Contents

1. Copyright Notice ......................................................................................................................... 4
2. Why Customize WebCenter? ...................................................................................................... 6
3. Where to Start the Customization? .............................................................................................. 7
4. Applying your WebCenter 7.X customization to WebCenter 10 or 12 ........................................ 8
5. Customizing the Look and Feel of WebCenter .......................................................................... 10
   5.1 Customizing Logos and their Hyperlinks ............................................................................ 10
   5.1.1 Customize Logos and Hyperlinks .................................................................................. 10
   5.1.2 Customize Logo for User or Company ......................................................................... 12
   5.2 Change the Color Scheme .................................................................................................... 14
   5.2.1 Color Scheme Reference ............................................................................................... 14
   5.3 Change the Menu Hover Color ........................................................................................... 15
   5.4 Change the Fonts ................................................................................................................. 15
   5.4.1 Font Scheme Reference .................................................................................................. 16
   5.5 Customize Static Text .......................................................................................................... 17
   5.5.1 What Static Text Items Can You Customize? ................................................................ 17
   5.5.2 Customize a Static Text Item ......................................................................................... 18
   5.5.3 Example: Customizing Static Text ................................................................................. 19
   5.6 Customize the Language Files ............................................................................................. 20
6. Customizing the WebCenter Menus ......................................................................................... 23
   6.1 Create a Custom Menu ......................................................................................................... 23
   6.1.1 Download an Existing Menu .......................................................................................... 24
   6.1.2 Structure of the XML Menu File .................................................................................... 24
   6.1.3 Remove Unused Menu Items ......................................................................................... 27
   6.1.4 Add Custom Menu Items ............................................................................................... 28
   6.2 Examples ............................................................................................................................... 29
   6.2.1 Going to Projects I Am invited to in One Click ............................................................. 29
   6.2.2 Customizing the Interface for Project Requests ............................................................. 30
   6.2.3 Removing the Keyword Search ...................................................................................... 37
   6.3 Will My Customizations Survive New Updates? ................................................................... 37
7. Customizing the Attributes ....................................................................................................... 38
   7.1 Create Image Label for an Attribute .................................................................................... 38
   7.2 Create HTML Info for an Attribute ..................................................................................... 39
8. Customizing the E-Mail Notifications ....................................................................................... 41
   8.1 Technology Used: XSLT ....................................................................................................... 41
   8.2 Migration of customized email from WebCenter 7.X to WebCenter 10+ ............................. 41
8.3 The E-Mail Template File Structure.......................................................................................................................... 45
8.4 Customizing E-Mail Language Files.......................................................................................................................... 46
8.5 E-Mail Notification Templates In Detail.................................................................................................................. 46
8.6 Customizing the Notification Templates Step by Step.............................................................................................. 47
  8.6.1 What You Need to Know Before You Begin........................................................................................................ 47
8.7 Start from the example................................................................................................................................................ 48
  8.7.1 Basic customization................................................................................................................................................. 49
  8.7.2 Advanced customization..................................................................................................................................... 53
1. Copyright Notice

© Copyright 2012 Esko Software BVBA, Gent, Belgium

All rights reserved. This material, information and instructions for use contained herein are the property of Esko Software BVBA. The material, information and instructions are provided on an AS IS basis without warranty of any kind. There are no warranties granted or extended by this document. Furthermore Esko Software BVBA does not warrant, guarantee or make any representations regarding the use, or the results of the use of the software or the information contained herein. Esko Software BVBA shall not be liable for any direct, indirect, consequential or incidental damages arising out of the use or inability to use the software or the information contained herein.

The information contained herein is subject to change without notice. Revisions may be issued from time to time to advise of such changes and/or additions.

No part of this document may be reproduced, stored in a data base or retrieval system, or published, in any form or in any way, electronically, mechanically, by print, photoprint, microfilm or any other means without prior written permission from Esko Software BVBA.

This document supersedes all previous dated versions.

PANTONE®, PantoneLIVE and other Pantone trademarks are the property of Pantone LLC. All other trademarks or registered trademarks are the property of their respective owners. Pantone is a wholly owned subsidiary of X-Rite, Incorporated. © Pantone LLC, 2012. All rights reserved.

This software is based in part on the work of the Independent JPEG Group.

Portions of this software are copyright © 1996-2002 The FreeType Project (www.freetype.org). All rights reserved.

Portions of this software are copyright 2006 Feeling Software, copyright 2005-2006 Autodesk Media Entertainment.

Portions of this software are copyright ©1998-2003 Daniel Veillard. All rights reserved.

Portions of this software are copyright ©1999-2006 The Botan Project. All rights reserved.

Part of the software embedded in this product is gSOAP software. Portions created by gSOAP are Copyright ©2001-2004 Robert A. van Engelen, Genivia inc. All rights reserved.

Portions of this software are copyright ©1998-2008 The OpenSSL Project and ©1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

Adobe, the Adobe logo, Acrobat, the Acrobat logo, Adobe Creative Suite, Illustrator, InDesign, PDF, Photoshop, PostScript, XMP and the Powered by XMP logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Microsoft and the Microsoft logo are registered trademarks of Microsoft Corporation in the United States and other countries.

SolidWorks is a registered trademark of SolidWorks Corporation.

Portions of this software are owned by Spatial Corp. 1986 2003. All Rights Reserved.

JDF and the JDF logo are trademarks of the CIP4 Organisation. Copyright 2001 The International Cooperation for the Integration of Processes in Prepress, Press and Postpress (CIP4). All rights reserved.
The Esko software contains the RSA Data Security, Inc. MD5 Message-Digest Algorithm.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems in the U.S. and other countries.

Part of this software uses technology by BestTM Color Technology (EFI). EFI and Bestcolor are registered trademarks of Electronics For Imaging GmbH in the U.S. Patent and Trademark Office.


All other product names are trademarks or registered trademarks of their respective owners.

Correspondence regarding this publication should be forwarded to:

Esko Software BVBA
Kortrijksesteenweg 1095
B – 9051 Gent
info.eur@esko.com
2. Why Customize WebCenter?

You can customize the look and feel of WebCenter. This information describes what can, and what cannot be modified, and includes instructions on how to make those changes.

Whenever you let your customers or partners work with your WebCenter-enabled website, you spread a message to them. Therefore, most WebCenter customers want to brand their site and take the best advantage of the time the WebCenter user spends using the site.

WebCenter allows changing color schemes, logos and images, and virtually every piece of text. You can even change the logo based on who logs in. This way, you can make the user feel treated personally. WebCenter gives you a high amount of flexibility in changing the look and feel. You can combine this with custom menus to change the navigation within the site.

Esko cannot guarantee robust functionality and consistent behavior when upgrading when you start changing the way WebCenter reacts to customer actions.

Some changes to the navigation are built into the product. Good examples are customizable menus, the choice of what document types are allowed, whether documents are shown in a list view or a grid view, or what CAD outputs are available. In future versions, we’ll further increase the flexibility. If you have suggestions, please contact Esko via your local Customer Services representative.
3. Where to Start the Customization?

**Customizing the Look and Feel of WebCenter**

The user interface to customize WebCenter's look and feel is the `customizationConfig.xml` file. All you need to do to apply your customizations is to modify this file accordingly and save it.Reload your page, and your change should be visible.

The `customizationConfig.xml` file should be placed into the customization directory on your Web server instance, which is `<WebCenter_Instance>\custom` directory (such as `c:\Artios\WebCenter\WebServer\tomcat\webapps\WebCenter_instance\custom`).

See *Customizing the Look and Feel of WebCenter* on page 10 for details.

**Customizing the WebCenter Menus**

You can customize the WebCenter menus to tailor what users see to what they need to do in WebCenter.

You can manage custom menus through the WebCenter user interface. Log in as an Admin, and go to Admin > Preferences > Menus.

See *Customizing the WebCenter Menus* on page 23 for details.

**Customizing the E-Mail Notifications**

All the e-mail notifications' related files are stored on the Application Server in the following path: `C:/Artios/WebCenter/ApplicationServer/EmailNotifications`.

Inside the root directory you can find two subdirectories - Default and Custom. Actually you will notice that there is also a third one - example - with an example customization.

To customize the e-mail notifications, modify the files in the Custom subdirectory.

See *Customizing the E-Mail Notifications* on page 41 for details.
It is possible to re-use your WebCenter 7.X customization in your WebCenter 10 or 12 installation. This section explains the changes that need to be made to make the customization work properly in WebCenter 10 or 12, along with new customization features that come with WebCenter 10+.

Moving your customization from WebCenter 7.X to WebCenter 10 or 12

Before upgrading from WebCenter 7.X, please make sure your custom folder is backed up in a different location. You can find the custom folder on your WebCenter Web Server: <WebCenter_Instance>\custom. You may have used different custom folders for different WebCenter instances.

After you've created your new WebCenter instance, copy the contents of the custom folder into the new, empty custom folder. The contents of the custom folder typically includes folders such as "images", "text" and "languages".

Attention:

When copying folders from a different operating system into your custom folder (especially from Mac OS to Windows), make sure the contents aren't encrypted.

1. Right-click the old folder you are copying into the custom folder, and select Properties.
2. In the General tab, click Advanced...
3. Make sure Encrypt contents to secure data is not selected and click OK.

Encrypted folders and files have green names, and can cause problems in your customization.

If you would like to make a totally new customization at this point, you can copy the contents of the custom_example folder into the custom folder and start making changes to this file.

In case you have copied your old customization into the new custom folder, please have a look at the new customizationConfig.xml file. You can find this file in the <WebCenter_Instance>\custom_example folder. Compare this new file to your old customizationConfig.xml file. The recommended way of proceeding is to copy the customized settings from your old file into the new one and use this new file in your custom folder. This way, you can use the new features that have been added.
Note: There may be sections in the customizationConfig.xml that are not actually used, these sections are typically greyed out in XML editors and can also be recognized because they are between: <!-- and -->

Changes to the customizationConfig.xml file

The new customizationConfig.xml file contains some additional features. One of the new items is MenuBackground, this section determines the background color of the menu when you hover over it. In this new section you can specify an image. Some examples can be found in the <WebCenter_Instance>\images folder. The name of these images is typically: top_navigation_selected_[color].png. In order for this customization to work in every browser, the image you would like to use also needs to be copied into the <WebCenter_Instance>\custom\styles\images folder. In WebCenter 10 and 12, some additional colors have been added. The new customizationConfig.xml file now provides additional information about where the colors are used so you can easily apply the desired values. For example, in WebCenter 12 you can now change the color of the text used in the WebCenter header.

The tab image section has been removed from the new customizationConfig.xml file. This image is no longer used since WebCenter 10. Also, some colors are not used anymore or are used very infrequently. If so, this is clearly stated in the Colors table.

Changes required to make your customization work

The old customizations you have created will largely work in WebCenter 10 or 12. After you have copied your old customizations into the new customizationConfig.xml file and possibly made some additional changes, it is a good idea to go over the Colors section. Since various colors have changed and new colors have been added, the color settings may have to be adapted.

Attention: After applying your customization, you may need to restart the TomCat service.
5. Customizing the Look and Feel of WebCenter

The user interface to customize WebCenter’s look and feel is the customizationConfig.xml file. All you need to do to apply your customizations is to modify this file accordingly and save it. Reload your pages to see your changes reflected.

Any settings from the customizationConfig.xml file overrule default WebCenter application settings.

The customizationConfig.xml file should be placed into the customization directory on your Web server instance, which is <WebCenter_Instance>\custom directory (such as c:\Artios \WebCenter\WebServer\tomcat\webapps\WebCenter_instance\custom).

In the following section we will list what is possible to customize and how to achieve that.

For your convenience there is a <WebCenter_instance>\custom_example directory in the web server instance including a customization example. This directory contains a commented customizationConfig.xml file and necessary additional files which will help you easily try most of the customizations described in this documentation.

To use the example files, simply copy content from this directory into the <WebCenter_Instance>\custom directory.

5.1 Customizing Logos and their Hyperlinks

A number of logos are displayed in WebCenter, many of them with a clickable hyperlink. You can change the logo image and the hyperlink target.

5.1.1 Customize Logos and Hyperlinks

In a standard WebCenter installation, the login window will look more or less like in the picture below (the actual appearance depends on your screen settings, the browser used and the version of WebCenter you have installed).
There are four images (and their hyperlinks) on this page you can change:

- The top logo @WebCenter. This logo appears on every page. It can be adapted by replacing it with any image of approximately the same height (40 pixels). In case you use an AppLogo image higher than 40px, you will stretch the top header along with this image. Once the user is logged in (and not before that), this image can be further changed to reflect your relationship. It is common practice to give the top logo a hyperlink to a more general web page (typically your company’s homepage).

- The light watermark can be replaced with any image or with no image at all. This watermark ONLY appears on the login page. It has no hyperlink.

- To the bottom left, there is a placeholder for another image (ProductLineLogo). The standard installation does not put anything there.

- To the bottom right is the Esko logo (and its hyperlink to www.esko.com). It can be replaced just like any other logo. We appreciate it when customers keep our logo visible. Esko stands for quality, innovation and communication. Your site will be more appreciated if you reinforce it by showing what technology you are using for servicing your customers and partners.

1. To customize the logos on the log on page, add the following tags to customizationConfig.xml:

```xml
<AppLogo
    RelPath="images\AppLogo.png"
    href="http://www.esko.com"/>

<BackgroundImage
    RelPath="images\BackgroundImage.png"/>

<ProductLineLogo
    RelPath="images\ProductLineLogo.png"/>
```
The following attributes can be used to provide the required image and optionally set the hyperlink:

- **RelPath** – defines the path of the logo image relatively to the custom directory.
- **href** – if the logo is also a hyperlink, you can specify its target.

2. To customize the logo for the normal pages, you can also use the following tag:

```html
<AppLogo
  RelPath="images\AppLogo.png"
  href="http://www.esko.com"/>
```

After replacing the logos, the following might be the result. Note that in this case the header and footer colors have also been customized.

### 5.1.2 Customize Logo for User or Company

Besides from customizing different WebCenter instances, it is also possible to create different customizations inside the same instance. Using this type of customization you are able to customize the **AppLogo** image, in the top left corner, for specific users or companies.
1. Inside your WebCenter instance’s custom folder, you can create individual folders. Each of these folders should contain their own customizationConfig.xml file and a folder called images in which the logo image is placed. These folders should be named according to the following conventions:
   - **USER** to apply for a user with a specific username
   - **Company_** to apply for a specific company
   - **Company_Location** to apply for a specific company and location

2. The customizationConfig.xml should only contain one tag:

   ```xml
   <AppLogo
     RelPath="images\AppLogo.png"
     href="http://www.esko.com"/>
   ```

   The following attributes can be used to provide the required image and optionally set the hyperlink:
   - **RelPath** – defines the path of the logo image relatively to the custom directory.
   - **href** – if the logo is also a hyperlink, you can specify its target.

In the example a special folder "ESKO" has been created. This folder contains a customizationConfig.xml file and an images folder with a different logo image. When the user with username "ESKO" now logs in, he will see his customized AppLogo image. All the other users will still see the image used in the top level customization.
5.2 Change the Color Scheme

The WebCenter color scheme has been chosen to be business-oriented and to reflect Esko’s corporate identity.

Many companies want to use their company colors in WebCenter. This is perfectly possible: the application uses several colors consistently throughout the application and each one of them can be replaced with another color. Use standard HTML-formatted color definition (e.g. #334444).

Tip:
The color definitions consist of 6 hexadecimal numbers, 2 for Red, 2 for Green, and 2 for Blue. The easiest method for determining the hexadecimal RGB values of a color is to create it in the Color Picker in Photoshop. You can copy and paste the hexadecimal values from the # field.

- Add the following tag to the customizationConfig.xml file:

```xml
<Colors>
  <Color id="1" hexVal="FF0000"/>
</Colors>
```

5.2.1 Color Scheme Reference

<table>
<thead>
<tr>
<th>ID</th>
<th>Color Name</th>
<th>Used in</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>esko-green (#AABB11)</td>
<td>top thin stripe (login page, welcome page, help pages); only limited use in WebCenter 10+</td>
</tr>
<tr>
<td>2</td>
<td>green1 (#749412)</td>
<td>text link actions</td>
</tr>
<tr>
<td>3</td>
<td>esko-grey (#334444)</td>
<td>main text</td>
</tr>
<tr>
<td>4</td>
<td>grey1_background (#EEEEEE)</td>
<td>thin light grey lines</td>
</tr>
<tr>
<td>5</td>
<td>grey2_background (#CCCCCC)</td>
<td>thin grey lines</td>
</tr>
<tr>
<td>6</td>
<td>grey3_background (#E6E6E6)</td>
<td>old table header background; only limited use in WebCenter 10+</td>
</tr>
<tr>
<td>7</td>
<td>white_background (#ffffff)</td>
<td>page content background, alternating rows, etc. - do not change in WebCenter 10+</td>
</tr>
<tr>
<td>8</td>
<td>folder (#ddeeef)</td>
<td>project folder selected</td>
</tr>
<tr>
<td>9</td>
<td>red_medium_bold (#FF0000)</td>
<td>overdue strong warning</td>
</tr>
<tr>
<td>10</td>
<td>orange_medium_bold (#FF7700)</td>
<td>overdue warning</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>unused in WebCenter 10+</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>unused in WebCenter 10+</td>
</tr>
</tbody>
</table>
5.3 Change the Menu Hover Color

Since WebCenter 10, it is possible to customize the menu hover color.

1. Add the following tag into the customizationConfig.xml file to change the menu hover color.

```xml
<MenuBackground RelPath="images/top_navigation_selected_blue.png"/>
```

2. Please make sure the file you selected in the previous step is available in both: `<WebCenter_Instance>\images` and `<WebCenter_Instance>\custom\styles\images` folders of the instance you would like to customize. You may need to copy the file into the latter manually for the menu customization to work on all browsers.

After customizing the menu color, the following might be the result. Note that in this case the header color has also been customized.

5.4 Change the Fonts

Fonts in WebCenter are based on a similar concept as the color scheme. One font in six sizes is consistently used through the application.

The default font sizes were chosen to optimally fit into the various WebCenter pages, so you probably wouldn’t need to change them unless you use different font that might look too small or too big.
Note: If you do decide to adjust the font size, we suggest that you adjust all font sizes proportionally if at all possible.

1. Add the following tag into the customizationConfig.xml file to change the font family to Courier New and increase one of the font sizes.

```xml
<Font name="Courier New">
  <Size id="4" pxVal="20"/>
</Font>
```

2. Add more Size tags to also change other font sizes. Normally you would probably want to adjust none or all of the sizes.

### 5.4.1 Font Scheme Reference

<table>
<thead>
<tr>
<th>ID</th>
<th>Default size</th>
<th>Used in</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>Links in the header like Log off, Contact etc., and also text in the footer: “WebCenter by Esko…”</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>Most of the dynamic text and labels such as project and document names, text within the table cells, document folder text, etc.</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>Menu text, button text, many links, etc.</td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>Static marketing text on the login page if not customized</td>
</tr>
<tr>
<td>5</td>
<td>16</td>
<td>Headings under the header such as Favorite Projects, My Work, Log In, etc.</td>
</tr>
<tr>
<td>6</td>
<td>28</td>
<td>Reserved; not used in the current version</td>
</tr>
<tr>
<td>7</td>
<td>20</td>
<td>Large project and document headers</td>
</tr>
</tbody>
</table>
5.5 Customize Static Text

There are a number of places in WebCenter where static text such as a marketing message is displayed. The software provides a straightforward mechanism for you to replace the text with your own HTML pages.

5.5.1 What Static Text Items Can You Customize?

Currently it is possible to replace the following (see screen shots below):

- Welcome message on the login page.
- Marketing message on the login page.
- Marketing message on the welcome page.
- Contact page (linked on top of each page)
5.5.2 Customize a Static Text Item

There are some limitations you need to take into account when preparing your HTML page. You need to make sure that your HTML page body is also valid XHTML. Therefore, the rules for HTML tags are a bit stricter. There are many tools and online Web pages where you can validate your HTML, such as http://validator.w3.org/. If the document body is not valid XHTML you may encounter runtime errors when accessing your customized page.

Note that the default WebCenter stylesheet is applied to the whole page, as well as to your custom HTML static text. That might be confusing because your text could look different than expected when inserted as static text on the page. For example, links might look bigger. To avoid such effects, make inline style references within your HTML. For example:
• To replace the pieces of static text, add the following tags into the customizationConfig.xml file:

```xml
<StaticText>
  <LoginWelcomeMsg RelPath="text/loginWelcomeMsg.html"/>
  <LoginMarketingMsg RelPath="text/loginMarketingMsg.html"/>
  <HomeMarketingMsg RelPath="text/homeMarketingMsg.html"/>
  <ContactPage RelPath="text/contactPage.html"/>
</StaticText>
```

Each of the StaticText sub-tags replaces the corresponding static text with your custom HTML document. Using the RelPath attribute, specify the path to your HTML document relative to the custom directory.

### 5.5.3 Example: Customizing Static Text

Let’s look at the following HTML with which we want to replace the login page welcome message:

```html
<html>
  <body>
    <h1 style="color:#79bcff;font-family:Verdana,Arial,Helvetica,sans-serif;font-size:25px;">Welcome to WebCenter 12!</h1>
  </body>
</html>
```

Now if you put the content of our HTML into the XHTML document as shown below and it passes the validation test, WebCenter should display your HTML correctly.

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
  <head>
    <title>XHTML should have head and title tag</title>
  </head>
  <body>
    <h1 style="color:#79bcff;font-family:Verdana,Arial,Helvetica,sans-serif;font-size:25px;">Welcome to WebCenter 12!</h1>
  </body>
</html>
```

Note that all relative references that you use in HTML should be relative to the Web server root directory which is the WebCenter instance directory, such as WebCenter_Inst. Below is an example of a customized login page marketing message that has a reference to an image.

```html
<html>
  <head>
    <title>Login Page Marketing Message</title>
  </head>
  <body>
    <h1 style="color:#79bcff;font-family:Verdana,Arial,Helvetica,sans-serif;font-size:25px;">
```

```html
</html>
```
5.6 Customize the Language Files

You can customize your own language files or add completely new language translations. The example below shows how to modify the customization configuration file to modify the default English language translation:

1. In the customizationConfig.xml file, add the following language tag:

```xml
<Languages RelPath="languages">
```
The `RelPath` attribute specifies the directory where the system looks for customized language files. `RelPath` should be specified relative to the custom directory. In this case, you are referring to the `languages` subdirectory in the custom directory.

The `Language` sub-tag has the same attributes as the corresponding tags in the default `config.xml` file. Also, the naming convention and structure of the language files are the same as for the standard language files.

2. Edit the `wcstrings_en.xml` file in the `languages` subdirectory to reflect your new language strings.

If you are not familiar with the default language files structure and usage, the easiest method is to copy the language file (from the default location `<WebCenter_Instance>\languages`) you want to modify to a subdirectory of the custom directory, such as `wcstrings_en.xml` in custom \languages. Edit the file and remove all tags except the ones you want to change.

The edited result could look like the example below:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<language xml:lang="en">
  <application name="JSP_Strings">
    <lang_string key="WCAUTHOR" value="WebCenter is another awesome product created by"/>
    <lang_string key="WELCOME" value="Information"/>
    <lang_string key="WELCOME_TAG" value="Information"/>
    <lang_string key="YOURE_LOGGED_IN_AS" value="Welcome to WebCenter! Have a wonderful day,"/>
  </application>
</language>
```

As you can see, we have changed the 'logged in as' text and the text in the lower left corner. Also the 'Welcome' link has changed to 'Information'. The picture below shows the result of our customization.
Welcome to WebCenter

WebCenter is a unique and powerful web-based tool for preproduction approvals and project lifecycle management. The highlights of WebCenter include:
- Easy & efficient online viewing and approval
- Efficient online tools & approval frameworks
- Integrated as locates workflow solutions and tools

The different modules in WebCenter have been developed focusing on the needs of the entire packaging supply chain. The vast experience and expertise of Esko in the field of packaging & print industry is incorporated in this great online tool.

The powerful modules in WebCenter support different needs of the packaging supply chain such as:
- Online graphical review & approval
- Online briefing & review portal with workflow automation
- Packaging intelligence tools & service
- Packaging end-to-end solutions

WebCenter as an online Review & Approval System

WebCenter as an online briefing & review portal on Automation Engine

WebCenter is another awesome product created by ESKO - Version 12.0 - Build #122.
6. Customizing the WebCenter Menus

You can customize the WebCenter menus to tailor what users see to what they need to do in WebCenter.

This way, you can make sure their user experience is as smooth as possible.

You can:

- remove unused menu items to simplify the user interface,
- add menu items that link to frequently used WebCenter pages not already part of a menu,
- change which page a menu item links to.

You can assign custom menus to users and / or companies (see "Edit a User" and "Edit a Company" in the Administration Guide).

You can manage custom menus through the WebCenter user interface. Log in as an Admin, and go to Admin > Preferences > Menus.

The Menus page contains the Default menu, and any custom menu you have already created.

You cannot edit or delete the Default menu, but you can use it as a basis for creating custom menus.

Tip:
To see what a menu looks like, click on its name. This shows the menu entries too.

6.1 Create a Custom Menu

You can create a custom menu for each category of users, to make it more convenient for them.

Note: Menus are XML files, so you will need an XML editor (you can find various freeware XML editors on the internet, for example XML Marker or Notepad++).

You will need to:

1. Determine which menus a particular group of users needs, and which menus they don’t need.
2. Download an existing menu to use as a starting point.
3. In an XML editor, remove the menu items that are not necessary for those users. This allows you to simplify the WebCenter interface, especially for occasional users. For example, you can create a menu that only contains My Work for approval users.

4. Add custom menu items, to link to the WebCenter pages of your choice.
This allows you to make the pages most used by your users more accessible. For example, you can add menu items linking to saved searches, or to project creation from a specific project template.

5. When you are done editing the menu file, save it under a meaningful name.

6. Upload your custom menu:
   a) click the Menu Upload button on the Menus page
   b) on the Menu Upload page, browse to your XML file,
   c) give your custom menu a Name and a Description,
   d) click Upload.

7. Assign the menu to the users (or company) you created it for.

8. Log out and log in to WebCenter as one of the users to which you gave the custom menu, and check whether the menu works.

6.1.1 Download an Existing Menu

You will use the default menu as a basis for creating custom menus.

1. Download the default menu:
   a) On the Menus page, click the Download Menu link corresponding to the Default menu.
   b) The (XML) menu file will typically open in your browser. You need to save it to a file. Use for example:
      • File > Save As in Internet Explorer,
      • File > Save Page As or Ctrl+S / Command+S in Firefox,
      • File > Save As or Command+S in Safari (make sure you select Page Source as Format),
      • Ctrl+S / Command+S in Google Chrome.

**Note:** Do not copy and paste the XML from your browser as this can give corrupted XML.

2. Open the menu file you downloaded in your XML editor.
   See Structure of the XML Menu File on page 24 to familiarize yourself with the file.

6.1.2 Structure of the XML Menu File

Each menu and menu entry is enclosed in a menuitem tag.
You can expand and collapse menus and menu entries (use + to expand and - to collapse).
Main Row Menus

The first `<menuitem>`, "NON_ADMIN", contains the menu items corresponding to the top menu bar (when not in Admin mode).

Task Type Details

The menu items inside the "TASKTYPEDETAILS" `<menuitem>` (under <!-- Task type details menu -->) correspond to the tabbed menu you get when editing a task type (only for Admin users).
You will probably not need to change this.

**Project Details**

The menu items inside the "PROJDETAILS" <menuitem> correspond to the tabbed menu you get on the Project Details page.

You will probably not need to change this.

**Template Details**

The menu items inside the "TEMPLATEDETAILS" <menuitem> correspond to the tabbed menu you get on the Template Details page (only for Admin users).

You will probably not need to change this.

**Document Details**

The menu items inside the "DOCDETAILS" <menuitem> correspond to the tabbed menu you get on the Document Details page.

You will probably not need to change this.

**Task Type Details (Read Only Mode)**

The menu items inside the "TASKTYPEDETAILSRO" <menuitem> correspond to the tabbed menu you get when looking at task types in read only mode.

You will probably not need to change this.

**My Users**

The menu items inside the "MYUSERS" <menuitem> correspond to the tabbed menu you get when going to My WebCenter > My Users and editing a user (for Project Managers with the User Can Create User right).

6.1.3 Remove Unused Menu Items

Remove the menus and / or menu entries that are not necessary for your target users.
For each item to remove, delete everything between `<menuitem>` and `</menuitem>`, including the `<menuitem>` and `</menuitem>` tags.

```
- <menuitem>
  <name>PROJECTS</name>
  <labeltag>PROJECTS_TAG</labeltag>
  <url>/myfavoriteproj.jsp</url>
- <menuitem>
  <name>MY_FAVORITE_PROJS</name>
  <labeltag>MY_FAVORITE_PROJS_TAG</labeltag>
  <url>/myfavoriteproj.jsp</url>
- <menuitem>
  <name>MY_ALL_PROJECTS</name>
  <labeltag>MY_ALL_PROJECTS_TAG</labeltag>
  <url>/myallprojects.jsp</url>
```

Attention: Make sure you are very precise and delete only corresponding `<menuitem>` and `</menuitem>` tags, so as not to break the XML structure!

### 6.1.4 Add Custom Menu Items

To add a top level menu, add a `<menuitem>` element, with the following elements in it:

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;name&gt;</code></td>
<td>Choose a unique name for your menu. This name is only used to identify the menu in the XML file.</td>
</tr>
</tbody>
</table>
| `<labeltag>` | Enter the language file tag corresponding to your new menu. You can use a tag already existing in the language files, or a new tag. This makes displaying your new menu in multiple languages possible: you enter the tag here, and in each language file you associate that tag with the text string that will be shown for that menu in that language. If for example you are adding a menu linking to a(n existing) saved search showing rejected documents:

1. Enter "REJECTED_DOCUMENTS_TAG" as `<labeltag>`,
2. Associate that tag with the "Rejected Documents" string in your English language file, with "Documents Rejetés" in your French language file, etc.

Note: If you are using a new tag, you need to add it to the custom language files, otherwise you will lose it with every upgrade. |
| `<url>` | Enter the page which will be shown when clicking the new menu. |

See Customize the Language Files on page 20 for more information about customizing the language files.
Typically this will be a standard WebCenter page which you otherwise would get by more complex navigation.

```xml
<scope>
    <admin_users/>
    <project_managers/>
</scope>
```

If you want all user types to see it, don’t enter a scope at all.

**Note:** Do not use spaces in the XML values.

For example, to create a menu showing the rejected documents saved search to all users, use this:

```xml
<menuitem>
    <name>REJECTED_DOCUMENTS</name>
    <labeltag>REJECTED_DOCUMENTS_TAG</labeltag>
    <url>dosavedsearch.jsp?searchName=RejectedDocuments</url>
</menuitem>
```

• To add a menu entry under a top level menu, place that bit of XML between the `<menuitem>` tags of the top level menu, as in the example below.

```xml
<menuitem>
    <name>SEARCH</name>
    <labeltag>SEARCH_TAG</labeltag>
    <url>projsearch.jsp</url>
    <menuitem>
        <name>REJECTED_DOCUMENTS</name>
        <labeltag>REJECTED_DOCUMENTS_TAG</labeltag>
        <url>dosavedsearch.jsp?searchName=RejectedDocuments</url>
    </menuitem>
</menuitem>
```

### 6.2 Examples

#### 6.2.1 Going to Projects I Am invited to in One Click

By default, the Projects menu of a normal (non Project Manager) user contains Favorite Projects and Projects I am invited to. However, many users don’t use favorites.
To make it quicker for them, you can remove the Favorite Projects and Projects I am invited to entries, and make clicking on the Projects menu go directly to Projects I am invited to (instead of going to Favorite Projects).

This makes going to Projects I am invited to a one-click operation.

1. Add a Projects menu (with no entries) going directly to Projects I am invited to for normal users:

   Just before...

   ```xml
   <menuitem>
       <name>PROJMGMNT</name>
   ...
   ```

   add:

   ```xml
   <menuitem>
       <name>PROJECTS_NONPROJMGMNT</name>
       <labeltag>PROJMGMNTPROJS_TAG</labeltag>
       <url>iamprojectmember.jsp</url>
       <scope>
         <normal_users/>
       </scope>
   </menuitem>
   ```

   **Note:** The "PROJMGMNTPROJS_TAG" `<labeltag>` already exists in the language files with the value "Projects" (in the English file), so you don't need to add it.

2. Change the scope of the old Projects menu (so normal users can't see it) by adding a `<scope>` element:

   ```xml
   <menuitem>
       <name>PROJMGMNT</name>
       <labeltag>PROJMGMNTPROJS_TAG</labeltag>
       <url>myfavoriteproj.jsp</url>
       <scope>
         <admin_users/>
         <project_managers/>
       </scope>
   ...
   ```

3. Save and upload your custom XML file.

### 6.2.2 Customizing the Interface for Project Requests

Often you might want to allocate a separate menu item for creating a project (or requesting a project) based on a fixed template.

You don’t need to call this button "Create Project"... Depending on your business process, you can use for example:

- New Sales Request
- Order Samples
- Create New Book
- Add Product
- New Campaign
If you only ever have one type of project request, you can Create a Project from a Fixed Template on page 31. If you have different project request types, Create a Project from Different Templates on page 31.

You can also Show Projects Created from the Request Template(s) on page 35 and Add Your Requests to Your "My Work" Page on page 35.

Create a Project from a Fixed Template

If your users only need to create one type of project (request), you can add a button that will bring them directly to the project creation page, with the template they need already selected.

1. In your XML menu file, remove all menu items the requesters won’t need.
2. Add a menu item (called for example “New Request”), that links to creating a project based on an existing template (called for example “New Box”).

```xml
<menuitem>
  <name>NEW_REQUEST</name>
  <labeltag>NEW_REQUEST_TAG</labeltag>
  <url>projmgmntnewproject.jsp?template=New%20Box</url>
  <scope>
    <project_managers/>
  </scope>
</menuitem>
```

**Note:** Make sure you enter your template name precisely (the `<url>` is case sensitive and needs %20 to represent a space).

3. Save and upload your custom XML file.
4. Assign the menu to the users (or company) you created it for.
5. Add a language string to the language files (see Add Custom Menu Items on page 28 and Customize the Language Files on page 20).
6. Restart TomCat.
7. Log in to WebCenter as one of the users to which you gave the “New Request” menu, and check whether:
   - you see the "New Request" menu,
   - clicking the menu opens the project creation page, with the "New Box" template selected.

**Note:** If this does not work, check all previous steps for spelling mistakes.

- If you don’t get the new menu item, you have added the information incorrectly to the XML file.
- If you get an empty menu item (no text), your XML file is correct but your language file is not.
- If all is scrambled, you have broken the XML. Go back to a previous version.

Create a Project from Different Templates

If your users need to create several types of projects (requests), you can add a button that will bring them to the project creation page, with a selection of templates to choose from.

1. Mark the templates they will need to choose from using a characteristic:
   a) Log in to WebCenter as an Admin and go to Admin > Characteristics.
b) On the Characteristics page, make sure **ROOT** is selected and click **New Characteristic**. Create a characteristic called "Request".

c) Go to one of the template the requesters will use (for example "New Box") and add the "Request" characteristic to it.

d) Repeat for all templates the requesters will use.

2. In your XML menu file, remove all menu items the requesters won’t need.

3. Add a “New Request” menu item, that links to the project creation page, and calls the “Request” template characteristic.

```xml
<menuitem>
    <name>NEW_REQUEST</name>
    <labeltag>NEW_REQUEST_TAG</labeltag>
    <url>projmgmntnewproject.jsp?templatecharacteristicpath=\Request</url>
    <scope>
        <project_managers/>
    </scope>
</menuitem>
```

**Note:** Make sure you enter your template name precisely (the `<url>` is case sensitive).

4. Save and upload your custom XML file.

5. Assign the menu to the users (or company) you created it for.

6. Add a language string to the language files (see Add Custom Menu Items on page 28 and Customize the Language Files on page 20).

7. Restart TomCat.

8. Log in to WebCenter as one of the users to which you gave the “New Request” menu, and check whether:

   - you see the “New Request” menu,
   - clicking the menu opens the project creation page, with all the templates that have the “Request” characteristic available.

**Note:** If this does not work, check all previous steps for spelling mistakes.

   - If you don’t get the new menu item, you have added the information incorrectly to the XML file.
   - If you get an empty menu item (no text), your XML file is correct but your language file is not.
   - If all is scrambled, you have broken the XML. Go back to a previous version.

### Create a Project and Run a CAD Standard

If you are an ArtiosCAD user, you can run a CAD standard at project creation (see Run a CAD Standard from WebCenter in the Administration Guide).

You can create a custom menu item that allows users to create a project, based on a specific project template (or a selection of templates), using specific ARD document templates to run a CAD standard.

1. Mark the ARD document templates the users will need to run a CAD standard using a characteristic:

   a) Log in to WebCenter as an Admin and go to Admin > Characteristics.
b) On the Characteristics page, create the characteristic to assign to your ARD document templates, and sub-characteristics if desired.

For example, create "Folding Carton" as a main characteristic and types of boxes as sub-characteristics.

![Characteristics Tree](image)

**Note:** The main characteristic can be as deep as you want in the characteristics tree, but it cannot contain more than two levels of sub-characteristics.

See *Create a Characteristic* in the Administration Guide for details.

Note: For more details on characteristics and sub-characteristics, see Create a Characteristic in the Administration Guide.

c) Add the characteristic or one of the sub-characteristics you just created to all ARD document templates you want to make available to the users at project creation.

For example, you can have an ARD document template with the "Tall rectangular box" and "Cereal box" characteristics, another one with the "Square box" characteristic, etc.

2. In your XML menu file, remove all menu items the requesters won’t need.

3. Add a menu item (for example "New Request") that links to creating a project based on an existing template (for example "New Box"), using ARD document templates with your chosen main characteristic (for example "Folding Carton") for running a CAD standard.

```xml
<menuitem>
  <name>NEW_REQUEST</name>
  <labeltag>NEW_REQUEST_TAG</labeltag>
  <url>projmgmntnewproject.jsp?template=New%20Box<br>&amp;documentcharacteristicpath=\Product%20Type\Folding%20Carton</url>
  <scope>
    <project_managers/>
  </scope>
</menuitem>
```
Note:
Make sure you enter your template and characteristic names precisely:

- the `<url>` is case sensitive,
- use `%20` to represent a space,
- use `&` between the reference to the project template and the reference to the document characteristic,
- use `\` before each level of the characteristic tree under ROOT (so `ROOT > Product Type > Folding Carton` is `\Product%20Type\Folding%20Carton`).

Tip:
If you want the requesters to select from a list of relevant project templates instead of using a fixed one, mark these project templates with a characteristic as explained in Create a Project from Different Templates on page 31.

Then specify the project template and document template characteristics to use in the XML:

```xml
<menuitem>
  <name>NEW_REQUEST</name>
  <labeltag>NEW_REQUEST_TAG</labeltag>
  <url>projmgmntnewproject.jsp?templatecharacteristicpath=\Request&amp;documentcharacteristicpath=\Product%20Type\Folding%20Carton</url>
  <scope>
    <project_managers/>
  </scope>
</menuitem>
```

4. Save and upload your custom XML file.
5. Assign the menu to the users (or company) you created it for.
6. Add a language string to the language files (see Add Custom Menu Items on page 28 and Customize the Language Files on page 20).
7. Restart TomCat.
8. Log in to WebCenter as one of the users to which you gave the "New Request" menu, and check whether:

   - you see the "New Request" menu,
   - clicking the menu opens the project creation page, with the "New Box" template selected (or with a list of all templates that have the "Request" characteristic available, depending on what you specified in the XML).
   - when clicking New in the Documents section, you can see the ARD templates that have the "Folding Carton" characteristic or one of its sub-characteristics, and you can filter on sub-characteristics (select up to two levels of sub-characteristics in Show and click Go).
Note: If this does not work, check all previous steps for spelling mistakes.

- If you don’t get the new menu item, you have added the information incorrectly to the XML file.
- If you get an empty menu item (no text), your XML file is correct but your language file is not.
- If all is scrambled, you have broken the XML. Go back to a previous version.

Show Projects Created from the Request Template(s)
Once you allow people to create requests, you will probably also want a “Requests” menu item, which shows all projects created from templates with the “Request” characteristic.

1. Add a “Request” menu item (under “NON_ADMIN” if you want it to be a top level menu, or under a top level menu item if you want a menu entry):

   ```xml
   <menuitem>
     <name>REQUESTS</name>
     <labeltag>REQUESTS_TAG</labeltag>
     <url>myallprojects.jsp?characteristicpath=\Request</url>
   </menuitem>
   ```

2. Save and upload (update) your custom XML file.
3. Add a language string to the language files.
4. Restart TomCat.
5. Log in to WebCenter as one of the users who has this menu, and check whether the menu works.

Add Your Requests to Your “My Work” Page
To complete the user experience of requests, you can add a My Work section that displays the list of requests the user created him/herself on their My Work page.

1. Log in to WebCenter as an Admin and do a project search looking for the “Request” characteristic.
2. Save this search as a Global search with the name "Requests".
3. Create a My Work section displaying this search:
   a) Go to Admin > Preferences > My Work and click Create New My Work Section.
   b) Fill the fields on the Create New My Work Section page as shown below and click Create.

   ![Create New My Work Section](image)

   **Note:** You can choose another scope than All Users, for example only a group of requesters.

Now people in the scope can add the "My Requests" section to their My Work page (via My WebCenter > My Work setup).
6.2.3 Removing the Keyword Search

To remove the ability to search using a keyword for non-Admin users, add the following under the "NON_ADMIN" <menuitem>:

```xml
<keyword_search><enabled>false</enabled></keyword_search>
```

This will remove the search box at the top right corner of the WebCenter window, and the Keyword Search option from the Search For Projects, Search For Documents and Search For Tasks pages.

6.3 Will My Customizations Survive New Updates?

The XML files you create are uploaded to the database and as such, they will survive WebCenter updates/upgrades. This means that the menus will remain as you have defined them.

However, new features added by a new version may mean extra menu items added in the default menu.

If you created your custom menu(s) from a full copy of the default menu, users having those menus will not see the menus added by the new version.

To prevent this, we recommend you work with an XML file containing only the “differences” between the default menu and your custom menu.

This means that you only put the part which you actually changed in the new XML file.

Examples

- If you only changed the top menu, your XML should only contain the "NON_ADMIN" menu item. This way, the other menus will stay the same.
- If you only want to remove a part of the project details tabbed menu, create an XML file with only the "PROJDETAILS" section (and change it to your needs).
- If you only want to add an item to the existing top level menu, just include that menu item in the XML file (nothing else). It will be added to the right of the top level menu.
7. Customizing the Attributes

From WebCenter 12, it is possible to customize the attributes. You can now change the attribute's label to an image, add an HTML help page, allow multiple values and more.

For more information about the Graphic Asset Browser and Cascading attributes, please consult the WebCenter Admin Guide.

7.1 Create Image Label for an Attribute

You can create Project request forms or Order forms with graphical labels by setting image labels for the specific Attributes in an Attribute Category. You can use this feature to make the form visually attractive and user friendly. In addition, you can add graphical (packaging) symbols and warnings to aid the user while filling in the forms.

1. Add the label images to the following folder on the WebServer: `WebCenter\WebServer\tomcat\webapps\WebCenter_Inst\custom\AttributeLabelImages`.
   Create a new folder `custom` if it does not exist yet and create a folder named `AttributeLabelImages` under this folder.

   **Note:**
   - For the best user experience, use .jpg images. It does recognize .gif,.jpeg and .png.

2. Open your Attribute Category and select the Attribute for which you want to give an image label.
3. In the cell details, enter the correct image name including the extension in the Label Image field and save your attribute category.

You can see image labels appearing instead of attribute names in the forms.
Tip: In case you wish to display a checkbox for the attribute, make sure you use an attribute with a restricted set containing only the values Yes and No.

7.2 Create HTML Info for an Attribute

When you design Project request forms or new Order forms, you might want to provide additional information to the user who is filling in the information. From WebCenter 12 onwards you can provide additional information as HTML help or info. This is especially useful when technical fields need to be supported with extended HTML help. There will be an info icon that is added next to the attribute name or attribute image label. When the user clicks on this icon, a popup HTML page with the information will appear.

1. Create an HTML file with the additional information you want to provide along with the attribute. You can create textual information as well as images to explain the attribute.
2. Place this HTML file in the AttributeHelp folder (WebCenter\WebServer\tomcat\webapps\WebCenter_Inst\custom\AttributeHelp).
3. Open the Attribute category and select the attribute for which you want to provide HTML help or info.
4. In the cell details, enter the correct file name (of the HTML file) in the Info Page field and save your Attribute Category.
If you have an HTML info page linked to an attribute, you can see an info icon next to the attribute in the form that you created. When you click on the info icon, you can see a pop-up window with the info to help the User.
8. Customizing the E-Mail Notifications

WebCenter users can determine the look and content of the e-mail notification messages themselves, thus allowing for customizable e-mail notification messages.

8.1 Technology Used: XSLT

Staying true to the technology already used and the web anchored environment of WebCenter the framework of e-mail notification templates is based on a XML/XSLT solution.

XSLT is an industry standard for transforming an XML source to any preferred output. It uses an XML-based structure to process a passed source XML document in order to transform it into a desired output (plain text format for e-mail notification messages’ purposes). Although working with XSLT requires some technical skills, it is a powerful and reliable tool.

In order to work with the XSLT template framework a user has to know:

- Structure of the XML document that is being processed by the XSLT template.
- Basic syntax and terminology of the XSLT technology.

Using XSLT also means that the templates are actually present in the form of physical files on the Application Server. They are modified on the file system directly using any text editor. There is currently no support for modifying the templates by using any part of the WebCenter user interface.

Note: You can find various freeware XSLT editors on the internet.

8.2 Migration of customized email from WebCenter 7.X to WebCenter 10+

Since WebCenter 10, the structure of the XML document containing the data for generating the email message, has been extended in order to support bundled email messages. This means that, if an approval action is done on a list of documents, there will only be one XML file being generated that contains the information about all the documents on which the action was done. See added example of such an XML file containing multiple documents. For customers who have customized their approval emails, the approval xsl transformations need to be updated so that the transformations will work with WebCenter 12 generated XML files. This update can be done manually, but we strongly advise to let the software take care of the update. The following section describes how the software handles the WebCenter 7.X customized emails.

Migration of the approval XSL transformation

When a notification of a particular type needs to be sent, the software will check whether there exists a _list.xsl file in the custom directory. If so, then this file is used for doing the transformation. If that file cannot be found, but the 7.X version of this file exists in the directory, it means that the customer upgraded from 7.X to Webcenter 10 or 12 and the xsl will not be compatible with the new code. This file will be updated by WebCenter and the file will be renamed to have the _list.
extension. The old file in the directory will be renamed to _unused. The _unused version of the file will not be used further on by WebCenter. If the 7.X version of the file can also not be found in the custom directory, then the default version of the file will be used, which sits in the default directory. For example: suppose that the approval cycle starts on a document and email notifications are sent out. The software looks in the custom directory whether it finds a cycle_started_body_list.xsl file. If so, this file is used to generate the email body.

Example: WebCenter 12 xml structure for the approval email

```xml
<email_data>
  <!-- Localized labels set-->
  <languages>
    <lang_string key="LABEL_EXAMPLE_TAG" value="Example Label" />
    <lang_string key="DOC_NAME" value="Document Name" />
    <lang_string key="REQUESTED" value="Requested by" />
    <lang_string key="TODO_LINK" value="Click here to go to Your ToDo list"/>
  </languages>
  <!-- Email body strings-->
  <lang_string key="APPROVAL_STARTED_TITLE" value="Approval Cycle Started on Documents" />
  <lang_string key="APPROVAL_FINISHED_TITLE" value="Approval Cycle Finished on Documents" />
  <lang_string key="USER_SUBMITTED_STATUS_TITLE" value="User Submitted Approval Status on Documents" />
  <lang_string key="ALL_USERS_APPROVED_TITLE" value="All Approval Users Approved Documents" />
  <lang_string key="APPROVAL_REJECTED" value="Rejected" />
  <lang_string key="APPROVAL_PENDING" value="Pending" />
  <lang_string key="APPROVAL_APPROVED" value="Approved" />
  <lang_string key="APPROVAL_APPROVAL_PENDING" value="Approval Pending" />
  <lang_string key="APPROVAL_FORCED_REJ" value="Forced Rejected" />
  <lang_string key="APPROVAL_FORCED_APP" value="Forced Approved" />
  <lang_string key="APPROVED_COMMENTS" value="Approved with Condition" />
  <!-- ... this section contains the content of the language file ...
  -->
</languages>
<host_url>http://webcenterserver/WebCenter_Inst</host_url>
  <!-- Basic Data - for each document where the particular approval action was applied, we have a approval_message node in the XML -->
  <!-- NOTE: OPTIONAL XML elements are not always available, thus it cannot be relied on their presence -->
<approval_message>
  <project>
    <project_name>Example Project Name</project_name>
    <project_description>an example of a Project Description...could be quite long - OPTIONAL</project_description>
    <project_manager>Barret, Syd - Example Project Manager</project_manager>
    <project_manager_email>project_manager@domain.com</project_manager_email>
    <project_customer>Pink Floyd Recordings Limited - Example Project Customer - OPTIONAL</project_customer>
    <project_status>Active - Project Status example</project_status>
    <project_due_date>2007-08-30 10:30:00.0 - example due date of the Project - OPTIONAL</project_due_date>
```
<approved_conditions_folio />
<pending>1</pending>
<pending_folio:test1</pending_folio>
<annotations />
<annotations_folio />
<blank />
<empty_folio />
<empty_folio />
<nr_pages>4</nr_pages>
<nr_rejected>1</nr_rejected>
<nr_approved>2</nr_approved>
<nr_approved_conditions>0</nr_approved_conditions>
<nr_pending>1</nr_pending>
<nr_annotations>0</nr_annotations>
<nr_blanks>0</nr_blanks>
<nr_empty>0</nr_empty>
</pagelist_approval_overview>
</approval_message>
<!-- details of the second document -->
<approval_message>
<project>
  <project_name>Example Project Name 2</project_name>
  <project_description>another example of a Project Description...could be quite long - OPTIONAL</project_description>
  <project_manager>Cobain, Kurt - Example Project Manager</project_manager>
  <project_manager_email>Kurt.cobain@esko.com</project_manager_email>
  <project_customer>Island - Example Project Customer - OPTIONAL</project_customer>
  <project_status>Non Active - Project Status example</project_status>
  <project_due_date>2011-08-30 10:30:00.0 - example due date of the Project - OPTIONAL</project_due_date>
  <project_link>http://myserver.com/webcenter/projdetails.jsp?menu_file=projsearchresults&amp;projectID=00002_0000013425 - link to the project defaults page</project_link>
  <project_attributes_link>http://myserver.com/webcenter/projdetailsattrs.jsp?menu_file=projsearchresults&amp;projectID=00002_0000013425 - link to the project attributes page</project_attributes_link>
  <project_tasks_link>http://myserver.com/webcenter/projdetailswctasks.jsp?menu_file=projsearchresults&amp;projectID=00002_0000013425 - link to the project task page</project_tasks_link>
</project>
<document>
  <doc_name>example2.pdf - example Document Name</doc_name>
  <doc_description>Example Document Description</doc_description>
  <doc_version>2 - example version number</doc_version>
  <doc_due_date>2011-08-30 10:30:00.0 - example approval due date of this document - OPTIONAL</doc_due_date>
  <doc_link>http://myserver.com/webcenter/docdetails.jsp?docVerID=00002_000000306 - example link to a document</doc_link>
  <doc_approval_link>http://myserver.com/webcenter/docdetailsapprovers.jsp?docVerID=00002_0000109718 - example link to a document's approval page</doc_approval_link>
  <viewer_link>http://myserver.com/webcenter/viewdocument.jsp?docVerID=00002_0000109718 - example link to the viewer</viewer_link>
8.3 The E-Mail Template File Structure

All the e-mail notifications’ related files are stored on the Application Server in the following path: C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications.

This is a root directory of the repository. All the subdirectories contain the required templates and language files. The structure of the repository is following:

Inside the root directory you can find three subdirectories - Default, Custom and Example. In the Example folder, you can find a customization example. Copy the contents of the Example folder into the Custom folder to try out.

Default directory is a container for all the default files related to e-mail notifications. Contents of this directory come with the WebCenter installation and are overwritten whenever any of these files/directories are updated (or new files/directories are added).

Note:
It is forbidden to modify the contents of Default directory, since doing so may lead to incorrect behavior in the process of creating the e-mail notification messages!

Custom directory is intended as a repository of the customized elements of the e-mail notification framework. In a clean WebCenter installation it comes empty and is populated only when a user decides to customize any part of the framework. In order to have the e-mail notifications customization framework to acknowledge and use the customized XSLT templates and/or language files it is necessary to mimic the file system structure of the default directory.

For example: a default XSLT template located on the following path
This default template will only be overridden by a customized template on a following path:

C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Custom/Approval/cycle_finished_body.xsl

It is not necessary to mimic the entire directory/file structure of the Default directory inside the Custom directory for the customized elements to be used by the system, but the user rather needs to supply the file system structure ONLY for the individual customized elements (this means, that if the user wants to customize just Approval Started notification message, he has to create an appropriate file system structure for this notification type only).

### 8.4 Customizing E-Mail Language Files

WebCenter in general uses XML based file as a source for all labels in the system with each of the labels set stored in a single file for each of the supported languages.

The same applies to the e-mail notifications framework. This allows for supporting localized labels inside the e-mail notification messages, based on the user profile preference of the notification message’s recipient.

In order to be independent on the rest of the application, the e-mail service utilizes its own E-Mail Language Manager and a separate set of XML language files.

These language files are stored on the following path: C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Default/Languages.

If you wish to customize the labels, you need to copy an appropriate XML language file (based on the language for which you want to customize the label for) to a corresponding path in the Custom directory. After doing so, you need to restart the JBoss service.

### 8.5 E-Mail Notification Templates In Detail

Each of the notification message types consists of two XSLT template files - one for the message’s body and one for the message’s subject.

For example the default Approval Cycle Finished notification message templates are following:

C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Default/Approval/cycle_finished_body.xsl

and

C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Default/Approval/cycle_finished_subject.xsl

All XSLT templates used for creating e-mail notification messages consume a single XML document as a source for the actual transformation that leads to a plain-text-formatted output for message body and subject. This XML source is supplied by the system when an appropriate action is triggered (i.e. a project manager has changed a project status; an approval cycle has been started on a document; a task was reassigned to a different user; etc.).

There are two basic parts of this XML source document:
• All supported String labels in XML format. These labels are inserted into the XML source document by E-mail Language Manager. They are localized based on the language preference of the message's recipient passed to the E-mail Language Manager.

• The actual data that makes up the "information part" of the message (i.e. project name/due date/project manager name etc.)

These two parts of a XML source are enough for the XSLT template to create an appropriate output.

8.6 Customizing the Notification Templates Step by Step

The most important aspect of smooth and successful e-mail notification customization process lies in basic knowledge of XSLT technology. The templates are rather simple and the XSLT constructs utilized are few. This makes any modification of the templates confusing at first sight, but is understood soon enough.

8.6.1 What You Need to Know Before You Begin

Structure of the Source XML

Even more important than knowledge of XSLT is knowledge of the source XML, since that is the cornerstone of X-Path usage.

To be able to work with the XSLT templates, the user has to know the structure of the XML document that is consumed by the appropriate XSLT file. Since the XML source that is used in e-mail notification message creation process is never stored anywhere on the disk and is held only in the server's memory at runtime, there is no easy way to investigate its contents and their XML structure. Since each of the individual notification templates share their “parent notification type” XML structure, it is required to learn only one XML document structure to be able to work with a whole set of the parent’s “subtype” (by parent notification type one should think about Approval Notification Type, while its subtypes are Approval Cycle Started Notification, Approval Cycle Finished Notification and Approval Status Submitted Notification).

This is why each of the "parent notification type" directories contains an example XML file that represents a model XML source document that is expected by each of the "subtype" XSLT templates. The actual values of the elements in such an example XML document are of no importance. It’s their XML structure that matters for anyone who wishes to customize the templates.

XPath

XPath is something like XSLT’s own proxy SQL query framework. While SQL queries are used to stalk through a database, XPath is used to navigate through a XML document. The XML structure of XML source used by notification templates is really simple so the XPath queries are also simple. To tell the truth all the user doing the template modifications needs to know about XPath is parent-child relation between XML elements.

XPath basic syntax is actually done through namespace navigation. This means that it is the same as navigating through a Windows file system or internet URLs.

Here are some examples:
Windows file system - C:/Artios/WebCenter/.
Internet URL - http://myserver/WebCenter_Inst/login.jsp

In the first example:

- C: part stands for volume name (you can consider this to be the root definition of the namespace through which it will navigate).
- Artios part is an element under this root
- WebCenter is a subdirectory of Artios (thus it is a child element of Artios which in turn is its parent element)

A similar logic applies for the second example:

- http:// part is a definition of a protocol that will be used for processing any element selected by this "path"
- myserver part is a root namespace; in this case it is a server by the name myserver, that should contain the selected element at the end of this "path"
- WebCenter_Inst part is context name of the web application that contains the selected element; myserver part of the path is its parent element
- Login.jsp is the selected element of this path; it is a child element of WebCenter_Inst and myserver elements

Now if you take a trivial XML structure like:

```xml
<car>
  <engine HP="500">
    <parts>Some parts</parts>
  </engine>
  <wheels>
    <tires>Bridgestone</tires>
  </wheels>
</car>
```

You can use a similar path (this time it’s XPath) structure to select (for example) the contents of tires element this way:

```
select = "//car/wheels/tires"
```

This (hypothetical) XPath expression selects the tires element, which returns its contents giving the result value Bridgestone.

If we analyze the path itself, it is very simple > car is the root element, wheels is its child element and tires is in turn child element of wheels element.

8.7 Start from the example

This section describes how to create customized e-mail notifications starting from the example that is delivered with WebCenter 12.
The easiest way to customize the WebCenter e-mail notifications, is to start from the example files. In the EmailNotifications folder, you can find an Example folder which contains an elaborate e-mail customization.

Note: This example has been tested for compatibility with the following e-mail clients: Outlook 2003, Outlook 2010, iOS Mail and GMail.

1. First of all, you may need to navigate to your ../EmailNotifications/Custom folder and create a back-up of your current customized e-mail notifications.
2. Place the contents of the ../EmailNotifications/Example folder inside the Custom folder.
3. Restart the WebCenter services in order to update all language labels.

8.7.1 Basic customization

At the root of the Example folder, you can find 3 important XSL templates for basic customization of the WebCenter e-mail notifications:

- header.xsl
- footer.xsl
- customize.xsl

These three template files contain the formatting of the header, the footer, style and content that is used in all the e-mail notifications sent by WebCenter. If you make changes to these files, the changes are reflected in all notification mails.

Another important file is: URL.xsl.

Header.xsl
The Header.xsl file contains the logo of your company. The same header is included at the top of every e-mail notification sent by WebCenter.

Logo image
In the example, the logo image is the same as the WebCenter AppLogo; the logo that is displayed in the top left part of the header. This image is located on the Web Server. You can replace the WebCenter logo by modifying the logoSource variable. You can use the host URL of WebCenter to build the URL, by first adding:

```xml
<xsl:value-of select="host_url" />
```

After that, you add the path to the image, for example: /myimages/logo.gif, like in the example. You could also use a web URL.

Note: You can only specify an image source a server location (for instance, a web server on the internet). A relative path to the e-mail customization folder will not work.

Logo link
By replacing the contents of the logoLink variable with the link to your company website, you change the link attached to the company logo. When a user clicks on the logo in the e-mail notification, they will be redirected to this specific URL.
Example:
Replace the existing variables:

```xml
<xsl:variable name="logoLink">
  <xsl:value-of select="host_url"/>
</xsl:variable>
<xsl:variable name="logoSource">
  <xsl:value-of select="host_url"/>
  /images/WebCenter-banner.png
</xsl:variable>
By:

```xml
<xsl:variable name="logoLink">
  http://www.mycompany.com
</xsl:variable>
<xsl:variable name="logoSource">
  http://www.mycompany.com/image.jpg
</xsl:variable>

Footer.xsl
The Footer.xsl file is similar to the header file. It is included at the bottom of every e-mail notification sent from WebCenter. You can modify the company name and address by changing the company and address variables at the beginning of the file.

Example:
Replace the existing variables:

```xml
<xsl:variable name="company" select="'WebCenter © Esko'" />
<xsl:variable name="address" select="'Kortrijksesteenweg 1095, B-9051 Gent, België'" />
By:

```xml
<xsl:variable name="company" select="'My company name'" />
<xsl:variable name="address" select="'My address'" />

Note: Pay special attention to the single quotes that are used in addition to the double quotes.

Customize.xsl
The Customize.xsl file contains two parts: styling and content.
Styling

The styling section contains all the elements that make up the layout of the e-mail notifications. You can modify the CSS of a specific part in order to change the look and feel of the notifications. For example, you can set a background color, text color, text alignment, font, font-size, etc..

Layout

Every e-mail notification follows the same layout pattern. The body of the e-mail notification contains the main table that consists of three parts: header, information table and footer.

1. main
2. header
3. information
4. message
5. section header
6. label
7. value
8. section header
9. footer

Header and footer

The header and footer information is imported from the Header.xsl and Footer.xsl files. The styling of these sections can be modified in Customize.xsl.

Information table

The information table consists of multiple parts. First of all, there is a message that describes the action. Next you find multiple sections describing the project, document and task changed information. Every section has a sectionHeader.

A section typically contains two columns: label and value.
Note: Not all sections are relevant for, and therefore present in, every notification, it depends on the action what is displayed. The same goes for label-value pairs, they are displayed depending on the action and information in WebCenter.

Example:
If you want to change the background color of the sectionHeader in all e-mail notifications to green, you only need to change the following value in the specific template:

'background-color: #EEEEEE;'  

The sectionHeader template in the Style.xsl file will now look something like this:

<xsl:template name="sectionHeader">
  color: #334444;
  background-color: #C9E57E;
  font: bold italic 14px Verdana,Arial,Helvetica,sans-serif;
</xsl:template>

Content
The second part, content, allows you to control the presence of certain sections in the notifications. You can display or hide the document, task and project sections. The section will be displayed when you set it to true and hidden if you set it to false.

Example:
If you want to show the document and task sections, but hide the project section, your variables will look like this:

<xsl:variable name="document" select="true()"></xsl:variable>
<xsl:variable name="project" select="false()"></xsl:variable>
<xsl:variable name="task" select="true()"></xsl:variable>

Note: Pay attention to the brackets used behind the true and false values, these are necessary.

URL.xsl
The URL.xsl file contains a template that is called for every URL in the e-mail notifications. This way, you can map the internal WebCenter URL to an URL that is accessible from the outside. This is sometimes needed because you want to access the links to tasks, documents and projects in the notification e-mails from the outside. The internal URL is specified in the internalURL variable and the corrected accessible URL is specified in the correctURL variable. Change both variables to your URLs to activate this feature for your WebCenter e-mail notifications

Example:
Replace the existing variables:

<xsl:variable name="internalURL" select="'http://intranet.esko.com'"/>
<xsl:variable name="correctURL" select="'http://accessible.esko.com'"/>
8.7.2 Advanced customization

For a basic customization, you can change the layout of the e-mail notifications by modifying the header, footer and/or style files. For a more advanced customization, you could alter every XSL of an e-mail individually.

For example, you could:

- Include or exclude certain label-value pairs in sections.
- Change or override the layout of an e-mail individually from the style file.
- Change the links attached to certain fields (for instance, the document name).

In order to execute such a more advanced customization, you should have knowledge of both XSL and HTML.