager > Catalogs > New Catalog** or click the **New** icon from the Catalog Console. The Create Catalog wizard appears. Complete each step of the wizard.

1. **Select Specification**: Choose the specification to use for the catalog and click **Select**. To edit the selected spec, click the **Edit** button or click **New** to create a new spec for the catalog.

2. **Select Display Attribute**: Select the spec attribute that is to be displayed for the catalog.

3. **Select Access Control Group**: Select the ACG used for the catalog and click **Select**. If desired, edit or create the ACG.

4. **Catalog Name**: Enter a name for the catalog and click the **Create Catalog** button.

5. **Select Hierarchy**: Select a hierarchy to use for the catalog and click the **Select** button. Click the **Edit** button to edit the current selected hierarchy or click **New** to create a new hierarchy.

A message box appears notifying of a successful catalog creation. Click the "Catalog Console" hyperlink or the back arrow icon to return to the Catalog Console page.

Delete catalogs

The Delete function allows a user to "clean things up" by deleting old or useless catalogs. It is important to note that once a catalog is deleted, there is no way to recover it. The catalog cannot be used or otherwise be viewed again.

When deleting a catalog, there are two confirmation dialog boxes that occur which makes it hard to delete a catalog by accident. When a catalog is confirmed for deletion, it is placed in the scheduler for completion.

Deleting a catalog

1. From the Catalog Console, select a catalog to delete by clicking the radio button next to the name of the catalog.

2. From the toolbar, click **Delete**. A dialog box appears confirming the operation. Click **OK** to delete the catalog or Cancel to quit the operation.

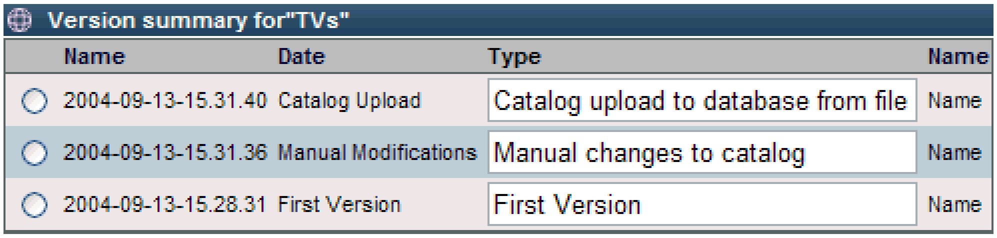
3. If **OK** was clicked, the "Choose a Catalog to Delete" wizard appears with a warning message notifying the user that the operation is not reversible. Click the **Delete** button to complete the deletion of the catalog.

4. Another dialog box appears to confirm the deletion of the catalog, Click **OK** to complete the catalog deletion.

5. A message appears notifying the user that the catalog deletion has been scheduled. Click the timer icon to view the schedule status.

Catalog Versioning

A version is automatically created when changes have been made to a catalog, whether it is an update import, change of catalog attributes, manual changes to items, etc. All catalog versions appear in the Version Summary table of the Catalog Attributes screen.



Version summary for"TVs"

	Name	Date	Type	Name
○	2004-09-13-15.31.40	Catalog Upload	Catalog upload to database from file	Name
○	2004-09-13-15.31.36	Manual Modifications	Manual changes to catalog	Name
○	2004-09-13-15.28.31	First Version	First Version	Name

Figure 3. 2 - Version summary table

Creating a catalog version

To manually create a tagged version of a catalog, which timestamps the catalog version, go to the Catalog Details screen:

- Left Pane - Right-click in the Left Pane and select Attributes from the short menu
- Catalog Console - select a catalog and click Attib.

Enter a name in the "Add a version with a name" field and click **+ADD**. The new catalog version is saved to the "Version Summary" table.

From the left p

Rename a catalog version

To rename a catalog version, select a catalog version from the Version Summary table, edit the version name, and click the **Rename Version** button from the toolbar.

View specific catalog version

Add a catalog in the Left Pane, click "Show On- Demand Filter". Select a catalog version to view and the catalog is populated with the catalog version data.

Note: The selection from the Version drop-down field matches that of the Version Summary table.

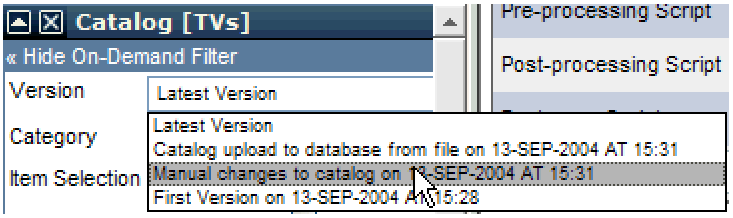

Catalog [TVs]
« Hide On-Demand Filter
Version
Latest Version
Category
Item Selection
Latest Version
Catalog upload to database from file on 13-SEP-2004 AT 15:31
Manual changes to catalog on 13-SEP-2004 AT 15:31
First Version on 13-SEP-2004 AT 15:28
Pre-processing Script
Post-processing Script

Figure 3. 3 - Change catalog version

Catalog differences

The Catalog Differences function allows the analysis between two catalog versions. Choose to view details on various difference types (added, modified, deleted, unchanged, or all).

From the Catalog Console, select a catalog and click the **Diffs.** button. The "Catalog Differences" wizard appears. Complete each step of the wizrd.

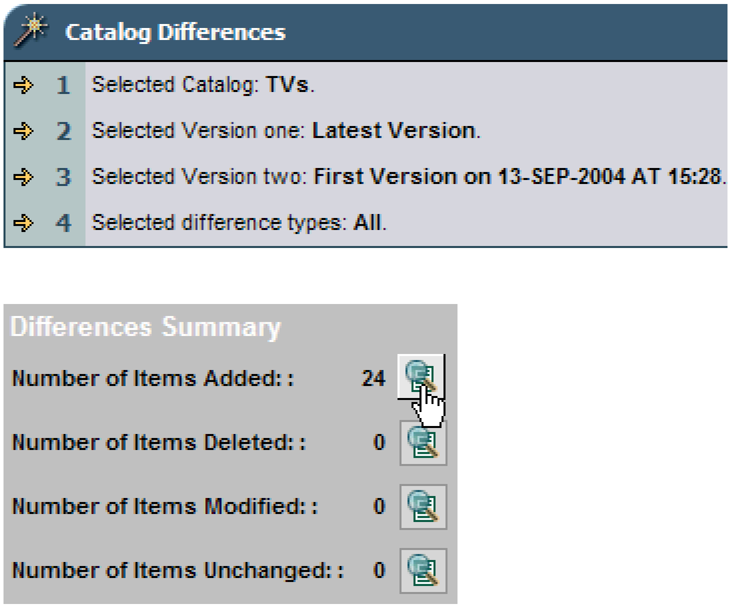
1. **Select Catalog:** Automatically filled with the name of the catalog selection from the Catalog Console.

2. **Select Version one:** Select the first catalog version.

3. **Select Version two:** Select a second catalog version

4. **Select difference type**: Click the Question Mark icon and select a difference type, then click **Save** to close the window, then click **Select** to apply the difference type to perform.

5. A Difference Summary table appears. Click on the View button to display the Item List details.



Catalog Differences

1 Selected Catalog: **TVs**.

2 Selected Version one: **Latest Version**.

3 Selected Version two: **First Version on 13-SEP-2004 AT 15:28**.

4 Selected difference types: **All**.

Differences Summary

Number of Items Added: :	24
Number of Items Deleted: :	0
Number of Items Modified: :	0
Number of Items Unchanged: :	0

Figure 3. 4 - Differences Summary

Catalog rollback

A catalog "Rollback" enables a user to change the version of a catalog that is currently being used to a different catalog version, which may contain modified items. This is especially useful if changes that were made to a catalog were later determined to be unnecessary or otherwise done by accident.

After clicking on the Rollback button, the user must select the version of the catalog that he or she would like to roll back to. A catalog is versioned just before and just after data is moved into (during an import) and out of (during an export) WebSphere Product Center.

By default, each version is given a name in terms of two properties: A. the operation that was performed on it and B. the date and time that the operation was performed.

USE WITH CAUTION!!! This operation is not reversible.

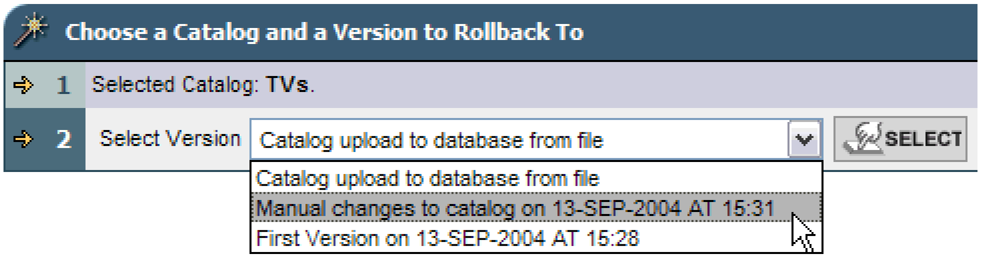
Note: Specs are not reverted during a catalog rollback so items will be displayed according to the latest specs. Attributes for spec nodes that have been deleted will not be shown although they existed in the rollback version.

Performing a catalog rollback

To rollback to a different catalog version, do the following:

1. From the Catalog Console, click on the radio button next to the catalog name and click the **Rollback** button.

2. Select a catalog version from the drop down selection and click the SELECT button. A message appears to remind you that the action is not reversible, proceed by clicking on the Rollback catalog button and then OK in the popup dialog box to confirm the action.



Choose a Catalog and a Version to Rollback To

1 Selected Catalog: TVs.

2 Select Version: Catalog upload to database from file

SELECT
Catalog upload to database from file
Manual changes to catalog on 13-SEP-2004 AT 15:31
First Version on 13-SEP-2004 AT 15:28

Fig 3.5 - Select catalog to rollback

3. The rollback is scheduled; click the status icon to view the status.

Note: When a rollback is completed, all catalog versions that occurred after the selected version is cleared from the Versions Table.

Catalog attributes

The attributes of a catalog can be viewed/modified through the Catalog Detail screen. There two methods to view a catalog's attributes.

- From the Catalog Console, select a catalog and click the Attrib button
- From the Left Pane, display a catalog and right-click next to the catalog name. Select "Catalog Attributes" from the short menu.

CATALOG ATTRIBUTES	
Name	Name of the catalog
Spec	Name of associated catalog spec
Primary Hierarchy	The primary hierarchy being used
Add Secondary hierarchy	Click the **+ADD** button to include a secondary hierarchy
Use Ordering	Check to apply ordering to the catalog.
Add a version with the name	Customize the name of a catalog version
Catalog display attribute	Select an attribute to display in the Left Pane
User Defined Core Attribute	Select the associated user defined core attribute collection for the

Collection	catalog
Entry Build Script	Build a script that runs after data retrieval for items/categories
Pre-processing Script	Build a script that manipulates the data before it is imported into a catalog
Post-processing Script	Build a script that manipulates the data after it is imported into a catalog
Post-save Script	Build a script that runs after an item is saved
Link Attributes	Displays nodes that have been linked to other catalogs
Version Summary	Displays all of the versions for the catalog

Scripts are created to validate and cleanse data before it is imported or exported. The catalog attributes allow for the selection of various script types to be processed at different stages (before import, after being saved, before export). The specs that are used for a catalog allow further manipulation to data such as applying validation or value rules. The combination of defining catalog attribute scripts and spec attributes allows for flexibility when manipulating data.

Although there are various types of scripts and rules that can be used to process data not every type needs to be defined and each type is performed in the specified order listed below:

- Entry Build script
- Pre-processing script
- Validation and Value rules
- Post-processing script
- Post-save script

Secondary hierarchies

Multiple hierarchies can be associated to a single catalog through the Catalog Details screen. The Primary Hierarchy cannot be removed.

Adding a secondary hierarchy

1. From the Catalog Console, select a catalog and click **Attribs**.

2. Select a hierarchy from the Add a Secondary hierarchy drop-down selection and click **+ADD**.

3. A message appears confirming the action.

4. If desired, add another secondary hierarchy.

5. To remove the associated Hierarchy, click the **Delete** button. A confirmation dialog box appears.

User defined logs

The function of the user-defined logs can be used to track deltas for catalogs or hierarchies. All changes are captured and viewed using the Log link in the Catalog or Hierarchy detail screen. Logs can be created for any of the available scripts that are built through the Catalog/Hierarchy Details screen.

Creating a user defined log for a catalog

1. From the Catalog Console, select a catalog and click **Attribs**. The details of the Catalog are displayed.

2. Click Logs and a table of user-defined logs appear. Click **New** to create a new user-defined log.

3. Enter a name and description for the log and check the Running Log checkbox, if desired, and click **+ADD**.

Note: If the Running Log box is not checked, the system captures the most recent change, whereas if the box were checked the system would capture all changes that have been made.

4. At the Catalog Details screen, select a script (i.e. Post-save script) to create and write it to a log. For more information on creating scripts, refer to WebSphere Product Center’s Scripting Documentation.

Figure 3. 19 - - Create a script that writes to a log file

Editing user-defined log for a catalog

1. From the Catalog Console, select a catalog and click **Attrib**. The details of the Catalog are displayed.

2. Click Logs and a table of user-defined logs appear. Click the **Edit** button next to the log name that is to be edited.

3. Make any changes and click **Save**.

Deleting user-defined log for a catalog

1. From the Catalog Console, select a catalog and click **Attrib**. The details of the Catalog are displayed.

2. Click **Logs** and a table of user-defined logs appear.

3. Click the **Delete** icon next to the log name that is to be deleted.

Creating user defined log for a hierarchy

1. From the Hierarchy Console, select a hierarchy and click **Attrib**. The details of the tree are displayed.

2. Click **Logs** and a table of user-defined logs appear. Click **New** to create a new user-defined log.

3. Enter a name and description for the log and check the Running Log checkbox, if desired, and click **+ADD.**

4. At the Category Details screen, select a script (i.e. Post-save script) to create and write it to a log. For more information on creating scripts, refer to WebSphere Product Center's Scripting Documentation.

Editing user defined log for a hierarchy

1. From the Hierarchy Console, select a hierarchy and click **Attrib**. The details of the tree are displayed.

2. Click **Logs** and a table of user-defined logs appear. Click the **Delete** icon next to the log name that is to be edited.

3. Make any changes and click **Save**.

Deleting user-defined log for a catalog

1. From the Hierarchy Console, select a hierarchy and click **Attrib**. The details of the tree are displayed.

2. Click **Logs** and a table of user-defined logs appear.

3. Click the **Delete** icon next to the log name that is to be deleted.

Ch 4 Content Authoring

Item edits are performed using the Standalone or Advanced Content Authoring screens. From either screen type, items can be added or edited. The Advanced Content Authoring screen allows multiple items to be added/edited at one time.

Differences between Standalone and Advanced content authoring screens:

Standalone Content Authoring screen	**Advanced Content Authoring screen**
• No Rich Search, Single Edit, or Multiple Edit Tab • No navigation to move to next or previous record	• Rich Search, Single Edit, and Multiple Edit tabs are available • Exit button

Setting content authoring screen type

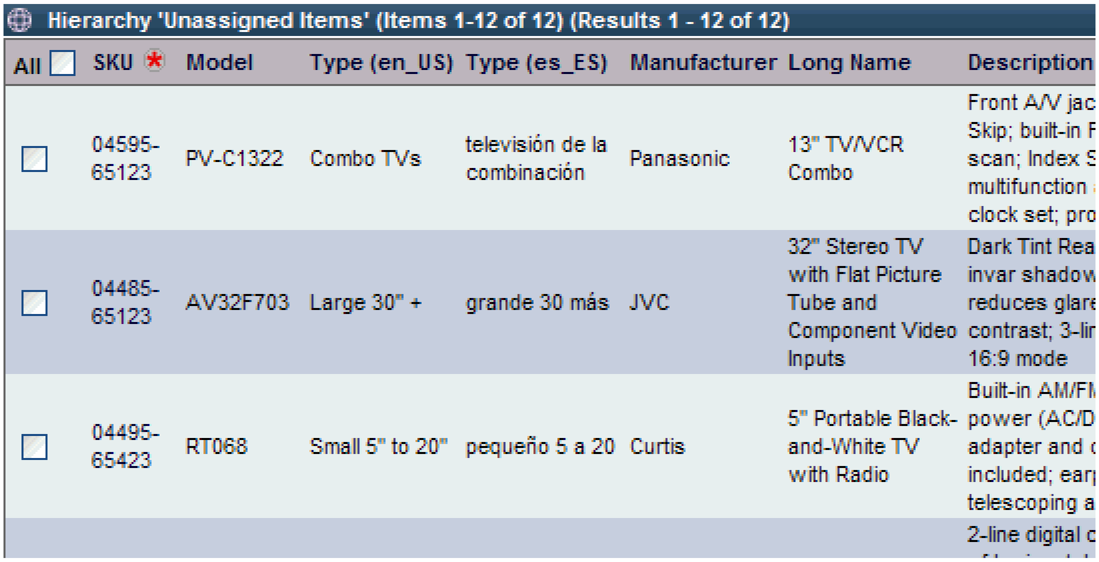
The type of content authoring screen used is set in the "Specific Screen Settings" configuration of My Settings. Find the "For editing and entering data, use:" row and select to use one of the following

- Stand Alone Content Authoring Screens
- Advanced Content Authoring Screens

The default setting is set to "Standalone Content Authoring screen".

Item List screen

When several items are selected in the Left Pane, they are displayed in an item list screen. For single items, they are displayed in a single edit screen. This screen allows a user to sort the list by any column of the item list table and to easily navigate through long lists of items.



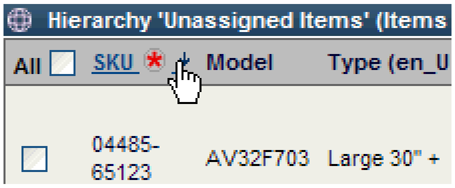
Hierarchy 'Unassigned Items' (Items 1-12 of 12) (Results 1 - 12 of 12)

All □	SKU *	Model	Type (en_US)	Type (es_ES)	Manufacturer	Long Name	Description
□	04595-65123	PV-C1322	Combo TVs	televisión de la combinación	Panasonic	13" TV/VCR Combo	Front A/V jac Skip; built-in F scan; Index S multifunction clock set; pro
□	04485-65123	AV32F703	Large 30" +	grande 30 más	JVC	32" Stereo TV with Flat Picture Tube and Component Video Inputs	Dark Tint Rea invar shadow reduces glare contrast; 3-lir 16:9 mode
□	04495-65423	RT068	Small 5" to 20"	pequeño 5 a 20	Curtis	5" Portable Black-and-White TV with Radio	Built-in AM/FM power (AC/D adapter and included; ear telescoping a
							2-line digital c

Figure 4.1 - Sample Item List screen

Sorting item list

If desired, items can be sorted in descending order by clicking on any Attribute Name column headings of the Item List table. The down arrow indicates that if clicked, the items are sorted in descending order. If the up arrow is displayed, click it to sort the item list in ascending order.



Hierarchy 'Unassigned Items' (Items

All □	SKU * ↓	Model	Type (en_U
□	04485-65123	AV32F703	Large 30" +

Figure 4.2 - Sort item list table

Navigating item list

For large item lists, use the navigation links at the bottom of the item list table. Select a page to view or skip to the middle or last page of the item list.

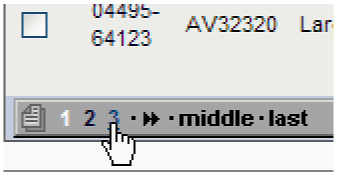


Figure 4.3 - Navigate item list

Item list buttons

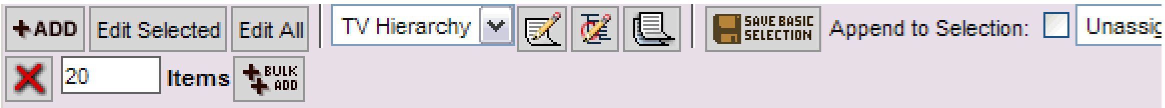


Figure 4.4 - Item list toolbar buttons

The following Item list buttons are described counter clockwise starting from the top-left corner.

Add	Enter a number of items to add (default is 1) then click the ADD button. In the Multiple Edit screen, the new items are added to the top of the list.
Edit Selected	Edit Selected Items
Edit All	Edit all items on item list
Hierarchy drop-down field	Browse or select Hierarchy
Edit hierarchy	Select a hierarchy from the drop-down filed and click this edit button to edit the hierarchy
Re-categorize items into different hierarchy	Select an item or group of items to re-categorize first then select a hierarchy from the drop-down selection. Click the re-categorize button to display a new window that allows the user to select a different

	hierarchy. The change is not committed until the Save button is clicked.
Copy item to a different category	Select an item or group of items to copy first then select a hierarchy from the drop-down selection. Click the copy button to display a new window that allows the user to select a different hierarchy. The change is not committed until the Save button is clicked
Save as Basic Selection	Click this button to save the current Item List as a basic selection in the Selection Console
Append Selection	Append the current item list to a saved selection. Check the "Append Selection" box and select a saved selection from the drop-down field box to add the current item list to the selection
Selection drop-down field	Select a saved selection to append an item list to
Delete	Deletes all an item. Be cautious when using this button. Once an item is deleted the action cannot be undone. Select an item or group of items, and then click the delete icon.
Number of items field and the Bulk Add button	Enter a number of items to add and click Bulk Add. The defined number of items to add appears in the Multiple Edit screen for data entry.

Single Edit screen

The Single Edit screen allows a user to edit a single item. This screen is displayed when a single item is selected for viewing or editing. For "Advanced Content Authoring", this screen is the Single Edit tab.

Single Edit screen buttons

There are a series of buttons, navigation tools, a status icon at the top and bottom of the Single Item screen.

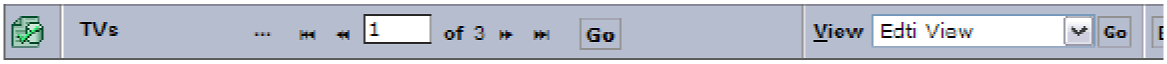


Figure 4.6 - Single Edit screen buttons (top screen)

The following Single Edit screen features are described left to right.

Left icon	Provides the status of the item. For example, if the item is synched with the database, or if an error has occurred
Catalog Name	The catalog name is displayed in this top bar
Navigation buttons and Go	Allows a user to navigate between pages of items. Type a page to go to and click Go.
View drop-down field and Go	To change a view for the item, select a view and click Go.
Bring to top	Allows a user to bring to the top of the order of items based on one of the following status: • Failed • Modified • New

	• Non-modified • Processing • Selected For example, if "Failed" is selected, all of the failed items are pushed to the beginning of the item list.

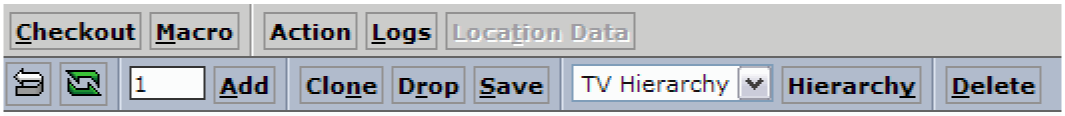


Figure 4.5 - Single Edit screen buttons (bottom screen)

The following Single Edit screen buttons are described counter clockwise starting from the top-left corner.

Checkout	Check out an item into a workflow
Macro	Run a macro against the item
Action	Preview item in a CSV, HTML, or Tab delimited format
Logs	View user defined logs for the item
Location Data	This button is disabled unless location data has been configured for the item
Undo Changes	Undo Change
Refresh	Refresh the screen information.
Add	Add an item
Clone	Select an item to clone and Click Clone. The item information is cloned into a new item with the primary key left blank

Drop	Drop an item from the catalog
Save	Save an item
Hierarchy drop-down field	By default, this field shows the current hierarchy for the item. To switch the hierarchy used, select a value from the drop-down field
Hierarchy	Add or replace category mappings
Delete	Delete item

Multiple Edit screen

The Multiple Edit screen allows a user to edit multiple items. This screen is displayed when a multiple items are selected for viewing or editing. This screen is only available through the "Advanced Content Authoring" screens in the Multiple Edit tab.

The Multiple Edit screens has many of the same features as the Single Edit screen except for the following screen feature unique to the Multiple Edit screen (see image).

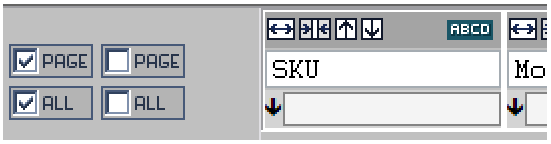


Figure 4.7 - Added feature in the Multiple Edit screen

Select Page	Select all items on this page only
Unselect Page	Unselect all items on this page only
Select All	Select all items in this selection/list
Unselect All	Unselect all items in this selection/list

Resize column widths	At the top of each column, there is a set of four resize options. This is used to Increase/reduce the size of the column or sort the columns ascending/descending
Attribute type indicator	At the top of each column is an image that indicates the attribute type
Down arrow next to enter value field	Enter a value for an attribute in the top row, check the items that are to take this new value, click the down arrow to set the new value for all checked entries

Item management

To edit content using either the Standalone or Advanced Content Authoring screens, simply change the information in any field and click Save to commit the changes. Errors will occur if the user does not have authorization to make edits in the specific fields or if view only fields have been applied.

Edit Items

Editing a single item

1. Select an item from the Left Pane or from an Item List. The item appears in a single edit screen.

2. Update the item information and click **Save**. If the item is not saved and the user attempts to leave the screen, a confirmation dialog box appears. Click "OK" to not save the item and to leave the edit screen.

Performing bulk edits

Editing multiple items can be done using the Multiple Edit Screen, which can be accessed through the Advanced Content Authoring screens. A single change can be made to multiple items, with the exception of the unique Primary Key.

If a field is grayed out and no changes can be made, the item has not yet been saved to the database, thus the information has not been updated. Once the information is saved to the database, it is made available for editing.

1. From an item list, select a group of items to edit and click **Edit Selected**. The group of selected items appears in a screen for edit.

2. The top row in the Item Edit screen is used to perform a single change through out multiple item with the same attribute. Enter a value in the first row of the item attribute and click the **ALL** icon. The item attribute is updated with the new value.

3. Click **Save** to save all edits.

Add Items

Users have the choice to add an item one at a time or in bulk. Similar to Bulk Edits, users can add multiple items. Adding multiple item are performed using the Multiple Edit screen. This allows the user to update a single attribute for multiple items if the value is the same.

Adding a single item

1. Use one of the following methods to add an item:

- From the Left Pane, right click on a hierarchy node and select **Add Item**
- From and Item List, click **+ADD**

2. Items are added through the Single Edit screen and are not committed to the system until clicking on **Save** before leaving the screen.

3. If a required field is not completed, an incorrect value was entered, or any other errors are performed, errors will display and the item is not saved.

Performing a bulk add

1. Multiple items can be added through the item list screen. Enter the number of items to add (defaulted to 20) and click **+BULK ADD**; the Multiple Edit screen is displayed.

2. The top row in the data entry screen is used to populate a single value to all item attributes. Enter a value in the first row of the item attribute and click the **ALL** icon or enter items one row at a time.

3. Once all items have been entered, click **Save** to add them to the catalog.

Copy Items

Items can be copied from one category to another category in the same catalog in the Left Pane. This does not move the item but creates an exact copy of the item to another category.

Items cannot be copied to the Unassigned folder.

Copy a single item

Note: If an item is copied to multiple categories and then the item is deleted from one of the categories, ALL items are deleted from all categories. Use the Remove selection from the short menu to remove an item from a single category.

1. Right-click on the item to copy and click **Copy**.

2. Right-click the category where the item is to be copied and click Paste.

Clone items

Cloning an item is different than copying an item as all the item attributes remain the same except for the Primary Key, which is left blank. Thus, a unique value must be entered for the primary key attribute.

Cloning a single item

Cloning a single item can only be done from the Single Edit screen.

1. From the Single Edit screen, click **Clone**.

2. The item attributes are cloned, except for the Primary Key attribute, which is left empty.

3. Enter a unique value for the primary key and click **Save**.

Cloning a bulk of items

Bulk cloning items can only be performed from the Multiple Edit screen.

1. From the Multiple Edit screen, select the items to clone and click **Clone**.

2. All items are cloned except for the primary key attribute.

3. Enter a unique value for the primary key attribute and edit any information. Click **Save** to commit changes.

Delete Items

Items can be deleted from the left pane or from the an item list. To delete multiple items, use the item list.

Deleting a single item

Note: If an item is deleted using the Delete selection from the short menu, all instances of the item in the catalog will be deleted. Use the Remove selection to remove an item from a category, not all categories in the catalog.

1. Use one of the following methods to delete a single item:

- Right click on the item in the left pane and click **Delete**
- Select an item from the item list and click **Delete**

2. When deleting items, a confirmation dialog box appears. Click **OK** to confirm the action.

Delete multiple items

Perform the deletion of multiple items in an Item List screen or Multiple Edit screen.

1. From the list of items, select the items to delete and click **Delete**.

2. A delete confirmation dialog appears, click **OK**.

Categorizing items

Items that are Unassigned can be categorized into their associated categories using the following steps:

1. From the Left Pane, open a catalog and view all items under "Unassigned Items".

2. To categorize a single item, click on an item and from the single edit screen, click the Hierarchy button, select a category, and click Save. The item is moved to the new category.

3. To categorize multiple items, select multiple items from the Left Pane and the items appear in an Item List. Select all items to categorize and lick the categorize button. Select a category and click **Save**. All the items are moved to the category.

Remove item from category

An item can be removed from a catalog using the short menu in the left pane.

1. Right-click on an item and select Remove.

2. A confirmation dialog appears confirming the item removal. Click **OK**.

3. The item is moved to the "Unassigned" folder unless the item has been assigned to multiple categories.

Recategorize Items

After an item is categorized, it can be changed to a different category from the Left Pane or from the Item List.

Recategorize items in the Left Pane

From the Left Pane, items can be moved from one category to another, but cannot be moved to the "Unassigned" folder. An item can only be moved to another category from the same catalog.

1. Right-click on the item to move and select **Cut**.

2. Right-click on the category to move to and click **Paste**.

3. The item is moved into the new category.

Recategorize from the Item List

1. From the Item List screen, select a single or multiple items and click on the categorize button.

2. All categories for the catalog appear in a separate window. Select a category, click **Save**, and items are moved to the new category.

Uncategorized Items

Uncategorized from the Left Pane

1. From the Left Pane, select an item, right click, and select "Remove" from the short menu.

2. The item is removed from the category and moved into the "Unassigned " folder, unless the item is a part of another category.

Uncategorized from the Item List

1. Select an item from the Item List and click the Categorize button.

2. From the popup window, DO NOT select any category and click Save.

3. The item is moved to the "Unassigned" folder.

Errors Saving Data

If an incorrect value is entered for a catalog attribute, an error dialog box appears describing the type of error where it has occurred. Click **OK**, re-enter a correct value type, and click **Save**.

Item Selections

Selections are created to return a list of items that can be used for the following purposes:

- Viewing and editing items
- Export a defined group of items to a specific destination

Users have the ability to create a basic item selection. A Basic Selection is a list of items where the primary key of the items is the criteria remembered by the selection.

The Item Selection Console is used to perform the following tasks:

- Create selections
- Delete selections
- Edit the selection criteria
- View the items in the selection

Item Selection console buttons

🔍	previews a selection
✎	edits a selection
✗	deletes a selection

Basic Selection

A Basic Selection is configured with the following objects:

- Catalog
- Catalog version
- Access Control Group
- Hierarchy

Note: Limit a selection to 5000 items or less. Larger selections may hang the application, causing a user to exit and log back into the application.

Create a new basic selection

Use the path **Product Manager > Selections > Selections Console**. The Item Selection Console appears. Click on the **New Basic Selection** and complete each step of the "Catalog Selection Editor" wizard.

1. Select Catalog

2. Select Version

3. Select Access Control Group

4. Select Hierarchy

5. Enter a name for the item selection in the Enter the Selection Name field.

7. Select a whole hierarchy, a specific hierarchy, or unassigned items.

8. To recursively select all sub-hierarchies, select a hierchy from the list and then at the bottom of the screen, click "On" and the **Go** button. To deselect all sub-hierarhcies, click "Off" and the Go button.

9. When the selection has been configured, click the **Save** button. The newly created selection is listed in the Selection Console.

Previewing selections

To preview a selection, click the Preview icon from the Item Selection Console. The results are displayed in an item list table format.

Hierarchies

Hierarchies are stored separately from catalogs within WebSphere Product Center. This allows a user to view and eventually export a catalog using a hierarchy of their choice. The Hierarchy Console allows the user to modify all of the hierarchies stored within WebSphere Product Center. The user also has the option of creating a new hierarchy from this interface.

Before a hierarchy can be created a primary spec must be created for the hierarchy, which is done from the Specs Console. The primary spec that is attached to the hierarchy should contain the following minimum attributes:

- Display Name
- Path - used by scripts to locate the category
- Primary Key - unique identifier for the category (as determined during spec creation)

There are two types of hierarchies that are used to organize information.

- Category Hierarchy - Used to categorize items for a catalog
- Organization Hierarchy - Used to create an organization of users

Creating a hierarchy

Used the following instructions to create either a category hierarchy or an organization hierarchy:

From **Product Manager > Hierarchies > New Hierarchy**, the Create Hierarchy wizard appears. Complete each step of the wizard.

1. Hierarchy Name: - enter a name for the hierarchy

2. Select Primary Specification: select a primary spec to use with the hierarchy

3. Select Display Attribute: select a display attribute from the drop-down selection

4. Select Path Attribute: select a path attribute from the drop-down selection

5. Select Hierarchy Type: select either "Category Hierarchy" or "Organization Hierarchy"

6. Select Access Control Group: select an ACG to use for the hierarchy

7. Once the hierarchy has been created, add it to the Left Pane and add sub-hierarchies (categories or users) as needed. Category hierarchies are listed in the left-pan drop down under "Hierarchies", and organization hierarchies are listed under "Organizations".

Managing Hierarchy Nodes

1. To edit a hierarchy, use the Left Pane to add, move, copy, delete, or view hierarchy attributes. Add a hierarchy from the Left Pane drop-down menu.

Note: Default hierarchies cannot be edited.

2. Click on a hierarchy node and view the hierarchy attributes.

3. To modify the hierarchies, right-click on a category and select an action from the short menu. If adding a category, make sure to right-click on the parent node where the new category will be added.

The short menu provides the user with five different functions to manipulate the hierarchy structure.

Add Category	Used to add a new category
Cut	Used to move an existing category to a new location within the hierarchy

Copy	Used to copy an existing category to a new location within the hierarchy without creating a new category
Paste	Move a category to a new location within the hierarchy after selecting Cut or Copy
Remove	Removed the category from use but does not delete it from the hierarchy.
Delete	Used to delete an existing category
Refresh	Used to refresh the hierarchy display

Hierarchy Rollback

Like catalogs, hierarchies have versions that can be rolled back.

- Select a hierarchy from the Hierarchy Console and click the **Rollback** button. Then select a version to rollback.

Hierarchy Delete

- Select a hierarchy from the Hierarchy Console and click the **Delete** button. An error will occur if the hierarchy is being used by another object (catalog, organization)

Hierarchy Mappings

Categories can be mapped between two hierarchies, which provide the ability to map an in-house categorization to a standard categorization without having to worry about individual item mappings.

For example, an internal tree called "Product Category" can be mapped to the standards of a "UNSPSC" tree.

When an item is created, the item can be mapped to a category in one tree. If the category is mapped to other trees, the item inherits the additional mappings.

Create hierarchy map

1. Use the menu path **Product Manager > Hierarchies > Hierarchy**

Map > Mapping Hierarchies, the category map wizard appears.

2. Select a hierarchy to map "from" and a hierarchy to map "to".

3. A list of categories appears. Click the double arrow icon next to the category to map.

4. A separate window appears with the second hierarchy. Find the category to map (multiple categories can be selected) and click **Save**.

Note: A green checkmark indicates where a mapping exists.

5. Continue steps 3-4 until all categories have been mapped.

Adding a hierarchy mapping

1. Use the menu path **Product Manager > Hierarhcies > Hierarchy Maps > Hierarchy Mapping Console**, click the edit tool button next to the row that shows the source and destination hierarchies.

2. Browse the tree to find the category that is to be mapped.

3. Click on the double arrow icon next to the category to map. A screen pops up with the second hierarchy.

4. Find the categories to map to, select it and click **Save**.

Delete hierarchy mappings

- Click the delete button next to the name of the hierarchy in the Hierarchy Mapping Console
- Edit the hierarchy mapping, click on a mapped category to view the list of mapped categories. Click the delete button to delete a single mapping.

Hierarchy name search

The Category search feature is implemented in various locations within WebSphere Product Center. This section discusses performing a search for category nodes in the Left Pane.

Features

- Criteria incorporates 'like' search
- Returns the paths to all matching nodes

1. Add a hierarchy to the Left Pane.

2. Click the **Show On-Demand Search** in the left pane to display the search criteria.

3. Define a criterion for the search, enter a string and select to search on the Display Name or Name. Click the green arrow button. The results are displayed in the Left Pane.

Note: Users can use a wildcard " * " when entering a string search. If a wildcard is not added as part of the search string, the system will look for an EXACT match.

Reports

WebSphere Product Center’s analysis module leverages the application’s powerful scripting engine to enable users to create and run custom reports. The report scripts are used to define how the information is ordered and formatted. Users can utilize these sophisticated reporting tools to analyze online sales activity along multiple dimensions: by customer, product/SKU, region, destination, etc.

A report structure for analyzing and reporting progress and results can easily be created once and then many instances of it can be run against different catalogs. The results might be delivered to a location of choice: by email, ftp, post, or XML connect. Copies of the outgoing reports are of course always stored in Document Store as well.

A user might analyze how catalogs are being processed, for example, by creating a report that summarizes all categorized items in all published versions of a particular catalog.

Report Console

The Report Console is the command center for creating, viewing, and editing reports, as well as scheduling them for delivery to user specified destinations. The console table lists the existing instances of reports in alphabetical order.

Name	User specified name for the instance of the report. A single report type might have multiple instances if it includes input parameters.
Type	User-defined script that generates the report is what defines its type. The user names the type during the report creation flow and then defines the report in the WebSphere Product Center script editor. All instances of this report type will now use the defined script as the template for generation.
Schedule	The Schedule icon takes the user to the Schedule Status Information screen where details on the report generated by clicking "Go" may be viewed in summary.
Action	The "Go" button schedules the report instances to be generated and distributed. The "preview" link simply generates the report in a new window for immediate preview.
Delivery Location	The delivery location tells the user what the chosen mode of distribution for the report, during the creation of the report instance, and links to the distribution flow where the details of that distribution

	may be modified.

Creating a report

Use the menu path: **Product Manager > Reports > Reports Console**. The Reports Console appears. Click the **NEW** icon and the Create/Edit Report wizard appears. Complete each step of the wizard.

1. Select Report Type: Choose a report type from the drop-down menu and click **Select**. In this step it is possible to create or edit a report type.

2. Report Name: Enter a name for the report and click **Next**.

3. Select Distribution: Choose a distribution type from the drop-down menu and click **Select**. In this step it is possible to create or edit a distribution type.

4. A message box indicates that the report was successfully created. Return to the Reports Console to view the newly created report.

Preview Report

From the Report Console, click the preview button under the Action column to generate an on-screen display of the report. This enables the user or administrator who creates it to view the report for accuracy before scheduling its distribution.

1. Find the report to preview from the Reports Console.

2. From the Action column, click the preview button.

3. A new window appears with the report details. If desired, send the report to a printer.

Generate Report

The "Go" button performs the action of generating and delivering the report. To manually generate a report, use the following steps:

1. Find the report to generate from the Reports Console.

2. From the Action column, click the **Go** icon to generate and distribute the report to its defined destination mode.

View Report Status

To view the status information of a report, do the following:

1. Find the report from the Reports Console.

2. From the Schedule column click the status button. Status information is displayed in the Schedule Status Information interface.

3. Click the **Job Details** button for more information regarding the generated report.

4. Click the **Refresh** button, to update the status information.

Delete Report

To delete a report, use the following steps:

1. From the Reports Console, find the report that is to be deleted.

2. From left side of the report name, click the delete button.

3. A dialog box appears confirming the deletion of the report. Click **OK** to delete the report or **Cancel** to quit the delete operation.

Lookup Tables

Lookup tables are provided to enhance the content management functionality available in WebSphere Product Center. They are used to perform search and replace functions within a catalog, and can also be used to validate data contained in specific catalog fields. Create standard tables, i.e. units of measure (UOM), currencies, or countries. Create custom replacements tables, i.e. BK = Black and BL = Blue.

The creation and management of lookup table records is very similar to standard item creation and management. The set of tools available to manage lookup tables, like available for catalogs, include Bulk operations (add,edit) and the Search tool.

Creating Lookup Tables

To manually create a lookup table:

- First, create a lookup table spec
- **Use the menu path Product Manager > Lookup Table > Lookup Tables**, the Lookup Table console appears.
- **Click** New, the Create Lookup Table wizard appears. complete each step of the wizard.

1. Select Type: Select "Single String Key".

2. Select spec: Choose a lookup table spec to use. In this step, it is possible to create or edit a lookup table spec.

3. Lookup Table: Enter a name for the lookup table.

4. From the Lookup Table console, click the preview button to view the newly created lookup table. Since it is a new table, no items are found.

5. Click the add or bulk add buttons to enter values for the lookup table.

Deleting items from lookup table

1. From the lookup table, select the items to delete by clicking on the box next to the item name.

2. Click delete button and the item is removed from the lookup table.

Search lookup tables

1. From the Lookup Table Console, select a lookup table and click SEARCH. The Search screen appears.

2. Enter a criterion in the Search Criteria table and click SEARCH. The results are displayed in the same screen.

Ch 5 Advanced Topics for Catalogs

This chapter covers advanced topics for creating catalogs and managing content within catalogs.

Inheritance

The development of inheritance functionality within WebSphere Product Center's product reduces the need to duplicate item information for similar items. Thus, a single item is able to obtain an attribute value from a separate category or from another item. Another advantage of implementing inheritance within a catalog is that the integrity of information is increased. There is no need to create similar attribute value throughout multiple catalogs, thus causing an opportunity for user error.

The concept of inheritance within WebSphere Product Center provides a natural classification for attributes and allows for the commonality of attributes to be explicitly taken advantage of in modeling and constructing product information. A catalog can be structure so that a single value can be shared throughout multiple catalogs or all instances of an attribute value can be pulled into one catalog.

Inheritance is a relationship between categories, items, and items to categories. One object is the parent (base/super class/ancestor/etc.) class of another. The use of inheritance provides an extension to product information, as opposed to reinvention. Although the natural state of inheritance allows an object to have the same behavior as another object, the ability to extend or tailor that behavior to provide special action for specific needs is provided through override controls.

Use the following application as an example. Both categories Televisions and Audio Components have similar attributes such as managing a name, a type, and a description. Rather than define the attributes in both of these categories separately, the shared attributes are placed under a new parent category called Home Audio and Video. Both Audio

Components and Televisions become a subcategory of the Home Audio and Video category, and both inherit similar Home Audio and Video category attributes.

Both Audio Components and Televisions categories can add additional attributes that are unique to them. For example, Audio components can have information related to CD players or Receivers. On the other hand, the Televisions category might want to keep track of whether the television is Digital, Projection, Combo, or Handheld.

Managing product information using inheritance provides an extension of information that would otherwise be cumbersome to create. Here is another example. A catalog has a parent category "Pentium 4 - 2GHz" and the child categories are organized by model number. Using inheritance, all items in each "Model" category use the following values:

Complete sys - Pentium® 4 2.0GHz CD COA Win98, 2000Mhz, ATX tower/400w, 1 yr warranty, (upgradeable)

To change this value for a particular model category, the governing inheritance rule can be overridden and the description can be change to read "Win2K" instead of "Win98".

Complete sys - Pentium® 4 2.0GHz CD COA Win2K, 2000Mhz, ATX tower/400w, 1 yr warranty, (upgradeable)

Defining Inheritance Rules

Inheritance rules are defined to direct which path an attribute uses to obtain its value. Once the path of ordered targets is defined, the inheritance rule must also specify whether to accumulate all data found in the path or to choose the data with the highest precedence, which by default is the first value found.

When defining targets, they must either be a Hierarchy or a Catalog, which has data for the given Attribute. The choice of target determines the type of inheritance relationship being applied. For example, if an item obtains its value from a target category, the Item to Category inheritance type is applied.

If a change is made in the Inheritance Rule of the target, all objects referring to it will automatically reflect the change. Thus, the inherited values will reflect the change in the rule itself. This dynamic inheritance allows the ability to change parent category attributes, which automatically updates all child items or categories.

The following example is an example of an Inheritance rule. Three targets have been defined and the use of dynamic inheritance is applied. If a rule change is made to the first target, all objects referring to it will append those changes.

Sub-specs

Sub-specs are used to support inheritance. Created in the spec console, Sub-specs are only used when assigned to a master spec (catalog specs, hierarchy specs and secondary specs). Sub-spec attributes are different from other specs in that they are not unique, do not have a link, and can only be made a primary key when they are used by a catalog spec.

Inheritance concepts

This section describes the different concepts related to inheritance. It is important to understand these concepts as they are applied to the implementation of inheritance within WebSphere Product Center.

Sharable Attributes

Attributes must be shared so that the same attribute can exist in different catalogs and hierarchies. The shared attribute is a requirement for inheritance, as a source attribute has to know which target attribute in a separate catalog or hierarchy to inherit values from.

Sharable attributes allow the ancestral inheritance from category attributes to item attributes and like node inheritance between item attributes in separate catalogs.

In order for an attribute to be shared, a Sub Spec is created to define all shared attributes, which are then and appended to a non-shared spec. When creating a new Catalog Spec, the option to add shared or non-shared nodes is available.

Like Node Inheritance

The concept of "Like Node Inheritance" is the dynamic inheritance of an attribute value from the same type attribute of an item with a like inheritance key (see next section) in a separate catalog. The like node works as an inheritance key for like node inheritance jumps. The receipt of inherited values for like attributes is based on a key attribute value of the item in another catalog.

Like Node Inheritance Key

A requirement for implementing inheritance between objects is the like node inheritance key. Two catalogs contain "like" nodes that is used to identify "like" items in different Catalogs. There must be a like node that is identified as the inheritance key in both hierarchies. This allows a category to jump between catalogs with like node inheritance.

Ancestral Inheritance

Ancestral Inheritance is the dynamic inheritance of an attribute value from the same attribute of a category node with which an item is associated. Category nodes inherit an attribute value from the same attribute in the parent category nodes.

Inheritance Path Definition

An Inheritance Rule defines where catalogs and hierarchies look for inheritable values; this target is defined by specifying the Inheritance Path. If no values are found, is null, then no information is used.

Extend Inheritance Path Based Upon Item Attribute Values

Individual items may have different inheritance paths; therefore the path to be followed is based upon an attribute value of an item.

Multiple Inheritances

Multiple Inheritances occur when an item or category inherits from more than one item or category. For example, a CD_ROM can inherit values from the "Internal components" category as well as "External components", or a 20 GB hard drive from the categories "EIDE" or "SCSI".

Dynamic Inheritance

Dynamic inheritance allows objects to change and evolve over time. If a new category is added under a parent category structure, the path of inheritance can change as well. Since a parent category would hold attributes for child categories, changing a parent attribute changes the attributes of the child categories. More specifically, dynamic inheritance refers to the ability to add, delete, or change parent category attributes which automatically/dynamically updates all child items or categories.

Overriding Inheritance

The previous sections detail how inheritance can be used to communalize a group of information, which can be updated from a single source. In the event that an attribute does not follow the inheritance rule that governs it, the ability to override inheritance is allowed.

Inheritance provides the ability to inherit values based on a user-defined hierarchy, referred to as "inheritance rules". The data is not only being passed through; it might need to be overwritten as well. On every object that inherits values, it is be possible to tell where the data is coming from, whether it is being inherited, or if it has been overwritten.

For example, the Televisions category has an attribute called "Description" which uses the following value:

> More to adore
> From personal TVs for a bedroom or kitchen to
> bay-window-size HDTV-ready showpieces for your
> state-of-the-art home theater; it is all here. If you really love

> TV, you've come to the right place.

Digital TVs is a subcategory of Televisions and uses the above value. The inheritance rule that governs Digital TVs can be overridden and have the following value assigned:

> Get the ultimate experience from one of our next-generation digital sets
>
> - Enjoy the biggest change to TVs since color
> - The lifelike picture will amaze your eyes
> - Digital sound will astound your ears

The following characteristics apply to overriding inherited values:

- If a value is edited upstream from attributes that inherit the value, the inherited values appear updated instantaneously
- The overwritten value is not be modified if an upstream value is modified
- If inherited data exists, users can view where the data is inherited from (catalog name, or hierarchy name) using the graphical display of the inheritance path.
- If there is overwritten data and inherited data, users have the ability to look at the inherited data and see where it is coming from
- If there is only inherited data, users can choose to override the content and enter new content
- If there is overwritten data, users have the ability to see a list of items/category that MAY inherit this data and DO override this data
- It is possible to tell when a value has been inherited or not
- It is possible to tell when a value could be inherited even if there is currently no data to be inherited
- It is possible to tell where a value could be inherited from
- When attributes are grouped, the inheritance rule is defined for the group

Precedence order:

- Overwritten value/provided value (if a value is provided, this is what will be saved)

- Default value/Sequence (if no value is provided, if the sub-spec is setup to have a default value or a sequence, this is what will appear)
- Inherited value (if no value is provided, and there is no default value or sequence for an attribute, the inherited attribute will appear)

Accumulate Inherited Values

This concept is used when an attribute retrieving a single inherited value is not sufficient. Therefore, it can be defined that an attribute follows the entire target path defined by an Inheritance rule and accumulates all values for display.

From the parent category	More to adore From personal TVs for a bedroom or kitchen to bay-window-size HDTV-ready showpieces for your state-of-the-art home theater; it is all here. If you really love TV, you've come to the right place.
From the child category	Get the ultimate experience from one of our next-generation digital sets • Enjoy the biggest change to TVs since color • ·13 The lifelike picture will amaze your eyes • ·14 Digital sound will astound your ears

Multi-occurrence inheritance

If a group of attributes has a multi-occurrence value greater than one, the option to accumulate all the values down the inheritance path, add occurrences, and override the inherited values is turned on by default.

Important limitations: Inheritance for multi-occurrence will only support accumulated inheritance for group nodes; it will not support non-accumulated multi occurrence, and it will not support inheritance of multi occurrence single attributes, or multi-occurrence within a group.

The following are characteristics for multi-occurrence inheritance:

- Default option is set to accumulate all the data found in a path, displaying all of the values found in the paths above
- Users will be able to override any value in the group; once a value is overridden it is possible to see the value that would be inherited or revert to it. This is done through the GUI, the same way it is done for single occurrence attributes
- Multi-occurrence inheritance remembers the occurrence it is inheriting from as if each occurrence had an internal counter (i.e. it is not the position or the value that is being overwritten but the specific occurrence)
- The specified maximum occurrence is enforced and will count the total of occurrences, inherited and native
- Overriding a multi-occurrence attribute is done through the same screen as overriding single occurrence attribute
- Nodes within multi-occurrence group can be localized
- It is possible to have group multi-occurrence inheritance with groups having up to 3 levels of sub-attributes in addition to localization
- A given item may have up to 50 grouped attributes with multi-occurrence inheritance and up to 15 occurrences
- If an attribute is changed from multi-occurrence to single occurrence and the accumulate option was turned on, the accumulate option is turned off
- If a record exceeds the maximum number of occurrence because of the inherited occurrences, this does not prevent saving the record, however it should prevent adding more occurrences for this record

Workflow support for inheritance

Items and categories can be updated as part of a workflow. The source of inheritance and inherited attributes need to be included in workflow.

The following are characteristics of workflow support for inheritance:

- If an attribute is updated in a workflow, the target values are inherited from the attribute
- The new value is visible once the attribute is checked back in
- If an attribute that inherits its value as part of a workflow, this attribute will see the checked in value of the source. Any changes to the checked in source appears on the inherited attribute even if the source attribute is in the same workflow and has been modified
- Data entry screens have the same inheritance icons and functions as they do in the main catalogs/hierarchies

Inheritance of localized attributes

Inheritance supports localized attributes. Localized attributes are to be considered as individual attributes. The following are characteristics for inheritance of localized attributes:

- It is possible to inherit localized attributes
- It is possible to override inherited localized attributes
- It is possible to define inheritance rules for the original attribute of the localized node

Importing and Exporting Inheritance rules

To facilitate updating or recreating an environment, users may import and export inheritance rules.

- Import new inheritance rules
- Update inheritance rules using import function
- Export existing inheritance rules

Inheritance Relationship Types

Before discussing the different types of inheritance relationships, this section will review the relationships between WebSphere Product Center objects, which are illustrated through the following diagram.

Note: This section uses the term Category Tree, the application refers to this as Hierarchies.

1. The hierarchy is a hierarchical arrangement of categories used to organize and classify items within catalogs.

2. A Catalog is used as a repository / container for Items, which is the most basic content object in WebSphere Product Center.

3. A Hierarchy is associated to a catalog and a single hierarchy can be associated with multiple catalogs, thus catalogs may be associated with multiple hierarchies.

4. Items can be mapped to any number of categories within any number of categorization hierarchies.

Inheritance is implemented within WebSphere Product Center using three types of relationships that expand a product’s information, which is shared with other related information types.

- Category to Category
- Item to Category
- Item to Item

Category to Category

A category-to-category relationship provides ancestral inheritance between categories in a hierarchy. With this relationship, a child category can obtain a value from the parent category, thus it is not necessary to provide the same value for all similar category attributes.

In the diagram below, a Child Category inherits its value from the Parent Category and the Grand Child Category inherits its value from either the Child or Parent Categories, as defined by the user-defined inheritance rule.

Values

The values for a category-to-category relationship are defined at the parent category level and are inherited by the child categories. There can be multiple child categories within the category hierarchy which all obtains its value from the parent category.

Defining Inheritance Rules

There is no need to identify an Inheritance Key as the category hierarchy is used to determine the data path. Inheritance rules are defined at the Hierarchy Attribute level (category spec) and must specify the target category. If desired, multiple target categories can be selected.

- Inheritance rules are defined at the Hierarchy Attribute Level
- Values defined at ancestral category level are inherited by child categories
- No Inheritance Key is needed since Hierarchy is used to determine data path

Characteristics

When creating a tree, there is an option to make inheritance available or not; this option must be selected to enable category ancestral inheritance

Inheritance can be setup for attributes that are in the Hierarchy Spec or in a Standalone Spec

To setup inheritance for attributes used only in Standalone Specs, the attribute must be assigned to at least one category

Inheritance works with multiple levels of category hierarchy

Item to Category

The values for an Item-to-Category relationship are defined at the Category level, which are inherited by items associated with the category. Therefore, there can be multiple items that obtain their values from the associated or assigned to a Category.

Values

The values for an Item-to-Category relationship are defined at the Category level, which are inherited by items associated with the category. Therefore, there can be multiple items that obtain their values from the associated or assigned to a Category.

Defining Inheritance Rules

Inheritance rules for an item are defined at the attribute level per catalog. The flow of inheritance traverses up the categorization hierarchy as defined by the hierarchy inheritance rule.

Characteristics

- Values defined at the category level are inherited by items associated with the category (immediate parent)
- Inheritance Rules are defined at the Attribute Level per Catalog
- Values are inherited directly from the Category to which an Item is assigned
- Traversal up the Categorization Hierarchy is defined by Hierarchy Inheritance Rules

In order to setup item ancestral inheritance, when creating a catalog, the following steps must be performed:

- Select a primary tree (or a secondary tree) where inheritance is available
- Specify that inheritance is turned on for the given catalog

Inheritance works with a combination of at least 10 levels of category hierarchy and 10 levels of catalog hierarchy

Changing category mappings updates the displayed inherited values

Item to Item

Two items in different catalogs can be related to each other via their primary key and can inherit data from each other. The value of one item is defined by "like" items in other catalogs. Items are unique to a catalog but can appear in multiple catalogs. One catalog might hold different values than another, but with an item-to-item relationship, the values can be inherited.

Value

The value of one item is defined by "like" items in other catalogs. Items are unique to a catalog but can appear in multiple catalogs. One catalog might hold different values than another, but with an item-to-item relationship, the values can be inherited.

Defining Inheritance Rules

Inheritance rules are defined at the attribute level (catalog spec) for the catalog that specifies the target items. In order to for an item to inherit values from another item, like nodes must be defined by specifying an "Inheritance Key" at the catalog level.

Characteristics

- Values defined by "like" items in other catalogs.
- Inheritance Rules are defined at the Attribute Level per Catalog
- The primary key must come from the same sub-spec
- When creating a catalog, to enable inheritance across items, inheritance must be turned on, which has to come from a sub-spec attached to the catalog spec
- Inheritance works with multiple levels of catalog hierarchy

Implementing Inheritance

Characteristics

To understand how inheritance is implemented within WebSphere Product Center, the following characteristics must be understood:

- Inheritance Rules are specified at an attribute level
- All items in a Catalog will follow the Inheritance Rule specified by the Catalog
- Items will have to define an Inheritance Key which will be used to identify "like" items in different Catalogs
- All Categories in a Hierarchy will follow Inheritance Rules specified by the Hierarchy
- Inheritance is always defined on Shared Nodes within a Spec
- Inheritance rules can differ between attributes – I.e. two attributes of an item may follow different inheritance paths

Inheritance Scenarios

The following examples provide scenarios of when inheritance can be beneficial.

Example 1

Multiple Inheritances occur when an item or category inherits from more than one item or category. For example, a CD_ROM can inherit values from the "Internal components" category as well as "External components", or a 20 GB hard drive from the categories "EIDE" or "SCSI".

Example 2

A parent catalog "Pentium 4 2Ghz" holds the following attributes:

- Category
- UNSPSC Code
- Product Name
- Product Type
- Description

The child category "Model" holds the following attributes.

- Image

- Detailed Description

The inheritance rule is defined at the parent hierarchy level, specifying which sub spec is to be shared, thus allowing the child categories to inherit the values of the parent category.

Applying the category-to-category inheritance relationship, the new structure would include all the categories from both trees. Thus, the values defined in the parent category are inherited by the child category.

- Category
- UNSPSC Code
- Product Name
- Product Type
- Description
- Image
- Detailed Description

Sub Specs – Inheritable Attributes

Sub Specs create an advanced method of maintaining data integrity. Nodes of a single sub spec can be shared in multiple catalogs and, which allows the ability to create a single defined attribute that is consistent throughout all associated catalog.

Creating inheritance within catalogs can only be accomplished by using Sub Specs. Inheritance is implemented using Sub Specs, which create a shared attribute set. Only Attributes in a Sub Spec can be used for inheritance. Sub specs do not have a primary key that is needed, but at least one attribute must be used

Creating Sub-Specs

The following characteristics apply when working with sub-specs:

- No primary key required (attribute can be designated a primary key after inclusion in a catalog spec)

- Create node hierarchies within sub-specs
- In the details for the parent node of the sub-spec, all specs that make use of the sub-spec are displayed in the "Associated Specs" section. When the nodes are removed from the master spec, they are also removed from the Associated Specs list.
- Maximum Length, Maximum Occurrence, Minimum Occurrence can be defined
- All field types are available (Binary, Currency, Date, Flag, Image, mage URL, Integer, Lookup table, Number, Number enumeration, Password, Period, Relationship, Sequence, String, String enumeration, Thumbnail image, Thumbnail Image URL, URL)
- All additional parameters are available (Default value, Default value for sequence start, Delimiter for category path, Help URL, Increment sequence by, Lookup Table, Minimum Length, Number enumeration, Pattern (regular expression), String enumeration, String enumeration rule, Validation rule, Value rule)
- Attributes can be made Editable, Hidden or Non- Persistent
- Localization is supported when using sub-specs

Assigning Sub-Specs

The following characteristics apply when assigning sub-specs:

The following specs (referred to as ‘Master Specs’) can utilize sub-specs:

- Primary Spec
- Secondary Spec

When one or both the master and the sub-specs are localized, both specs must have the same locales associated with them to be able to assign the sub-spec

A single sub-spec can be assigned to any number of master specs

Any number of sub-specs can be assigned to a single master spec

The attributes of a sub-spec can appear on a master spec only once

Users have the option to add sub-specs when adding nodes to a master spec

When a sub-spec is added to a spec, all the unique attributes from the sub-spec are added to the bottom of the master spec and the name of the sub-spec will not appear. If there are duplicate nodes, an error message is displayed and all the nodes that can be added are added

It is not possible to edit sub-spec nodes in a master spec

It is possible to view the originating sub-spec of an attribute. In the master spec, click on a sub-spec node to view its details

It is possible to remove a node that was added through the sub-spec

If a sub-spec has already been added, re-adding it will look for nodes that are not in the master spec (whether they are new or had previously been deleted) and add them to the bottom of the master spec

Modifying sub-spec attribute parameters modifies their behavior in the master specs in which they are used

Applying Inheritance

This section details each step that is performed to apply inheritance rules. The following steps assume that no pre-existing specs, catalogs, and categories will be used and everything is created from new.

- Create Primary Spec for Inheritance
- Create Hierarchy with Inheritance
- Create Catalog with Inheritance
- Define Inheritance Rules
- Enter Targets
- Enter Items into a Catalogs
- Enter category data and category inheritance

Create Catalog Spec For Inheritance

The goal of this section is to create two catalogs (parent and child) that use the same Sub Spec.

Create a Sub Spec

1. From the Specs Console, click Sub-spec from the Spec toolbar then click **New**.

2. Add nodes to the spec, as needed. When all nodes have been added, click **New**.

Note: A Sub Spec cannot be deleted if it is in use.

Create new Catalog Spec

Goal: Create two catalog specs for the parent child catalogs and add nodes from the sub spec created in the previous section.

1. From the Specs Console, click **Catalog** from the Spec toolbar then click **New**. The Spec Tree Wizard appears.

2. Enter a name for the first Catalog Spec, then click **Next**.

3. Add non-shared nodes as needed, then add shared nodes. To add a shared node, select the newly created sub spec from the popup window. All nodes from the sub spec are inserted into the Catalog spec.

Note: Share nodes are denoted with a hand in the attribute type icon.

4. Select a Primary Key and click on **Save**.

5. Create a second Catalog spec and select the same sub spec.

Create Hierarchy with Inheritance

Goal: Configure a Hierarchy for inheritance.

1. From the Hierarchy Console, click **New**. The Create Hierarchy Wizard appears.

2. Enter a name for the Hierarchy.

3. For "Select Inheritance Available", select **Yes**, and click **Next**.

4. For the purposes of this document, select No for "Select Use Code for Aggregation and Syndication".

5. Click create hierarchy and the new tree appears in the Hierarchy Console.

Note: There is no inheritance icon that appears next to the hierarchy name until an inheritance rule is defined for it.

Create Catalog with Inheritance

Goal: In this step, two catalogs are created with inheritance using the catalog specs created in the previous sections.

1. From the Catalog Console, click **New**. The Create Catalog Wizard appears.

2. Complete each step of the Wizard.

- Select a Catalog Spec to use
- Select a Node to use as the Inheritance Key. The values that appear in the drop-down list are the same from the Sub Spec created in the previous section
- Select an Access Control Group for the Catalog
- Enter a Catalog Name
- Select the Hierarchy that was created in the previous section

3. Create another Catalog using the same inheritance key selected in Step 2.

When both catalog have been created, they appear in the Catalog Console. Catalogs that have been configured with inheritance have the Inheritance icon next to the catalog name.

Define Inheritance Rule

Goal: Edit the Inheritance Rules for both catalogs so that the inheritance paths (targets) are properly defined.

1. Click on the Inheritance icon next to the catalog. The Inheritance Rule

table appears. Click **New**.

2. Complete the Define Inheritance Targets Wizard.

- Select a catalog spec node to define a target path, click **Select**.
- Select No for "Accumulate for this catalog"

3. A table appears showing the available targets. Select the target paths, more than one target can be selected. Click Save.

4. Define inheritance rules for each catalog, as needed.

5. View all Inheritance rules by using the menu path Data Model Manager > Specs/Mappings > Inheritance Console.

Use Inheritance Feature

Now that everything is setup with Inheritance rules, Items can be entered into catalogs or categories. The following sections step through how different inheritance relationship types being applied.

Enter Items into Catalogs

1. Enter an Item into the one catalog and add data to the attributes.

2. Enter an Item with same SKU into the other Catalog.

3. Press Refresh to get Inherited Data.

The item shows the values from the target path. The following functions are available:

- View the inheritance path in a separate window
- View the data from the inherited item
- The "I" button denotes that the attribute is using its inheritance rule. Click the "O" button to override the attributes governing rule.

View Inheritance Path

To view the inheritance path of an attribute, click the view button, a separate window appears displaying the path for each attribute.

Overriding Inheritance Rule

To override an inheritance rule, click on the "O" button and change the attributes value. The original value can be returned by clicking the ‘I’ button, but if it is saved, the original value will not return.

Note: When overriding a value

Enter Category data and Category Inheritance

Use the following steps to add category data, and then view how information is inherited

1. Create a Hierarchy Spec and add to Hierarchy with Inheritance.

2, Update Inheritance Rules to include Hierarchy

3. Add Data into the Categories with Items assigned to them.

4. Add / Modify Data in Items to see Inheritance scenarios

Issues and Limitations

Implementing inheritance within a catalog structure can have some implications with existing catalogs. Test inheritance with data in a test environment to ensure stability.

Inheritance Key = Primary Key

When identifying an inheritance key, it is recommended to use an attribute that is set to be the Primary Key. This ensures that the value is unique and does not create. Remember, a Sub-Spec does not have a primary key, yet a defined node can be the same primary key used for a catalog.

Non-Unique Inheritance Key

Although an Inheritance Key can be defined as a non-unique attribute, it is not recommended. The results of selecting a non-unique inheritance key are undeterminable. Where an attribute obtains its value can differ from case to case. Therefore, it is suggested to select an attribute that is unique.

Relationship Performance

When structuring a catalog to use inheritance, the use of an inheritance relationship can provide different performance results. The following list of inheritance relationship is rated by performance, with the first being the fastest.

- Item-to-Item
- Item-to-Category
- Category-to-Category

Entry Build Scripts

Entry Build Scripts are usually executed when values are entered into the data entry screen. An entry build script can be defined at the Catalog attribute screen. Advanced WebSphere Product Center users with good amount of scripting experience should use this feature.

Avoiding Circular Reference

It is not possible to create a circular reference when implanting inheritance and WebSphere Product Center does not allow a user to do so. For example, there are three items (Item 1, Item 2, and Item 3). With inheritance defined, Item 3 looks to Item 2 for a value and then to Item 1. It would not be possible to define an inheritance path that looks for a value in Item 3.

Deleting Parent Values

When deleting attribute values that have inheritance rules associated to them, it is recommended to first view all associated inheritance paths and double-check the data integrity of other catalogs. When using this feature for large catalogs, it should be expected for the operation to take a good deal of time to complete as the application is directed to find all associated instances.

How to Migrate Catalogs to use inheritance

If it is desired to implement inheritance with a current set of catalogs, a new catalog structure must be created with shared nodes from Sub Specs. It is recommended to contact WebSphere Product Center on the best way to architect a catalog with inheritance implemented.

Other limitations to note

- It is not possible to search on inherited values
- Inheritance between items within a given catalog is not available
- Inheritance across trees is not available
- Inheritance of empty values is not available
- Multi-occurrence inheritance may cause to exceed the number of occurrences and make an item invalid without going to the item; multi occurrence inheritance should only be used with a large number of maximum occurrence and without the intention of enforcing the maximum occurrence.
- Deleting multi-occurrence inherited is not possible
- Inheritance will not support inheritance for multi-occurrence fields within a group
- If an inheritance rule exists for a multi-occurrence group, and the spec is modified to no longer be multi-occurrence, the inheritance rule will have to be deleted and recreated
- It is possible, in the GUI, to tell not what item/category is inheriting from an item, but what category or item COULD be inheriting, up to a maximum of 50 category or items
- For category-to-item inheritance, items can only inherit from their immediate parent
- Nodes within sub-specs cannot be made unique and cannot be created as links
- Sub-specs cannot be used by specs other than catalog specs, hierarchy specs, and secondary specs
- Sub-specs cannot be added under a group in a master spec

- If the same sub-spec attribute is inherited more then once by the same item, only one instance of the attribute will be shown; it will not be possible to predict which instance will be shown
- If an item or a category has more then one possible inheritance source, it will not be possible to predict which value will be inherited
- When adding a new item or category, the inherited values will not appear until after saving and refreshing the screen
- Inherited values will not appear on a screen automatically; users will need to refresh the screen to see the inherited values
- It is not possible to create distinct inheritance rules for attributes within a group, which includes localized nodes
- Inheritance does not support attributes of type FLAG unless the source catalog's flag is set to "false". The work around for this is to simply use the STRING_ENUM type with three choices: YES/NO/NA
- Currently, Multi-value inheritance is ONLY supported for Grouped Multi-values. Therefore, NON_GROUPED, MULTI_VALUED attributes cannot have inheritance on them
- Inheritance DOES NOT support value rule nodes

Customized Help Text (Context sensitive)

WebSphere Product Center provides the ability to create customized help text with the option to make it context sensitive. The help feature is defined at the catalog spec level, which identifies the link to the customized help file. Thus, a separate location can be used to store and maintain all customized help files and WebSphere Product Center is linked to the file location.

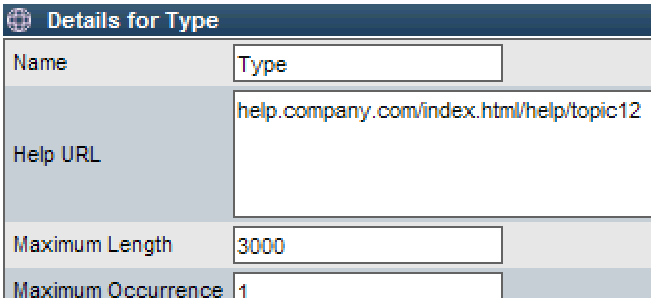
- Help files open in a standard separate window and remains open until the user closes it
- Only one modeless window is opened for customized help files, when a new file is accessed the old page is replaced with the new content

Creating help text

1. From the Specs Console, select a primary spec and click on the node that will have a help text defined. This includes localized nodes, which allows the definition to localized help files.

2. Select **Help URL** from the drop-down field and click the Plus icon.

3. Enter the custom help file location in the Help URL field.



Details for Type

Name	Type
Help URL	help.company.com/index.html/help/topic12
Maximum Length	3000
Maximum Occurrence	1

Figure 5. 1 – Access context sensitive help

4. Click Save to store the changes made to the spec.

5. When a user adds or edits items in the catalog, the help icon is displayed next to the nodes that have been defined with a Help URL (Single Edit Screen only). To access the help file, click the button next to the node name in the Single Edit screen, and select "Help about attributes".

Create Context Sensitive Help

By default, customized help files are accessed when accessing a node from the Single Edit screen. If it is desired for the help file to activate when the field is accessed in the Single Edit screen, update the user defined settings. Set the following option:.

- From **Home > My Settings**, on the Specific Screen Settings table, select "Yes" for "Always display help text.

Views

Catalog Views allow users to personalize which item attributes are editable, read only, or not visible. Views are created for the following purposes:

- Create a more efficient or task-specific view of items when working in a catalog
- Often used to create groups of attributes related to a specific data entry or data maintenance process (i.e. promotion related attributes)
- Created multiple views to be used with each catalog
- Create views shared by all users
- Catalog privileges to determine which attributes a user can choose from to create a view

Catalog views can also be configured to customize the way a catalog is displayed in the Left Pane, Item Lists, or Content Authoring Screens. Views can be configured in the following manners:

- Hide a specific item attribute
- Re-order how attributes are displayed on the screen
- Restrict modification of certain attributes

Views and core attributes

Views are defined by attribute collections that include the following types of collections:

Generated Default core attributes	System defined attributes that are retrieved and saved for every object and include only the attributes that are created to making sure an item is not saved to the database in violation of key rules, which include: • Primary key • Path attribute (for categories only) • Any required attributes (either from a primary or secondary spec)
User defined core attributes	Attributes that are required to be retrieved and saved for every object. These include attributes that need to be validated or calculated every time an item or category is saved.

Views work with core attributes in the following manner:

- Only attributes in the current view are retrieved and saved
- Core attributes (default or user defined) are always retrieved and saved, even if they are not in the view
- System view is modified to display only core attributes
- The benefit is improved user response time and reduced server loads due to focused data retrieval and processing

The benefit is improved user response time and reduced server loads due to focused data retrieval and processing

System Default Views

Each hierarchy is defined with a default view; this view includes all attributes applicable to the hierarchy. The view can be a combination of Default core attributes and User defined core attributes.

Fields with default values, unique fields, or sequences should be considered for the user defined core attributes or a "new item view". If a field with a default value is not in the core attributes or in the view, the field is never populated unless the field was brought up from with a different view.

View management

Views are created, deleted, and modified through the Catalog console and can be used in any are that provides a drop-down list of views.

Creating a catalog view

1. From the Catalog Console, select a catalog and click the **Views** button. The catalog view wizard appears. A catalog must be selected.

2. Enter a name for the new catalog view and click **Next**.

3. Choose an attribute from the left column and click the **Add Viewable** or **Add Editable** buttons. The selections appear in the right column. There are four view configuration areas:

- Edit Item: choose the fields that are to be made editable
- Bulk Edit: choose the fields that you can add bulk items to (Usually, all fields are chosen since this is an addition of new items to your item catalog)
- Item List: choose the fields that are to be displayed within an item list
- Item Popup: choose the fields that appear in a popup window
- Item Location: choose the fields that appear in item locations data

4. When all selections have been made, click the **Save** button. A message displays stating that the new view has been successfully created.

5. To use the new view, make a catalog selection from the Catalog Console and select the view from the Views column..

Deleting a catalog view

Any catalog view can be deleted, except for the System Default view. To delete a catalog view, it must first be selected from the Catalog Console.

1. Select a catalog view from the Views column and click the **Edit** button, the Catalog View Create/Edit page displays.

2. Click the **Delete** button and a confirmation dialog box appears, click **OK**.

Modify a catalog view

To modify an existing view:

1. Navigate to the Catalog Console

2. For the catalog for which you wish to modify a view, select the view from the Views column.

3. Click the **Edit** button next to the view.

Using a catalog view

A catalog view can be changed from the Catalog Console, Single Edit, or Multiple Edit screens.

To use an existing view:

1. Navigate to the Catalog Console

2. For the catalog for which you wish to view, select the view from the Views column.

3. Upon selecting the view, the view is automatically applied and will be remembered upon the next logon. Views can also be changed in the Single and Multiple Edit screens..

Tabbed Views

The customized views created for a catalog restrict the attributes that are viewable or editable and are displayed in a basic layout. WebSphere Product Center has the ability to create Tabbed Views for a catalog, which decreases the need to scroll through a catalog's long list of attributes.

Tabbed views are created from a current Catalog View. It is possible to display the same attribute throughout multiple Tab Views, except for the Primary Key, which always appears on the top of each Tab View.

Creating a tabbed catalog view

When creating a catalog view, it is possible to create a tabbed view to manipulate further how items are displayed.

1. From the Catalog Console, select a catalog and click the Views button. A catalog must be selected.

2. Enter a name for the new catalog view and click Next.

3. Select all attribute collections that are to appear in all tab views and click Add Viewable or Add Editable. If an attribute is not selected here, it will not appear when creating a Tab View.

4. Click Save, to create the catalog view before creating a tab view.

5. Click the Tab View button to access the Tab View page and click New to create a new tab view. A list of all selected attributes from the Customized view appears in the Tab detail table.

6. Enter a name for the Tab name, which will appear on the tab label, select the attribute collections to appear in the tab and click Save.

Note: Add>> is used to add attribute collections to the Tab View.

The tab is listed in the Tab Views table. A new Tab View must be created for each tab.

Edit Tab View

To add or delete attribute collections from a Tab view, the tab view must be edited.

1. From the Catalog Console, select the Catalog view that has been customized with Tab Views and click the edit button..

2. Click on Tab View button and the Tab for Catalogs table appears.

3. Click the radio button next to the Tab View that is to be edited and click the edit button.

4. Add or remove attributes to the Tab View as needed and click Save to commit the changes.

Delete Tab View

If a catalog view is no longer needed and deleted, all associated Tab Views are deleted. To delete a single tab view, do the following:

1. From the Catalog Console, select the Catalog view that has been customized with Tab Views and click the edit button.

2. Click on Tab View button and the Tab for Catalogs table appears.

3. Click the radio button next to the Tab View that is to be deleted and

click the delete button.

Reorder Tab View

The order of the Tab Views can be rearranged

1. From the Catalog Console, select the Catalog view that has been customized with Tab Views and click the edit button.

2. Click on Tab View and the Tab for Catalogs table appears.

3. Click the radio button next to the Tab View that is to be reordered and click the up or down buttons.

Linking Catalogs

WebSphere Product Center’s Linking feature is only available to Catalog specs. In order to create a catalog with information from two or more other catalogs, create a child catalog that has a link to one or more other catalogs.

By linking one catalog (the "source" or "master" catalog) to a second catalog (the "destination" or "child" catalog), items within the destination catalog can be supplemented by the attributes of respective linked items in the source catalog.

- Allows shared attribute values to be maintained in one source/master catalog
- The "source Attribute Name" is the attribute in the child catalog that references the key in the master catalog
- The "Destination Attribute" is the name of the key in the master catalog

Purpose and Pre-Requisites of Linking Catalogs

Purpose of linking catalogs:

- To create a relationship between items in the "master" catalog with

items in the "child" catalog
- Inherit Attributes from a child catalog for syndication

Pre-requisites

- Have a linked attribute in the "master" catalog
- Have a "child" catalog

Example 1:

- The master catalog can be an item catalog
- The child catalog can be a customer catalog
- The catalogs are linked via the attribute "customer id"
- Prices can then be recorded by a specific customer in the customer catalog and linked to the master catalog

Example 2:

Multiple customer catalogs share the same description and list price, however, the customer price is different for each

- Create a single master catalog to maintain the master item number, description, and list price
- Create a child catalog for each customer which maintains only the customer price
- Link these customer-specific child catalogs to the master catalog

Example 3:

Linking catalogs "normalizes" the product information

- Placing the data in multiple catalogs enable WebSphere Product Center to present, access, and update the items and attributes much more rapidly

Design Catalog Architecture

When designing the architecture of linked catalogs, decide where the information for a catalog is going to be retrieved.

- One item in two catalogs (merging the attributes for one item)
- Merged list of items

There are two ways to link catalogs, which provide different types of relationships to one another.

- 1 - Link multiple child catalogs to a parent/master catalog. This is useful in the case of having multiple catalogs for various marketplaces and pulling all the items into the master catalog.
- 2 - Create a catalog from two or more catalogs, a child catalog must be created with a link to one or more catalogs

Note: The important thing to note is that an attribute in a child catalog can have at most one link. Therefore, to link two catalogs to one catalog there must be a link in each of the child catalogs.

Case 1

Both links in the child catalog should be selected (probably with the same SKU). They syndication script used will determine how the data is handled.

Case 2

In this case, only one of the two catalogs would be linked. When the catalogs are exported, the import script defines the data output. In the following figure, the Child catalog is linked to the Master Catalog.

Steps to link catalogs

There are several steps to take when creating a link between catalogs:

- Select the source attribute
- Select the catalog to link to
- Select the destination attribute
- Create a link between catalogs

Select the source attribute

The source attribute is the attribute from the sub-catalog that is mapped to the "master" catalog. If there is only one attribute of type link, this is the only attribute that is available in the drop-down. If no attributes of type link exist, no drop-down field is available.

Select the catalog to link to

In the list of catalogs, select which catalog the attribute links. It is possible to select from any of the existing catalogs in the system, except the catalog that a link is currently being created for.

Select the destination attribute

When selecting a catalog, the field of the destination attribute shows, by default, the primary key for the master catalog. This is the field used to link to the sub catalog. To create a link from an item in the sub catalog to an item in a master catalog, enter a value in the source attribute of the item in the sub catalog that corresponds to the primary key of an item in the master catalog.

Create a link between catalogs

The following steps provide an example of linking a catalog for the following use case:

Multiple customer catalogs share the same description and list price, however, the customer price is different for each.

1. Create a single master catalog to maintain the master item number, description and list price

2. Create a child catalog for each customer that maintains only the customer price

3. View the catalog attributes for the Catalog that contains the source attribute

4. From Category Detail screen, select the Source Attributes Name, Product Catalog, and the Destination Attributes Name (all defining the link to the other catalog).

5. Click on "Add" to Add the New Link

Catalog to Catalog Exports

To facilitate the loading of print catalogs (or any catalog) from another catalog, use Catalog-to-Catalog Exports. A Catalog-to-Catalog Export loads the categorization and items from one catalog into another catalog.

Setting up a catalog to catalog export

Use the menu path: **Product Manager > Catalogs > Catalog-to-Catalog Export > New Catalog to Catalog Export**. The Catalog-to-Catalog Export wizard appears. Complete each wizard step.

1. **Select Catalog Source**: Select the source catalog used for the export

2. **Select the group of items to export:** the entire catalog or a saved selection

3. **Select the Catalog-to-Catalog Export Type**: either all items in a version, the differences between two versions, or the differences since the last version

4. **Select the Catalog destination**: If no catalog destination has been created, one must be created before going to the next step.

5. **Select Catalog to Catalog mapping**: Map the fields in the source catalog to the fields in the destination catalog

6. **Select the Catalog-to-Catalog Export Script**: which includes an auto-generated script by default

7. **Select approving authority**: Optional - Select the approving authority for the export

9. **Select the Catalog Export name** - Enter a name for the catalog export

After all steps of the catalog to catalog export has been completed, run the Catalog-to-Catalog Export from the Export Console. The export is sent to the scheduler like any other export job.

Ch 6 Localization

WebSphere Product Center provides localization that adheres to ISO’s Internationalization Standards. Locales are created based on language/country pairs, which provide variances across countries (i.e. U.S. English, British, English).

Localization is configured at the spec level (primary, lookup table, or secondary spec), therefore, it is possible to localize a single node or multiple nodes and set localized display names for each node. Once a spec has been configured with localization and associated with a catalog, localized data can be imported into or exported from WebSphere . Catalogs that contain localized nodes can be set to view a specific set of locales for an item set displayed in the user interface through the user's settings.

Features

Localizing spec node values for a primary, lookup table, or secondary specs

- Supports attaching language + country locales to any node in a primary spec, lookup table spec, secondary spec
- Locales incorporate language + country pairs, which provides for language variances across countries (e.g.: American English vs. British English)

Localize catalog display names

- Enables entering a locale-specific display name for all localized catalog attributes, or using a default display name

Display catalog items in one or more locales depending on the user's settings, role, and views

- User can set desired locales in a User Setting dropdown box

- Any locale created in the user's environment is automatically loaded into the User Settings
- User can restrict display via a User Setting to one locale or all available locales
- User can also setup catalog views to display one or more locales, which are displayed as they are ordered in the spec
- Catalog Access Privileges apply to all localized catalog nodes

Restrict locales

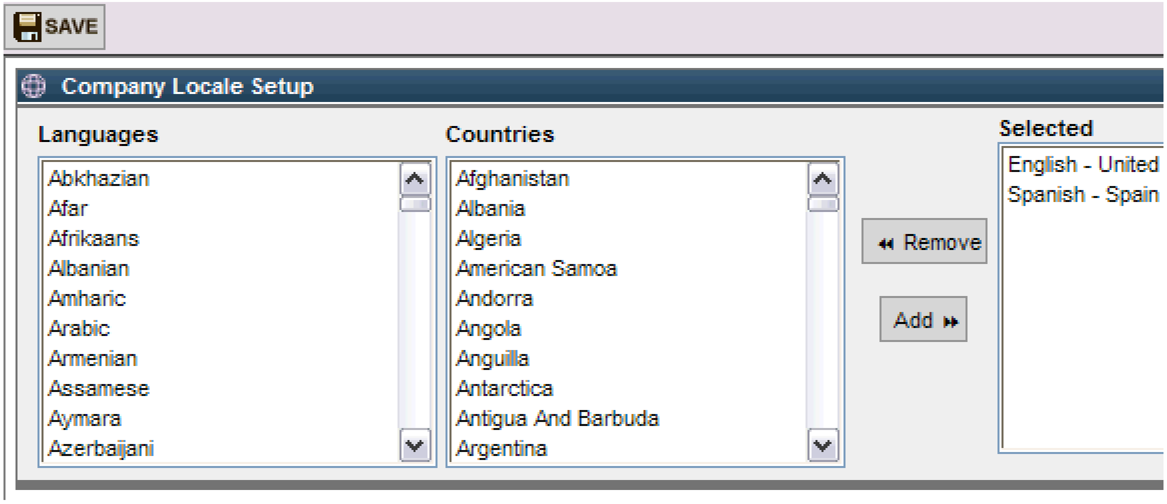
The restriction of locales that are displayed in the user interface is controlled through company attributes, user settings and/or role configurations.

- **Company Attributes** - Define the available locales for the company. This setting defined what locales are available in the user settings and role configuration.
- **Role configuration** - The locale setting can be defined for a Role, hence the user’s locale is based on the role they are assigned. The user can only control which locales are displayed based on what is available for their assigned role.
- **User settings** - A user can set the desired locale(s) using "Restrict the displayed attributes in item and category screens to the selected locale" within their user settings. Any locale created in the user’s environment is automatically loaded into the User Setting’s "Available Locales". Restrict a user's view to a single or multiple locales (if more than one locale has been created).

Setting available locales for the company

Before any data can be localized, the available locales must be set for the company. This task should be performed by the WebSphere Product Center administrator.

1. Use the menu path **Data Model Manager > Security > Company Attributes**; the "Company Locales setup" table appears.



Selected

English - United
Spanish - Spain

Figure 6.1 - Company Locale setup table

2. Click on a language from "Languages" and a country from "Countries", then click **Add** to add the locale to the "Selected column".

3. To remove a locale, click on a locale from the "Selected" column and click "Remove".

4. When all locales have been selected, click **Save**. All selected locales are available to users in the company.

Restricting locales to display for a role

Restricted locales can be defined for a role. When a user is assigned to the role, the user's graphical interface will be restricted to the locales defined for the role. Therefore, the user's locale is based on the role that they are assigned to and the role locale restriction overrides control from the user settings. From the list of roles to restrict or make available in the user settings, the user is able to control which locales are restricted based on the locales that are made available through the role definition.

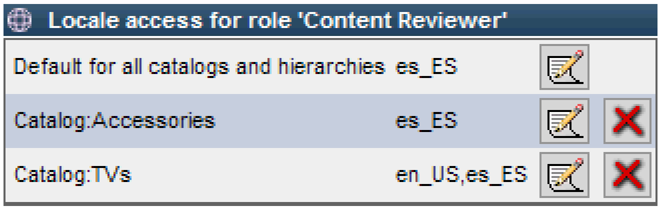
1. Use the menu path: **Data Model Manager** > **Security** > **Role Console**; the Role Console is displayed.

2. Select a Role to edit and click **Edit** or simply click on the name of the role to get to the edit role screen.

3. When a new role is created, it is defaulted for all categories and

hierarchies to use all available locales for the company. To change the locales available for this default setting, scroll to the bottom of the screen to the "Locale access for role" table. Click on the edit button and add/remove locales and click Save.

4. To create a new locale access for the role, click the **New Locale Access** button to display the "Containers To Locales Map" wizard. This allows the restriction of locales to a specific object (catalog or hierarchy. Add/remove locales and click **Save**.



Locale access for role 'Content Reviewer'

Default for all catalogs and hierarchies	es_ES	✎	
Catalog:Accessories	es_ES	✎	✗
Catalog:TVs	en_US,es_ES	✎	✗

Figure 6.2 - Locales access configured for role

5. To apply locale restrictions defined by the role, assign a user to the role using the User Console.

Restricting locales to display in user settings

1. Use the menu path **Home > My Settings;** the several user setting tables appear.

2. Select one of the following settings to control which locales are displayed in the user's graphical interface:

Locale for user interface display	Select a locale to use for the User Interface. This only applied to the graphical user interface.
Locale for Item and Category Data Display	Set the locale to display item or category data.
Restrict the displayed attributes in item and category screens to the selected locale	This option restricts the display of a selected locale for attributes in item and category screens.

3. Click **Save**, to commit the settings.

Localized specs

Localization is defined at the spec level for primary, lookup table, and secondary specs. This section details how to configure localization for a spec.

The implementation of localization within WebSphere Product Center provides the proper structure to support database localization, because all locales are stored as separate strings. The data for the node is moved to the leaf nodes.

The display names that are created will appear in the item list/edit screens based on the locale setting in the user settings. The actual translation of the data is done manually, but is stored in each individual leaf node.

Note: The localized check box option is not available if it has already been localized or localization is not available.

Configure spec for localization

1. From the Specs Console, select a primary, lookup table, or secondary spec to localize and click **Edit**.

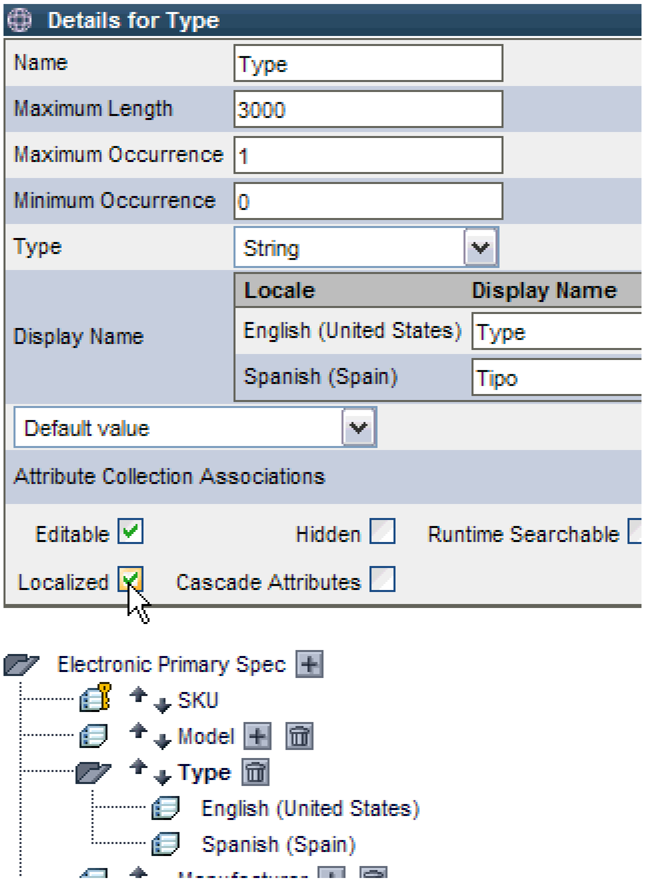
2. Select the **Localized** checkbox in the "Details for ... spec" table. The "Localization Information" section appears.

3. Select the locales from the "Available Locales" column and add them to the "Selected Locales" column. Click **Save** to save the spec.

4. Click on a spec node. Enter localized display names in the "Display Name" section. This allows for the display name for the spec node to appear in different languages.

5. Click the Localized checkbox for the spec node. Notice that the node has been localized for all locales defined for the spec. Continue to

localize each node of the spec, if desired.



Details for Type

Name	Type
Maximum Length	3000
Maximum Occurrence	1
Minimum Occurrence	0
Type	String

Display Name

Locale	Display Name
English (United States)	Type
Spanish (Spain)	Tipo

Default value

Attribute Collection Associations

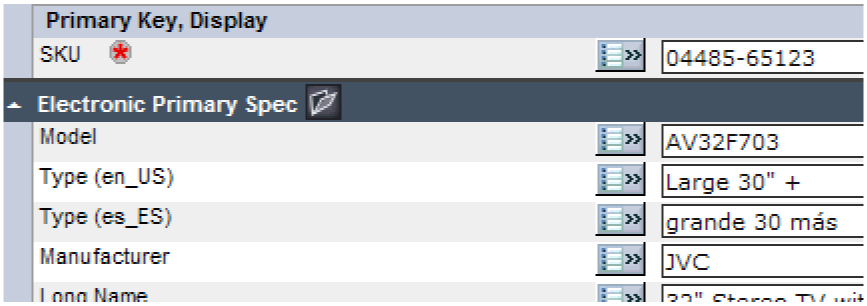
Editable ☑ Hidden □ Runtime Searchable □

Localized ☑ Cascade Attributes □

Figure 6. 3 – Localized node

Entering localized data

Once the a spec has been localized, data can be entered separately for each locale for an item. All locales appear in the catalog item list, unless specified otherwise in the user's setting or role. If no localized display name was created for the locale, the default name appears.



Primary Key, Display

SKU	04485-65123

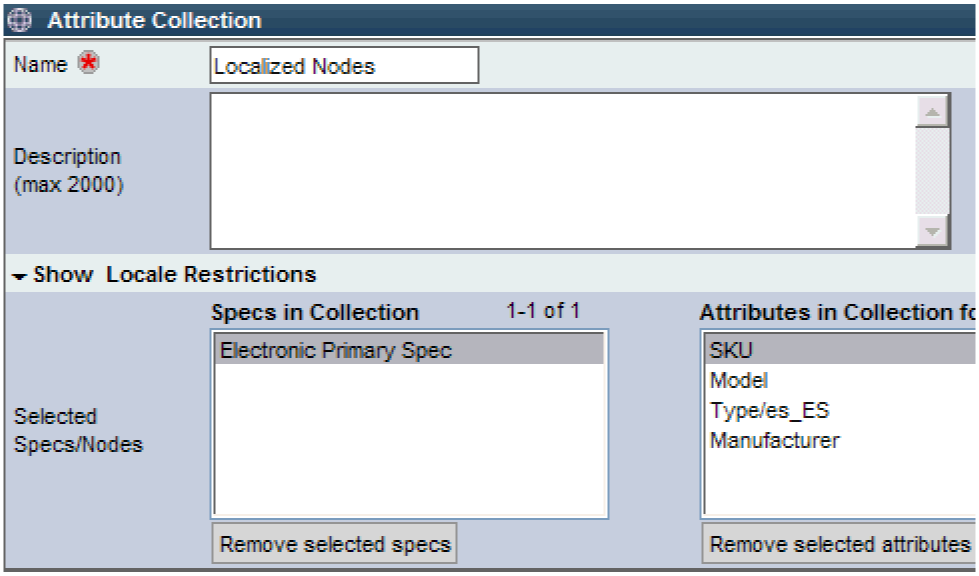
Electronic Primary Spec

Model	AV32F703
Type (en_US)	Large 30" +
Type (es_ES)	grande 30 más
Manufacturer	JVC
Long Name	32" Stereo TV wit

Figure 6. 4 - Data for localized node entered manually

Creating attribute collections with localized nodes

When defining an attribute collection, it is possible to include a node for a specific locale. When searching for a spec and/or attribute from the create attribute collection screen, localized nodes are indicated by "(L)'.



Attribute Collection

Name * : Localized Nodes

Description (max 2000):

▾ Show Locale Restrictions

Selected Specs/Nodes

Specs in Collection 1-1 of 1

- Electronic Primary Spec

Attributes in Collection fc

- SKU
- Model
- Type/es_ES
- Manufacturer

Figure 6.5 - Select node for specific locale for attribute collection

Import/Export to localized nodes

The ability to import/ into localized nodes is made simple through the use of WebSphere Product Center’s Mapping Console. If a source file contains fields that are localized for various regions they can be mapped to the proper localized nodes of the catalog. Thus, the same can be done for exports.

Localization issues

There are no locale selections in My Settings

The company has not been configured for any locales. Use the menu path **Data Model Manager > Security > Company Attributes**, the Company Locales Setup screen appears.

Select the required locales for the company. The selected locales appear as a selection in the user's settings.

Part III –Collaboration Manager

Part III includes chapters that cover each of the components that are available within the Collaboration Manager module.

- Imports
- Exports
- Collaboration Areas
- Queues
- Web Services
- Document Store
- Data Sources
- Routing
- Generated Files

Ch 7 Import Console

To centrally store and manage product information, a master catalog is created in WebSphere Product Center. The following sections describe each step that is performed to collect the data from various sources and to assemble the information into organized catalog structures.

- Overview of the Import Console
- Obtaining a data source
- Define items that are needed to create an import feed
- Setting up an import feed
- Load incoming file into WebSphere Product Center

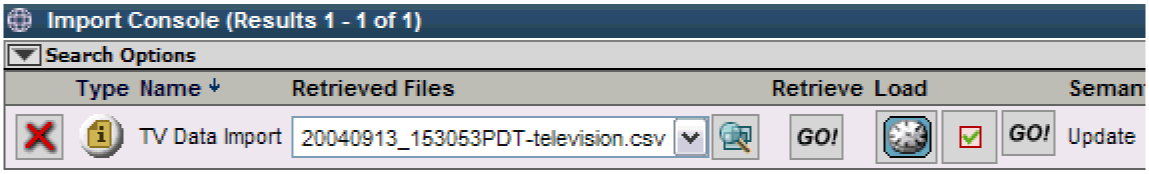
Import overview

An import is a feed that inputs data from an external source into WebSphere Product Center. An import is first configured manually and can then be run on a scheduled or on-demand basis. WebSphere Product Center allows the import of multiple types of data (items, binary files, category trees, categorization mappings, etc.) from multiple sources to accomplish multiple purposes (update, replace, delete).

The Import Console is the tool that is used to create catalogs from various sources of data. The import console acts as the switchboard for importing all data into WebSphere Product Center. From here you can view, modify and create data that is being fed into WebSphere Product Center.

To create an import, user needs to define and perform the numerous data and file feeds into WebSphere Product Center. The import console presents a list of feeds that have been created.

To access the import console, use the menu path **Collaboration Manager > Imports > Import Console**.



Import Console (Results 1 - 1 of 1)

Search Options

Type	Name	Retrieved Files	Retrieve	Load	Seman
	TV Data Import	20040913_153053PDT-television.csv	GO!	GO!	Update

Figure 7.1 - Import console

Customizing the console view

The console can be customized to sort or hide columns. The settings are saved to the user’s profile. To customize the view of the Catalog Console, do the following.

1. In the top right corner of the Catalog console, click the configure button . The Configure table appears in a popup window.

2. To sort a column, make a selection in the Sort Column. Only one selection can be made.

3. To hide a field, make a selection in the Hide column. Multiple selections can be made.

4. In the "Other Options" table, set the sorting order to display in ascending or descending order and set the number of rows to display per page.

5. To save the customized settings, click **Save**. The Catalog Console appears with the new customized settings.

Sorting a list of imports

- The imports in the Import Console can be sorted by the "Type" of import or by the "Name" of the import. Click the "Name" or "Type" column heading name to sort.

Console Columns

The following table lists each of the columns of the Import Console.

Type	Icon indicating the type of import,

	refer to the previous section "Import Console Icons" for a list of import types
Name	The name of the import
Retrieved Files	A drop-down selection of source files that have been retrieved into the WebSphere Product Center document store
Retrieve	Execute the manual loading of the data source into WebSphere Product Center
Load	Uploads the data source into a specified catalog
Semantics	Indicates if the import is of type Update, Replace or Delete.
Source	The name of the source file used
File Spec/Import Script	Name of the file spec or import script used
Map	Click on the icon to view or edit the file to catalog mapping
Ctlg/Cat Tree	The catalog and catalog tree used for the import
ACG	The ACG used for the import

Import Preparation

Before an import is created in the Import Console, there are several steps to perform in preparation of an import. This section describes the following import preparation tasks before an import is created and scheduled:

- Obtain data source
- Make data available
- Review data file
- Create required objects to setup an import

Obtain data source

Before data can be imported into WebSphere Product Center, the source files must be situated in an accessible directory or FTP site. Review the data file to ensure the most accurate product information and that the format uses one of the following data sources available for import into WebSphere Product Center:

- Character Delimited
- Tab Delimited
- Comma Separated Values (CSV)
- Fixed Width
- XML

Note: The source file must conform to one of the formats mentioned in the list above before continuing with the next step.

Make data available

Check that the source data file is available for access into WebSphere Product Center. If the files are to be uploaded through FTP, make sure that the proper permissions are applied to the FTP site. Contact the appropriate team members, as needed.

Review data file

Analyze the source data file and make note on the file structure, fields, and associated attributes. This information is required to build the File Spec in the next step.

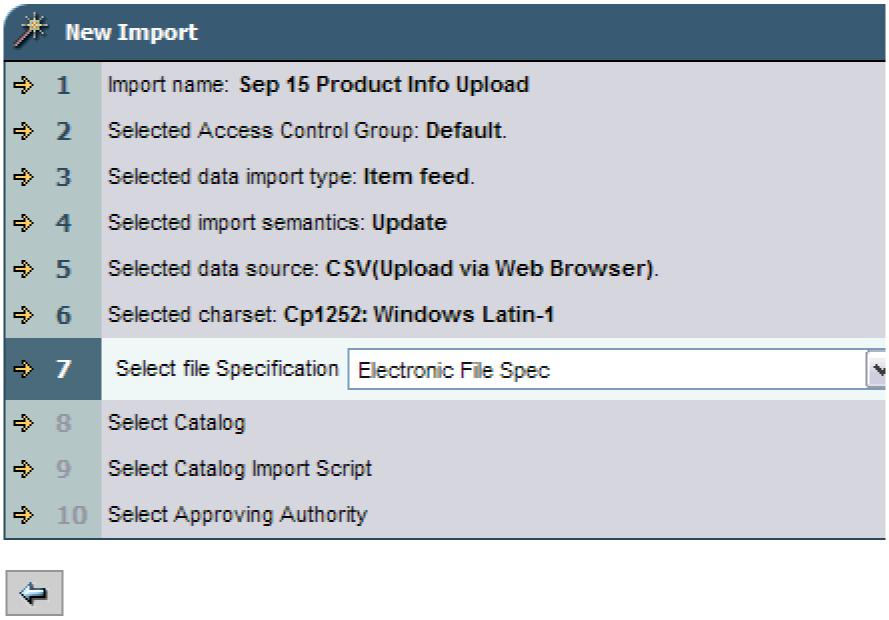
Create required objects to setup an import

There are several objects that are required to be present prior to setting up an import. Many of these objects can be created during the setup of an import, but it may be easier to have these tasks completed beforehand. The following objects are the minimum requirements to set up an import:

File spec	A file spec is required in order to configure an import. Please refer to the Data Model Manager section on creating a file spec.
Primary spec	A Primary Spec is required in order to configure the format of the WebSphere Product Center catalog structure. This may include specs for Hierarchies. Refer to the Data Model Manager section on creating a primary spec.
Spec mapping	In order to properly route data to the correct catalogs, the file spec needs to be mapped to the right catalog/hierarchies. Please refer to the Data Model Manager section on mapping specs.
Create catalog using primary spec	Once the file, catalog, and other associated objects have been created and mapped, the next step is to create the catalog. Please refer to the section on Product Manager for more information on creating a catalog.

Define Import Feed

After setting up the items for an import as described in the section "Import preparation" , define how the external file information is going to be populated into WebSphere Product Center catalogs. The definition of an import feed is done by creating a new import.



New Import

1 Import name: **Sep 15 Product Info Upload**

2 Selected Access Control Group: **Default**.

3 Selected data import type: **Item feed**.

4 Selected import semantics: **Update**

5 Selected data source: **CSV(Upload via Web Browser)**.

6 Selected charset: **Cp1252: Windows Latin-1**

7 Select file Specification: Electronic File Spec

8 Select Catalog

9 Select Catalog Import Script

10 Select Approving Authority

Figure 7.2 - New Import wizard

Create a new import

Use the menu path **Collaboration Manager> Imports > New Import** or click **New** in the Import Console. The New Import wizard appears. Complete each step of the Create Import wizard to create an import.

1. Import Name

Enter a name for the import and click the Next button.

An import name must be provided to distinguish from other names. Duplicates are not allowed. The feed name chosen should be comprehensive.

For example, the names "Item Attributes from Mainframe" or "Images for Catalog X" are good names.

2. Select Access Control Group

Choose Access Control Group from the Select Access Control Group drop-down menu. If Access Control Group does not exist, create one by clicking on NEWbutton.

3. Select Data Import Type

Choose a data import type from the drop-down menu and click the Select button. The following values are available for selection:

Binary Feed	Feed of binary files for each item in a catalog (i.e. images, docs and PDFs).
Hierarchy feed	Hierarchy feed includes a hierarchy that is designed to organize all of the items within the catalog for easy browsing.
Item feed	Item feed contains the essential descriptive information that describes each catalog item to be found within the product catalog.
Item to category map feed	Item to category map feed maps items that already exist in a catalog to a hierarchy that will be uploaded as part of the incoming data. This feed can also be used to upload a file that contains new items and the appropriate categorization information directly into a new catalog. The hierarchy will be updated whenever any new category is added.

4. Select Data Import Semantics

Choose an import feed semantic and click the Select button.

The import feed semantics specifies how the incoming file will affect the items already found in the destination catalog.

Update	Updates the item found in the catalog with the item that is specified in the incoming file and adds items to the catalog that only exists in the incoming file.
Replace	Replaces the content of the current catalog with the content of the incoming file.
Delete	Deletes from the catalog any items specified in the incoming file.

5. Select Data Source

Choose a data source name and click the Select button. A data source defines which data source is to be imported into WebSphere Product Center.

In this step, a user can modify or create a new data source. To create a new data source, click "New" and to edit a data source, click the Edit button.

The steps of creating a new data source include specifying a name for the data source and selecting a data source type. The data source name should be as descriptive as possible. Data source types include the following options:

- Have WebSphere Product Center retrieve via FTP
- Take from Document Store (including FTP)
- Upload via Web Browser

6. Select Character Set

The next step is to select the encoding of the incoming file from the "Select charset" drop-down menu. Select the character set that matches the one used in the incoming file. This is particularly important when incoming files contain characters outside the 26 letters of the alphabet and the 10 digits. The system default is "Cp1252: Windows Latin-1".

Note: Failure to select the appropriate character set may result in problems during import.

7. Select File specification

The file specification defines the attributes and format of the incoming file. Either choose an existing file specification from the drop-down menu or click **New** to create a new file specification.

8. Select Catalog

The catalog defines where the data will be imported into WebSphere Product Center. Either select an existing catalog or click **New** to create a new catalog.

9. Select File to Catalog Mapping

Either select an existing mapping or click **New** to create a new mapping. To create a new mapping (or modify an existing mapping) use the drop-down selection field to match each field in the file specification (left column) to match a field in the catalog (right column). For advanced manipulations (field concatenation, string manipulations, etc.) click on the Expression Editor button.

10. Select a Catalog Import Script

A simple catalog import script is always generated by default. Either select the default generated script or click **New** to create a new import script.

11. Select Approving Authority

The final step in creating an import feed is to select an approving authority. This optional step allows the selection of a user who is given the responsibility to review the import job and approve it before it is generated. Select which user should be notified when then feed is initiated.

12. Return to Import Console

At this point, the import feed is configured and the last step is to run the import job to populate the catalog where it can be managed within WebSphere Product Center.

Run Import

Once an import has been defined, the import can be manually scheduled or set to run on demand. Before running an import, the external data source file must first be obtained.

Load file into Document store

On the Import Console, there are two GO buttons for each import feed, one in the Retrieve column and the other in the Load column.

To retrieve the external incoming file, click on the GO button in the Retrieve column of the appropriate import feed. The retrieved column will show "No files for feed." When the files are fetched, a selection of files appears in a drop-down menu.

Once the file has been fetched, it is placed into the Document Store. The Document Store is a repository for all files and data that are moved into or out of WebSphere Product Center. This provides the user with an opportunity to preview all files before they are actually uploaded into WebSphere Product Center.

Start import

The second of the two GO buttons on the Import Console will upload the fetched file from the document store into the specified catalog. From the Load column, click the GO Button.

An alert can be configured to notify, via email or through the Alert module of the Home Page that the load has completed. If an approving authority has been assigned, the feed can be approved or rejected through the user's Home Page Alert module inter. The product information can then be managed through the Product Manager Module.

Managing imports

Modifying an import

To modify an import, it is best to delete the old import and create a new one. For example, there is no method of selecting a new file spec. To use a different file spec, create a new import and select the appropriate file spec.

Deleting an import

A feed can be deleted by clicking on the delete button in the first column of every feed.

Note: Deleting a feed is not reversible. A dialog box will appear to verify the deletion of an import feed.

Binary Feeds

Binary files must be compressed before they are imported into WebSphere Product Center. For example, to import a group of images, a file of *.zip is created in a source directory. Once the compressed file is retrieved, it is stored into WebSphere Product Center's Document Store and then uncompressed into the docstore's directory /public_html/ctg_files. Binary files can include such files as jpg, gif, or pdf.

Figure 7. 4 - Uploading binary feeds

Create a binary import

The following steps create an import feed for binary files.

Use the menu path, **Collaboration Manager > Imports > Import Console**, the Import Console appears. Click New to access the "New Import" wizard . Complete each step of the wizard.

1. Import Name – Enter a name for the binary feed

2. Select Access Control Group – Select an access control group used for the feed.

3. Select import type – Select "Binary feed"

4. Select import semantics – Select "Update"

5. Select data source – Select a data source for the binary feed or create a new one

Keep path from zip archive Y or N

Always select No. All image files are stored in the "public_html/ctg_files", in the Document Store. If you preserve the path of the archive, the images will extract into the wrong directory.

6. Enter docstore path

Leave empty and click Next. The docstore path is assumed and there is not need to enter a path. When the application unzips the file, it will place all images into the ctg_files directory.

7. Select action on file name and path

Accept the default **Neither** - This option is sued the case of the images from downstream systems. In this case there is no problem with the images files being either uppercase/lowercase.

8. Select Approving Authority – Select a user to approve the import, if necessary.

This completes the creation of the Import Feed. The next step is to upload the compressed file into WebSphere Product Center.

Upload Compressed File

After the binary import feed is defined, the next step is to retrieve the compressed file into WebSphere Product Center. The feed should show "No files for feed" in the "Retrieve Files" column.

1. From the Import Console, locate the newly created feed and click Go! from the Retrieve column.

2. Click Browse, select the compressed file, and then click Upload. The path to the zip file appears in the "File" field.

3. Return to the Import Console or to view the file in the Document Store, select the Document Store from Collaboration Manager menu.

All uploaded files are stored in a sub-directory of the organization's "feed_files" directory, which is named after the name of the import feed.

Once the compressed file is uploaded to the Document Store, it is uncompressed.

Uncompress Binary Files

1. From the import console, click Go! from the "Load" column. WebSphere Product Center, unzips the images into the ctg_files directory.

2. To check the status of the process load, click the "check the status of the update" hyperlink.

When the load is completed, the Schedule Status Information shows the status "Completed Running". Click the document store hyperlink to view all of the uncompressed binary files.

Lookup table imports

An existing import table can be imported from an external data source into WebSphere Product Center. Creating an import for a lookup table is similar to creating any other import using the "New Import" wizard.

Create a lookup table import

From the Import Console, click **New**. The "New Import" wizard appears.

1. Select a unique feed name.

2. Select an Access Control Group.

3. Select the data type "item feed".

4. Select the relevant feed semantics, data source, and file specification

5. From the "Select Catalog" field, select the lookup table to be imported

6. Define the file to lookup table mapping

7. Select the import script. Create a new import script if necessary.

8. Select the approving authority, if necessary.

9. Save the new import feed.

10. Run the import

Ch 8 Export Console

WebSphere Product Center allows you to create catalogs customized for any internal or external destination. The catalogs can be automatically published to customers, partners, and print catalogs, or to a corporate website. The following sections describe the method used to distribute or publish product information to any destination.

- Export Overview
- Prepare catalog for export
- Create export
- Run an export
- Managing exports

Export overview

Exports are created to distribute or publish product information to the customers (internal or external). With WebSphere Product Center's flexible architecture, businesses can connect to multiple marketplaces with multiple catalogs, using a single repository of data.

Using WebSphere Product Center's Export Console, the following tasks are performed:

- Export catalogs into any format
- Publish variances
- Publish subsets of content
- Merge destination-specific data and standard information

Export Console Columns

Type	Icon indicating the type of syndication, refer to Export Console Icons section
Name	Name of the Export job
Catalog Items	The name of the catalog and group of items that are to by syndicated

Version	The version of the catalog that is to be syndicated
Mapping	View/edit the mapping characteristics between the catalog spec and the destination spec
Destination Spec	Name of the destination spec
File Type	The name of the Catalog Export script used and parameter values set.
Hierarchy	The name of the hierarchy used, click to view or edit.
Job Info	Detailed information on the syndication job. View the job's run time, status, warnings, and a list of generated files.
Approval	If approval is needed before syndication, the status appears in this column as accepted or rejected
Alerts	If an alert has been configured for the syndication, it appears in this column
Destination	The syndication's destination mode

Prepare catalog for export

The following sections describe the items that are needed before a catalog can be setup for an export. Although several items can be performed through the wizard for setting up an export, it is beneficial to have the items prepared beforehand to simplify the setup process.

Destination Spec

When an export job is created in WebSphere Product Center, a destination spec is needed to define the exact requirements of the destination file. Similar to a file spec or primary spec, each node of the destination spec is defined with attributes. During the data export, the destination spec will generate a file that adheres to a set of predefined requirements.

Item Selection

The Export component of the Data Model Manager allows for the basic selections of a group of items from a catalog to be pushed to a specific destination. The basic selection includes the catalog, catalog version, and the hierarchy used for the export.

Catalog to Destination Map

The Catalog to Destination Specification map is similar to the File to Catalog Map, used for imports, in that it instructs WebSphere Product Center with how the fields in the catalog should be routed. The fields in a catalog are mapped to the fields of the destination file. This mapping must occur for a successful syndication.

Hierarchy

As per the explanation given in the Import chapter, hierarchies are stored separately from catalogs within WebSphere Product Center. This allows the user to view and eventually export a catalog using a hierarchy of their choice.

Destination Mode

A catalog, once exported, can be sent out to multiple destinations for purposes of approval and/or verification. The distribution component of syndication allows a user to select the destinations to where the catalog is to be sent. The catalog can be sent out in the form of an email (in the body of the email) or FTP.

Create Export

When creating a new export, the "Create New Export" wizard walks through each step for setting up an export, which helps to simplify the complete process.

During each step in the wizard, the following functional buttons are available for use:

✎	Edit
NEW	Create New
SELECT	Select
Next	Continues to the next step
⇦	Returns to the Export console

Setting up an Export

Use the menu path **Collaboration Manager > Export > New Export** or from the Export Console click the **New** button, the Destination Upload wizard appears Complete each step of the wizard driven GUI.

1. Export Name - Enter a new name for the export and click **Next**. Create a name that is unique and can be easily identified.

2. Select a Destination Specification - The specification itself dictates the format, length and necessity of each field of data that will eventually be exported. If needed, the user can click the appropriate buttons to create or edit a destination specification.

3. Select Catalog - Specify the catalog to be exported by selecting any of the previously created catalogs from the "Select Catalog" drop-down field. Typically, the catalog selected for export is targeted to a specific customer. That is, it will only include information that is specific to particular customers.

This information might include data such as pricing, contract effective dates and customer product IDs. Using the catalog to destination specification map, WebSphere Product Center can export data from either a master or child catalog.

4. Select Version - Select the Catalog Version Not only does the user have the opportunity to select the catalog to be exported, they can also select the version of a catalog. WebSphere Product Center automatically saves every version of each catalog that is uploaded and modified. It also creates a new version for a catalog when a feed or an export is run. This feature allows a user to export an older version of the same catalog in the event that, for example, some items were added to a catalog that are not supposed to be sent to a particular customer.

- Latest version - ensures that every time the job is run, the latest product information is exported, including all updates. This selection is "dynamic".
- Any other version - ensures that every time the job is run, the same specific catalog version is exported. This selection is "static".

5. Select Group of Items - Select a basic selection of a group of items from an entire catalog to be pushed to a specific destination. The basic selection includes the catalog, catalog version, and the hierarchy used for syndication. The item selections can be viewed and created through the Item Selection console of the Product Manager module. An example of a selection maybe used to export only a particular product line.

6. Select Hierarchy - This feature allows a user to export each item's categorization within one of the selected catalog's hierarchies. Although selecting the hierarchy itself will determine which categories will be exported, it does not ensure that the categories themselves are sent with the output file. The categories themselves must be included as part of this export's catalog export script. The catalog export script requires configuration of an authorized user and the output file will not include categories unless it has been written into the catalog export script.

7. Select Syndication Type - By selecting the export type, WebSphere Product Center provides the user with the option of selecting whether to syndicate an entire catalog or simply an update to the catalog. This allows the export of catalog variances.

- All items in a version
- Content differences between two specific versions
- Content differences since last syndication
- All images and binary files for items in a version
- Updated images and binary files between two specific versions
- Updated images and binary files since last syndication

If a "differences" export type is selected, select a second version of the catalog to generate a delta against. Differences types include:

- All
- Added
- Deleted
- Modified
- Unchanged

8. Select Catalog to Destination Mapping - Either select an existing mapping or click **New** to create a new mapping. To create a new mapping (or modify an existing mapping), use the drop-down fields to match each field in the catalog (left column) to match a field in the destination spec (right column).

9. Select charset - Select a Character set to choose the encoding of the outgoing file. This is particularly important when outgoing file contain characters outside the 26 letters of the alphabet and the 10 digits.

The default is Cp1252:Windows Latin-1.

10. Select Distribution - Distributions are designed to permit the notification of individuals inside and outside of the application itself. A distribution is similar to an email alias. Building these sorts of distributions lists within WebSphere Product Center makes it possible to reach people outside of WebSphere Product Center. If needed, create a new distribution mode.

11. Select a Catalog Export Script - Catalog Export scripts are used during an export. They can be used to perform advanced on-the-fly operations on the data held in the catalog before it is actually exported to an output file. The modifications made to the content through the scripting engine at the time of export and are not applied to the catalog but rather simply applied to the output file, as a one-time content modification.

Similar to an import, the export to an external target file can take two forms: either the fields in the catalog map on a one-to-one basis to the external target file, or the fields in the catalog require some modifications before they are exported into the external target file.

All exports require the use of a script. Contrary to imports, selecting a script during export is a mandatory step.

12. Select Approving Authority - Allows the selection of a user who has the responsibility of reviewing the export job and when the job is executed. Choose from a list of system users and click Select.

13. Return to Syndication Console

Once all of the above steps are completed, a message box appears notifying the addition of the export profile to the Export console. Click the back arrow button at the bottom o f the screen to return to the Export Console. The export job is now set up and ready for generation.

Run an export

After the export job is configured, it can be generated manually or automatically through the Scheduler. If the job is dependent on approval, the approving authority must accept the job before it can be generated.

Manually run an export

To manually execute an export job, do the following:

1. Use the menu path **Collaboration Manager > Export > Export Console** to view a list of all export jobs.

2. Find the export job to execute. Click on the checkbox located in the leftmost column.

3. Click the **GO** button. If an approving authority has been set, the export job starts but the file is not exported to the destination until it has been authorized. The status of the export job can be viewed through the Scheduler's Jobs Console, within the Data Model Manager module.

Verifying export progress

While an export is running, clicking on the green animated icon allows a user to review the progress of the export. A progress bar is shown to indicate how much of the export has completed. To interrupt an export, click on the red icon on the left of the export; a confirmation dialog box is displayed.

When an export is completed, the progress bar disappears. A summary is offered to include details such as a hyperlink to all the files that were created during the export and errors/warnings that occurred during the export.

Export management

View Output

When an export is generated, it is saved to the Document Store before it is sent to the defined destination. If desired, the results of the generated files can be viewed and printed.

Viewing export results

To view the output results, do the following:

1. From the Export Console, click the status button from the Job Info column, the Schedule Status Information table appears.

2. Click the hyperlink in the Return Value column, the Task Approval Information interface appears with a list of generated files. Each file is hyperlinked and when clicked, a new window opens displaying the file details.

Review export file

Standard exports produce one file entitled 'catalog.out'. This file can be reviewed and saved to a local or network drive.

Reviewing an export file

- To review an export file, click the 'catalog.out' hyperlink from the export results screen and the file opens up in a new browser windwo.

Approve Output

If an approving authority has been assigned to an export job, it must be approved before the job can be executed. From the approver's My Task List, click the Pending Approval icon in the Approval column, the Task Approval Information interface appears. Accept or Reject the export job and provide a reason.

View and Modify Export Components

From the Export console screen, the user is permitted to view and/or modify virtually every component of an export job. The user is allowed to view the list of customers that the catalog is to be sent to, the catalog to marketplace specification mapping and the dates and times that the catalog is scheduled for syndication. The user can modify the catalog being exported by clicking on the catalog name.

Export status

To view the status of an export job, click the schedule button in the Job Info column, the Schedule Status Information appears. Then click the View Details button in the Jobs Information column.

Once a catalog has been executed for export, a user can check the following job details:

- Job information
- Schedule information
- Running Time
- Status of the job
- Return Value

Reviewing export status

To review the job status of an export job, do the following:

- From the Export Console, click the schedule button in the Job Info column. The Schedule Status Information table appears displaying detailed information on the syndication job.

Delete exports

Export jobs can be created and deleted as needed.

Deleting an export

To delete an export job, do the following:

1. From the Export Console, find the syndication job to delete. Click on the checkbox located in the leftmost column.

2. Click the Delete button and the export job is deleted from the console.

Troubleshooting exports

If an import or an export runs into problems, hyperlinks to Errors or Warnings are displayed on the job summary.

- Errors are critical problems that prevented the successful processing of an item (i.e. primary key not present or mandatory field not present).
- Warnings are non-critical problems that should not have prevented the successful processing of an item (i.e. field too long).

Sometimes, a job will fail to complete to run and instead of a summary, an error will be displayed. Record precisely in which circumstances the problem occurred and take note of the error message for further analysis.

Ch 9 Queues and messaging

The new Queues feature was implemented to maintain multiple queue types and multiple messaging protocols. A queue serves as a gateway to handle inbound and outbound messaging with external sources/destinations, including EAI platforms and web servers.

Note: Continue to the section "Implementing Messaging Framework" for information on setting up a Messaging Framework.

For reference, the following terms are defined:

- **Queue** - a WebSphere Product Center construct serving as a receiving and transmission point for messages. A script supports each Queue.
- **Message** - an XML document provided by UCCnet, EAI platform, data pool, or other message source

With the new queue feature, messages can be used as part of a process, in that when jobs are imported or exported, status messages can be sent to all required parties.

Queues Console

Accessing the Queue Console

Use the menu path **Collaboration Manager > Queues > Queue Console**, the Queue Console displays.

View Queue Details

To view the details of a queue click on the **Queue Name** and the "Queue Detail" screen appears. The following information is provided:

- Queue Name
- Description

- Protocol
- Script

View Message in Queue

The Queue Console has a column called Messages with a hyperlink to the number of messages that have been received by a queue. Click on the number of messages to view the contents of the message.

Search for message in a Queue

1. To search for a message in a queue, use the menu path Collaboration Manager > Queues > Message Console. The Queue Messages Search screen appears.

2. Select a value for the following fields

- Arrival Date From
- Arrival Date To

3. Click Search and the results appear in the Queue Messages table below.

Creating a Queue

1. Use the menu path Collaboration Manager > Queues > New Queue

2. Enter the required information:

Queue Name: Enter a name for the queue

Description: Enter a description for the queue

Protocol: Select from the list of messaging protocols that will be used to tie the external source to/from the queue.

Script: Select from a list of pre-written scripts that can be run to route messages to/from the queue. A typical message for destinations/sources include:

- A workflow collaboration area, which is supported by an in or out

script in the workflow step
- A catalog that is supported by a pre-processing, post-processing, or post-save script

Implementing messaging framework

The implementation of WebSphere Product Center's messaging framework allows for integration with the following target EAI platforms:

- IBM WBI
- SeeBeyond
- Tibco
- WebMethods

The platform of choice should provide a reliable transport mechanism and a consistent interface for programs to communicate with each other across different systems or platforms.

WebSphere Product Center's messaging framework was designed to support the following process:

- Receiving a message containing an item set, including providing acknowledgement messages upon receipt
- Sending a message containing an item set, including receiving acknowledgement messages after the transmission

WebSphere Product Center ships with the ability to parse and create XML documents, and to send messages to, and collect messages from an EAI platform queue. A message is defined as an XML document provided by an external source. All this functionality is accessible to the WebSphere Product Center scripting engine. To enable interaction between WebSphere Product Center and the EAI platform adapter, it is necessary to install scripts that take advantage of this functionality.

WebSphere Product Center supports queues, which are a construct for serving as a receiving and transmission point for messaging. The WebSphere Product Center queue serves as a gateway to handle inbound and outbound messaging with external sources/destinations and each queue is supported by WebSphere Product Center Scripting Operations.

A queue provides the following functionality:

Enables setting up a message transmission protocol to tie the external message source to/from the queue and provides the following message transmission protocols:

- MQ
- JMS Point-to-Point
- JMS Publish and Subscribe
- HTTP
- HTTP/S

Runs a script to route messages to/from the queue. Typical message destinations/sources include -

- A workflow collaboration area -Supported by an in or out script in a workflow step
- A catalog - Supported by a pre-processing, post-processing, or post-save script

The following list depicts the framework for the messaging functionality:

- An external source sends a message to the EAI Platform. The matter in which this is handled will vary depending on the external source and has no effect on how WebSphere Product Center will retrieve the message.
- The setup of the EAI platform should include an inbound and outbound queue. This setup is done by a third party and the only information WebSphere Product Center needs to know in order to retrieve and send messages, is the identification of the queues that will be accessed by WebSphere Product Center.
- As files are placed into the EAI platform queue, a corresponding queue setup on WebSphere Product Center is able to retrieve the message via a supported protocol (MQ, HTTP/s, or JMS).
- Using WebSphere Product Center Scripts, the message of the body is parsed to obtain the message type, message ID, and

message source. This information is routed to a target WebSphere Product Center catalog that has been created to contain all messages.

- The events that occur in D are tracked using a workflow and when the message has been recorded successfully, an event triggers the Acknowledgement Workflow to check out the new record from the Message Catalog into the collaboration area setup.
- The Acknowledgement Workflow sends an acknowledgement message to the message source for confirmation. When this is done, the Acknowledgement Workflow checks in the message back into the catalog.

In order for this process to function, a messaging framework must be built.

Building a messaging framework

The following process outlines a suggested method of integrating an EAI platform framework. This process may be customized based on specific requirements.

Receiving a message

This section describes the process of receiving a message containing an item set, including providing acknowledgement messages upon receipt. The two processes, Setup and Runtime support the process of receiving a message. The process is generic and applicable to most purposes.

<u>Setup</u>

1. A technical business process analyst creates the following:

2. A technical business process analyst builds an inbound Queue containing a script. The script supports three functions - message receipt, message body parsing, and routing.

Message Receipt

The message receipt section of the script supports the following functionality:

- Obtains message from source via a supported protocol - including MQ, HTTP/S, or JMS
- Parses message body to obtain message type, message ID, message source
- Creates record in Message Catalog containing message type, message ID, sender ID, and datetime
- Triggers event to Acknowledgement Workflow. See below for the Acknowledgment Workflow functionality.

Message Body Parsing

The message body parsing section of the script supports the following functionality:

- Includes as parameters the source-to-destination map name and the target catalog name
- Parses the message body per the source-to-destination map name and the target catalog name to render an item set

Message Routing

The routing section of the script supports the following functionality:

Adds/modifies/deletes items in the item set from the target catalog

- A technical business process analyst sets up an Acknowledgement Workflow for sending a receipt acknowledgement message. The workflow has the following functionality:
- Checks out new record from Message Catalog into a collaboration area in a step in the Acknowledgement Workflow.
- A next step in the Acknowledgement Workflow sends an acknowledgement message to the message source containing the message ID, sender ID, datetime, and any required commands by the message source (e.g.: Received).
- A next step in the Acknowledgement Workflow

checks in the message catalog record.

Runtime

Once the setup has been properly configured, the following runtime events should occur.

1. Queue receives message via message receipt section of queue script

2. Message receipt section of queue script parses message body to obtain message type, message ID, and sender ID

3. Message receipt section of queue script creates record in Message Catalog containing message type, message ID, sender ID, and datetime

4. Message receipt section of queue script checks out new record from Message Catalog into a collaboration area in a step in the Acknowledgement Workflow.

5. A next step in the Acknowledgement Workflow sends an acknowledgement message to the message source containing the message ID, sender ID, datetime, and any required commands by the message source (e.g.: Received).

6. A next step in the Acknowledgement Workflow checks in the message catalog record.

7. Message body parsing section of queue script parses the message body (using a new script operation cited below) per the map name and the target catalog name to render an item set

8. Routing section of queue script adds/modifies/deletes items in the item set from the target catalog using existing script operations for catalog adds/modifies/deletes

Create WebSphere Product Center inbound/outbound queue

WebSphere Product Center inbound and outbound queues are created using the Queue Console. Before any queue is created, a Trigger script must be created from the Scripts Console. The trigger scripts appear in the drop-down Trigger Script Path field of the New Queue screen.

1. In the Queue Console, click **New**.

2. In the Queue Detail screen, enter a queue name, description; select a protocol and a trigger script path. The trigger script is created in the Scripts Console of type "Message Queue Processor".

3. Click **Save**.

Messaging Script Operations

WebSphere Product Center script operations provide a flexible way to write WebSphere Product Center script applications with defining capabilities in the arguments of each script operations. The following script operations identified in this section are used to support the messaging feature supported by WebSphere Product Center using MQ or JMS. These methods allow the import and export of messages from an external queue.

Note: The scripting operations listed in this section are subject to change. Refer to the Script Sandbox for the most up-to-date script operations.

MQ Scripting Operations

When creating a script application, the script operation mqGetQueueMgr returns a handler to MQQueueManager, with that handle, multiple MQ operations can be made before calling mqDisconnect to release the handle.

mqGetQueueMgr

- Prototype: MQQueueManager mqGetQueueMgr(String hostname, String port, String channel, String queueMgrName)
- Description: Creates and returns a new MQ queue manager with the given properties.

mqDisconnect

- void MQQueueManager::mqDisconnect()
- Disconnects from the given queue manager.

mqSendTextMsg

- Prototype: MQMessage MQQueueManager::mqSendTextMsg(String msgText, String queueName, String queueOpenOptions, String messagePutOptions)
- Description: Sends a message provided in the String msgText over queueName. Returns the MQMessage

  Note: If trying to send a reply to a given message using mqSendReply, an error will return if the mqSendTextMsg is used. To avoid this, use mqSendTextMsgWithReply

mqSendTextMsgWithReply

- Prototype: MQMessage MQQueueManager::mqSendTextMsgWithReply(String msgText, String queueName, String replyQueueName, String queueOpenOptions, String messagePutOptions)
- Description: Sends a message provided in the String msgText over queueName. The reply queue is specified. Returns the MQMessage object.

mqGetTextFromMsg

- Prototype: String mqGetTextFromMsg(MQMessage mqMessage)
- Description: Returns a string containing the entire content of a MQMessage, including headers.

mqGetReceivedMsg

- Prototype: MQMessage MQQueueManager::mqGetReceivedMsg(String queueName, String queueOpenOptions, String messageGetOptions)

- Description: Receives a message from queueName. Returns the message, as a MQMessage, or null.

  Note: When messages are retrieved, they are removed from the queue. Unless specifying a message ID, the first message in the queue is obtained.

mqSendReply

- Prototype: MQMessage MQQueueManager::mqSendReply(MQMessage receivedMsg, String msgText, String passedInQueueOpenOptions, String passedInMessagePutOptions)
- Description: Sends a reply to the given message, without indicating success or failure.

mqSendReplyWithStatus

- Prototype: MQMessage MQQueueManager::mqSendReplyWithStatus(MQMessage receivedMsg, String msgText, String status, String passedInQueueOpenOptions, String passedInMessagePutOptions)
- Description: Sends a reply to the given message, setting the feedback field to indicate the given status. Status must be one of the following (in upper or lower case): SUCCESS, FAIL, VALCHANGE, VALDUPES, MULTIPLE_HITS, FAIL_RETRIEVE_BY_CONTENT, BO_DOES_NOT_EXIST, UNABLE_TO_LOGIN, APP_RESPONSE_TIMEOUT, NONE.

  Note: Only one status value can be used.

mqGetXMLMessageContent

- Prototype: String mqGetXMLMessageContent(String orgXmlMsg)
- Description: Discards any garbage at the beginning of the input string to get a XML document. More precisely, behaves as follows: If the input string is of the form A + B, where B is a valid XML document and A is any (possibly empty) string, this

operation returns B. Otherwise, returns null.

Note: Use this method to pares incoming messages.

mqGetResponseToMsg

- Prototype: MQMessage MQQueueManager::mqGetResponseToMsg(MQMessage outgoingMessage, String queueOptions, String messageOptions)
- Description: Gets the response to the given message from the given queue.

mqGetMessageDiagnostics

- Prototype: String mqGetMessageDiagnostics(MQMessage message)
- Description: Returns a string containing diagnostic information about the given message.

mqGetMessageId

- Prototype: String MQMessage::mqGetMessageId()
- Description: Returns the ID of the given message as a String containing a hexadecimal number.

mqGetReceivedMsgByMessageID

- Prototype: MQMessage MQQueueManager::mqGetReceivedMsgByMessageID(String queueName, String messageId, String passedInQueueOpenOptions, String passedInMessageGetOptions)
- Description: Finds the message in the given queue with given message ID. The ID is passed in a a String containing a hexadecimal number. Returns null if there is no such message in the given queue.

JMS Script Operations

When creating a script application, the script operation jmsGetConnectionFactory returns a handler to QueueConnectionFactory, with that handle, multiple JMS operations can be made before calling jmsDisconnect to release the handle.

jmsGetContext

- Prototype: Context jmsGetContext(String url, String jndiFactory)
- Description: Creates a JMS context.

jmsGetConnectionFactory

- Prototype: QueueConnectionFactory Context::jmsGetConnectionFactory(String jmsFactory)
- Description: Creates and returns a jms connection factory with the specified context.

jmsGetMQConnectionFactory

- Prototype: QueueConnectionFactory jmsGetMQConnectionFactory(String mqQueueManager, String mqHostname, String mqChannel, Integer mqPort)
- Description: Creates and returns a JMS connection factory for communicating with MQ queues. Note that you do not need a Context to get an MQ connection factory whereas you need a Context for connecting to other JMS queues.

jmsGetQueueByName

- Prototype: javax.jms.Queue jmsGetQueueByName(Context ctx, String name)
- Description: Returns a javax.jms.Queue object from the given JNDI Name and Context.

jmsGetQueueConnection

- Prototype: QueueConnection QueueConnectionFactory::jmsGetQueueConnection()
- Description: Returns a JMS queue connection from the given connection factory.

jmsGetQueueSession

- Prototype: QueueSession QueueConnection::jmsGetQueueSession()
- Description: Returns a JMS queue connection from the given connection factory.

jmsDisconnect

- Prototype: void QueueSession::jmsDisconnect(QueueConnection qcon)
- Description: Disconnects from the given queue manager.

jmsCreateTextMsg

- Prototype: Message QueueSession::jmsCreateTextMsg(String msgText)
- Description: Creates a new JMS TextMessage using QueueSession information with the text provided.

jmsSendMsg

- Prototype: Message QueueSession::jmsSendMsg(Message msg, String queueName[, HashMap properties, Message messageToReplyTo])
- Description: Sends a message MSG over queue with name queueName and returns MSG or null. If a MESSAGETOREPLYTO is provided, the reply to queue and message id are read from it. PROPERTIES is a map from string keys to string values. There are three special keys "WPC_REPLY_TO_QUEUE", "WPC_COPY_CORRELATION_ID_BYTES", and "WPC_COPY_CORRELATION_ID". If WPC_REPLY_TO_QUEUE is provided, then it overrides the QUEUENAME or replyto queue in MESSAGETOREPLYTO provided. replyto queue in MESSAGETOREPLYTO overrides QUEUENAME. "WPC_COPY_CORRELATION_ID" and "WPC_COPY_CORRELATION_ID_BYTES" copy over correlation id from MESSAGETOREPLYTO to MSG. Both can be provided. Their values need to be boolean (as opposed to strings -

as described above)

jmsReceiveMsgFromQueue

- Prototype: JMSMessage QueueSession::jmsReceiveMsgFromQueue(javax.jms.Queue queue, Integer timeout[, String messageSelector, JMSMessage messageToReceiveReplyFor])
- Description: Receives a JMS Message. Times out after TIMEOUT milliseconds. If INBOUNDQUEUE is not null, looks on that queue. If INBOUNDQUEUE is null, and MESSAGETORECEIVEREPLYFOR is not null, looks on the queue defined in the Reply-To field of MESSAGETORECEIVEREPLYFOR. If INBOUNDQUEUE is null and MESSAGETORECEIVEREPLYFOR is null, throws an AustinException. We now know which queue will be used. If MESSAGESELECTOR and MESSAGETORECEIVEREPLYFOR are both null, selects the first message from that queue. Otherwise selects the first message from the queue (if any) fulfilling all of the conditions defined by MESSAGESELECTOR and MESSAGETORECEIVEREPLYFOR. If MESSAGETORECEIVEREPLYFOR is not null, rejects any message not having a correlation ID equal to MESSAGETORECEIVEREPLYFOR's message ID. If MESSAGESELECTOR is not null, rejects any message not fulfilling the condition defined in messageSelector. If no appropriate message is found, returns null.

jmsGetTextFromMsg

- Prototype: String Message::jmsGetTextFromMsg()
- Description: Returns a string containing the entire content of a JMS message, including headers.

jmsGetMessageID

- Prototype: String Message::getJMSMessageID()
- Description: Returns a string containing the JMS message id.

jmsGetMessageCorrelationID

- Prototype: String Message::getJMSMessageCorrelationID()
- Description: Returns a string containing Correlation Id for the JMS message.

jmsGetMessageProperties

- Prototype: HashMap Message::getJMSMessageProperties()
- Description: Returns a hashmap from string property names to string values for those priorities.

jmsSendMsgToQueue

- Prototype: JMSMessage QueueSession::jmsSendMsgToQueue(JMSMessage msg, javax.jms.Queue outboundQueue [, HashMap properties, JMSMessage messageToReplyTo,])
- Description: Sends message MSG and returns MSG or null. The message is sent to the queue specified by OUTBOUNDQUEUE, unless OUTBOUNDQUEUE is null. If OUTBOUNDQUEUE is null, MSG is sent to the reply-to queue of MESSAGETOREPLYTO, if MESSAGETOREPLYTO is provided. If OUTBOUNDQUEUE is null and MESSAGETOREPLYTO is not provided, throws an AustinException. If MESSAGETOREPLYTO is provided, the message id is read from it. PROPERTIES is a map from string keys to string values. There is one special (non-JMS) key: WPC_INCOMING_REPLY_QUEUE. WPC_INCOMING_REPLY_QUEUE indicates a javax.jms.Queue object to which an external application should send replies to this message.

jmsSetMessageText

- Prototype: void Message::setJMSMessageText(String msgText)
- Description: Sets the provided text for the JMS TextMessage. Only JMS TextMessage type is supported.

Web Services

Web Service Console

The Web Service console provides users the ability to create and manage web services by used of SOAP_HTTP. A WSDL script can be written to define the service and an implementation script is created to control how the service is executed.

Web Service console columns

The Web Service console contains the following columns.

Name	Name of the Web Service. Click on the name to view the details of the web service.
Description	Brief description used to describe the Web Service
Protocol	Currently, SOAP_HTTP is the only available protocol
Transactions	Number of transactions for the service. Click the number to view the transaction or click the Delete button to remove the web service from the console

Managing Web Services

Accessing the Web Service Console

To access the Web Services console, use the menu path **Collaboration Manager > Web Services > Web Service Console**.

Creating a new web service

Use the menu path: **Collaboration Manager > Web Services > New Web Service**. The "Web Service Detail" screen appears.

Enter the appropriate information into the following fields:

Web Service Name	Enter the name of the web service. This name becomes part of the URL of the SOAP service. It must not contain any white space. For example,
Web Service	Enter a description for the web service.

Description	
Protocol	The protocol used for the web service. Currently, SOAP over HTTP is the only supported protocol. The default value is “SOAP_HTTP”.
URL	Provides the URL where the service may be accessed. This field is automatically populated after the web service has been saved.
WSDL URL	The URL where the WSDL for the web service may be accessed. This field is automatically populated after the web service has been saved.
WSDL	Enter WSDL for this service. A WSDL document is a description of the interface, URL and protocol of the service in XML format. You must enter this document manually, but we provide a sample WSDL document below. You must enter valid XML for the web service to save successfully.
Implementation script	Enter a Trigo script that implements this service. The incoming parameters for the service are available in the array variable "soapParams". The return value for the service must be a String, and can be written to the "out" writer variable. To return a SOAP fault, write the fault code to the "soapFaultCode" writer variable and write the fault message to the "soapFaultMsg" writer variable. A sample implementation script is provided below.
Store requests?	If this is checked, Trigo will store the parameters of all incoming requests in the docstore. They will be available from the Transaction Console.
Store replies?	If this is checked, Trigo will store the content of all responses in the docstore. They will be available from the Transaction Console.
Deployed	If this is checked, the service will be deployed. Otherwise this service will not be available.

A sample implementation script and WSDL document

The following design of the sample implementation script and WSDL document checks the number of parameters of the incoming SOAP request. If there are exactly four parameters, it returns a string listing those parameters. If there are more or less than four parameters, it throws a SOAP fault.

Implementation Script

```
nParams = soapParams.size();
if (nParams != 4)
  {
    soapFaultCode.writeln("WRONG_NUM_PARAMS");
    soapFaultMsg.writeln("Wrong number of parameters.  This service requi
  }
else
  {
    out.writeln("Success.");
    for (i = 0; i < nParams; i++)
    {
      out.writeln("Parameter " + (i + 1) + " is " + soapParams[i]);
    }
  }
```

WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://my.trigo-instance.com/soap/serv
                  xmlns="http://schemas.xmlsoap.org/wsdl/"
                  xmlns:apachesoap="http://xml.apache.org/xml-soap"
                  xmlns:impl="http://my.trigo-instance.com/soap/services/
                  xmlns:intf="http://my.trigo-instance.com/soap/services/
                  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding
                  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
                  xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
                  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:message name="invokeRequest">
    <wsdl:part name="param1" type="xsd:string"/>
    <wsdl:part name="param2" type="xsd:string"/>
    <wsdl:part name="param3" type="xsd:string"/>
    <wsdl:part name="param4" type="xsd:string"/>
  </wsdl:message>
  <wsdl:message name="invokeResponse">
    <wsdl:part name="invokeReturn" type="xsd:string"/>
  </wsdl:message>
  <wsdl:portType name="CheckParams">
    <wsdl:operation name="invoke" parameterOrder="param1 param2 param3 pa
      <wsdl:input message="intf:invokeRequest" name="invokeRequest"/>
      <wsdl:output message="intf:invokeResponse" name="invokeResponse"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="CheckParamsSoapBinding" type="intf:CheckParams">
    <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/s
    <wsdl:operation name="invoke">
      <wsdlsoap:operation soapAction=""/>
      <wsdl:input name="invokeRequest">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/enc
                       namespace="http://DefaultNamespace"
                       use="encoded"/>
      </wsdl:input>
      <wsdl:output name="invokeResponse">
        <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/enc
                       namespace="http://my.trigo-instance.com/soap/servi
                       use="encoded"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="CheckParamsService">
    <wsdl:port binding="intf:CheckParamsSoapBinding"
               name="CheckParams">
      <wsdlsoap:address location="http://my.trigo-instance.com/soap/servi
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Managing transactions

Accessing the Transaction Console

To search for all web services transaction, view the Transaction Console using the menu path **Collaboration Manager > Web Services > Transaction Console**.

Viewing a web service transaction

1. From the Transaction Console, view the list of transactions from the Web Service Transactions table.

2. Click on the View button from the **Response** or **Request** columns. The transaction details appear in a new browser window.

Searching a web service transaction

1. From the Transaction Console, select a date range from the **Arrival Date From** and **Arrival Date to** fields of the "Web Service Transaction Search" table.

2. Click the **Search** button. All transaction results appear in the "Web Service Transactions" table below the search table.

Ch 10 - Document Store

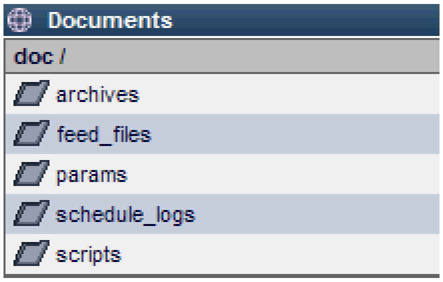
The Document Store is the area within WebSphere Product Center where every incoming and every outgoing file is stored. This includes import scripts, reports, and specification files. For auditing purposes, search capabilities are provided.

Every file that goes through WebSphere Product Center is copied and stored into the Document Store. Thus, the Document Store can be utilized as a backup engine.

The Document Store is structured in that the GUI provides hyperlinks to files that are stored on the Oracle database. Essentially, they are pointers to the location of the files.

Through the document store, users can manage files in the following manner:

- View a file's content and details
- Control access to a file
- View a document's audit log
- Delete files



Documents

doc /

archives

feed_files

params

schedule_logs

scripts

Figure 10.1 document store

Accessing the Document Store

The document store can be accessed through the following toolbar path, **Collaboration Manager > Document Store**.

The document store is presented with a structure of file directories. Click on any folder/sub-folder to view the files located within the directory. The full document path is displayed.

Viewing Document Details

The Document Store provides details on each document. This includes audit log information which shows when and who has accessed the document.

- To view a document's details, click the View icon.

View Document Contents

- To view the content of the document, click the Get Content hyperlink from the document details table. The contents of the document appear in a new window.

Control Access to Documents

The Document Store allows the control of access to a document. An Access Control Group (ACG) can be selected, or a new ACG can be created.

1. To control which user group can access a document, click the lock and key button next to the document name. The Docstore Access Details table appears.

2. If an ACG has already been created, select an ACG from the drop-down field. Otherwise, click New to create a new ACG. A user group can be controlled to create, view, or delete the document.

3. Once the ACG is selected, click **Save** to keep the control setting for the document.

Delete Documents

The Document Store allows the deletion of files through the document store, Click on the garbage can icon next to the name of the document to delete it from the document store.

Note! Once the file is deleted, it cannot be restored! A dialog box appears to confirm the delete. Click Cancel to cancel the delete operation.

Ch 11 - Data Sources

A data source is a WebSphere Product Center Entity that defines how data will be imported into WebSphere Product Center. External data can originate from various locations or databases and be accessed in a number of different ways. From example you might want to import data from a database, a file found on an FTP server, or data from your local file system. Each of these options requires specific configuration parameters in order to access the data. A Data Source encapsulates these parameters, and lets them be manipulated as a single named entity so that once the source details are setup it can be re-used throughout the product.

Creating a data source

To create a new data source click on the New button from the Data Source console. (existing data sources can also be modified using the edit tool button). The "Create Data Source" wizard appears. Complete each step of the wizard.

1. Data Source - Enter a name for the data source.

2. Select Data Source Type - The steps of creating a new data source include specifying a name for the data source. As in the case of an import, the name should be as explanatory as possible. Ex: "Upload through Browser" or "Retrieve by FTP". After selecting a data source type, click **Select**.

Possible data source types include the following:

Have Websphere Product Center retrieve via FTP By selecting this option, the user is telling WebSphere Product Center to connect to a specified FTP server to retrieve the data file to be uploaded. WebSphere Product Center requires the user to specify the following additional information so that it can connect to the FTP server and retrieve the correct file: the server address and port number, the username and password, the file name and directory in which it is located.
Take from Document Store (including FTP)

The Document Store is an area of WebSphere Product Center that stores every single document that has either been imported into WebSphere Product Center or that has been generated by WebSphere Product Center for export. It is therefore possible to fetch files from that document store into a feed.
Updated via Web Browser This option is used to manually import a file into WebSphere Product Center using the "Browse" function to explore and select a file residing on your hard disk or on any network accessible drives.

3. Provide details about the data source - The details to provide regarding the data source depends on the selection made in Step 2.

If data source type selected is...	**Additional information required**
Have Websphere Product Center retrieve via FTP	Enter values for the following: • Server address • Server port • Username • Password • Filename • Directory
Take from Document Store (including FTP)	Enter the full path to the document in the Websphere Product Center Document Store
Updated via Web Browser	No additional details are required.

4. After completing each step of the "Create Data Source "wizard, return to the Data Console.

Editing a data source

- From the Data Source console, find the data source to delete and click the edit button. The Create Data Source appears. Change the data source configuration, as needed.

Deleting a data source

- From the Data Source console, find the data source to delete and click the delete button. The data source is removed from the Data Source Console.

Ch 12 - Routing

Distributions are designed to permit the notification of individuals inside and outside of the application itself. A distribution is similar to an email alias. Building these sorts of distributions lists within WebSphere Product Center makes it possible to reach people outside of WebSphere Product Center.

The Routing Console displays all of the previously created distributions stored within WebSphere Product Center. From here, a user can modify or delete any current distributions and create new distributions or distribution group. A distribution group is simply a named group of previously built distributions.

Distributions

A catalog, once exported, can be sent out to multiple destinations for purposes of approval and/or verification. The distribution component of exports allows a user to select the destinations to where the catalog is to be sent. The catalog can be sent out in the form of an email (in the body of the email) or ftp. Essentially, a distribution is also referred to as the destination mode.

Create New Distribution

To create a new distribution, do the following:

- Click **Collaboration Manager > Routing > Routing Console**, the Routing Console interface appears.
- Click the **New Distribution** button. The "Distribution" wizard appears. Complete each step of the wizard.

1. Distribution Name - Enter a name for the distribution and click **Next**.

2. Select Distribution type - Choose a distribution type and click

Next. Select from one of the following choices:

- Ariba catalog upload
- Custom
- Email
- FTP
- Post

3. Distribution Mode - Depending on the distribution type selected, enter the required information as needed. Click **Save** to store the distribution settings.

Create Routing Group

- Click **Collaboration Manager > Routing > Routing Console**, the Routing Console interface appears.
- Click the **New Distribution Group** button. The "Distribution Group" table appears. Complete each step of the wizard.

2. Enter a name for the distribution in the Group Name field.

3. Select a distribution names from the available list of distributions and click **add** or **remove** to create the group.

4. Click **Save** to store the distribution group setting. The new distribution group appears in the Routing Console.

Delete Distribution

- To delete a distribution or distribution group from the Routing Console, click on the delete button next to the name of a distribution or distribution group.

Part IV - Data Model Manager

Part IV covers each of the components that are available within the Data Model Manger.

- Scheduler
- Specs/Mappings
- Attribute Collections
- Scripting
- Security
- Alerts
- Staging Area
- Workflows

Ch 13 Scheduler

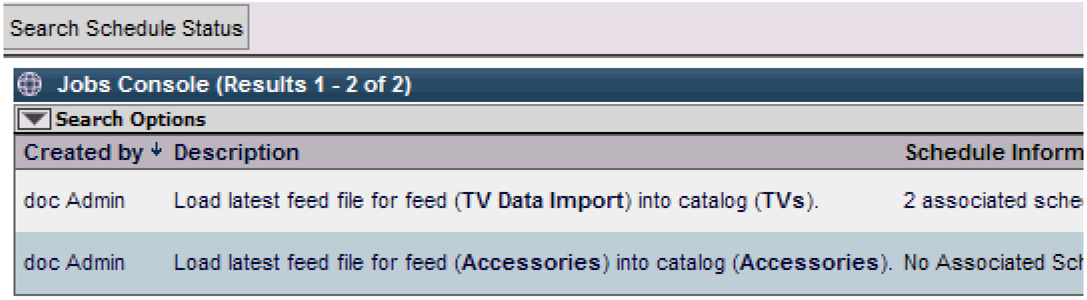
The Scheduler component is used to track the status of activities within WebSphere Product Center (i.e. imports, exports, reports). The Jobs Console provides a unified view to manage all scheduled jobs that can be executed based on a defined timetable and monitored with status information.

Scheduler terminology

Jobs	A job is an import, export, or report that is created in their respective consoles. Multiple scheduled can be identified for a single job.
Schedule	A schedule is defined for a job. Multiple schedules can be created for a single job.
Scheduler	The Scheduler component of the Data Model Manager allows a user to view information on all schedules that are associated to the various jobs that have been created.

Jobs Console

The Jobs console displays jobs that are scheduled to run, which include imports, exports, and reports. Users can disable a job, compare jobs, view the status of a job, or update schedule information.



Created by ↓	Description	Schedule Inform
doc Admin	Load latest feed file for feed (**TV Data Import**) into catalog (**TVs**).	2 associated sche
doc Admin	Load latest feed file for feed (**Accessories**) into catalog (**Accessories**).	No Associated Sch

Figure 13.1 - Jobs Console

Jobs Console columns

The following table lists the columns that are available in the Jobs Console.

Created by	Identifies the creator of the job. This column can be sorted by clicking the arrow next to the column heading name
Description	Description of the job
Schedule Information	Lists the number of associated schedules. Click to view all schedules associated to the job
Action	Functional buttons available for each export job listed

Jobs Console buttons

The following table lists the functional buttons from the Jobs Console screen.

⚙	View the status of the job
+ ADD	Update the schedule information
☒	Disable a job so that it does not execute as scheduled
COMPARE	Compare two different instances of a job or to a different job
Search Schedule Status	Click to access the Schedule Status page to search by the status of scheduled jobs

Accessing the Jobs Console

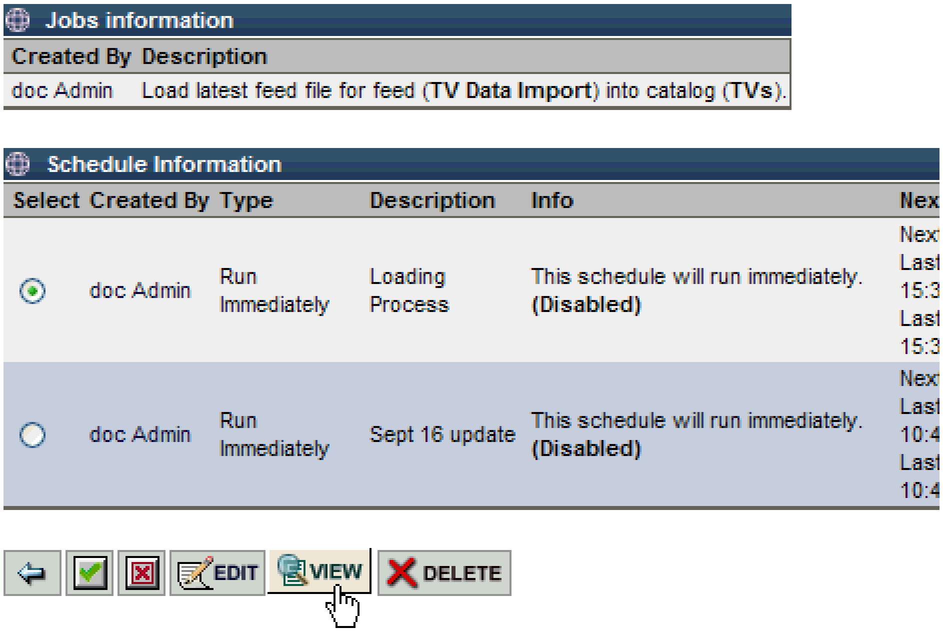
Use the menu path: **Data Model Manager > Scheduler > Jobs Console**. The Jobs Console interface appears.

Viewing job information

From the Jobs Console click on the description for the job to take you to a screen that provides a calendar of the day the job was executed along with additional details related to the job.

Update schedule information

From the Jobs Console, click the associated schedules hyperlink for the job to take you to a screen that provides view additional job information and the ability to update the schedule information for a job.



Jobs information

Created By	Description
doc Admin	Load latest feed file for feed (**TV Data Import**) into catalog (**TVs**).

Schedule Information

Select	Created By	Type	Description	Info	Nex
●	doc Admin	Run Immediately	Loading Process	This schedule will run immediately. **(Disabled)**	Nex Last 15:3 Last 15:3
○	doc Admin	Run Immediately	Sept 16 update	This schedule will run immediately. **(Disabled)**	Nex Last 10:4 Last 10:4

Figure 13.2 - Schedule information

Select a job from the Schedule Information table and click one of the following action icons:

✓	Enable selected schedules
☒	Disable all associated schedules
EDIT	Edit scheduled job
EDIT	View Schedule Status Information
DELETE	Delete scheduled job
COMPARE	Compare schedules that have already been executed

Viewing job status

From the Jobs Console, click the status button from the Action column for a job. The Schedule Status Information table appears displaying detailed information for the selected job.

Comparing Scheduled Jobs

The Compare action allows a user to compare jobs that have already been executed.

1. Select a job from the Jobs Console and click the **Compare** button from the Action column. The "Runs Progress Comparison For Job" table displays with the job statistics for the job.

2. From the table, find the job to compare with and click the [< Compare] hyperlink in the column heading.

3. There are six components that can be used to analyze the performance of a job. Select one of the following action buttons.

PROGRESS CHART	Displays the current status percentage bar, and also two charts to plot progress over time.
PERFORMANCE	An expandable/collapsible table to provide insight into all the operations that were performed during the job run.
DEBUG REPORT	Displays useful debug information for certain job types.
% PROGRESS TABLE	Displays the duration from the start of the job it took to reach each percentage point. The durations are in milliseconds.
COMPARE JOB RUNS	Displays a comparison between different jobs.
COMPARE SCHEDULE RUNS	Expanded versions of the Progress Table. Extra columns for the durations for each job/schedule run are also displayed.

1) Progress Chart: This displays the current status percentage bar, and

also two charts to plot progress over time.

2) Performance: An expandable/collapsible table to provide insight into all the operations that were performed during the job run. If the job is still running, the row with a yellow background was the last updated row (operation last performed during the job run)

3) Debug Report: Displays useful debug information for certain job types. To write messages to this debug report, use the logDebug script operation

4) Progress Table: Displays the duration, in milliseconds, from the start of the job it took to reach each percentage point. Also, displayed on this screen for comparison are the average duration times for the job and for the schedule.

5) Compare With Job Runs: Compare the progress between jobs that have been executed at different times.

6) Compare With Schedule Runs

Displays an expanded version of the Progress Table. There are extra columns for the duration of each scheduled job. If desired, delete the information displayed for this job. Change the focus to a different job by clicking the " Compare" link.

Searching the jobs console

User can search for a job in the Jobs Console using the search feature.

1. From the Jobs Console, click the Search Options button located above the table heading row.

2. Select to search by the "Created by" or "Description" column. Add a search criterion (use * for wildcard) and view the results in a table.

Disabling/stopping a scheduled job

1. From the Jobs Console, click the disable button, located in the Action column. A dialog box appears confirming the job will be disabled.

2. Click **OK** to disable the job, or **Cancel** to continue the job.

Issues with stopping an import in progress

Stopping an import in progress can be somewhat risky at certain points in the import process and could possibly damage a catalog. If a job is stopped at 75% completion, a rollback must be done manually.

If the job has surpassed 75% and the job is killed, extra steps need to be taken to make sure data is not in an inconsistent state. This can be done by first performing a catalog difference with the last version before the import. If data appears in the catalog, a rollback needs to be setup and executed to return the catalog to its previous state.

Entry Processor Status

This component displays the status of all item macros or saves that have completed running, currently running, or items that have caused an error.

Accessing the Entry Processor Status page

Use the menu path: Data Model Manager > Scheduler > Entry Processor Status. The Entry Processor Status Search interface appears.

Search for entry processor status

1. From the Entry Processor Status page, select a value from the Current Status drop-down field.

- ALL
- Completed Running
- Error Completing
- Running

2. Select a value from the Processor Type drop-down field (Item Marco or Item Save).

3. Select a time frame to search from the Date From and Date to fields.

4. Select a user from the Created By drop-down field. The default is the current user.

5. When the search parameters have been selected, click **Search** and the results display in the Entry Processor Runs table.

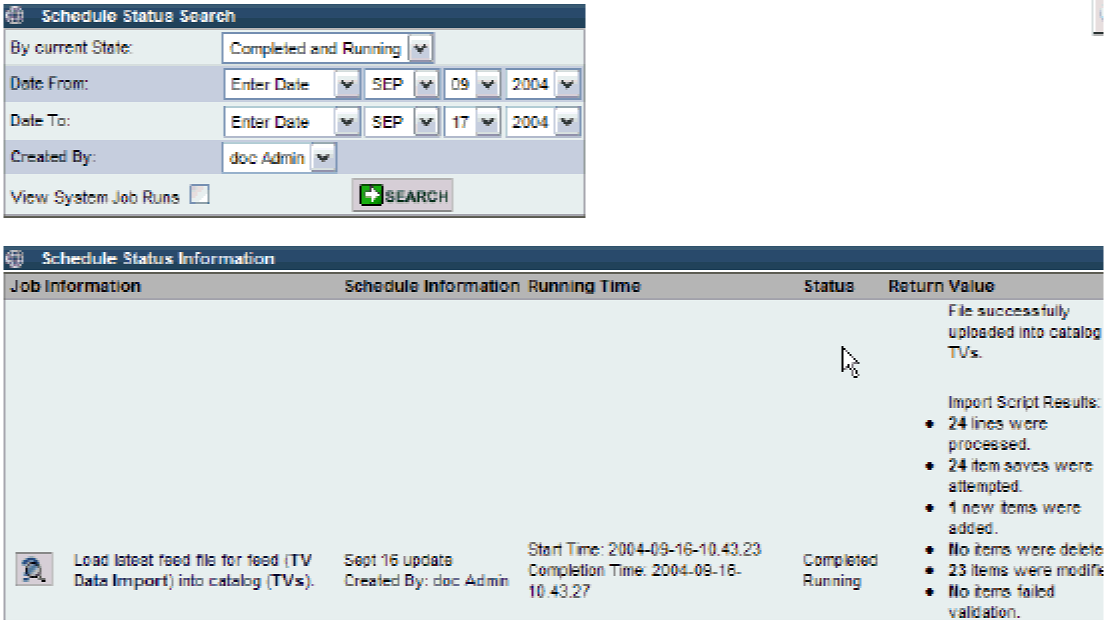
6. To view the error log for any item processor runs, click the view button in the Error Log column.

Schedule Status

Schedule information can be obtained on all job types (import, export, reports), which can also be obtained from the Jobs Console. Within this component, the user has the option to view system job runs.

Accessing the schedule status page

Use the menu path: **Data Model Manager > Scheduler > Schedule Status**. The "Schedule Status Search" and "Schedule Status Information" interfaces appear along with a table of scheduled status information on the most recent jobs executed.



Schedule Status Search

By current State:	Completed and Running
Date From:	Enter Date · SEP · 09 · 2004
Date To:	Enter Date · SEP · 17 · 2004
Created By:	doc Admin
View System Job Runs □	SEARCH

Schedule Status Information

Job Information	Schedule Information	Running Time	Status	Return Value
Load latest feed file for feed (TV Data Import) into catalog (TVs).	Sept 16 update Created By: doc Admin	Start Time: 2004-09-16-10.43.23 Completion Time: 2004-09-16-10.43.27	Completed Running	File successfully uploaded into catalog TVs. Import Script Results: • 24 lines were processed. • 24 item saves were attempted. • 1 new items were added. • No items were delete • 23 items were modifie • No items failed validation.

13.3 Schedule Status page

Searching the status of a schedule

To search the status of a schedule, do the following:

1. From the "Schedule Status" page select one of the following values from the By current State drop-down field:

- Completed and Running
- All
- Completed Running
- Error Completing
- Running
- System Error

2. Select a time frame to search by selecting values from the Date From and Date To drop-down fields.

3. Select a user in the Created By drop-down field.

4. OPTIONAL: Check the View System Job Runs box to include jobs that were executed by the system automatically.

6. Click the Search button and the results of the searched schedules appear in a Schedule Status Information table.

Job approvals

If a job has been generated and awaiting an approval, the job is placed in the approval workflow. The approving authority is notified through the Alerts Module of their My Task List and the generated file can be reviewed before it is approved or rejected.

Approving Scheduled Jobs

1. From the "My Task List" of the approving authority for the job, the

Approval Module displays all of the jobs that have been requested for approval or submitted for approval from a user.

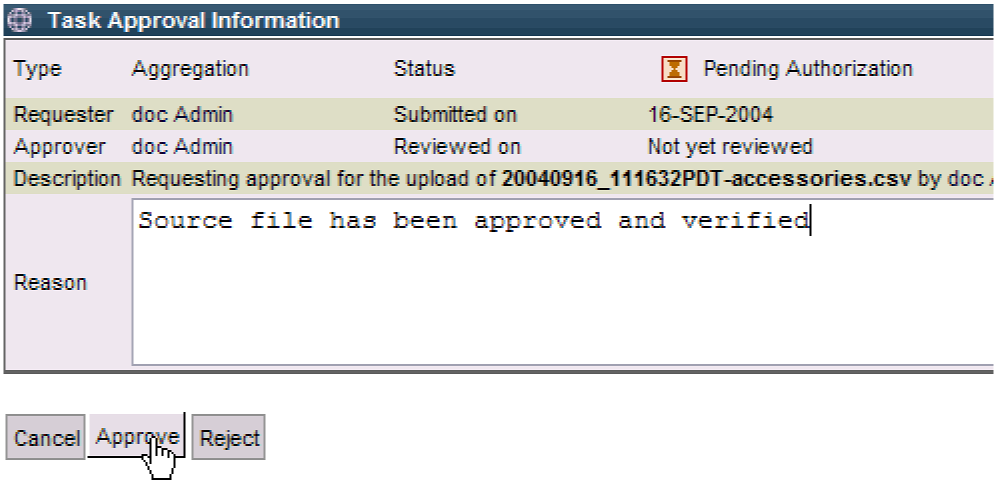
Note: If the Alerts Module is not configured to display, edit the My Task List settings to do so.

The following is a list of authorizing status icons.

✓	Accepted
⌛	Pending Authorization
🚫	Rejected

2. If the job status is in pending status, click on the Pending icon. The Task Approval Information screen appears.

3. Accept or Reject the job by clicking on the associated button. If desired, enter a reason for your decision. The job status is updated in the user's task list and in the associated console for the job (i.e. Import Console).



Task Approval Information

Type	Aggregation	Status	⌛ Pending Authorization
Requester	doc Admin	Submitted on	16-SEP-2004
Approver	doc Admin	Reviewed on	Not yet reviewed
Description	Requesting approval for the upload of **20040916_111632PDT-accessories.csv** by doc		
Reason	Source file has been approved and verified		

Figure 13. 4 – Approving a job

Ch 14 Specs/Mappings

The Specs/Mappings component of the Data Model Manager provides access to the Specs Console and the Spec Map Console.

Specifications ("Specs")

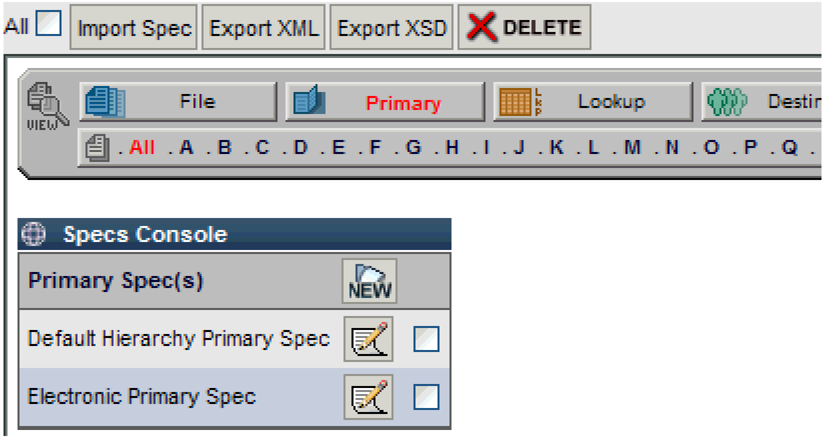
The following is a list of spec characteristics:

- A model for how data is stored, calculated, and managed within WebSphere Product Center
- A model for data as it resides outside of WebSphere Product Center
- Flexible data templates that are used for data validation and can be changed on-the-fly
- Can be updated and maintained by a selected group of users

Specs Console

The Specs Console allows a user to easily navigate and view the following specifications.

- File
- Primary
- Lookup
- Destination
- Secondary Specs
- Script


All □
Import Spec
Export XML
Export XSD
DELETE
VIEW
File
Primary
Lookup
Destir
. All . A . B . C . D . E . F . G . H . I . J . K . L . M . N . O . P . Q .

Specs Console

Primary Spec(s) NEW

Default Hierarchy Primary Spec □

Electronic Primary Spec □

Figure 14.1 - Specs Console

Accessing the Specs Console

To access the Specs Console, use the menu path
Data Model Manager > Specs/Mappings > Specs Console.

Navigating the Specs Console

To navigate between different spec types, use the Spec Navigation bar.

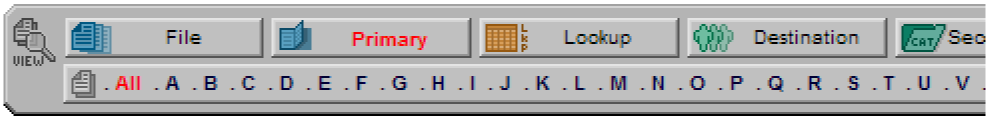

VIEW
File
Primary
Lookup
Destination
Sec
. All . A . B . C . D . E . F . G . H . I . J . K . L . M . N . O . P . Q . R . S . T . U . V .

Figure 14.2 - Spec navigation bar

The top-level buttons can be clicked to view its associated spec type

To search a spec by name for a specific spec type, select a spec type and a letter from the alphabet selection on the navigation bar. All specs starting with the selected letter appear. Click "All" to view a list of all specs for a selected spec type

Spec Types

File	Defines data elements of incoming data
Primary	Defines item attributes, category hierarchy attributes, and organizational hierarchy attributes.

	These specs can be attached to catalogs, category hierarchies and organization hierarchies
Lookup	Defines lookup table record attributes
Destination	Defines data elements at destination
Secondary specs	Secondary specs are used as item hierarchy specs or as standalone specs. The secondary specs are attached to categories. If a secondary spec is attached as a item hierarchy spec, the attributes will be available to all items under the category. If a secondary spec is attached to the category as standalone specs, it will be available only at the category level for a specific category
Script	An input spec that defines attributes to be passed to a script (input parameters)

Spec Management

Creating Specs

To create a spec, select a spec type from the Spec toolbar and click the **New** button located in the spec table heading. Refer to the following sections on creating and defining specs (Attribute Management).

Editing Specs

If desired, a spec can be edited from the Specs Console by clicking on the Edit button . Edit can be made from the spec tree and saved by clicking the Save button at the bottom of the screen.

Add Nodes

When a spec is created, nodes are be added in a treelike structure and parameters are defined for each node, such as the field length and the data type. If needed, additional parameters can be added (i.e. maximum length, required flag, data type, etc.)

Add attributes to a spec

1. To add a node, click the Plus icon on the spec tree.

2. Enter a name for the node and click the Plus icon, located at the end of the entry field. The attribute is added to the Spec Tree and the details box appears. The next step is to define the node parameters.

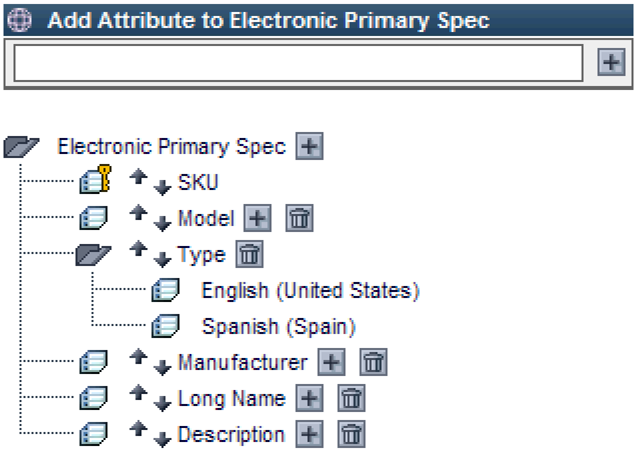
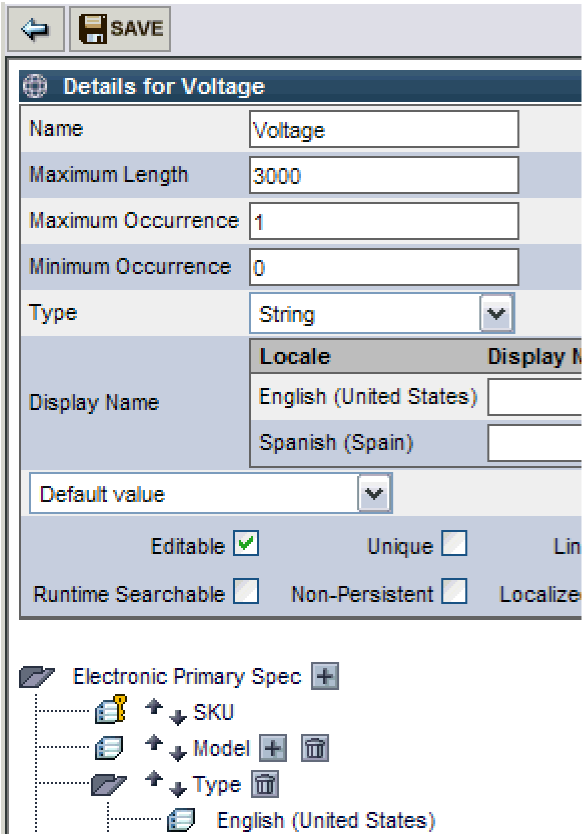


Figure 14.3 - Adding a node

3. To define node parameters, enter information in the Details table and click Save to commit the change.



Name	Voltage
Maximum Length	3000
Maximum Occurrence	1
Minimum Occurrence	0
Type	String

Display Name

Locale	Display N
English (United States)	
Spanish (Spain)	

Default value

Editable ☑ Unique □ Lin

Runtime Searchable □ Non-Persistent □ Localize

Figure 14.4 - Define node parameters

4. To add additional attributes, choose a value from the drop-down selection under the Display Name field, then click the Plus icon. The value is added in the Details box. Enter a value for the attribute and click **Save**.

5. Enter information into the new attribute field and continue adding nodes and node parameters, as needed.

6. When all of the nodes and node attributes are added, click the **Save** button at the bottom of the screen to store the spec.

Note: If the Save button is not clicked, the newly created nodes will not appear in the spec.

Edit or delete nodes

Nodes on a spec tree can be edited or deleted at anytime. Be careful when making any changes to a spec as it will change all the objects that are using the spec. For example, if a node is deleted from a primary spec, any catalogs that use the spec will no longer have the node in the catalog along with their values. This action cannot be reversed.

Editing node

To edit an attribute, click on the node name on the spec tree. Make modifications and click the **Save** button.

Deleting a node

To delete a node, simply find the node in the spec tree structure and click the garbage can icon next to it.

Spec Node parameters

A node parameter can be in any type of spec, a number of parameters may be set for each attribute (node). These parameters define how data is to be calculated, stored and/or validated for that attribute (node).

All specs are created in a tree structure that is built by adding and defining node parameters. This section explains how to add and define spec nodes.

Node parameters

Defining node types are important to create and maintain a standard structure for product information. With multiple users handling the data, a miscue in data entry will produce an error message.

When adding nodes to a spec, they can be defined with the following parameter characteristics:

Data Types	Strings, Numbers (Integer, Decimal, Currency), Enumerations (Number, String), Images (Binary), Dates, Flags, URLs, Groupings, etc.
Editable	An attribute can be made editable or not. If unchecked, the value for this field is viewable and cannot be edited

Unique	Creates a requirement for the node to be unique in a catalog. If a user attempts to enter a duplicate value, an error occurs. Option only available in Primary specs.
Link	Defines a node as the "source attribute" or "foreign key" to the master catalog. Option only available in Primary specs.
Hidden	Defines a node that is not displayed and is useful for nodes that are used as placeholders for intermediate values. When this box is checked, the node is hidden and when unchecked it is displayed. Although this feature hides a node from the item view screens, it is exposed to scripts and searches.
Runtime Searchable	Check this box to include the node for runtime searches. If the box is not checked then it is available to background searches. Use this option for common searchable nodes.
Non-Persistent	Provides the ability to make a node not constant
Localized	Set attribute for localization. Company must be set with the desired locales.

Node parameter limitation

There is a limit to the number of characters in a value/validation/string-enumeration rule. It can be at most 3000 bytes, which in most cases translates to 3000 characters (including spaces). It can be less than that depending on the characters used e.g. Japanese characters etc. can be more than one byte long.

Adding scripts

Each node within a spec is associated to a node of any type. Therefore, scripts can be created to perform rules for a node.

- Validation rule
- Value rule
- String or number enumeration rule

Node data types

Each node within a spec is associated with a data type, which controls the following:

- Display properties in the user interface
- Storage in the database
- Presentation in an export
- Search capabilities
- Implicit validation on Import/ Export/ Data Entry

The available data types differ between each spec type. This section lists the different data types available in WebSphere Product Center.

Binary Function: Store binary data, i.e. PDF files, Image files. Available in Specs: All Rules associated: None After an attribute has been defined as Binary, the catalog attribute can be used to upload a binary file into WebSphere Product Center. Click the Edit icon from the Item Edit screen and the Upload Image screen appears.
Currency Function: Define currency fields. By default, the value is rounded to the second decimal place Available in Specs: All Rules associated: Default value, Minimum length, Pattern, Validation rule, Value Rule When adding a value to the attribute, it is rounded to the second decimal point.
Date Function: Define date fields Available in Specs: All Rules associated: Date format, Default value, Minimum length, Validation rule, Value rule Apply the spec to an object and the date format appears in the format as defined. Select from a set of pre-defined date values or enter a date manually.
Flag Function: Holds one of two states, true or false Available in Specs: All

Rules associated: Default value, Validation rule, Value rule

A checkbox is created for selection. An empty box could be searched for as a null value for the node.

Image

Function: Used to store images
Available in Specs: All
Rules associated: None

After a node has been defined as "Image", the catalog attribute can be used to upload an image file into WebSphere Product Center. Click the Edit icon and the Upload Image screen appears.

Image URL

Function: Used to store a URL location of an image. WebSphere Product Center fetches the URL from the defined location
Available in Specs: All
Rules associated: Default value, Minimum length, Patter, Validation rule, Value rule

In the catalog item view, a URL for a graphic can be entered. Click on the Preview button to view the image.

Integer

Function: Used to store only integer numbers. Numbers with decimals are rounded to form an integer.
Available in Specs: All
Rules associated: None

Any of the natural numbers, the negatives of these numbers, or zero must be entered for this attribute. If an integer is not entered for this attribute, an error occurs.

Lookup table

Function: Associate to a lookup table, which provides a drop-down selection of lookup tables that have been created
Available in Specs: All
Rules associated: Value rule

Number

Function: Used to store double numbers, i.e. number which can hold decimal values. (For example, 10.98)
Available in Specs: All
Rules associated: Default value, Minimum length, Pattern, Validation rule, Value rule

A number is defined to allow a number value with a decimal value. If no decimal value is given, it given a value of ".0". Negative values are allowed.

Number Enumeration

Function: Used to create a list of number data types. Negative and decimal values are allowed.
Available in Specs: All
Rules associated: Number enumeration rule

1. Set the data type to "Number Enumeration".

2. The details table adds a Number enumeration row. Click the "CLICK HERE" hyperlink and a separate dialog box appears.

3. Add a number in the Spec Enum Details table and click the Plus icon.

4. Continue to add number values to the enumeration and click Close Window when finished.

5. The values entered are displayed in a drop-down selection format and are listed in ascending order.

Password

Function: Refers to an alphanumeric field which hides the contents to the user
Available in Specs: All
Rules associated: Default value, Minimum length, Pattern, Validation rule, Value rule

When entering a value, it is displayed with asterisks. This is useful when values are to be hidden.

Period

Function: Creates two Date fields, Start Date and End Date Available in Specs: All Rules associated: Default value, Minimum length, Validation rule, Value rule The GUI creates a field with a Start and End Date value.
Relationship Function: Used to hold a link to another item in the same or another catalog Available in Specs: Catalog Spec Rules associated: Default Value, Validation rule, Value rule The GUI allows a user to select a catalog and a key attribute to link two catalogs. The Key value allows for a link relationship to be established.
Sequence Function: Used to create a numbered sequence field Available in Specs: Catalog, Lookup table, Category, Hierarchy Rules associated: Default value, Increment sequence by, Default value for sequence start, Minimum length, Validation rule, Value rule Set a rule associated to the data type, i.e. set the Default value for sequence start at "1". When a new item is added, the sequence field automatically enters "2". This field is not editable to the user.
String Function: Holds a string or character data Available in Specs: All Rules associated: Default value, Minimum length, Pattern, Validation rule, Value rule If desired, create an associated rule for the attribute. The GUI displays an empty field for data entry.
String Enumeration Function: Create a string enumeration data type that holds a list of string data types Available in Specs: All Rules associated: Default value, Minimum length, Pattern, String enumeration rule, Validation rule, Value rule

Associate a rule with the data type, which displays all values in a drop-down selection field.
Thumbnail image Function: Stores a thumbnail image Available in Specs: All Rules associated: Default value A user is able to load a thumbnail image and view it from the GUI.
Thumbnail Image URL Function: URL link to a thumbnail image. WebSphere Product Center will fetch the image from a defined URL. Available in Specs: All Rules associated: Default value, Value rule The field allows for an address to be entered for the Thumbnail Image URL. Set a Maximum length for the attribute.
URL Function: Define a URL field and appears as a link on the item list and item detail screens Available in Specs: All Rules associated: Default value, Minimum length, Pattern, Validation rule, Value rule The field allows for an address entered for a URL.

Setting node rules

Once a data type is selected, additional rules can be added to further define the characteristics for an attribute.

Default Value Function: Define a default value for an attribute. Available in Specs: All
Default value for sequence start Function: Define a default value for a sequence.

Available in Specs: All
Rules associated: Sequence

Help URL

Function: Define a Help URL; used to customize help.
Available in Specs: All

Although WebSphere Product Center has general help topics available for application assistance, the ability to have an attribute link to a URL can be utilized to create customized help topics.

Increment sequence by

Function: To increment a sequence by a defined value
Available in Specs: All
Rules associated: Sequence

Lookup Table

Function: Defined an associated lookup table. No value is available if no lookup tables exist.
Available in Specs: All
Rules associated: Lookup Table

Minimum Length

By default, the Maximum Length is provided. To set the node to a minimum length is must be defined explicitly from the attributes details screen.

1. A minimum length can be defined by selecting "Minimum Length" and click the Plus button.

2. Enter a value in the Minimum Length field.

3. Click save to update the spec.

Number Enumeration

Function: To defined a enumerated number attribute:

1. Select Number Enumeration Rule and click +, the Number Ennumeration field appears.

2. Click "CLICK HERE" to access the pop up window used to create a number

enumeration..

3. After creating a number enumeration, click Close.

Occurrences To Display

Function: To defined the number of occurrences that are displayed, if an attribute is defined as multi-occurrence:

1. Select the Occurrences To Display Rule and click +.

2. Enter a value in the Occurrences To Display field.

Pattern (regular expression)

To defined an attribute with a pattern for a regular expression.

1. Select the Pattern (regular expression) Rule and click +.

2. Enter a pattern for a regular expression.

String Enumeration

Function: Create a string enumeration data type that holds a list of string data types
Available in Specs: All
Rules associated: Default value, Minimum length, Pattern, String enumeration rule, Validation rule, Value rule

1. Set the data type to "String Enumeration".

2. Click "CLICK HERE" for a pop up window used to create a string enumeration

String Enumeration Rule

Function: Create a string enumeration rule for data type that holds a list of string data types
Available in Specs: All
Rules associated: Default value, Minimum length, Pattern, String enumeration rule, Validation rule, Value rule

1. Set the data type to "String Enumeration Rule".

2. Click "CLICK HERE" for a pop up window used create a string enumeration rule.
Validation Rules Function: To defined a validation rule for a node. 1. Select Validation Rule and click +, the Validation Rule field appears. 2. Click "CLICK HERE" to access the Validation Rule Editor. 3. Create a Validation Rule for the attribute and click Save.
Value Rule Function: Create a value rule for the node

Grouped nodes

Nodes can be grouped on any spec tree. When grouping nodes, it is important to create all major groupings before adding sub-nodes to each group. Use the Single Edit screen to view grouped nodes as the multi-edit screen will not appear in the Multiple Edit screen.

Creating grouped nodes

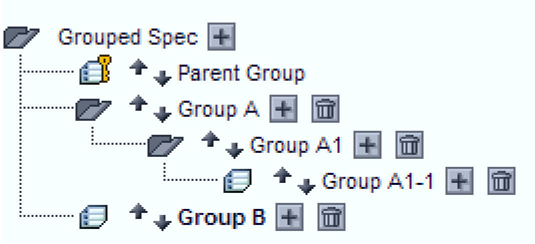


Figure 14.5 - Grouped nodes

The following steps create an example of grouped nodes.

1. When creating a spec, click the root node and add the first level group node (i.e. Group A).

2. Click the "Group A" attribute to add a sub-node, "Group A1"

3. Click the "Group A1" node to add a sub- attribute, "Group A1-1". The

spec tree appears with three group levels.

4. Add a node under any of the grouped nodes. Before adding sub-nodes, all parent level groupings must be created before adding sub-nodes.

Note: When grouped nodes have been created, use the Single Edit screen to display groups as they do not appear in the Multiple Edit screen.

File Specs

A file spec is created to define the file structure of an external data source. The file spec is needed for the importation of a source file into WebSphere Product Center. Analyze the structure of the data source and create a file spec that emulates the structure.

If the data source changes at a later time, the file spec can always be modified to reflect the changes.

Creating a File Spec

- From the Specs Console, select **File** from the spec navigation bar, the Specs Console appears with a list of previously created file specs, if any.
- Find the Files Spec(s) column heading and click **New**. The Spec Tree interface appears.
- Complete each step of the wizard driven GUI.

1. Select Spec Type

By default, File Spec displays in this step.

2. Enter File Spec Name

Enter a unique name for the new file specification. Choose an intuitive name for easier retrieval. The system will not allow duplicate File Spec names.

3. Specify File Format

Select one of the following formats and click the **Select** button:

- Character Delimited - Used for a file that uses a custom character to separate the fields in the incoming file. For example, a file could use the character # to separate the fields
- Tab Delimited - Used for a file that uses the TAB character to separate the fields in the incoming file.
- Comma Separated Values (CSV) - Used for a file that uses the comma to separate the fields in the incoming file.
- Fixed Width - Used for a file that uses a constant space for all fields in the incoming file. Such a file is easily recognizable as all fields are nicely aligned in columns.
- XML - Used for a file that uses the XML format.
- line(s) at the tope of the file. - enter a value for the number of lines to ignore in the heder.

The file spec is created and nodes can be added as needed. When all nodes have been added, click **Save** to store the file spec.

Primary Specs

A Primary Spec is needed in order to configure the format of a WebSphere Product Center catalog structure. The primary spec is mapped to a file spec, defining how information should be routed from data source to the catalog. Primary specs are also used to define hierarchies associated to catalogs.

Creating a Primary spec

- From the Specs Console, select **Primary** from the spec navigation bar, the Specs Console appears with a list of previously created primary specs, if any.
- Find the Primary Spec(s) column heading and click **New**. The Spec Tree interface appears.
- Complete each step of the wizard driven GUI.

1. Select Spec Type - By default, File Spec displays in this step.

2. Select Name - Enter a unique name for the primary spec and click **Next** to display the spec tree.

3. Define the primary spec with nodes defined with parameters as needed.

4. Set Primary Key Field - A unique identifier is required when creating a catalog spec. Select a node as the primary key by clicking on the checkbox. A primary key must be identified before the spec can be saved.

5. When all attributes have been added, click **Save** to store the spec.

Note: Once a primary key is selected and the spec has been saved, the primary key cannot be changed. If a new primary key needed, create a new spec.

Hierarchy Specs

Hierarchies (or Taxonomies) are used in WebSphere Product Center with catalogs as a way to classify the items that are stored within. It can be compared to how library books are placed in certain areas of the building, such as "Non-Fiction", "Reference" and "Periodicals." The books are classified in a way that makes them easier to browse, both for the library patrons and for the librarians.

In the same way that a librarian uses this categorization scheme, WebSphere Product Center provides users with the tools to build and modify multiple hierarchies to help them organize the items stored within their catalogs.

Before hierarchies can be created, it must have an associated Primary Spec. Once a Primary Spec has been created, a hierarchy can be built using the Hierarchy Console.

Create Hierarchy Spec

Creating a hierarchy spec is the same as creating a primary spec. Use the same instructions as creating a primary spec.

Build Destination Spec

When an export job is created in WebSphere Product Center, a destination spec is created to define the exact requirements of the destination file. Similar to a file spec or primary spec, each attribute of the destination spec is defined. During the exportation of data, the destination spec generates a file that adheres to a set of predefined requirements.

A set of pre-defined destination specs is available (i.e. Ariba, Yahoo Shopping, Commerce One) and is not editable.

Create Destination Spec

- From the Specs Console, select **Destination** from the spec navigation bar, the Specs Console appears with a list of previously created destination specs, if any.
- Find the Destination Spec(s) column heading and click **New**. The Spec Tree interface appears.
- Complete each step of the wizard driven GUI.

1. Select Spec Type - By default, Destination Spec displays in this step.

2. Select Name - Enter a unique name for the destination spec and click **Next** to display the spec tree.

3. Define the destination spec with nodes defined with parameters as needed.

4. When all attributes have been added, click **Save** to store the spec.

When creating an export, a primary spec can be mapped to the destination spec. All available destination specs are displayed when creating a Spec Map or an Export job.

Spec imports and exports

Any spec can be imported and exported from the Specs Console. Specs can be exported in XML or XSD formats and can be imported into another WebSphere Product Center instance. The export feature is a great way to back up all specs so they do not have to be recreated from scratch.

Importing a spec

1. Select a spec type from the spec navigation bar and click on the **Import** button. The "Upload of a Spec XML or XSD" appears in a separate window. To import all the currently displayed specs appearing in the Spec Console, click the All checkbox.

Note: The spec type of the imported spec file must be selected. If Primary is selected in the Specs Console and the user attempts to import a File Spec, an error occurs.

2. If selecting a XML Document, click Browse and select a file to import then click Upload. The document is stored in the Document Store.

3. If selecting an XSD Schema Definition, the "XSD Options" section appears. Enter the required information and click **Upload**.

4. The imported spec appears in the Specs Console. If an error occurs, it is possible that the imported file type does not match that of the Spec Console.

Exporting a spec

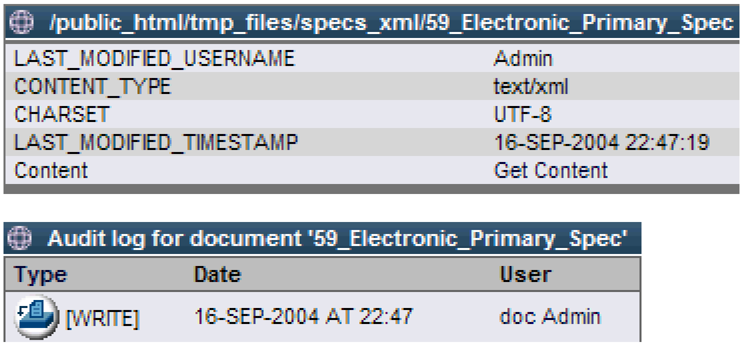
Select a spec type from the spec navigation bar and select a spec to export from the Specs Console. To export all the currently displayed specs appearing in the Spec Console, click the All checkbox.

1. To export a spec, simple check the box next to the spec in the Specs Console and click on the Export XML or Export XSD button. The Specs Export Results interface appears.

2. A system-generated name is given to the exported spec and is stored in the Docstore. A new window appears providing details of the spec export results.

Note: If exporting to file type XML, it is saved to the docstore. If the same file is exported to file type XSD, it will overwrite the previous XML file and replaces it with the XSD file.

3. Click on the export file name hyperlink in the "Spec Export Results" window to view the spec file details. View the last modified information, view content link, and the audit log for the exported file.



/public_html/tmp_files/specs_xml/59_Electronic_Primary_Spec

LAST_MODIFIED_USERNAME	Admin
CONTENT_TYPE	text/xml
CHARSET	UTF-8
LAST_MODIFIED_TIMESTAMP	16-SEP-2004 22:47:19
Content	Get Content

Audit log for document '59_Electronic_Primary_Spec'

Type	Date	User
[WRITE]	16-SEP-2004 AT 22:47	doc Admin

Figure 14.6 - Spec export results

4. Click the "this link to docstore" hyperlink to navigate to the Docstore directory where the file was saved.

Spec Maps

Spec Maps are created to define how information from one source is going to route to another source. For example, a catalog spec, which defines the fields of a catalog, is mapped to a destination spec, which defines the fields of a destination. Therefore, when the catalog is exported using the Catalog spec to Destination Map, the information is routed in the proper structure.

Spec Map Console

The Mapping Console displays all of the previously created maps of the following types:

- Catalog Spec to Destination
- Catalog to Catalog

- Catalog to Destination Spec
- File Spec to Catalog
- File Spec to Catalog Spec

The following are the icons in the Mapping Console.

✕	Delete mapping
✎	Edit mapping
🔍	View mapping
New Map	Create a new mapping

The following are the columns in the Mapping Console

Type	Type of mapping
Name	User defined name for the map
Source	The source spec used
Destination	The destination spec used

Accessing the Spec Map Console

To access the Specs Console, use the menu path **Data Model Manager > Specs/Mappings > Specs Maps > Spec Map Console**.

Creating Spec Map

All specs maps are created the same way. This section describes how to create a catalog to destination spec map.

The Catalog to Destination Spec map is similar to the File to Catalog Map in that it instructs WebSphere Product Center with how the fields in the catalog should be routed during an export. The fields in a catalog are mapped to the fields of a destination file. This mapping is used when defining an export of the catalog to a destination.

- From the Spec Map Console, select Catalog to Destination Spec from the drop-down field and click the **New Map** button. The Source to Destination Mapping wizard appears.

- Complete each step of the wizard driven GUI.

1. Select spec map type - This defaults to the type of spec selected from the Spec Map Console

2. Select Catalog Source - Select the catalog that is to be exported and click **Select**.

3. Select Destination Spec - Select the destination spec that is to be used and click Next, the Source Mapping table appears.

4. Enter name for the spec map in the Enter map name field.

5. Select a node from the Selected Source Attribute drop-down field and click **+ADD** to map it to a destination attribute.

6. To add an expression to the mapped attributes, click the expression button. The "Scriptlet Editor" interface appears.

7. When all spec nodes have been added, click **Save**.

Ch 15 Attribute Collections

The implementation of Attribution Collections increases the efficiency of data model management, performance and becomes very apparent when working with a large number of attributes (500+). Attribute Collections were introduced to simplify the management of large sets of attributes. Instead of working with an entire attribute group, it is possible to work on a functional subset of attributes.

The subset of attributes can be used to create views, tabs, workflows, inheritance rules, privileges, etc. Associating a subset is more efficient than associating individual attributes.

With the use of Attribute Collections, data is modeled in a more efficient and organized manner. Instead of managing a large amount of attributes for an item, managing them in subsets creates a more manageable dataset.

Attribute Collections were designed to improve performance when fetching and saving an item category where only the attribute selected for a view are retrieved and saved. Limiting the fetching and saving to the attributes in the view could cause issues by saving invalid items. Therefore, core attributes were introduced.

There are two types of core attributes:

- Default core attributes
- User define core attributes

When fetching and saving a record, the attributes included come from the default core attributes, the user defined core attributes and the superset of the attributes included in the single edit and multiple edit views

Default Core Attributes are system defined attributes that are retrieved and saved for every object and include only the attributes that are critical to making sure an item is not saved to the database in violation of key rules; this includes:

- The primary key
- The path attribute (for categories only)
- Any required attributes (either from a primary or a secondary spec)

For other attributes, it is not possible for the system to determine if they are needed or not. In some cases, some validation errors should run for every item, and therefore should be included in the core attributes. For this reason, user-defined core attributes can be added to the total set of core attributes. These attributes are included in an Attribute Collection. For each catalog and category tree, it is possible to associate an Attribute Collection as the user defined core attributes.

User Defined Core Attributes: attributes that are required when an attributes needs to be retrieved and saved for every object; these core attributes are defined per container and could include attributes that need to be validated or calculated every time an item or category is saved; one set of user defined core attributes will be associated per container. When creating user defined core attribute collections, include any required attributes located in secondary specs.

Note: It is recommended for the user defined core attributes to be kept at a minimum to achieve optimal performance. Therefore, remove any unnecessary attributes if they are not needed.

What is an Attribute Collection?

An attribute collection is a group of attributes that will be associated or behave the same way in a given context.

For example, a subset of attributes can be created for an electronic product catalog with an attribute set for features. This section can have multiple feature "types" (technical, marketing, etc.) Therefore, an attribute subset can be created for "technical" attributes and "marketing" attributes.

When defining an Attribute Collection, there are two types available for definition:

- **General** – used for a collection of attributes that are not associated with inherited attributes (i.e. views, tabs, workflow, etc.). Selecting this type allows the selection of primary or secondary spec attributes.
- **Inheritance** – used for a collection of attributes that are used to define inheritance rules. Selecting this type allows the selection of sub spec attributes only.

When defining an Attribute Collection, if it is known that the attributes will not be inherited, then select the type "General", otherwise select "Inheritance".

Note: For type "Inheritance", refer to the section "Attribute Collections for Inheritance Rules"

What is the difference between an attribute group, a spec and a subspec?

Subspec	**Primary/Secondary Spec**	**Attribute Collections**
A sub-spec is a collection of attributes that could be edited at different levels in the data model. Sub specs are used for inheritance purposes.	Primary specs and secondary specs are full or partial item or category templates. They define the object that will belong to it. They can be made of sub-spec attributes or individual attributes Note: What was formerly known as hierarchy specs and catalog specs have been replaced by Primary Specs.	Attribute Collections are a group of specs and attributes that will behave the same way in all contexts

Where are attribute collections used?

When the association of an attribute is made to an Attribute Collection, the attribute is made available wherever the Attribute Collection is used. Attribute Collections are used to define the following:

- Access privileges
- Views
- Tabs
- Workflow steps
- Inheritance rules

When are attribute collections created?

An Attribute Collection is defined as a group of attributes that will be associated or behave the same way in a given context. Specifically, a new attribute collection is created to group attributes that share the same characteristics for the following purposes:

- Security – Create a collection of attributes that is used to setup Access Privileges, thus following the same security guidelines (i.e. create a group for all the merchandising attributes that should be edited only by the merchandising users)
- Function – Create a collection of attributes that will be edited or viewed together, as part of the same process, or step; these groups will be used to set up Views, Tabs and Workflow steps (for example, create a group for all the attributes needed for the creation of an item by a supplier in a supplier product introduction workflow)
- Behavior – Create a collection of attributes that inherit their content following the same path (for example, create a group for all marketing attributes that are used for a worldwide catalog and are inherited by a regional catalog)

What is the attribute picker?

The attribute picker is a portion of the attribute collection edit screen that is used to search and select specs and/or nodes to add to an attribute collection. It is planned for this feature to be used in other areas of the application in a future release.

What changes have been made to the GUI to support Attribute Groups?

Several changes were made to the GUI to support Attribute groups and are identified in the following sections.

Attribute Collection Setup/Console

Attribute Collections are organized in a typical WebSphere Product Center Console format, thus the following standard characteristics apply:

- Console displays Attribute Collection names and descriptions
- Sort names and descriptions by clicking on the column headings
- Console paginates when more than twenty groups exist
- Configure Console to sort and hide columns
- Available buttons to Add, Delete, or Edit an Attribute Collection
- Attribute Collection names are hyperlinked, click to edit a collection
- Click on the checkbox next to an Attribute Collection to delete, then click the Delete button
- Attribute Collections cannot be deleted if the group is associated with a view, a tab, an access privilege, an inheritance rule, or workflow. If this is the case, deletion is not permitted and if attempted, a pop up message indicates where the group is associated

Attribute Collection Console

The Attribute Collection Console is accessed using the menu bar options:

- Console: **Data Model Manager > Attribute Collections > Attribute Collection Console**
- Create New Attribute Collection: **Data Model Manager > Attribute Collections > New Attribute Collection**

Attribute Collection Screen

The attribute screen is used to create a new attribute collection or edit an existing one. The screen is segmented into three parts:

- Attribute Collection – Enter Attribute Collection information, display specs and attributes that are associated to the attribute collection
- Attribute Picker - Search for specs or attributes
- Results – View the search results for specs or attributes and make selection for attribute collection

Attribute Collection Information

- Each Attribute Collection requires a name. This can be modified at any time
- A description is optional and can have up to 2000 characters
- Locales can be selected for the given collection; if needed, locales can be added/removed after the collection has been set-up. The implication of selecting a locale means that any localized attributes added to the spec already in the collection will be part of the collection only if they belong to the selected locale.
- The attribute collection type can be either "Inheritance" or "General"
- When creating an attribute collection, click Save in order to display the bottom portion of the screen that allows for searching and adding individual specs, individual attributes or localized node parents. Adding a spec will not add the individual attributes of a spec, rather all the attributes in the spec at the moment the group is called upon will be considered part of the group. Adding individual attributes creates a static list of attributes. It is also possible to add a grouping node when a node is localized.

Attribute Associations to Attribute Collections

View Attribute Collection Association

- To view the Attribute Group(s) that an attribute is associated with, click on the icon next to "Attribute Collection Associations" from the attribute's Detail screen. The "Attribute Collection Associations" screen is displayed.

- Click to view Attribute Group association
- Attribute Collection association screen
- Edit Attribute Collection Associations
- Removing attributes or specs from an attribute collection is performed in the Attribute ollection definition screen.

Note: Generated Default Core Attributes cannot be edited, as they are system generated.

Attribute Picker

The Attribute Picker screen is new to 4.2 and is used to define attribute collections. In future releases, it is planned for this feature to be used throughout the application to select attributes.

- Specs are displayed alphabetically
- Attributes are presented in the order they were setup in the spec

Searching for an Attribute

The search feature in the Attribute Picker was designed to reduce the need to browse through longs lists of attributes. Several search methods are available to the user and are located in the Search section of the Attribute Picker. All results are displayed at the bottom of the screen.

Search by field	A user can search using one of the two following fields: • A Spec Name field where a portion of a spec name can be entered • An Attribute Path field where a portion of an attribute name can be entered
Search Type	• Specs and Attributes - Display results of Specs and Attributes • Specs Only – Display results of Specs
Search by Spec Type	Select a Spec Type to search for • Search by Locale(s) • Select to search by Locale

Results:

The results will either present specs only or specs and attributes depending on the option selected

A locale selection list presents a list of the locales for this company where multiple locales can be selected

A file spec list that allows selecting one or multiple attributes; the list of spec type will be filtered depending on the context (for example, to create a group of type inheritance, the only spec type available should be "subspec")

When a locale is selected and the Spec Only option is selected, the specs that are listed are the spec that have that locale associated with the spec, regardless of whether there are any nodes localized or not

When a locale is selected and the spec and attribute option is selected, the specs that are listed are the specs that have that locale associated with the spec and have localized nodes for the selected locales

The list displays up to 50-100 results per page, with the ability to navigate back and forth between pages. For each page, up to 20-50 attributes are displayed, with a scroll bar to view the rest of them. Going from one page to the next will lose selected attributes. The list will present a list of specs, leaf nodes and localized nodes. Localized node should have a special indication next to them.

Selecting Specs and Attributes

After a search is performed for a spec and/or attribute, the results are displayed in the results section of the Attribute Picker. Select from the list of results and click Selected, or just click All to select the complete list of results. It is possible to also right click on the spec and select "Add Spec" from the short menu.

Localized parent nodes can be selected for an attribute collection.

Attribute Collection for catalog access privileges and views

This section describes how to associate an Attribute Collection to Catalog Access Privileges and Views

Access Privilege Setup

- Previously, access privileges were setup by selecting individual attributes., simply select an Attribute Collection
- Use the search field if there exists a large list of Attribute Collections
- Only Attribute Collections of type "General" is available for selection for access privileges

Setting up Views

Similar to catalog access privileges, Views are created by selecting Attribute Collections, rather than individual attributes. The following are general characteristics of creating views:

- Attributes Groups are ordered, which determine the order in which the specs are displayed.
- The order of the attributes is dependent on their order in the Primary or Secondary Spec to which they belong

Note: Only Attribute Collections of type "General" are available for selection.

Tab Views

Previously, it was necessary to select individual attributes to create Tab Views. This can be can be a huge task when working with items with many attributes. With the introduction of Attribute Collections, the user simply selects from a list of attribute collections to create a Tab View. Any changes to the Attribute Collection will be reflected in the Tab View.

Note: If changes were made to an Attribute Collection, it is necessary to log out and log back in for the changes to take effect in the Tab Views.

If an entire Attribute Collection is not needed, it is possible to select individual attributes as needed.

Ch 16 Scripting

WebSphere Product Center's Scripting Engine allows for extremely sophisticated data manipulations during the import or export of information to and from WebSphere Product Center. With this added flexibility to your product information management capabilities, users can do the following:

- Apply business rules to standardize data
- Define calculated fields
- Run custom reports
- Perform rules-based cleansing of data
- Create validation rules

Scripts Console

Using the Scripts Console, users can view, create, and edit scripts of the following types:

Catalog Differences Export	The Catalog Differences Export script allows the update of product information to an external destination. The script does not provide a full update, but only the changes that have been made since the last version of the catalog. For example, if the images for a line of products have changed, these changes can be updated without having to update the entire catalog.
Catalog Export	Catalog Export scripts are used during syndication. They can be used to perform advanced on-the-fly operations on the data held in the catalog before it is actually exported to an output file. The modifications made to the content through the scripting engine at the time of syndication are not applied to the catalog but rather simply applied to the output file, as a one-time content modification. Similar to aggregation, the syndication to an external target file can take two forms: either the fields in the catalog map

	on a one-to-one basis to the external target file, or the fields in the catalog require some modifications before they are exported into the external target file. All syndications require the use of a script. Contrary to aggregation, selecting a script during syndication is a mandatory step.
Catalog Import	Catalog Import scripts are used during aggregation and can be used to perform advanced operations on incoming data before it is imported into a catalog. Simple scripts are generated from WebSphere Product Center, with no customization, but can be user specific with as many modifications as necessary. The aggregation of an external file is taken in one of two types: • Mapped on a one-to-one basis to the fields of a catalog • Modifications performed before being imported into a catalog
Catalog Preview Script	A Catalog Preview script is used to create a user-defined preview of a catalog. The script defines how the catalog is displayed.
Catalog Script	A Catalog script is a sequence of operations that a user specifies to be run at the time of item creation and edit. This function provides another layer of functionality over the attribute level operations available via catalog specs.
Catalog to catalog export script	A Catalog-to-Catalog Export script allows the automation of an export where information from one catalog is exported into another catalog.
Custom tools	Create a script to work with custom tools. Functions related to the custom tools can be created.
Distribution script	Distributions scripts are used to create a custom distribution that is not addressed by the built-in WebSphere Product Center distributions (i.e. Ariba Catalog Upload, FTP, HTTP POST and email.)
Entry build script	The Entry Build script allows a user to execute a script within the data entry screens. For example, a script could be written

	to replace all strings with a certain value.
Entry Macro script	The Entry Macro script allows a user to execute a script within the data entry screens. For example, a script could be written to replace all strings with a certain value.
Entry Preview script	The entry preview script allows a user to create a sample view of a current item set, which can be executed from the data entry screens. For example, a script could be written to view how an item would display using an XML format.
Hierarchy Import script	This function is used during aggregation. Although a user can create a hierarchy manually, the Hierarchy Import script allows the building of a full hierarchy out of an incoming flat file.
Hierarchy script	The Hierarchy Script allows a user to build a hierarchy without having to manually create one.
Images and Binary Files Differences Export script	Images and Binary Files Export script take catalog images or binary type files and exports them through syndication. With the Images and Binary Files Differences Export script, a user can export changes that have been made from a previous catalog version.
Images and Binary Files Export script	Images and Binary Files Export script take catalog images or binary type files and exports them through syndication. In most cases, images and binary files are handled differently; therefore the scripts allow these types of files to be exported based on the external systems requirement.
Lookup Table Import script	Lookup table scripts are very similar to aggregation scripts in that they are used to parse an incoming text file. When triggered through the lookup table interface, they are used to populate the contents of a lookup table instead of a catalog.
Order export script	Create a script to execute order distributions.
Order import script	Create a script to execute order imports.
Order status update import script	Used to create a status update on order imports.
Queue message processor	Create scripts to process queue messages created in Websphere Product Center
Report script	Report scripts are used to create custom reports. When creating a report in WebSphere Product Center, a script is

	needed to define the report output. The report script is used to define how the information is ordered and formatted.
Secure Trigger script	Similar to a regular trigger script with additional security.
Trigger script	Trigger scripts are created to avoid the need to populate the same script operations in multiple places. The scripts are stored in the Document Store and can be called from another script function. Used to externally trigger events, i.e. import, exports, etc., in WebSphere Product Center.
Web Service Implementation script	Create a script to implement web services
Workflow step	This script is used to automatically create workflow steps, which can then be viewed using the Workflow Console.

Creating New Scripts

To create new scripts do the following:

1. Click **Data Model Manager > Scripting > Scripts Console**. The Scripts Console appears with a user-friendly navigation bar.

2. From the navigation bar, select a type of script to create. The script console appears with a list of associated scripts. For example, if "Catalog Export Scripts" was selected, a list of created catalog export scripts appears, if any.

3. To create a new script of the selected type, click the **New** button, from the Scripts Console interface. A wizard-driven GUI appears.

4. Follow each step of the wizard. In step four, the script editor appears. This is where script is created.

5. When the script is complete, click the **Save** at the bottom of the screen.

Script Sandbox

The Script Sandbox provides an expression builder with a library of available script operations with a prototype and description (The Prototypes and Description fields do not appear until "Script Operations" is selected.

Users can create sample scripts, which can then be run and tested to see if the results are valid. Click "Run Script" to compile and return an expected value or error, which is viewable to the user. This is a great way to test a script before it is implemented.

Ch 17 - Security

The security menu selection is located under the Data Model Manager. It includes the following menu selections.

- User Console
- Role Console
- Company Attributes
- Access Control Group
- Access Privileges
- Activity Logs

Roles and Users

The management of users within WebSphere Product Center is controlled through a set of roles that are created through the Administer Roles component of the Security module.

Rule: Privileges are not set to the individual, rather to the role that they are assigned to. If a user is assigned to multiple roles, they inherit the privileges from each role.

Customized roles can be created (i.e. content reviewer, content approver, catalog manager) with permissions to specific WebSphere Product Center functionalities and/or objects. Thus, to apply the privileges for a customized role, assign a user to the role.

Use WebSphere Product Center's Access Control Groups (ACGs) to set the permissions in accordance with which user(s) can view/edit specific catalogs. Assign a user to one or multiple ACG's, depending on the user's responsibilities. If needed, group various roles to a single object.

Additional control for catalog access is available through the Catalog Access Privilege Console. A set of privileges can be set for which roles can view/edit specific columns in a catalog.

Roles and Privileges

Roles are created to control a user's privileges to catalog management. Privileges are not set to the individual, rather to the role that they are assigned to. The objective of creating an ACG helps to control user's privileges, but is created to group a set of users to a single object.

Note: Objects cannot be mapped to more than one ACG.

Access control privileges are used as follows:

- Each role can contain multiple users
- A user can belong to multiple roles
- Each access control group contains a number of objects (in this case catalogs)
- A catalog can only belong to one Access control group

User Management Scenario

John has spent the time creating a set of users and catalogs. Now he would like to specify which catalogs each user can access and define their privileges through the use of roles and ACGs.

Assuming that the users and catalogs have already been created and the catalogs have not been assigned to an ACG, the following sections will step through the following tasks:

- Create a New Role
- Create a New Access Control Group
- Assign an ACG to an Object (Catalog)
- Assign User to a Role

Create a New Role

1. Use the menu path: **Data Model Manager > Security> Role Console**. The Role Console table appears.

2. Click **New** and enter a Role Name and Role Description, which are required fields.

3. For the Access Control Group, select 'Default'.

4. Select a set of privileges for this role. (Note: The privileges that a role can have for a specific access control group are defined later; they will be a subset of the privileges given here.)

5. Click **Save** (top of screen).

Summary

After saving the new role, it appears in the Role Console table, see figure below. Notice that the Assigned to column contains the number of users assigned to the role.

Note: When managing users, each user must be assigned to at least one role in their User Profile.

Create a New Access Control Group

1. Use the menu path: **Data Model Manager > Security > Access Control Groups > Access Control Group Console** and click **New**.

2. Enter a Name and Description for the new ACG.

3. Select a role from the drop-down menu.

4. Select the set of privileges for the role selected. (Note: These privileges are to control what the user can do.)

5. Click **Save**.

Assign ACG to an Object

The following will apply an ACG to a catalog.

1. Use the menu path: **Data Model Manager > Security>Access Control Groups>Object to Access Control Group Map**. A wizard-driven GUI appears.

2. Select the object type "Catalog." Select a catalog from the Select Object drop-down list.

3. Select an ACG. This will assign the catalog to the control group.

Summary

At this point, roles have been created and grouped into different ACGs, which have been mapped to a catalog. Now that all of the privileges have been set, users can be assigned to any role and all privileges to the selected role applies.

Assign Users to a Role

1. Use the menu path: **Data Model Manager > Security >User Console**. The Current Users table appears.

2. Click on a user hyperlink to view the user's profile. From the Roles for current user table, select all roles assigned to the user.

3. Click **Modify Role Info**.

Summary

The privileges given to the user are determined by the role they have been assigned to and to the Access Control Group the role belongs to.

Creating User

Before a user is created, at least one role must exist in the application.

1. From the left pane, select an Organization Hierarchy. Right-click on the name of the organization and select Add user from the short menu. The New User screen appears.

2. Enter the required details in the User Profile interface.

3. Enter the password for the user.

4. Assign a role(s) to the user. Multiple roles can be selected based on

responsibility.

5. Once all the information needed is entered, click Save to store the information.

6. The last step is to enable the user in the system. New users always default to Disabled.

Enable User

When a new user is created, they are disabled. To allow the new user to access the application, they must be enabled.

From menu **Data Model Manager > Security > User Console**, click on the **Disabled** button. The button changes to Enabled.

Access Control Groups

Access Control Groups - Use WebSphere Product Center's Access Control Groups (ACGs) to set the permissions in accordance with which user(s) can view/edit specific catalogs. Create an access group and assign access privileges to each role in the ACG. Map the ACG to an object. Then, assign a user to one or multiple roles, depending on the user's responsibilities.

ACGs can be applied to the following:

- Catalogs
- Collaboration areas
- Hierarchies
- Selections
- Workflows

Privileges can be placed on various objects by creating rules that restrict access to a group of roles. The rules are enforced to all users assigned to the roles.

Map ACG to object

The wizards used to create catalogs and hierarchies require an association with an ACG. This is true with other objects created in WebSphere Product Center. The following objects listed in the table below require an association with an ACG. The right column describes how an ACG is mapped to the object.

Object	How ACG is associated
Catalogs	Data Model Manager > Security > Access Privileges > Catalog Access Console Or Data Model Manager > Security > Access Control Groups > Object to Access Control Group Map
Collaboration Areas	Associate ACG during the creation of a collaboration area
Hierarchy	Data Model Manager > Security > Access Privileges > Hierarchy Access Console Or Data Model Manager > Security > Access Control Groups > Object to Access Control Group Map
Selections	Associate ACG when creating a selection
Workflows	Associate ACG during the creation of a workflow

Example: Associate ACG with Catalog

In order to enforce access control to a catalog, the catalog must be mapped to an ACG, which is done during the creation of a catalog.

1. Use the menu path, **Product Manager > Catalogs > Catalog Console**, to display the Catalog Console.

2. Click **New** to create a new catalog.

3. For the "Select Access Control Group" step, create and ACG or select an existing ACG.

Example – Apply Access Control for Selections

The ability to control a user's access to view selections, edit selection rules, and deleting selections can be restricted based on the access privilege definition of the role that is assigned to the user. In order to restrict access to a selection, the role defined with restricted access privileges must be associated with the ACG that the Item Selection is using.

Therefore, a single selection can be made available to a specific ACG and all roles that are part of the ACG will have access to the Item Selection. A user is allowed access to the Item Selection once they are assigned to the role.

Troubleshooting

If the user cannot view item selection that is defined by the ACG, check the following:

- Make sure the user has been enabled
- Check to make sure the user has been assigned to the proper role
- Check the access privileges for the role have been properly defined as described in the step "Create New Role".
- Check to see if the user belongs to an ACG that allows access to the specific catalog. A user can be setup for access to an items selection, but with no access to the catalog, the user will not see any information.

Creating a new role and assign to ACG

For each role, there are three areas of security that can be implemented:

- Group Access – restrict access to the role for each associated ACG
- System-wide Access – restrict access to various application features
- Locale Access – restrict access to one or more available locales

When setting group specific access for a role, it is recommended to select access for the ACG "Default". ACG – Default is created by default and when no custom ACG is selected for an object it uses the ACG "Default". All objects that can be associated to an ACG require that an ACG is associated with the object. Therefore, it is important to create a set of group access privileges for the role.

1. Use the menu path **Data Model Manager> Security > Role Console**, the "Role Console" dialog appears.

2. Click **New** and enter a name and description for the Role. For this exercise, use the name "Basic View".

3. From the Group specific access for role table, select the group access for each Access Control Group.

Note: These changes are also updated in the ACG Console.

4. Click **Save** to commit settings. A message appears indicating a successful role creation.

5. Scroll down to the "System-wide access for role" table; click the "Edit Screens" hyperlink to access the Edit Screen Access page.

6. Select the screens that are to be made available to the role. The following screens must be selected as a bare minimum requirement:

- View WebSphere Product Center Main' screen
- View ' WebSphere Product Center' screen
- View 'Catalog Navigation Pane' screen
- View 'My Home' screen
- View 'Collaboration Console' screen

7. Click **Modify** to commit settings.

Creating Access Control Groups

ACG's are mapped various objects, which then enforces a set of security rules defined in the group's roles. Objects required that an ACG be selected and if a customized ACG is not desired, select the ACG "Default".

1. Use the menu path **Data Model Manager > Security > Access Control Groups> Access Control Group Console**, the ACG console appears.

2. Click **New** and enter a Name and Description for the ACG. For this exercise, the ACG is named "E".

3. Select any Role from the drop-down list. A new role will be created in the next section, which will be added to the ACG.

4. In the Access Control Group table, select the following checkboxes:

- Catalog – List
- Catalog – View
- Catalog – Search
- Selection – List

5. Click **Save** to create the new ACG.

Enforce Access Control for a User

To enforce the access control rules that have been set up for an ACG, the user must be assigned to a role member of the ACG.

Assigning a role to a user

Once a user and role have been created, use the User Console to assign a role to a user.

1. Use the menu path **Data Model Manager > Security > User Console**; the Current Users table appears.

2. Click on a user name

3. Scroll down to the Roles for Current User table and select the role "Basic View", which was previously created.

4. Click Modify Role Info to commit the new user profile.

Note: Any user assigned to the Basic View role will have view only access to Items selection created in the next section.

Access Privileges

Setting Access Privileges are an extension of the security rules defined for ACG's. Using WebSphere Product Center's Catalog Access Privileges Console, users have the ability to restrict associated attribute collections of a catalog to one or multiple role members.

For example, when defining the access privileges for a catalog, it is possible to enforce View and/or Edit privileges for an attribute collection of a catalog, therefore providing complete control of which catalog attributes a role can view or edit. If locales have been implanted, it is possible to restrict attributes based on the available locales.

Setting Catalog Access Privileges

A role can be restricted to any catalog based on a set of defined privileges that are configured in the Catalog Access Privilege Console. Users assigned to the role will be restricted to the catalog access privileges.

Create rules to allow a role to enforce viewable and/or editable privileges to any catalog. Privileges will need to be defined for each role that needs access to the catalog.

1. From the menu path **Data Model Manager > Security > Access Privileges > Catalog Access Console**, click the **New** button next to the name of the catalog from which access privileges will be created. The Catalog Access Privileges wizard appears.

2. Select a Role from the drop-down field. Only roles that are members of the ACG tied to the selected catalog are displayed.

3. From the Catalog Access Privileges Editor, select the attribute collections as viewable or editable.

Note: A "V" appears next to the attribute denoting view privileges. A "V+E" denotes view and edit privileges.

4. To remove a rule, click on an attribute collection from the Selected box and click Remove.

5. When all of the privileges have been defined, click **Save**. A message indicates the privileges were successfully saved.

6. If desired, create privileges for all associated roles. Each role that has been defined with privileges appears in the Catalog Access Privileges Console.

7. To edit the privileges for a role, click on Edit icon from the Catalog Access Privileges Console, make the changes in the editor, and click Save.

Removing Catalog Access Privileges

To remove all catalog access privileges for a role, do the following:

1. From the Catalog Access Privileges Console, click the edit button to edit the role.

2. Highlight all of the attributes from the Selected Attributes box and click **Remove**.

3. Click **Save**. Return to the Catalog Access Privileges Console and attribute collection is removed.

Edit Role Access

Restrict access privileges for roles, which apply to any user assigned to the particular role. Changes made in the Edit Role Access screen are reflected in the associated Access Control Group Details and System-wide Access pages.

Editing role access

1. To edit role access, use the menu path **Data Model Manager > Security > Roles Console**. The "Roles Console" table appears with a list of roles that have been created.

2. Select a role to edit and the Edit Role Access page displays. Each Access Control Group that is associated to the role is displayed in separate columns.

3. Select specific access privileges for each Access Control Group. In the table "System-wide access for role" click on the Edit Screens link to restrict access to specific application screens.

Note: Refer to the table below for role access descriptions.

Group specific access for role	
Catalog	
list	Allows display of catalogs in Catalog Console and in lists throughout WebSphere Product Center. If not selected, the Catalog Console states "No catalogs found."
edit catalog views	Allows creating, deleting, and editing of catalog views
view items	Allows view only access to catalog items
add items	Allows new items to be created. If not selected, all buttons and short menus used to add items are disabled
modify items	Allows items to be modified * Note: If unselected, "add items" and "recategorize items" must be unselected
delete items	Allows the deletion of items. If not selected, the DELETE button in the Item Edit Screen is disabled
recategorize items	Allows the recategorization of items in catalogs.

	If not selected, the Categorize button in the Item Edit screen is disabled
summary items	Not functional. Will be removed in future release
export	Allows the export of a catalog items or item-category attribute values from a catalog
attributes	Allows access to the attributes page via the Attributes button in the Catalog Console
differences	Allows the display of differences between catalogs
rollback	Allows the rollback of a catalog
search	Allows a basic or rich search on a catalog
delete	Allows the deletion of a catalog from the Catalog Console
run preview script	Allows running a preview script for an item (i.e. Item HTML Preview, Item Tab Delimited Preview)
Hierarchy	
list	Allows display of hierarchies in Hierarchy Console and in lists throughout WebSphere Product Center If not selected, the Hierarchy Console states "No hierarchies found."
edit hierarchy views	Allows creating, deleting, and editing of category views
view hierarchy nodes	Allows view only access to hierarchies * Note: If unselected, "add categories", "modify category names", and "modify category attributes" must also be unselected.
add hierarchy nodes	Allows new categories to be created
modify hierarchy node attributes	Allows modification to hierarchy node attributes * Note: If unselected, the "add categories"

	must also be unselected
delete hierarchy nodes	Allows the deletions of categories
recategorize hierarchy nodes	Allows the recategorization of categories
summary hierarchy nodes	Not functional. Will be removed in future release
specmap hierarchy nodes	Allow
attributes	Allows the view of hierarchy attributes
rollback	Allows the rollback of a hierarchy
delete	Allows the deletion of a hierarchy
Selection	
list	Allows the display of selections in the Selection Console
edit rule	Allows the creation of rules applied to a selection
delete	Allows the deletion of a selection
Import	
list	Allows the display of imports in the Import Console
perform import	Allows the initiation of importing catalog items or item-category attribute values into a catalog
delete	Allows the deletion of an import
SelectionMembers	
View items	View item selections
add items	Add items to selection
modify items	Modify items in a selection
delete items	Delete items in a selection
recategorize items	Recategorize items in a selection
view hierarchy nodes	View hierarchy node in selection
add hierarchy nodes	Add hierarchy node in selection
modify hierarchy node attributes	Change hierarchy node attribute in selection
delete hierarchy nodes	Delete hierarchy node in selection
recategorize hierarchy nodes	Recategorize hierarchy node in selection

specmap hierarchy nodes	Create specmap hierarchy nodes in selection
DocStore	
view files	View files in the docstore
delete files	Delete files in the docstore
PurchaseOrderExport	
list	Allows the display of purchase order exports in the PO Export Console
export	Allows the initiation of a purchase order export
delete	Allows the deletion of a purchase order export
Workflow	
list	Allows the display of workflows in the Workflow Console
edit	Allows workflow to be edited
delete	Allows the deletion of a workflow
CollaborationArea	
list	
checkout entries	Allows entries to be checked out in the Collaboration Console

System-wide access for role	
Spec	
modify spec	Allows modification of any spec
modify spec map	Allows modification of any spec mapping
Screen	Edit Screens (click to edit screen access privileges)
view	Allows access to the screens selected in the "Edit Screens" above. If this box is not checked, the list of selected screens is not available to the role.
Script	
create modify scripts	Allows the creation of scripts. When this is

	not selected, the New button in the "Scripts Console" does not appear
Scheduler	
view company jobs	Allows display of jobs in the Jobs Console
Security	
modify users	Allows creating, deleting, editing of users
modify roles access	Allows creating, deleting, and editing of roles

"Local access for role" is used to select from a list of available locales for a role.

Locale access for role	
Available Locales	List of available locales that have been setup in "Administer Company Attributes"
Selected Locales	List of selected locales that have been made available for the role

Edit Screen Settings

It is possible to restrict a role to specific WebSphere Product Center screens. In the System-Wide Access table, click on the Edit Screens hyperlink and the Role Info table appears with a hierarchical list of screens.

Each listed screen can be restricted to a role by leaving the checkbox next to the screen name empty. Thus, a checked box allows access to the screen. When all restrictions to screens have been made, click Modify to update the changes.

Minimum Requirement Settings

Although the behavior of the Edit Screens is fairly straight forward, there are a few special cases to note and are described in the following sections.

A user's home page is made up of various screens; therefore access to each screen must be provided. The following settings are the minimum requirements for a user to login and view the home page.

- View WebSphere Product Center Main' screen
- View WebSphere Product Center screen
- View 'Catalog Navigation Pane' screen
- View 'My Home' screen
- View 'Collaboration Console' screen

With the access privileges to the above screens, the user home page displays the WebSphere Product Center Main, Navigation Pane, and the Collaboration Console.

If one of the above screen permissions is unchecked, instead of getting the Collaboration Console you will get the error message "You do not have the privilege to access this page".

The following sections define each of the screen settings in the Role Info table.

Activity Log

The activities performed by users can be monitored through WebSphere Product Center's Activity Logs. Monitor which pages the user has visited, which catalogs have been edited, and have an instance of an activity notify another user via email. When a new user is created, they are automatically added to the list of the users in the Activity Log.

Configure Activity Log

1. **Use the menu path: Data Model Manager > Security > Activity Logs > Activity Log**.

2. Monitor a user's activity, have activities notified, or track deletion activities by clicking on the appropriate box selection.

3. To receive an update via email, click the Update Notification Email

check box and enter an email address.

4. When all activities have been configured, click Update.

View User Activity

From the Users Monitored table, select Sessions, Log, or Summary to view current activities for a user.

- The Sessions link provides a list of pages visited by the user.
- The Logs link displays a log of pages visited by a user.
- The Summary link displays an overview of the number of times the user has visited a page.

Notify Users

From the Activity Log screen, a message can be created and sent to all users or only to those users currently logged in to the application The Users Monitored table displays all current users. The message is sent to the email address defined in the user's profile or the notification email defined in the Users Monitored table. This email can be different than the one defined in the user's profile.

Send a Message to Users

1. Scroll to the end of the Activity Log screen. Enter a message in the Notify Users table.

2. Select to send the message to all users listed in the Activity Log box or only to the users who are currently logged on.

3. Click **Send** and the message is sent to each user's email address.

Alerts

The alerts functionality is in essence WebSphere Product Center's messaging system. Alerts can be tied to any type of event and can be used to notify specific users or group of users that a specific event has occurred.

Events include a successful export job to a problem during an import job. WebSphere Product Center supports a number of different alerts that accommodate a wide variety of events.

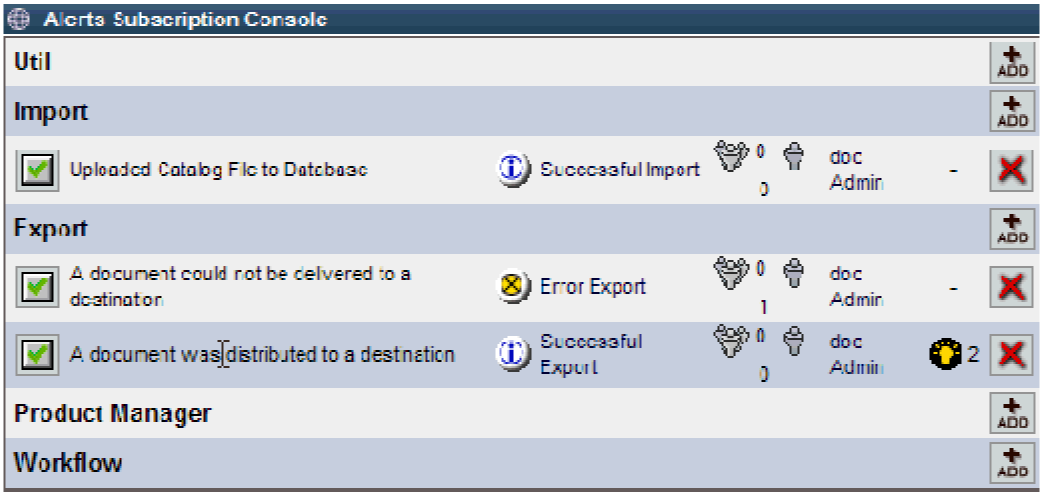
Alerts Console

The Alert Console is the area where all alerts are managed in WebSphere Product Center. From this screen, it is possible to subscribe to alerts and browse through alerts that have been triggered. It is also where alerts can be configured and users can be associated to specific alerts.

The Alert Console is broken down into multiple sections that correspond to the various functional areas of the application. The alerts can also be monitored from the Alerts display in the left quick access bar.

Accessing the Alerts Console

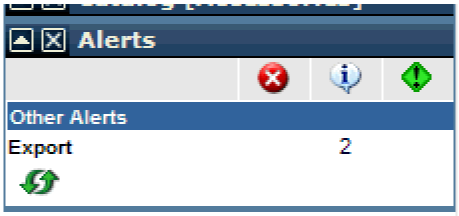
Use the menu path: **Data Model Manager > Alerts > Alerts Console**.



Alerts Subscription Console

Util						ADD
Import						ADD
☑ Uploaded Catalog File to Database	Successful Import	0 0	doc Admin	-		☒
Export						ADD
☑ A document could not be delivered to a destination	Error Export	0 1	doc Admin	-		☒
☑ A document was distributed to a destination	Successful Export	0 0	doc Admin		2	☒
Product Manager						ADD
Workflow						ADD

Figure 18. 1 - Alerts subscription console



Alerts

Other Alerts		
Export	2	

Figure 18. 2 - Alerts displayed in the Left Pane

Displaying alert activities

The Alerts Subscription Console displays a user's subscribed alerts. To display all alert activities, use the menu path **Data Model Manager > Alerts > Alerts Display.** A table appears "Current Results" that show all of the alerts that have triggered.

Subscribing to an alert

Alerts can be individually set up for each user through the Alerts Subscription Console.

Use the menu path: **Data Model Manager > Alerts > Alerts Console**.

1. Select Event Group

Available event groups available correspond to the functional areas of the application, including:

- Util
- Import
- Export
- Product Manager
- Workflow

Select an event group and click **+ADD** and the "Add A New alert Subscription" wizard appears.

2. Select Event Type

An event type defines the precise nature of the alert to be configured. The list of event types is dependent upon the group selected in the first step.

3. Select Event Conditions

Specify additional parameters that trigger an alert under very specific circumstances. This step can be skipped if no parameters are necessary

4. Alert Description

Provide a description for the alert that will make it easy to locate when reviewing a long list of pre-configured alerts.

5. Select Distribution

Click the question mark button and select a distribution group that is to be notified when the alert is triggered or click **New** to create a new distribution group. Click Select to go to the next step.

Note: The user setting up the alert is automatically subscribed for the alert, and therefore a user who selects their own name from the user list while subscribing to an alert will get the alert twice!

6. Select Users

Instead of selecting a distribution group, select the user(s) that is to be alerted when the event occurs.

Once the alert subscription wizard is completed, alerts can be viewed from the Alerts Subscription Console.

Note: Alerts can be viewed from the Left Pane if it has been added.

Viewing alerts results

Alert results can be viewed from one of three areas:

1) Alerts Console - Click on the alert name to view results

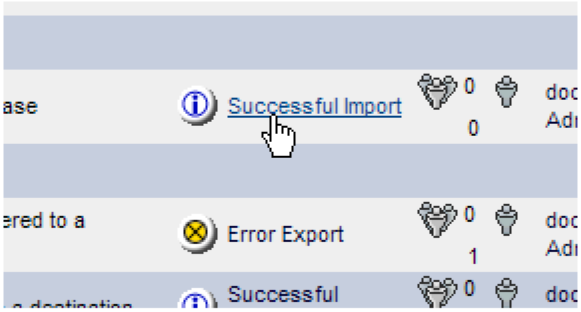
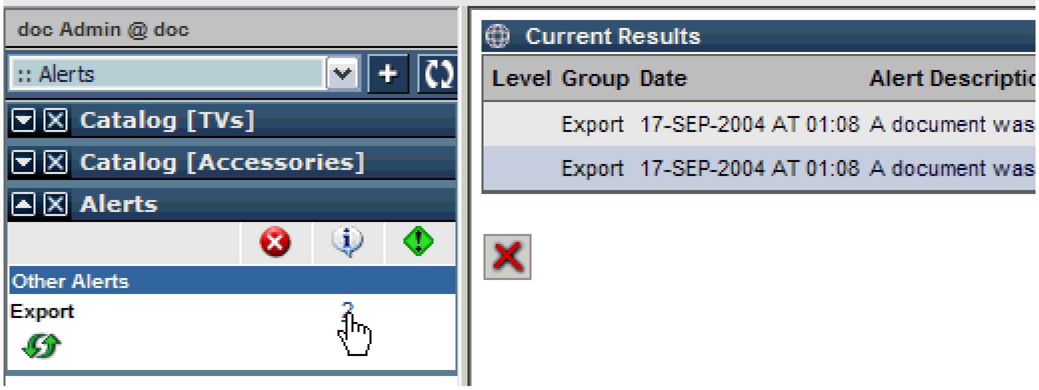


Figure 18.3 - View alert results from the Alerts Console

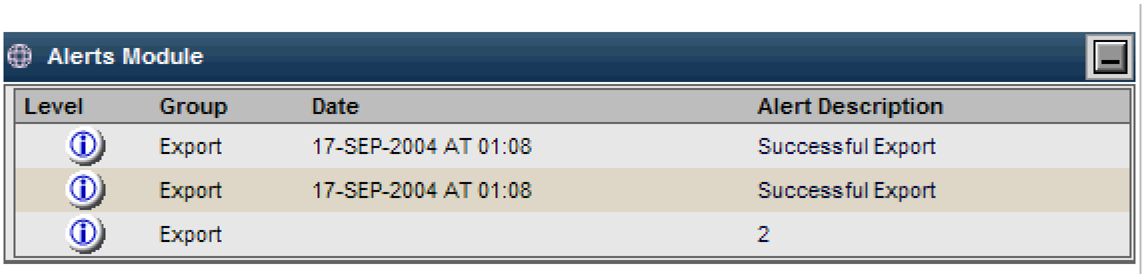
2) Left Pane - click on the alert number in the Left Pane and the results appear in the right pane



Level	Group	Date	Alert Descriptic
	Export	17-SEP-2004 AT 01:08	A document was
	Export	17-SEP-2004 AT 01:08	A document was

Figure 18.4 - View results from the Left Pane

3) My Task List - From the Alerts Module, click on the Alert Description name to view results.



Alerts Module

Level	Group	Date	Alert Description
ⓘ	Export	17-SEP-2004 AT 01:08	Successful Export
ⓘ	Export	17-SEP-2004 AT 01:08	Successful Export
ⓘ	Export		2

Figure 18.5 - View results from My Task List

Ch 19 Staging Area

Business processes are supported and enforced through the use of a staging area. For example, Product Managers can launch data changes on an item level into a temporary staging area, identify jobs that would ask for manager approvals with alerts that will notify pertinent parties, which then can be rolled back granularly if rejected, and that will automatically trigger a differences report for managers and monitors.

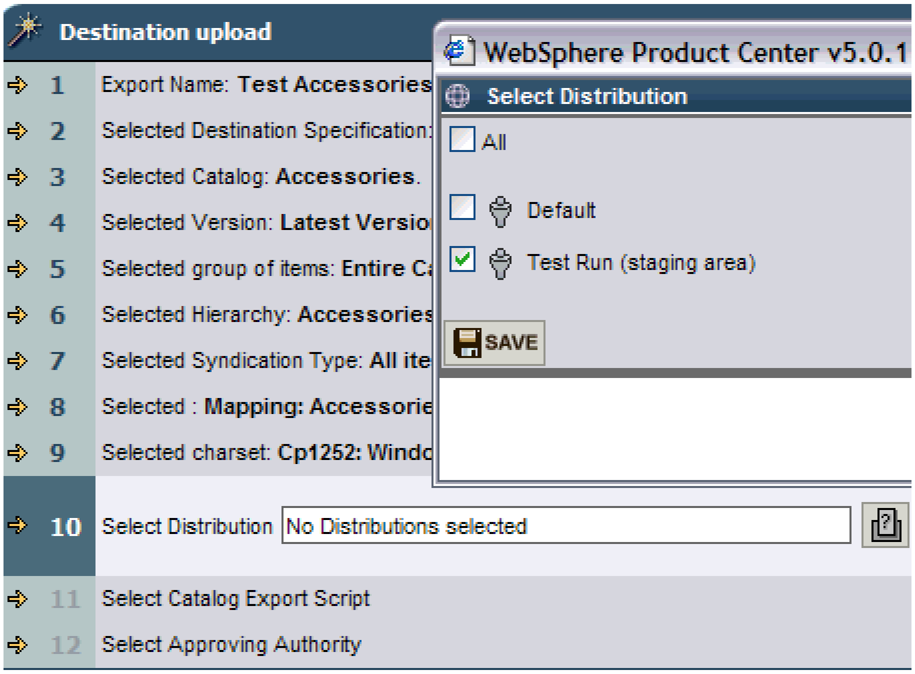
When a staging area is created, nothing appears in the Staging Area Console. When creating an Export using the "Create Staging" wizard, Step 9 selects a distribution. This is where the Staging Area is selected and when the export runs, it sends it to the selected staging area and appears in the Staging Area Console.

Creating a staging area

To create a staging area, do the following:

1. Use the menu path Data Model Manager > Staging Area > Staging Area Console, to display the Staging Area Console and then click **Create new Staging Area**. The Create Staging Area wizard appears.

2. Enter a name for the new staging area and click **Next**. A message box appears notifying of a successful creation.



Destination upload

1 Export Name: **Test Accessories**
2 Selected Destination Specification:
3 Selected Catalog: **Accessories**.
4 Selected Version: **Latest Versio**
5 Selected group of items: **Entire Ca**
6 Selected Hierarchy: **Accessories**
7 Selected Syndication Type: **All ite**
8 Selected : **Mapping: Accessorie**
9 Selected charset: **Cp1252: Windo**
10 Select Distribution: No Distributions selected
11 Select Catalog Export Script
12 Select Approving Authority

WebSphere Product Center v5.0.1

Select Distribution

□ All

□ Default

☑ Test Run (staging area)

Figure 19. 1 - Selecting a staging area for exports

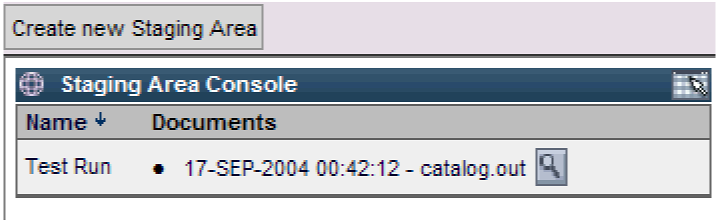
Viewing staging area details

As files are added to the staging area, they are listed in the Staging Area Console.

1. Click on the name of the staging area to view all of its associated files in the Docstore.

2. Click on the generated document name to view its contents.

3. Click on the view button to display the audit log for the document.



Staging Area Console

Name	Documents
Test Run	• 17-SEP-2004 00:42:12 - catalog.out

Figure 19. 2 - Staging area console

Ch 20 - Workflow

The PIM process can be managed through the definition of a workflow. The Workflow Console is used to create a workflow process, containing multiple instances that are viewable through the definition display.

A workflow instance can be created to appear in the Workflow Console and based on the status, an alert can be sent to notify that an approval is needed before it is escalated to the next step in the workflow.

This chapter summarizes the Workflow feature using the following key questions:

- What is a WebSphere Product Center workflow?
- How is a workflow setup?
- How does data move through workflow steps?
- What is the available task list/status functionality?
- What is the available workflow reporting functionality?

Each question is provided with a high-level response and is provided with a more thorough discussion in the Workflow Technical Details section.

What is a WebSphere Product Center workflow?

A WebSphere Product Center workflow implements a business process in either the Product Center application or a separate WebSphere Product Center application. WebSphere Product Center's workflow component offers a set of screens to setup task lists/status screens, and reporting functionality.

Example business processes:

Within the WebSphere Product Center core application:

- Add item
- Modify item

Within a WebSphere Product Center Item Synchronization application:

- UCCNet Item Add

Within a WebSphere Product Center Supplier Self Service application:

- Supplier file submittal

How is a workflow set up?

A Business Process Analyst uses the UI screens to build a series of steps corresponding to a specific business process. Although it is possible to configure most steps without any scripting, further workflow definitions can be performed using scripting for any workflow step.

There are a variety of predefined step types for each workflow step including:

- Modify
- And Approval
- Or Approval
- Automated
- General

Based on the step type, it is possible to setup parameters for the step. These available parameters include

- The roles or users with access to the step
- The editable attributes in the step
- The exit values for the step (including escalation)
- The email notifications at the step
- The timeout for the step
- The script for the step

If needed, a nested workflow can be defined by having a step feed another workflow, or a step can accept data from another workflow. A step can also call out to external systems via HTTP, MQ, JMS, FTP, or SMTP.

How does data move through the workflow steps?

Catalog or hierarchy attribute values move through the workflow steps in a collaboration area. A collaboration area is a "mini-catalog" that supports regular catalog/hierarchy functionality - including content authoring screens, views, spec validation rules, and scripts.

Note: WebSphere Product Center Workflows currently only support handling catalog and hierarchy attribute values, not handling specifications for attributes.

Insert data into a collaboration area via either "checking-out" an existing attribute value from a main catalog/hierarchy, or via importing new values into a collaboration area.

For example, a user may checkout one attribute of an item into a collaboration area in one workflow (e.g.: English short description), while checking out another attribute of the same item into a different collaboration area in another workflow (e.g.: French short description).

A checked-out attribute is available as read-only in the main catalog. There is a lock symbol on the item in the Catalog or Hierarchy Multiple Edit screen to indicate that an attribute of the item is checked-out. As a read-only attribute, the attribute can be viewed or exported from the main catalog/hierarchy but not modified. Only the parties with access rights to the modification steps in the collaboration area containing a checked-out attribute can modify a checked-out attribute.

Note: It is possible to setup a main catalog/hierarchy as fully read-only, while forcing all attribute value changes to be made in workflows.

If the Add Items box in any step is checked, it is feasible to import new items into the collaboration area at that workflow step. All items imported into a collaboration area are validated by the same import validation as for an import into a main catalog/hierarchy. It is not possible to save invalid records into a collaboration area, as it is not feasible to save invalid records into a main catalog.

After a set of items completes its transit through the workflow, it is possible to "check-in" new or modified records into a main catalog/hierarchy. It is also possible for a user to drop an item + attribute from a collaboration area at any time (where dropping an item releases the item + attribute in the main catalog for editing). After all records in the collaboration area complete their passage through the workflow, it is possible to set a property of the collaboration area to automatically delete an empty collaboration area. An administrator can also manually delete an empty collaboration area. The system retains the history of a deleted collaboration area for reporting.

What is the available task list/status functionality?

The workflow includes a standard collaboration console that pictorially represents the status of data in each collaboration area in each workflow step.

A business process analyst can supplement the standard collaboration console with custom scripted screens generated by the Invoker.

The collaboration console/task list is available for any user in their default Home Page. If a user has access to any step in a workflow, the user will have access to the collaboration console for that workflow. The collaboration console indicates the number of items at any step in the workflow. The user can directly interact with items in green by clicking on the green number at any step. The user can see the number of items at any step with a red number, but cannot interact with the items in that step.

As well as maintaining the status of a collaboration area, the system supports an item history for each item in the collaboration area. A user in the collaboration area can click on an item to see the changes to the item at each workflow step, approvals/rejections, and user comments.

What is the available workflow reporting functionality?

The workflow includes an extensive audit trail. It stores every attribute change at each workflow step for each collaboration area in the database. With the provided script operations it is possible to build extensive attribute-level lifecycle reports. Example reports are -

- Time to Shelf Report - showing the time for newly introduced products to move from receipt to syndication into external systems

- Cost per Managed SKU Report - measuring the time and number of resources required to move product from receipt to delivery
- Price Change Report - indicating all price changes at each workflow step with each user's name, datetime of change, and comments
- User Throughput Report - showing the number of items processed by each user over time at each workflow step
- Approval Chain Report - presenting all approvals within a given workflow
- Current User Status Report - showing a snapshot of the number of items at each workflow step for a given user
- Escalation Report - indicating all items that were escalated by timeout during a time period

Workflow Technical Details

The following sections summarize the technical details for WebSphere Product Center Workflows:

- Workflow Setup Steps
- Data Movement and Task Lists/Status
- Reporting

Workflow Setup Steps

A business process analyst sets up the overall workflow in the Workflow Setup Console and Edit Workflow Step screens.

There are two key characteristics for all workflows:

1) All workflows automatically include Initial, Success, and Failure steps. Timeout steps are also available by default.

- An Initial step is always the first step in a workflow
- A Success step attempts to check in all items that reach this step
- A Failure step drops all items that reach this step
- A Timeout step places all item that reach this step into a "fix-it" holding area for review

2) A workflow will only save as long as the process moves from the Initial step to a Success, Failure, or Timeout step without a break in the flow.

It is not necessary to have a route from the Initial step to each of the Success, Failure, and Timeout steps. But for a workflow to be valid, all paths from the Initial step must lead to a Success, Failure, or Timeout step.

WORKFLOW SETUP FOR TYPICAL BUSINESS PROCESS

A typical process for a business process analyst to set up a workflow is:

0. User creates a workflow flowchart in a program such as Visio.

1. Open the Workflow Console screen

2. Press New to create a new workflow. Reach the Edit Workflow Details screen.

3. Name the workflow.

4. Provide a Description of the workflow (optional).

5. Set the Access Control for the workflow. This Access Control determines which roles can view, edit, or delete this workflow.

6. Determine the Container Type supported by the workflow.

There are two supported Container Types - Catalog or Hierarchy. A workflow supporting a Catalog can support a collaboration area containing the attributes directly supported by catalogs - catalog attributes and item-category attributes. A workflow containing a hierarchy can support a collaboration area containing the attributes directly supported by hierarchies - hierarchy attributes and category secondary attributes.

7. Press Add Step to define the first step after the Initial step (if any - as it is possible to complete a workflow by mapping the Initial step directly to a Success step). In this example, the second step is Modify Price.

8. The Add Step button opens the Edit Workflow Step screen.

9. Provide a Name for the step

10. Provide a step Description (optional)

11. Select the step Type.

In this example the step type for the Modify Price step is Modify. There are two broad types of steps - Steps that involve user interaction and steps that do not involve user interaction.

The Step Type Table below describes the available step types, the exit values available at each step, whether performers are available at a step, whether nodes are accessible at a step, whether a deadline is available for the step, whether notifications are available at a step, and whether a script is available for the step.

12. Select the Exit Value if the Exit Value is not predetermined for the step type. In this example with a Modify step type, the Exit Value is predefined as DONE.

If a step involves user interaction, the Exit Value is the text displayed on the button(s) that enable moving to the step mapped to the Exit Value.

If a step does not involve user interaction, each outcome in the script within the step should map to an Exit Value.

13. Select the Performers at the step if Performers can be selected for the step type. A performer is a role and/or user who is allowed to perform the action supported by the step (the action could be modify, and_approval, or_approval, dispatch to another step, etc.). Performers are the only roles/users who can access the step.

It is possible to combine roles and users in any step. If a user is within a role and both the user and role are mapped to a step, the user will be able to act on behalf of the role.

Note: In order to deselect a selection in this popup window, press the

CTRL key, then left click on the selection.

14. Optionally select the Nodes for the step if Nodes can be determined for the step type.

The Nodes are the catalog or hierarchy attributes available for editing within the step. These attributes must be available with the specification of a given catalog or hierarchy. For a catalog spec, the attributes can include catalog attributes and item-category attributes. For a hierarchy spec, the attributes can include hierarchy attributes or category secondary attributes.

If the container is catalog, it is possible to add nodes from multiple catalog specs. Likewise if the container is hierarchy, it is possible to add nodes from multiple hierarchy specs.

15. Optionally set the Deadline for the step if a Deadline can be determined for the step type. Upon reaching a Deadline, an item will move to the step mapped to Timeout.

There are two available Deadlines for a step -

- Duration based Deadline at a Step - which will move items in a collaboration area from the current step to the step mapped to Timeout after the item remains in the step for the duration time. Duration based deadlines can be fractions of a Day or Hour.
- Date based Deadline at a Step - which will move all items in a collaboration area from the current step to the next step upon reaching the date

Note: There is also a Deadline available for the whole collaboration area that can be set upon loading items into the collaboration area. With this deadline for the whole collaboration area, all items in the collaboration area have the same Deadline.

16. Optionally set if it should be possible to Add Items to the step. If the Add Items box is checked, it will be possible to run an import feed into a collaboration area at that step.

Note that if the business process analyst setting up the workflow enables adding items to a step after approval steps, the items would not go through the approval steps.

17. Optionally set the Notifications for the step. Notifications are available for every step type. Notifications are emails that are triggered upon Entry into the step or upon the Deadline for the step. The business process analyst enters email addresses into the notifications boxes. The system sends predefined emails to the addresses upon step Entry or reaching a step Deadline.

If the business process analyst wishes to send custom emails to users, it is possible to configure custom emails via the script in a step.

18. Optionally set the Script for the step. Access the script functionality by saving the step, then by pressing the Add Script button. Any step can have a Script. There are three methods available in a Script - IN(), OUT(), and TIMEOUT(). Timeout is equivalent to Deadline. It is not necessary to include a script in each method. It is necessary to map each exit value to a script function.

It is possible in the script step to use any WebSphere Product Center script operation. We expect that customers will frequently use the script step for the following purposes:

- Route workflow records based on certain criteria (e.g.: margin > 10% is mapped to Exit Value = FINAL APPROVAL else is mapped to Exit Value = SPECIAL APPROVAL)
- Run an Invoker trigger script to send workflow data to an external product via HTTP, MQ, JMS, UCCnet, SMTP, or FTP or receive workflow data from an external product
- Run an Invoker trigger script to push data into custom HTML pages or receive data from custom HTML pages
- Create reports such as an add/modify report

19. Repeat Steps 7-18 for the remaining steps in the workflow. In this example the remaining step is Approve Price.

20. In the Select Next Steps screen, map each step to appropriate next step based on the step Exit Value. In this example we need to setup the following mappings:

- Initial Modify Price
- Modify Price Approve Price
- Approve Price/Approved Success
- Approve Price/Rejected Modify Price

21. Setup the pictorial representation of the workflow in the Edit GUI screen. This screen enables a user to depict the steps and the flow between steps in a picture. There is a link to this picture in the Edit Workflow Details screen.

To access the screen, press the blue magnifying glass button in the toolbar in the Edit Workflow Details screen.

The screen displays all of the steps created above. There is a tool tip on each step displaying the step path, description, type, transitions in, and transitions out.

Position each step on the screen by clicking on the step, then clicking on the appropriate box on the screen. Use the lines in the Transition Library to connect steps.

23. Save the workflow.

Step Types

The tables on the following pages contain all of the workflow step types with an explanation of each step type, which is following with a description.

Base System Steps

Step Type	Initial
Description	A workflow always starts with an Initial step and must end in either a

	Success, Failure or Timeout step. There is only one instance of an Initial step per workflow.
Exit Values	SUCCESS
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	Yes (if the user wishes to create new records in the workflow by running an import feed into the Initial step, the user must check the Add Entries box in the Initial step)
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Success
Description	If records reach the Success step in a workflow, the system attempts to check the records into the core container (catalog or hierarchy) connected to the collaboration area tied to the workflow.
Exit Values	SUCCESS

Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Failure
Description	If records reach the Failure step in a workflow, the system drops the records from the collaboration area.
Exit Values	FAILURE
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes

Script?	Yes

Step Type	Fixit
Description	This step is a special step used to repair entries. An user can send an entry in any step to a fixit step for not satisfying a requirement.
Exit Values	FAILURE
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

User Steps

Step Type	And_Approval
Description	An approval step where all the performers must approve an record before it moves to the next step. It only

	takes one approver to reject the record.
Exit Values	APPROVED REJECTED [TIMEOUT]
Exit values editable?	No
Performers	At least one
Nodes	No
Can Add Entries?	No
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Type	Or_Approval
Description	An approval step where it only takes one of the performers to approve a record before it moves to the next step. It only takes one approver to reject the record.
Exit Values	APPROVED REJECTED [TIMEOUT]
Exit values editable?	No

Performers	At least one
Nodes	No
Can Add Entries?	No
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Type	Dispatch
Description	This step is used when you want a user to decide which next step should be taken. Note that this is a view only step. A user cannot modify attributes.
Exit Values	DONE [TIMEOUT]
Exit values editable?	Yes
Performers	At least one
Nodes	No
Can Add Entries?	No
Deadline	Yes
Notifications	Yes

Script?	Yes

Step Type	Modify
Description	Use this step when you want users to modify a set of records.
Exit Values	DONE [TIMEOUT]
Exit values editable?	No
Performers	At least one
Nodes	At least one
Can Add Entries?	Yes
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Type	General
Description	Use this step when you want users to modify a set of records.
Exit Values	DONE

	[TIMEOUT]
Exit values editable?	Yes
Performers	At least one
Nodes	Yes
Can Add Entries?	Yes
Deadline	Yes
Notifications	Yes
Script?	Yes

Automated Steps

Step Type	Automated
Description	Use this step to automate a task. The logic of this step is captured in the IN() and OUT() functions of the script. Please see below for the Step Transition information that explains the execution sequence of the IN() and OUT() functions.
Exit Values	DONE
Exit values editable?	Yes
Performers	No
Nodes	Yes (It is necessary to include a Node

	in an Automated step when a workflow only contains Automated steps and the user desires to check out attributes into the workflow.)
Can Add Entries?	Yes
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Wait
Description	This step is used when you want records to wait for a user or script to move them to the next step. This step can also be used for checking-in entries back into the source container at a specific date. For instance, if you want the entries to be merged with your source container only on Nov 15, you would insert a wait step with a deadline of Nov 15 before the Success step.
Exit Values	DONE [TIMEOUT]
Exit values editable?	Yes
Performers	No

Nodes	No
Can Add Entries?	No
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Type	Make Unique
Description	Use these step when you want to remove every other copy of a record in other branches of the worfklow (usually after a split). This ensures that a record reaching this step is in this step and this step only.
Exit Values	DONE
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Merge
Description	Use this step to merge several steps after a split. Note that if you have n steps pointing to the merge step, then n copies of the record must go through the merge step before this record can move to the next step. Use the condenser to reduce the number of incoming steps...
Exit Values	DONE [TIMEOUT]
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Condenser

Description	This step is used in front a merge step to reduce the number of entries pointing to a merge step. You achieve this by having several steps pointing to the condenser…
Exit Values	DONE [TIMEOUT]
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Condenser
Description	This step is used in front a merge step to reduce the number of entries pointing to a merge step. You achieve this by having several steps pointing to the condenser…
Exit Values	DONE

Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	No
Notifications	Yes
Script?	Yes

Step Type	Partial_Undo
Description	This step is used to undo the changes done to nodes in this workflow. What really happens is that the values of these nodes are re-fetched from the main catalog when a record enters this state.
Exit Values	DONE [TIMEOUT]
Exit values editable?	No
Performers	No
Nodes	At least one. These nodes will be re-fetched from the main catalog.

Can Add Entries?	No
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Type	Nested_Workflow
Description	This step is used to include another valid workflow as a step. The exit values for this step are the same as the termination exit values for the included nested workflow.
Exit Values	SUCCESS FAILURE TIMEOUT
Exit values editable?	No
Performers	No
Nodes	No
Can Add Entries?	No
Deadline	Yes
Notifications	Yes
Script?	Yes

Step Transitions

Step transition for automated steps:	1/ The IN() function is executed (could be empty) 2/ The OUT() function is executed (could be empty). The OUT() function should set the exit value of the records. If the step only has one exit value, it is selected by default. 3/ Using the workflow graph (which maps each exit value to one or multiple next steps), records are routed to the next step
Step transition for user steps:	1/ The IN() function is executed (could be empty) 2/ The records in this step will be shown in the Advanced Content Authoring screens 3/ There, the performers will select records and assign one of the step exit values to this set of records. 4/ The IN() function is executed (could be empty). The IN() function has a chance to modify the exit value before an record really leaves this step. 5/ Using the workflow graph (wich maps each exit value to one or multiple next steps, records are routed to the next step.

- NOTE: It is possible to enable backward data movement in a workflow by inserting steps that point to an earlier step. If a step in a workflow has a deadline, the step is automatically mapped to a TIMEOUT exit value. The workflow designer may map the TIMEOUT exit value to a step in the workflow. If the workflow designer leaves the TIMEOUT exit value unmapped, the system maps the TIMEOUT exit value to the FIXIT step.

Nested Workflows

It is possible to nest a workflow within another workflow. Here is the process:

- Create both the main workflow using the above process. Save the main workflow.
- Create the nested workflow using the above process. Save the nested workflow.
- Edit the main workflow (by for example selecting the main

workflow in the workflow console, then pressing the Edit button).

- In the top toolbar, select the nested workflow in the dropdown box. Press the Add Workflow button.

  Note: It is not possible to nest a workflow of a different container type. So it is not possible to nest a hierarchy workflow within a catalog workflow.

- Map the exit values in the nested workflow to the appropriate steps in the main workflow.
- Save the main workflow.

Part V - System Administration

This section covers each of the components that are available within the System Administration.

- Audit
- DB Admin
- Performance Info
- Properties
- Log Files
- System Status
- Import Environment
- Size Distribution

Ch 21 Application Support Tools

WebSphere Product Center provides an administrator with the core tools to assist in the support, maintenance and troubleshooting of WebSphere Product Center. For example, view system log files, check system status, and analyze database performance.

Audit

WebSphere Product Center provides a log file that stores details on every recorded event within WebSphere Product Center. The Audit Log Search function allows the ability to view all activities that have been performed by a user, within a defined timeframe. The results table is sorted from the latest date to the most current date. View when a user has logged in or out, when items were modified, which roles were created, which files were imported, and more.

Searching audit logs

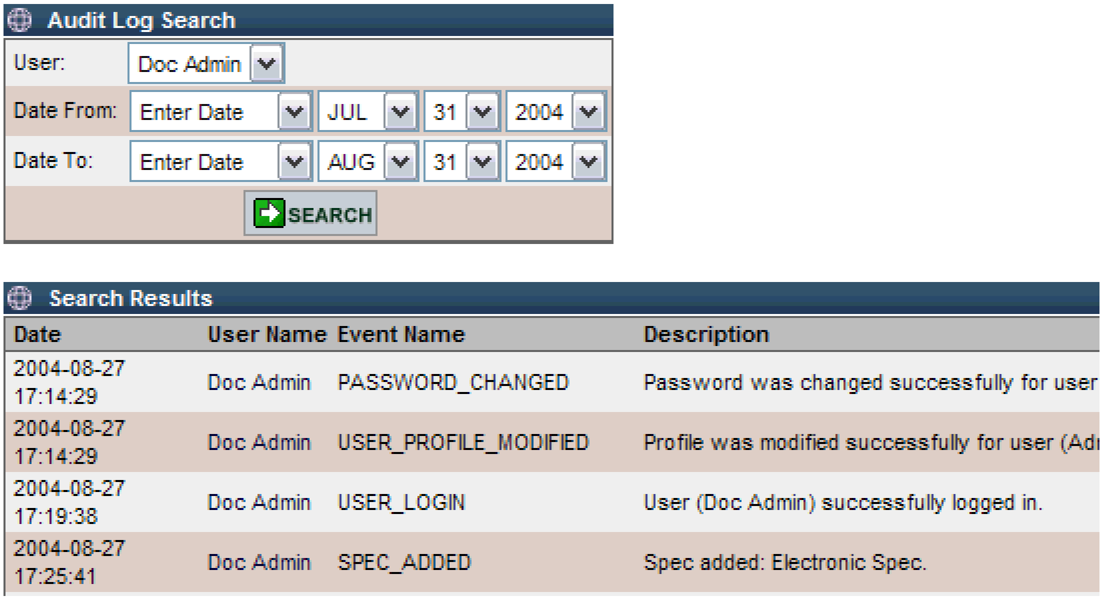
To view audit details, use the following menu path:

System Administrator > Audit

1. From the **Audit Log Search** interface, select a user to audit from the User drop-down field.

2. Select a date from the **Date From** and **Date To** drop-down field.

3. Click Search to display a list of all events associated to the specified user.



Audit Log Search

User:	Doc Admin
Date From:	Enter Date JUL 31 2004
Date To:	Enter Date AUG 31 2004

Search Results

Date	User Name	Event Name	Description
2004-08-27 17:14:29	Doc Admin	PASSWORD_CHANGED	Password was changed successfully for user
2004-08-27 17:14:29	Doc Admin	USER_PROFILE_MODIFIED	Profile was modified successfully for user (Ad
2004-08-27 17:19:38	Doc Admin	USER_LOGIN	User (Doc Admin) successfully logged in.
2004-08-27 17:25:41	Doc Admin	SPEC_ADDED	Spec added: Electronic Spec.

Figure 21.1 - Audit Log Search

DB Admin

The DB Admin component allows a qualified DBA to run simple SQL queries. Enter the query in the Sql Command box and click Save.

Warning! Be careful when using this tool as it allows the execution of commands that may jeopardize the integrity of the database. Allow access to this System Admin module to qualified users.

Running simple SQL

To used DB Admin, use the menu path:

System Administrator > DB Admin.

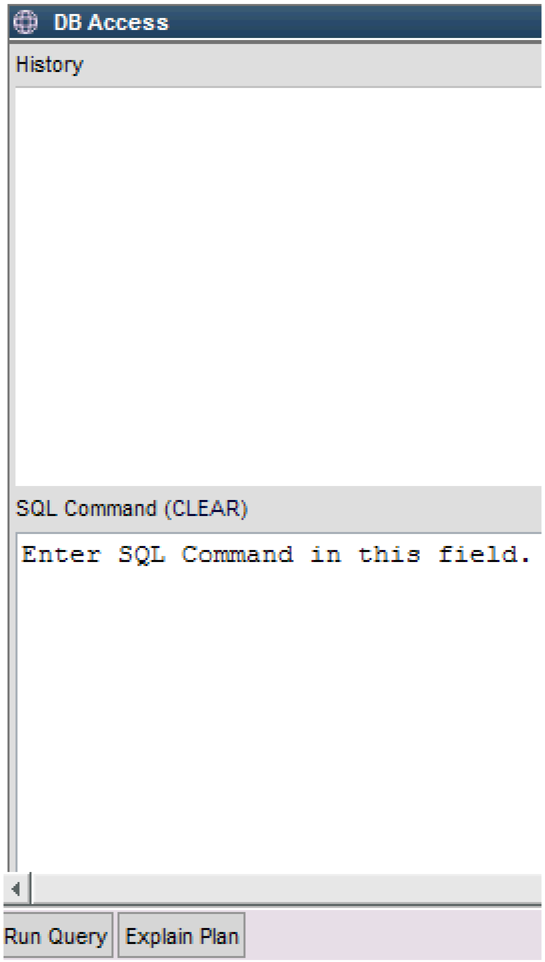


Figure 21.2 - DB Admin interface

Queries are executed against the database to pull information that is displayed in WebSphere Product Center. You can type SQL Command in the SQL Command space and click on Run Query button on the top left corner of the screen to execute the query. To find additional details click on Explain Plan button.

Performance Info

WebSphere Product Center provides several tools that allow for the analysis of application and database performance. These tools were designed in the System Administration module and should only be made available to authorized users.

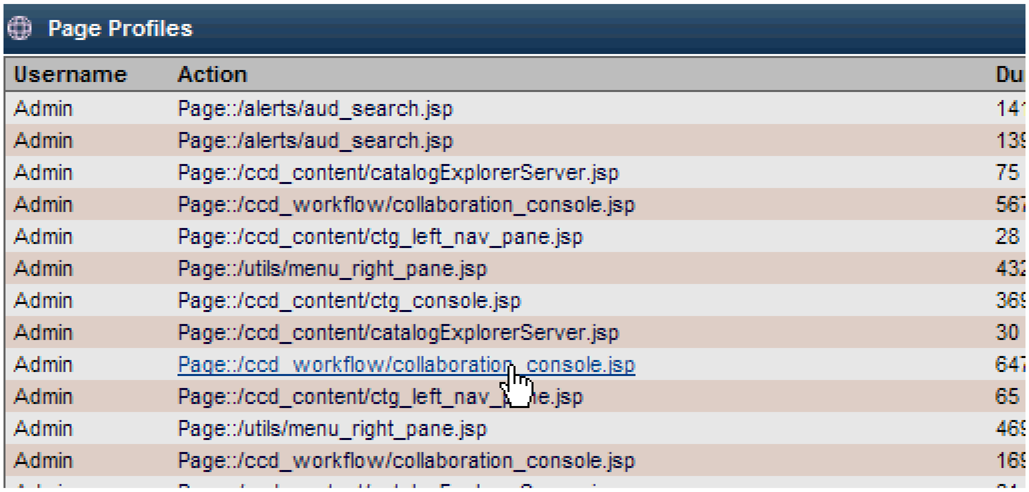
Profiling

This component provides profiling data for every page and executable within WebSphere Product Center. When first accessing this area, the Page and Executable Profiles appear in separate tables.

To view WebSphere Product Center's profiling statistics, use the following menu path:

System Administrator > Performance Info > Profiling

To view additional detail on a specific profile, double-click a page in the Action column. The page details appear in a separate page.



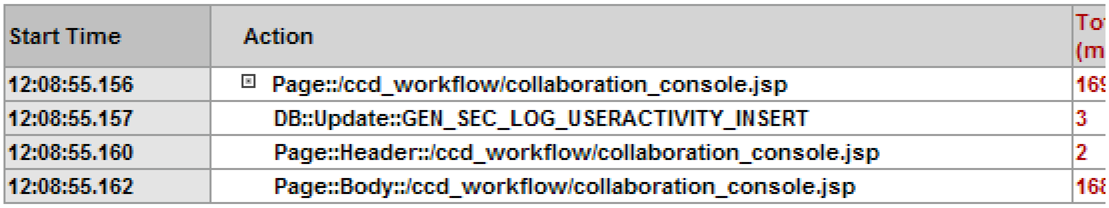
Username	Action	Du
Admin	Page::/alerts/aud_search.jsp	141
Admin	Page::/alerts/aud_search.jsp	139
Admin	Page::/ccd_content/catalogExplorerServer.jsp	75
Admin	Page::/ccd_workflow/collaboration_console.jsp	567
Admin	Page::/ccd_content/ctg_left_nav_pane.jsp	28
Admin	Page::/utils/menu_right_pane.jsp	432
Admin	Page::/ccd_content/ctg_console.jsp	369
Admin	Page::/ccd_content/catalogExplorerServer.jsp	30
Admin	Page::/ccd_workflow/collaboration_console.jsp	647
Admin	Page::/ccd_content/ctg_left_nav_pane.jsp	65
Admin	Page::/utils/menu_right_pane.jsp	469
Admin	Page::/ccd_workflow/collaboration_console.jsp	169

Figure 21.3 - Page profile details

Username	The user who accessed the page
Action	The page that was accessed
Duration	The time it took to process the page.
Start Time	The time the page was accessed

Click on any Action line to view additional details for analysis. Users can review the time it took to run an export and analyze which areas are bottlenecked.

Note: Each dash (-) represents one level underneath. Therefore, a line showing two dashes (- -) would indicate two levels of details.

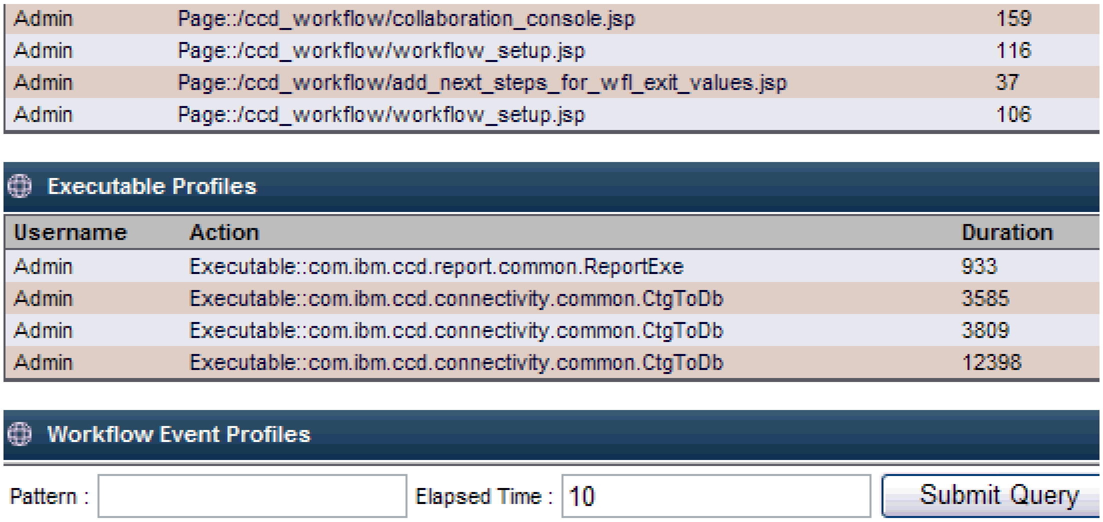


Start Time	Action	To (m
12:08:55.156	⊟ Page::/ccd_workflow/collaboration_console.jsp	169
12:08:55.157	DB::Update::GEN_SEC_LOG_USERACTIVITY_INSERT	3
12:08:55.160	Page::Header::/ccd_workflow/collaboration_console.jsp	2
12:08:55.162	Page::Body::/ccd_workflow/collaboration_console.jsp	168

Figure 21.4 - Page analysis

Start Time	The time of the event
Action	Description of the action.
Total Time (ms)	The total time it took to complete the action
# Iter.	The number of times the action had to be performed
Avg Time per Iter. (ms)	The average time it took for multiple iterations

To flush all profile details, click ☒ from any profile table heading. A confirmation dialog box appears, click OK and the information is cleared from the table and begins tracking current audit events.



Admin	Page::/ccd_workflow/collaboration_console.jsp	159
Admin	Page::/ccd_workflow/workflow_setup.jsp	116
Admin	Page::/ccd_workflow/add_next_steps_for_wfl_exit_values.jsp	37
Admin	Page::/ccd_workflow/workflow_setup.jsp	106

Executable Profiles

Username	Action	Duration
Admin	Executable::com.ibm.ccd.report.common.ReportExe	933
Admin	Executable::com.ibm.ccd.connectivity.common.CtgToDb	3585
Admin	Executable::com.ibm.ccd.connectivity.common.CtgToDb	3809
Admin	Executable::com.ibm.ccd.connectivity.common.CtgToDb	12398

Workflow Event Profiles

Pattern : Elapsed Time : 10

Figure 21.5 Profile analysis

Performance

This component is used to monitor the performance of java virtual modules by noting the time it takes for page completions. If desired, all information can be flushed from the database to test current conditions.

To monitor WebSphere Product Center's Performance, use the following path:

System Administrator > Performance Info > Performance

From the Performance Search interface, select a search value and click **Search**.

The Current Performance Result table appears with performance measurements for each JVM (Max, Min, Avg, and number of visits.

To clear all performance measurements, click Flush to DB at the bottom of the table. All details will be purged from the DB.

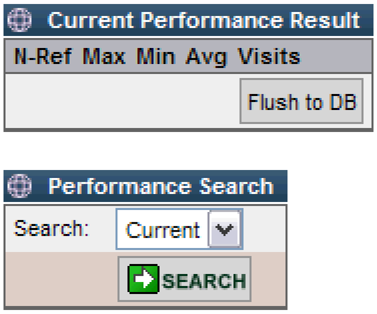


Figure 21.6 - Performance table

N – Ref	The name of the JVM
Max	The longest time it took for a JVM to Execute
Min	The shortest time it took for a JVM to execute
Avg	The average time it took for a JVM to execute
Visits	The number of visits made to the JVM

Database Performance

Check the performance of your java virtual modules by noting the times for various DB calls. This component is useful in analyzing in detail where certain processes are taking a long time, which can then be used to determine what types of changes can be made to improve performance. There is two table displayed with results: Current Performance Results and Query Performance Statistics.

For example, analyze the time it takes to process a script and find an unnecessary operation that takes a long time to run, the script can then be modified to maximize the processes time.

To monitor WebSphere Product Center's database performance, use the following path:

System Administrator > Performance Info > Database Performance

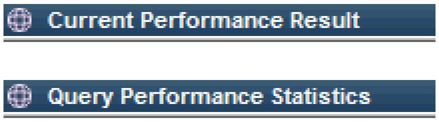


Figure 21.8 - View DB performance

Current Performance Results

Page	Name of WebSphere Product Center page
Total Time	Total time (ms) DB spend used to pull this page
Hits	How often the page was accessed
Avg. Time	The average time it took to pull the page
Last	The time it last took the DB to

	process the page
Avg. DB Time	The average time it took for the DB to process the page.
% Time in DB	The percent of the DB's time was used to pull the page
Queries	View the queries that were executed to process the page

Query Performance Statistics

The queries list in the table is hyperlinked to display the details of the queries.

Query	The name of the query that was executed
Total Time Taken	The total time it took to execute the query
Times Executed	The number of time the query was executed
Average	The average time to execute the query
Maximum	The maximum time to execute the query
Minimum	The minimum time to execute the query

Caches

An Administrator can view the details on the specs that are being cached in one instance per server instance. Through the WebSphere Product Center GUI, an Administrator is able to flush the cache when needed.

From the menu path System Administrator > Performance Info > Caches, view the Cached Specs details.

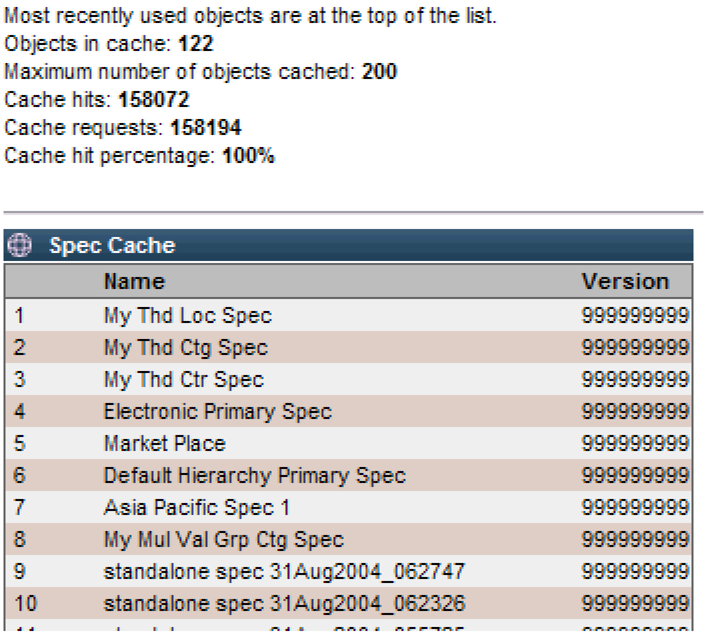
Example of a catalog using basic specs (1 & 2)

When a new catalog is viewed, the previous specs are moved down in the Cached Specs table.

When returning to the previous catalog, the previous cached specs return to the beginning position of the Cached Specs table.

When a spec is changed, the older version of the spec is not automatically removed from the cache, as some operation such as an export may be using it. Specs are only removed from the cache in the following cases:

- If they have not been among the most recently used
- The cache is full (has maximum number of entries)
- Another spec is added to the cache



Most recently used objects are at the top of the list.
Objects in cache: **122**
Maximum number of objects cached: **200**
Cache hits: **158072**
Cache requests: **158194**
Cache hit percentage: **100%**

Spec Cache

	Name	Version
1	My Thd Loc Spec	999999999
2	My Thd Ctg Spec	999999999
3	My Thd Ctr Spec	999999999
4	Electronic Primary Spec	999999999
5	Market Place	999999999
6	Default Hierarchy Primary Spec	999999999
7	Asia Pacific Spec 1	999999999
8	My Mul Val Grp Ctg Spec	999999999
9	standalone spec 31Aug2004_062747	999999999
10	standalone spec 31Aug2004_062326	999999999

Fig 21.9 - Cache statistics

Cache Size

The size of the cache is set in the common.properties configuration file.

Example:

```
# Limit for cache size for specs
# Larger will take more memory, so this should not be set too large.

max_specs_in_cache=5
```

The size of the specs cache is set according to available system resources:

- The amount of memory on the server
- The number of different specs that are commonly used

Review the Spec Cache page to examine the number of specs in the cache and if commonly used specs are falling out of the cache consistently. Then set the Spec Cache value based on current needs.

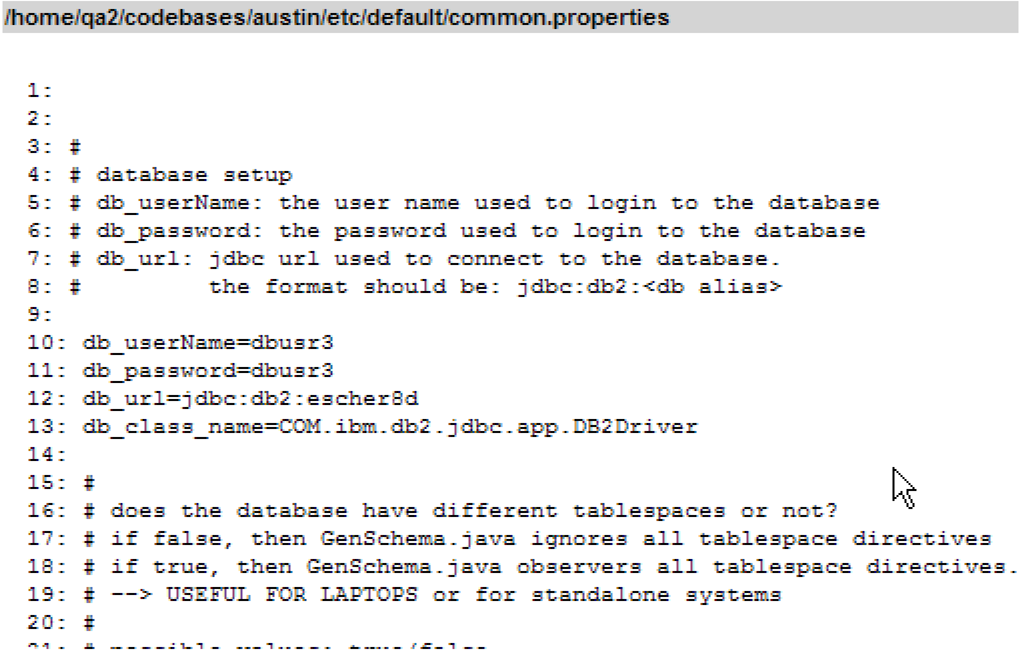
Properties

This component allows a user to view the current application configuration files, detailing the file parameters. This feature allows for a view-only display. (For more information on the WPC's configuration files, refer to the WebSphere Product Center: Installation Guide.)

To review the application's configuration files, use the following path:

System Administrator > Properties

The configuration files are for display only. Scroll through the page to view all configuration file details.



/home/qa2/codebases/austin/etc/default/common.properties

```


#
# database setup
# db_userName: the user name used to login to the database
# db_password: the password used to login to the database
# db_url: jdbc url used to connect to the database.
#         the format should be: jdbc:db2:<db alias>

db_userName=dbusr3
db_password=dbusr3
db_url=jdbc:db2:escher8d
db_class_name=COM.ibm.db2.jdbc.app.DB2Driver

#
# does the database have different tablespaces or not?
# if false, then GenSchema.java ignores all tablespace directives
# if true, then GenSchema.java observers all tablespace directives.
# --> USEFUL FOR LAPTOPS or for standalone systems
#
# possible values: true/false
```

Figure 21.10 - Review configuration files

Log Files

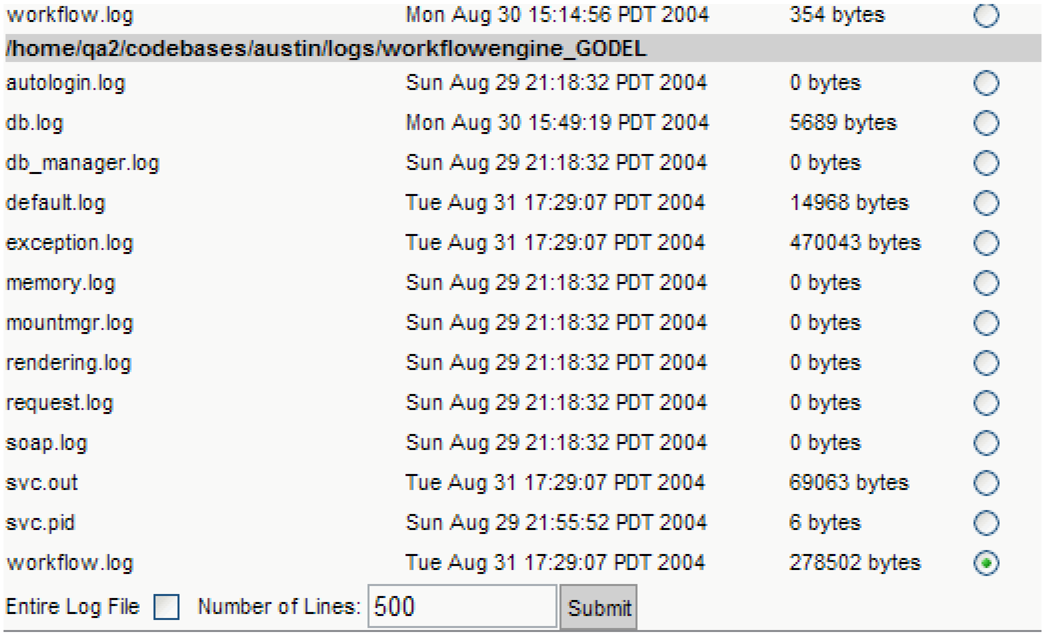
This component is useful for troubleshooting system issues. View the tail end of a log or the entire log if desired.

View log files

To view the application's log files, use the following path:

System Administrator > Log Files

Select a system log and click Submit. Users can select to view the entire file or a defined number of lines of the tail end log.



workflow.log	Mon Aug 30 15:14:56 PDT 2004	354 bytes	○
/home/qa2/codebases/austin/logs/workflowengine_GODEL			
autologin.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
db.log	Mon Aug 30 15:49:19 PDT 2004	5689 bytes	○
db_manager.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
default.log	Tue Aug 31 17:29:07 PDT 2004	14968 bytes	○
exception.log	Tue Aug 31 17:29:07 PDT 2004	470043 bytes	○
memory.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
mountmgr.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
rendering.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
request.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
soap.log	Sun Aug 29 21:18:32 PDT 2004	0 bytes	○
svc.out	Tue Aug 31 17:29:07 PDT 2004	69063 bytes	○
svc.pid	Sun Aug 29 21:55:52 PDT 2004	6 bytes	○
workflow.log	Tue Aug 31 17:29:07 PDT 2004	278502 bytes	●

Entire Log File □ Number of Lines: 500 Submit

Figure 21.11 - Select a log file to view

System Status

The complete WebSphere Product Center system consists of five types of services running concurrently.

Admin	admin - the admin server (starts/stops modules on remote machines)
Scheduler	scheduler - the scheduler (runs background jobs)
Workflow engine	workflow – WebSphere Product Center's workflow engine
Event Processor	eventprocessor - the event processor (dispatches events between all the modules)

Queue Manager	queuemanager - the queue manager (sends documents outside of WebSphere Product Center)
Application Server	appsvr - the application server (serves jsp-Java Server Pages)

Each service can be controlled through the WebSphere Product Center GUI and status information can be obtained for each application service. Users can view the performance metrics displayed in a table format.

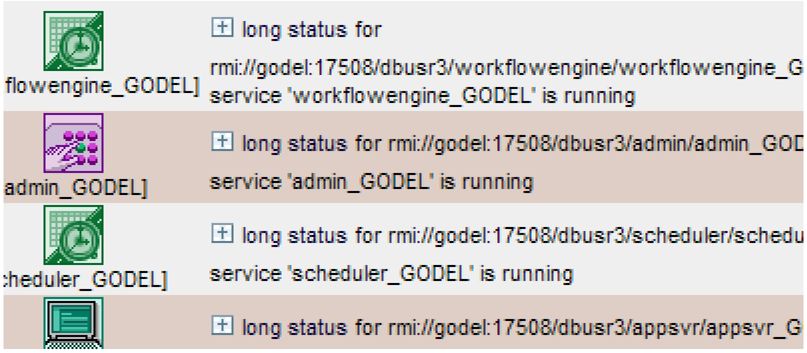
Note: The status provided is the WebSphere Product Center GUI can also be obtained using the RMI Status scripts executed from the command line.

View Service Status

To view the Service Status table, use the menu path:

System Administrator > System Status

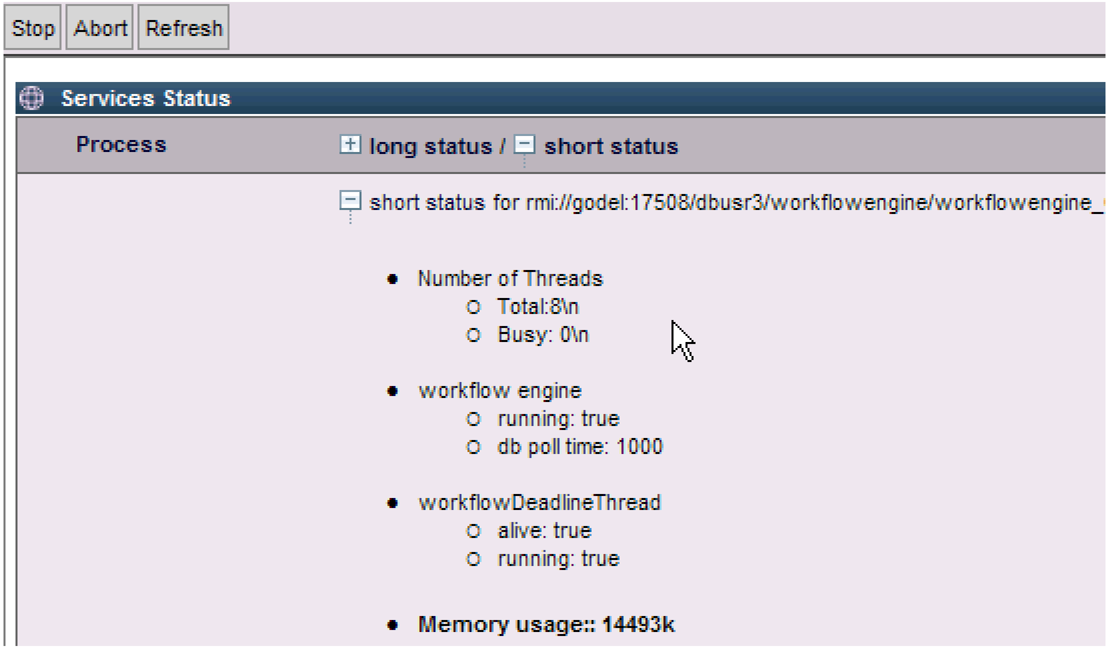
The short status of each service is displayed in the Service Status table. To view expanded details for a service, click the **long status** hyperlink.



flowengine_GODEL]	⊞ long status for rmi://godel:17508/dbusr3/workflowengine/workflowengine_G service 'workflowengine_GODEL' is running
admin_GODEL]	⊞ long status for rmi://godel:17508/dbusr3/admin/admin_GOD service 'admin_GODEL' is running
:heduler_GODEL]	⊞ long status for rmi://godel:17508/dbusr3/scheduler/schedu service 'scheduler_GODEL' is running
	⊞ long status for rmi://godel:17508/dbusr3/appsvr/appsvr_G

Figure 21.12 - Short status of a service

The long status of a service displays information such as the connection pool status and details on each db thread.



short status for rmi://godel:17508/dbusr3/workflowengine/workflowengine_

- Number of Threads
  - Total:8\n
  - Busy: 0\n
- workflow engine
  - running: true
  - db poll time: 1000
- workflowDeadlineThread
  - alive: true
  - running: true
- **Memory usage:: 14493k**

Figure 21.13 - Service status information

Stopping Service

To stop a service, select a service by checking the box next to the service name and clicking the Stop button.

Aborting Services

To abort a service, select a service by checking the box next to the service name and clicking the Abort button.

Important Note On Abort and Stop

Which one should be used? Stop or abort?

- **Abort** - Guarantees that the service will be shut down, but it cannot guarantee that a currently running task will not be interrupted.
- **Stop** - Guarantees that "if" the service is stopped, it will be stopped smoothly after every currently running task has been stopped first.

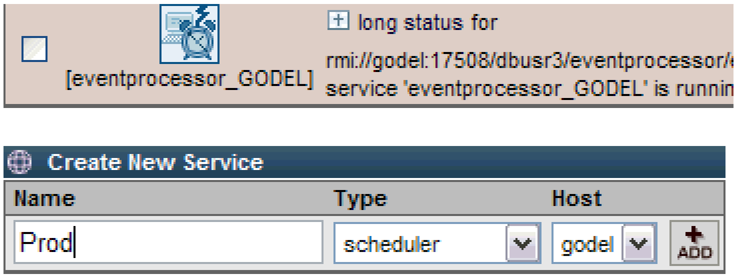
Refreshing status information

To obtain the most current status for a service, select a service by checking the box next to the service name and click the Refresh button.

Creating a New Service

Services can run in a cluster of workstations. The different machines in the cluster are defined in the file **admin_properties.xml** (For more information, refer to the WebSphere Product Center: Installation Guide). If desired, a user can define a service on a different host through the WebSphere Product Center GUI.

Below the Service Status table is a **Create New Service table**. Enter a name for the new service, the service type, select a host, and click the **ADD+** button.



□ [eventprocessor_GODEL]

long status for
rmi://godel:17508/dbusr3/eventprocessor/
service 'eventprocessor_GODEL' is runnin

Name	Type	Host
Prod	scheduler	godel

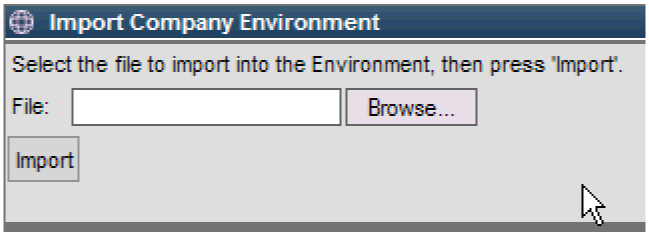
Figure 21.14 - Create a new service

It is not possible to create more than one instance of a service on a machine. If no service name is specified, the default name is used:

- "scheduler" for the scheduler.
- "eventprocessor" for the event processor.
- "queuemanager" for the queue manager.

Import Environment

If a company environment has been packaged for import, the Import Environment component allows a user to browse for the packaged company environment file and import it into the current company. This feature can be used to port a company that was created in a test environment into a production environment.


Import Company Environment

Select the file to import into the Environment, then press 'Import'.

Figure 21.15 - Importing a company environment

Size Distribution

The WebSphere Product Center database designates a tablespace for all the files stored in the Document store. When a file is stored, a new record in the Oracle DB is created and the file is stored as a BLOB (Binary Large Object) file.

Viewing distribution files in document store

To view the distribution of files within the document store, use the following path:

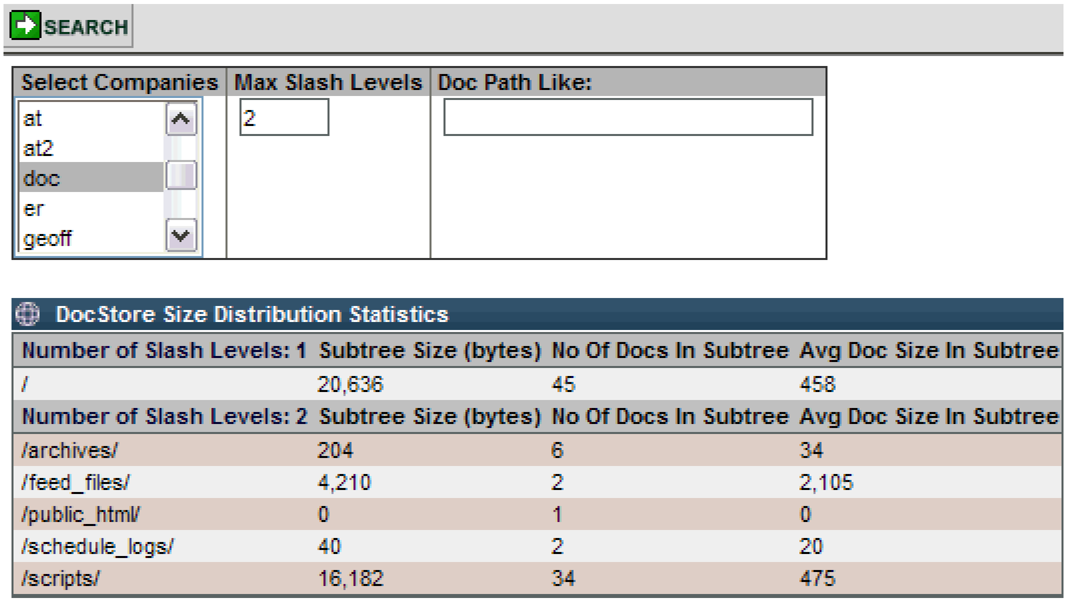
System Administrator > Size Distribution

1. Select a company or set of companies from the **Select Companies** selection box.

2. Determine the level of subdirectories to search and enter the number in the **Max Slash Levels**.

3. If desired, view file statistics of a specific path. Enter a path in the **Doc Path Like** field.

4. Click the **Search** button to view the size distribution statistics.



Select Companies	Max Slash Levels	Doc Path Like:
at at2 doc er geoff	2	

DocStore Size Distribution Statistics

Number of Slash Levels: 1	Subtree Size (bytes)	No Of Docs In Subtree	Avg Doc Size In Subtree
/	20,636	45	458
Number of Slash Levels: 2	**Subtree Size (bytes)**	**No Of Docs In Subtree**	**Avg Doc Size In Subtree**
/archives/	204	6	34
/feed_files/	4,210	2	2,105
/public_html/	0	1	0
/schedule_logs/	40	2	20
/scripts/	16,182	34	475

Figure 21. 16 - DocStore Size Distribution Statistics

Notices

IBM may not offer the products, services, or features discussed in this document in all countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Burlingame Laboratory
Director IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not necessarily tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information may contain examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples may include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries, or both:

IBM
the IBM logo
AIX
CrossWorlds
DB2
DB2 Universal Database
Domino
Lotus
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.

This product includes software (Axis, Jakarta Commons Collection, Jakarta Commons DBCP, Jakarta Commons Pool, Jakarta Commons Regular Expression, Log4J, Regexp, Xalan, Xerces, XML4J) developed by the Apache Software Foundation (http://www.apache.org/).

Apache Software License
Version 1.1
Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER

CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

a. You must give any other recipients of the Work or Derivative Works a copy of this License; and

b. You must cause any modified files to carry prominent notices stating that You changed the files; and

c. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

d. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

CUP Parser Generator Copyright Notice, License, and Disclaimer

Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

JLEX COPYRIGHT NOTICE, LICENSE AND DISCLAIMER.

Copyright 1996-2003 by Elliot Joel Berk and C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Java is a trademark of Sun Microsystems, Inc. References to the Java programming language in relation to JLex are not meant to imply that Sun endorses this product.