

WebCenter 22.07

Customization Guide

07 - 2022

Contents

- 1. Copyright Notice..... 4**
- 2. Why Customize WebCenter?..... 6**
- 3. Where to Start the Customization?..... 7**
- 4. Applying your WebCenter 12.X or 14.X customization to WebCenter 16.X or later..... 8**
- 5. Customizing the Look and Feel of WebCenter Using the Customizations Page..... 11**
 - 5.1. Viewing and Editing a Customization..... 11
 - 5.1.1. Load a Theme.....12
 - 5.1.2. Reset to Default Look and Feel.....12
 - 5.1.3. Customize Logos and Hyperlinks..... 12
 - 5.1.4. Customize Colors..... 13
 - 5.1.5. Customize Fonts..... 13
 - 5.1.6. Customize Static Text..... 14
 - 5.1.7. Forgot Password..... 14
 - 5.1.8. Fixed Navigation Bar..... 14
- 6. Customizing the Look and Feel of WebCenter Using the customizationConfig.xml file..... 15**
 - 6.1. Customizing Logos and their Hyperlinks..... 15
 - 6.1.1. Customize Logos and Hyperlinks.....15
 - 6.1.2. Customize Logo for a User or Company.....18
 - 6.2. Custom Language Flags for Text Content..... 19
 - 6.3. Change the Color Scheme..... 19
 - 6.3.1. Color Scheme Reference..... 20
 - 6.4. Change the Fonts..... 20
 - 6.5. Customize Static Text.....20
 - 6.5.1. What Static Text Items Can You Customize?.....20
 - 6.5.2. Customize a Static Text Item..... 23
 - 6.5.3. Example: Customizing Static Text.....24
 - 6.6. Customize the Language Files.....25
 - 6.7. Forgot Password..... 26
 - 6.8. Fixed Navigation Bar.....27
- 7. Customizing the WebCenter Menus.....28**
 - 7.1. Create a Custom Menu..... 28
 - 7.1.1. Download the Default Menu..... 29
 - 7.1.2. Structure of the XML Menu File.....29
 - 7.1.3. Remove Unused Menu Items..... 33
 - 7.1.4. Add Custom Menu Items.....33
 - 7.2. Examples.....35

7.2.1. Going to Projects I Am invited to in One Click.....	35
7.2.2. Customizing the Interface for Project Requests.....	36
7.2.3. Removing the Keyword Search.....	42
7.3. Will My Customizations Survive New Updates?.....	43
8. Integrating an External Web Connection into the User Interface.....	44
8.1. Integrating an External Web Connection into a Top-Level Menu.....	44
8.2. Integrating an External Web Connection into a Project Menu.....	46
8.3. Integrating an External Web Connection into a My Work Section.....	48
9. Customizing the Attributes.....	50
9.1. Create Image Label for an Attribute.....	50
9.2. Create HTML Information for an Attribute.....	52
10. Customizing the E-Mail Notifications.....	54
10.1. E-Mail Notification Templates In Detail.....	54
10.2. Start from the example.....	54
10.2.1. Basic customization.....	55
10.2.2. Advanced customization.....	60
10.3. Technology Used: XSLT.....	60
10.4. The E-Mail Template File Structure.....	60
10.5. Customizing E-Mail Language Files.....	61
10.6. Customizing the Notification Templates Step by Step.....	62
10.6.1. What You Need to Know Before You Begin.....	62
11. Customizing the Translation Service.....	64
12. Customizing the Onboarding Hook.....	65

1. Copyright Notice

© Copyright 2022 Esko Software BV, Gent, Belgium

All rights reserved. This material, information and instructions for use contained herein are the property of Esko Software BV. 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 BV 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 BV 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 BV.

This document supersedes all previous dated versions.

PANTONE® Colors displayed in the software application or in the user documentation may not match PANTONE-identified standards. Consult current PANTONE Color Publications for accurate color. PANTONE® and other Pantone trademarks are the property of Pantone LLC. © Pantone LLC, 2015

Pantone is the copyright owner of color data and/or software which are licensed to Esko to distribute for use only in combination with Studio Visualizer. PANTONE Color Data and/or Software shall not be copied onto another disk or into memory unless as part of the execution of Studio Visualizer.

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 Best Color Technology (EFI). EFI and Bestcolor are registered trademarks of Electronics For Imaging GmbH in the U.S. Patent and Trademark Office.

Contains PowerNest library Copyrighted and Licensed by Alma, 2005 - 2007.

Part of this software uses technology by GlobalVision. ArtProof and ScanProof are registered trademarks of GlobalVision Inc.

Part of this software uses technology by Qoppa Software. Copyright © 2002-Present. All Rights Reserved.

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

Correspondence regarding this publication should be forwarded to:

Esko Software BV

Raymonde de Larochelaan 13

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.

Do provide your suggestion to 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. To apply your customizations, modify this file accordingly and save it. The changes are visible after a little while (this can take up to a few minutes).

The **customizationConfig.xml** file should be placed into the customization directory on your Webserver 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 Using the customizationConfig.xml file](#) on page 15 for details.

It is also possible to customize several of these elements from the Customization page (from WebCenter 12.1) in the Admin section without editing the customizationConfig.xml file manually anymore. See [Customizing the Look and Feel of WebCenter Using the Customizations Page](#) on page 11 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 28 for details.

Since WebCenter 12.1, it is also possible to customize menus using the Menu Builder in the WebCenter Admin section.

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 54 for details.

4. Applying your WebCenter 12.X or 14.X customization to WebCenter 16.X or later

It is possible to re-use your WebCenter 12.X or 14.X customization in your WebCenter 16.X or later installation. This section explains the changes that need to be made to make the customization work properly in WebCenter 16.X or later, along with new customization features that come in WebCenter 16.X, 18.X, 20, 21.X or 22.X.

Moving your customization from WebCenter during Upgrade (12.X or 14.X to 16.X or later)

Before upgrading your WebCenter installation, 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 have 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".

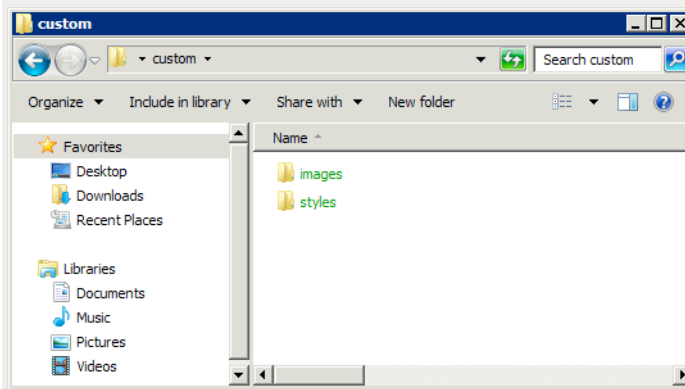


Attention:

When copying folders from a different operating system into your custom folder (especially from Mac OS to Windows), make sure the contents are not 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, as well as some features that have been removed. For example, the **MenuBackground** image is no longer used in 16.X. Instead, the Menu background and font colors are calculated automatically based on a single Menu Color, allowing for much easier customization.

A new feature in WebCenter 16.X is the "full screen Background Image" for the Login page. Also, it is now possible to change the default background colors for the buttons.

Changes required to make your customization work

The old customizations you have created will largely work in WebCenter 16.X. 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 colors may have been added or removed, the color settings may have to be adapted.



Attention: After applying your customization, you may need to restart the Tomcat service.

Moving your customization from WebCenter 18.X

1. Backup phase (during uninstallation)

When uninstalling WebCenter 18.0 or later, in the Web server the installer inspects the list of Tomcat web instances and records the names of all the custom WebCenter instances while backing up 'custom' folder including the default one - *WebCenter_Inst*.

Common App Data root folder is used to store the customization files backup on the target Web server host. The default path on most Windows server is

```
C:\ProgramData\Esko\WebCenter\WebInstanceCustomData
```

This folder is the root and each of the WebCenter web instance custom folder files are stored under a subfolder that takes the name of the appropriate instance.

The list of web instances that were found and later re-deployed is stored in the registry among the other WebCenter upgrade values.

2. Re-deploy phase (during upgrade installation)

When performing an upgrade installation of a WebCenter Web server on the same host the installer gives the User the following options:

Web Server System
Please enter the options for your WebCenter application server.

Please enter your options.

Application server system host:

IIS web site that the web server will be deployed in:

Redeploy web instances:

- **Redeploy web instances with customizations:** This is the default option. Re-deploys the custom instances marked during the previous uninstallation as well as the custom folders backup.
- **Redeploy web instances without customizations:** Re-deploys the custom instances marked during the previous uninstallation and does not retain the backed-up custom folders
- **Do not redeploy automatically:** Does not re-deploy any of the custom instances and their custom folder backups.

5. Customizing the Look and Feel of WebCenter Using the Customizations Page

In the **Admin > Preferences > Customizations** page, you can customize a number of WebCenter user interface elements visually.

The current customization for your instance is always visible when you first load the Customizations page. You can alter this customization, reset the customization to default (the standard WebCenter look) or load a theme, which gives you a starting point for your new customization.

You can customize images, colors, fonts and static text in this page.

You can also enable the **Forgot Password** option, that displays a **Forgot Password?** link on the WebCenter **Log In** page, and allows users to reset their password.

You can also customize WebCenter *using the [customizationConfig.xml](#) file*.

System History Logs

All changes/events in this page are recorded and stored in the **System History** page for Administrator users. Read this page on how to access the *records*.

5.1. Viewing and Editing a Customization

Admin users are able to view and edit an existing customization on the Customizations page.

On the Customizations page the following items can be customized:

- **Images:** Application logo, Company logo, Product Line logo, Login page Background image and full-screen Background image.
- **Colors:** Different colors can be set for various elements using a color picker.
- **Fonts:** The current font type can be changed and a base size for the font can be entered. Other sizes are calculated proportionally to allow for easier customization.
- **Static Text:** Customized messages on the login page, welcome page and contact page by uploading a valid XHTML file.

The following steps describe how to view and change the current WebCenter instance's customization:

1. Go to the WebCenter instance you would like to customize and open the Customization admin page.
2. In case any customization was done before, those files and values are displayed, if not, the page shows the standard WebCenter values.
3. Change images, colors, the font that is used and upload a static text file.
4. After making your changes save your customization by clicking on the **Save** button.



Note: After saving your customization, there is no need to restart the TomCat server, as the changes are applied upon saving or after loading a theme. Since WebCenter 16.X, the customizations are re-applied by the system and the effect is not shown immediately. Reloading the customization may take as long as 2 to 5 minutes.

5.1.1. Load a Theme

On the Customizations page, there is an option to load a theme. The first theme (Default) is the WebCenter default look.

1. Click **Load Theme** for a particular theme to load it.
2. After confirmation the theme is installed.



Note: Loading the theme, as applying a customization, may take up to a few minutes.

5.1.2. Reset to Default Look and Feel

From the Customization page, it is always possible to revert back to the standard WebCenter look and feel.

1. Either click the button **Reset to Default** at the bottom of the page, or pick **Default** from the **Themes** section.
2. After confirmation, the default WebCenter look and feel is applied and used again.



Note: You may need to do a deep refresh of your browser to see the changes.

The option to reset to default is also available for several individual items, like colors and images. In this case, only that particular item will revert to the default look. Sometimes the 'default look' is to not show the element, for example the Product Line image is not there by default.

5.1.3. Customize Logos and Hyperlinks

It is allowed to change several images on the Customizations page. An image can be changed by uploading a new file, or reset an image to the default WebCenter value. Any changes to the images are saved after clicking the page's main **Save** button.

There are four images (and their hyperlinks) on you can change:

- **Application Logo:** This is the top left logo and it appears on every page. It can be replaced by uploading a new image with approximately the same height (50 pixels). In case you use an AppLogo image higher than 50px, you will stretch the top header along with this image. The recommended file type for this image is: GIF or PNG. It is common practice to give the top logo a hyperlink to a more general web page (typically your company's home page). In WebCenter 16.X you can choose between showing the Application Logo in the Menu bar (default, more compact), or displaying it above the Menu bar for wider images, by applying the setting **Show the application logo above the menu bar**.
- **Company Logo:** To the bottom right of the login page is the Esko logo (and its hyperlink to www.esko.com). It can be replaced by uploading a new image with approximately the same height (28 pixels). We appreciate it when customers keep our logo visible.

- **Product Line Logo:** To the bottom left, there is a placeholder for another image (ProductLineLogo). The standard installation does not put anything there. It is possible to upload a new image with a height of approximately 28 pixels.
 - **Background Image:** The default background image can be replaced with any image or with no image at all. This watermark appears only on the login page. It has no hyperlink. The recommended file type for this image is JPG or PNG. By disabling the setting **Use full screen background image**, the image is displayed as a smaller rectangle on the page as in older versions of WebCenter.
1. To upload a new image click the **Upload** button and select an image from your system. Make sure you select an image with roughly the correct dimensions and resolution (72 ppi).
 2. To reset an image to the default value, click the **Reset** button. Note that sometimes there is no default value, like for the Product Line Logo.
 3. To add or modify an image link, enter a URL (starting with http://) in the **URL** field.

5.1.4. Customize Colors

It is possible to change various colors on the Customizations page. Any changes to the colors are saved after clicking the page's main **Save** button.

There are several colors listed in the Colors section. Each with a title reflecting its purpose. For example, you can change the footer color on the login page by changing the value for **Login Footer Color**.



Note: Menu colors are all determined by the value set as **Menu Color**. The system automatically calculates shades of your chosen color and determines the font color (black or white) based on the optimal contrast with the calculated colors, to allow for easy and fast customization of the Menu bar.

1. In the **Colors** section, you can see an overview of the currently selected colors.
2. To change a color, you can either paste a hexadecimal color code in the color field, or you can use the color picker by clicking the color patch.
3. After opening the color picker you can move the little circle selecting the color around to change the saturation and brightness within the same hue. It is also possible to edit the hexadecimal color code manually, or to enter a new value for the RGB and HSB values.
4. After selecting a color, you can exit the color picker by simply clicking outside the color picker pop up, or by clicking the little **color circle icon**.
5. By using the **Reset** button, you can revert a specific color to the original WebCenter value.

5.1.5. Customize Fonts

In the Fonts section, you can change the currently used font to a different font type and change the font base size.

Any changes to the fonts are saved after clicking the page's main **Save** button.

1. To change the font type, enter a new value in the **Font** field. Make sure you use a commonly used font, which is installed on your user's systems.
2. To change the font sizes, simply enter a new value in the **Base Font Size** field. All other font sizes (larger/smaller) are calculated based on this base font size, to allow for easier customization.

5.1.6. Customize Static Text

There are a number of places in WebCenter where static text such as a marketing message is displayed. You can replace these sections with your own valid HTML pages. Any changes to the static text section are saved after clicking the page's main **Save** button.

Currently it is possible to replace the following sections:

- Welcome message on the login page (band above the main login section)
- Marketing message on the login page
- Marketing message on the welcome page
- Contact page (linked on top of each page)

The Static Text section shows an overview of the static text customizations that are currently applied. The static text section can be filled in with HTML pages containing text, images and inline CSS. In case no customization has been done, you will see "Default" next to the static text name. In case there is a customization, **Preview** is visible and a **Reset** button is also shown.

1. To upload a new static text file, click the **Upload** button and select a valid HTML file from your system.



Note: Ensure that your HTML page does not contain !DOCTYPE else it could affect the functioning of WebCenter.

2. To preview a static text file, click the **Preview** button.
3. To reset a static text item and remove the customization, click the **Reset** button.

5.1.7. Forgot Password

To add a **Forgot Password?** link to the WebCenter **Log In** page, do the following:

1. Go to **Admin > Preferences > Customizations**.
2. At the bottom of the **Customization** page, select **Show Forgot Password link** in the **Other** section.
3. Click **Save**.

Alternatively, you can add this link when [customizing the user interface](#).

When you click the **Forgot Password?** link, you are asked to supply user name or email address. You are then sent an email with a temporary password and a link to the **Log In** page.

After logging in, you are prompted to change the password.

5.1.8. Fixed Navigation Bar

To fix the Navigation / Menu bar in the WebCenter application, do the following:

1. Go to **Admin > Preferences > Customizations**.
2. At the bottom of the Customization page, select **Use Fixed Navigation Bar** in the **Other** section.
3. **Save** the changes.

When a user scrolls down any page, the navigation bar is fixed on the top of the page.

6. Customizing the Look and Feel of WebCenter Using the customizationConfig.xml file

You can customize WebCenter's look and feel using 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.

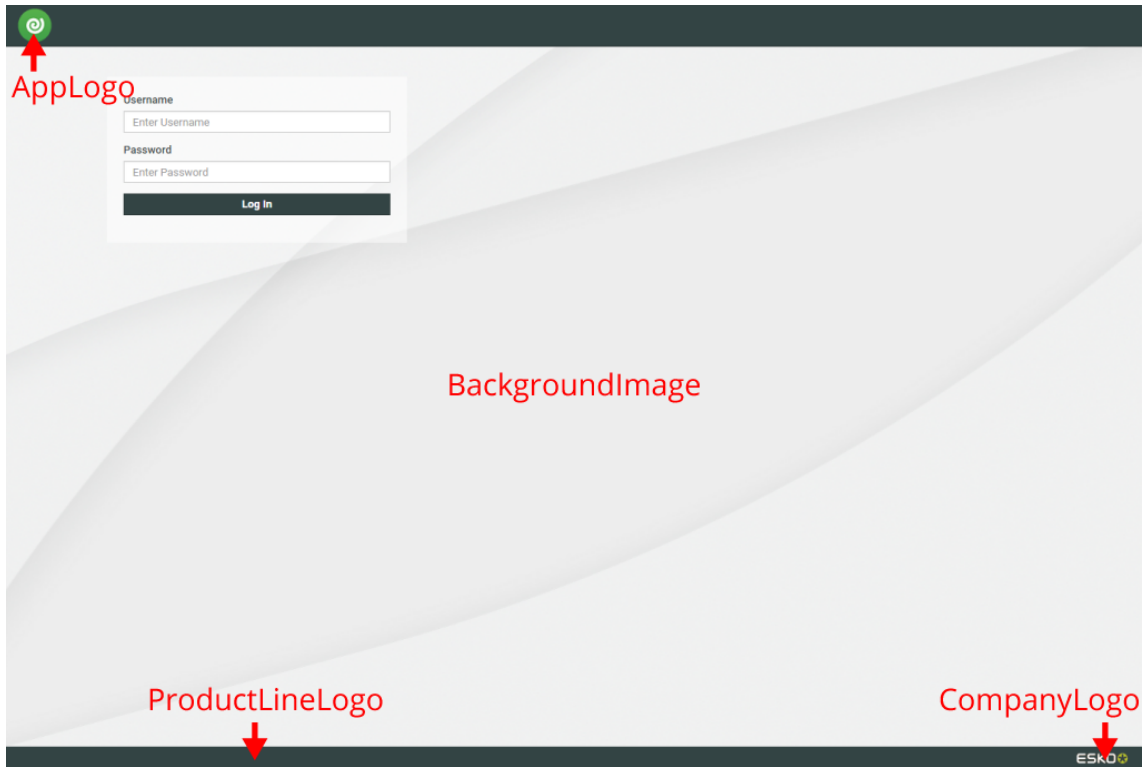
To use the example files, simply copy content from this directory into the `<WebCenter_Instance>\custom` directory.

6.1. 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.

6.1.1. 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 icon**. This logo appears on every page. It can be adapted by replacing it with any image of approximately the same height (50 pixels). In case you use an AppLogo image higher than 50px, 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 home page).
- The Background image (BackgroundImage) can be replaced with any image or with no image at all. This image **ONLY** appears on the login page. It has no hyperlink. Since WebCenter 16.X, it can be set as a full-screen image as well.
- To the bottom left, there is a placeholder for another image (ProductLineLogo). The standard installation does not put anything there.
- To the bottom right (CompanyLogo) 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:

```
<AppLogo
  RelPath="images/DEFAULT_WebCenter-banner.png"
  href="homepage.jsp" />
<CompanyLogo
  RelPath="images/DEFAULT_logo_esko.png"
  href="http://www.esko.com" />
<BackgroundImage
  RelPath="images/BackgroundImage.png" />
<FloatAppLogo
  value='false' />
```



```
<FullScreenBackground
  value='true' />
```

The settings **FloatAppLogo** and **FullScreenBackground** can be used to move the Application Logo above the Menu bar and set the Background Image to full-screen respectively.

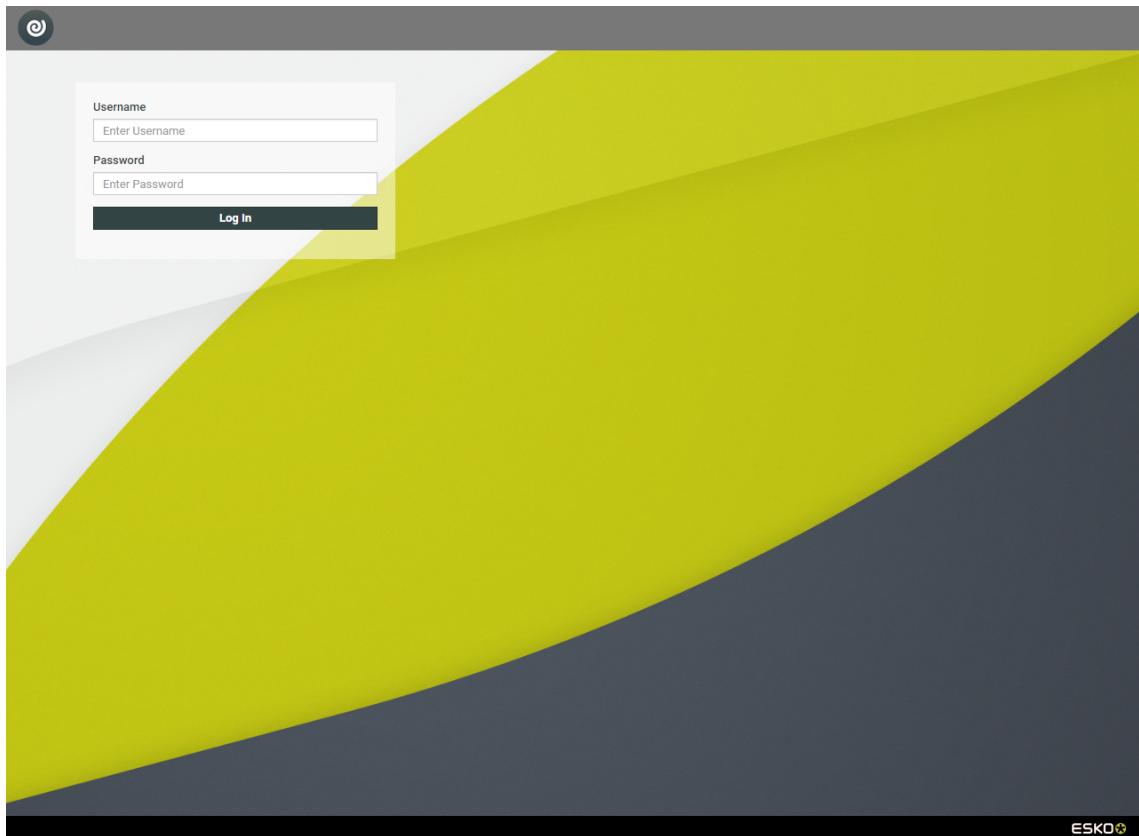
The following attributes can be used to provide the required image and optionally set the hyperlink:

- **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:

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

The following is an example of a Login page with customized Application Logo and Background Image. Note that in this case the header and footer colors have also been customized.



6.1.2. Customize Logo for a 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 user name,
- **Company_** to apply for a specific company,
- **Company_Location** to apply for a specific company and location.

For example, for a user called "John Doe" (with the user name "JOHNDOE") at the company "Esko", that has two locations: "Ghent" and "Singapore", you should name the folders:

- JOHNDOE (the user name) when applying the customization to that user specifically,
- Esko when applying the customization to all users at the "Esko" company,
- Esko_Ghent when applying the customization to users at the "Esko" company and "Ghent" location.



Note: Please make sure you use the exact user, company and location names, keeping the correct case and any spaces or special characters they may contain. For example, user names should be in capital letters.

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

```
<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 "JOHNDOE" folder has been created. This folder contains a `customizationConfig.xml` file and an `images` folder with a different logo image. When the user with the user name "JOHNDOE" 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.

6.2. Custom Language Flags for Text Content

You can now use customized flag images for the Master and Translation languages used in the Packaging Content Management (PCM). To use custom languages for the master or translation languages list, customized country flags can be stored in the custom folders in the Web Server. Since the images are available in the custom folder, these images are retained after every WebCenter upgrade.

To use custom images for flags for each of the languages,

1. Store the custom images in `custom/country_flags` and `custom/country_flags/big` folder in the WebServer.
2. Along with this language list is also required to be updated to use these flags.
3. Ensure full path for the flag is mentioned in language list excel sheet for the image as `custom/country_flags/file name`.

LANGUAGE_CODE	DESCRIPTOR	FLAG_FILENAME
fr-CA	French - Canada	custom/country_flags/CA.png
fr-FR	French - France	FR.png



Note: Flags which are available with WebCenter as default are mentioned with file name only.

6.3. Change the Color Scheme

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

It is possible to use your company's brand colors in WebCenter. 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:

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

```
</Colors>
```

6.3.1. Color Scheme Reference

ID	Default Value	Used in
8	fafafa	Project Selected Folder
13	334444	Main Header and Menu
15	fbfbfb	Footer Background Color
16	334444	Footer Login Page
18	334444	Primary Button
19	ffffff	Default Button
20	5f7e7e	Link Color
21	3e5353	Link Hover Color

6.4. Change the Fonts

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

The default font sizes were chosen to optimally fit into the various WebCenter pages. You only need to adapt the base font size and all the other font sizes adapt proportionally.

Add the following tag into the `customizationConfig.xml` file to change the font family to Open Sans and increase the base font size.

```
<Font name="Opens Sans">
  <Size id="3" pxVal="14"/>
</Font>
```

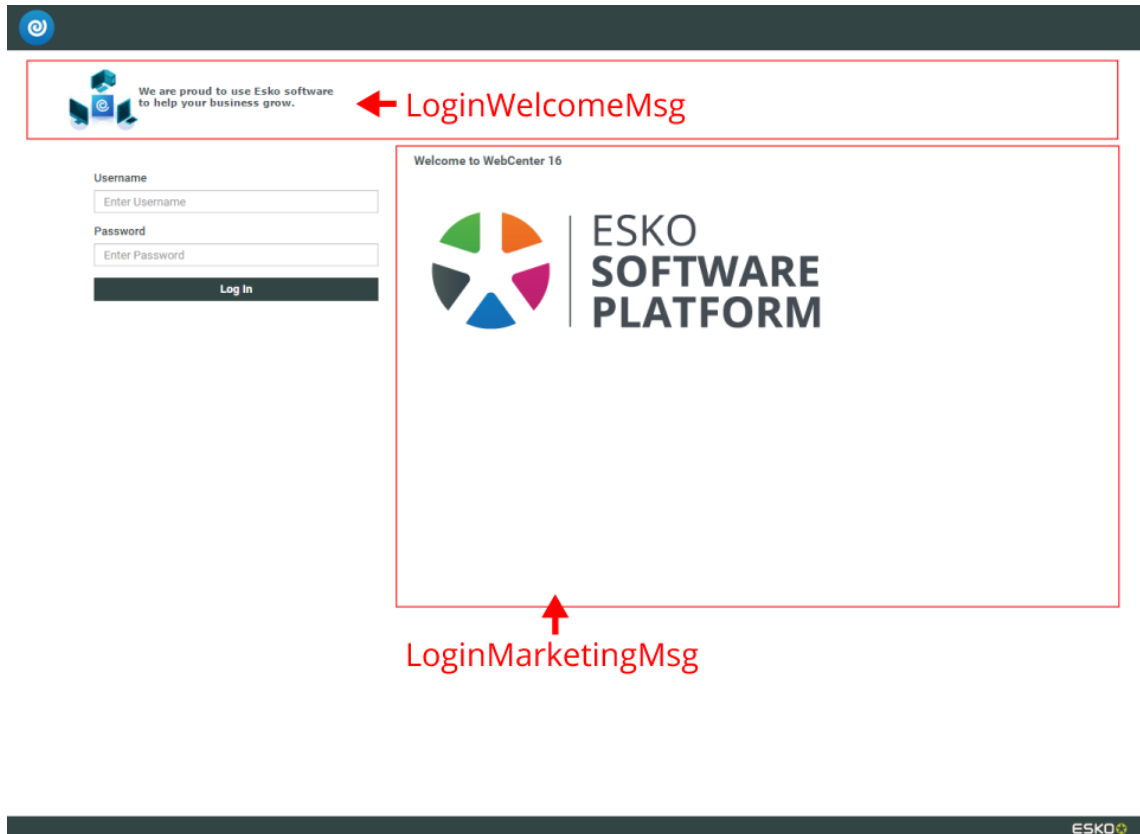
6.5. Customize Static Text

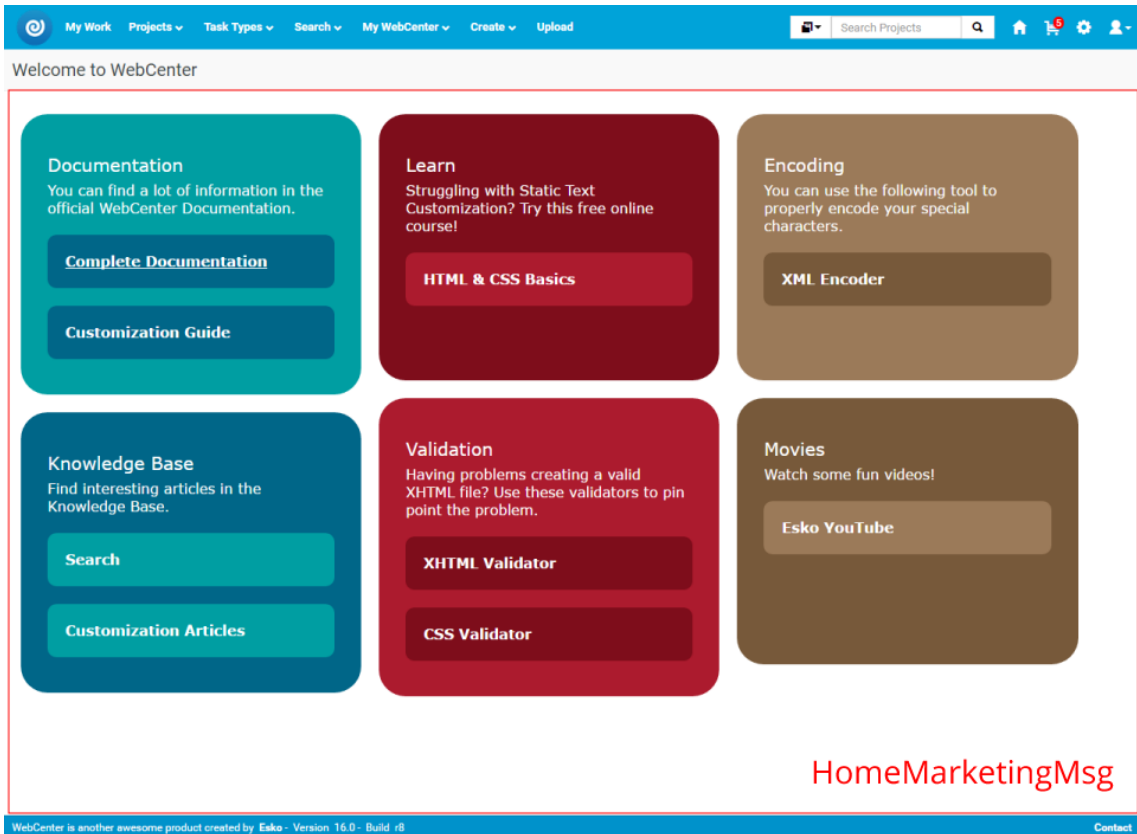
There are a number of places in WebCenter where static text such as a marketing message is displayed. The software provides a straightforward mechanism for you to replace the text with your own HTML pages.

6.5.1. 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 in the footer, at the bottom of each page)

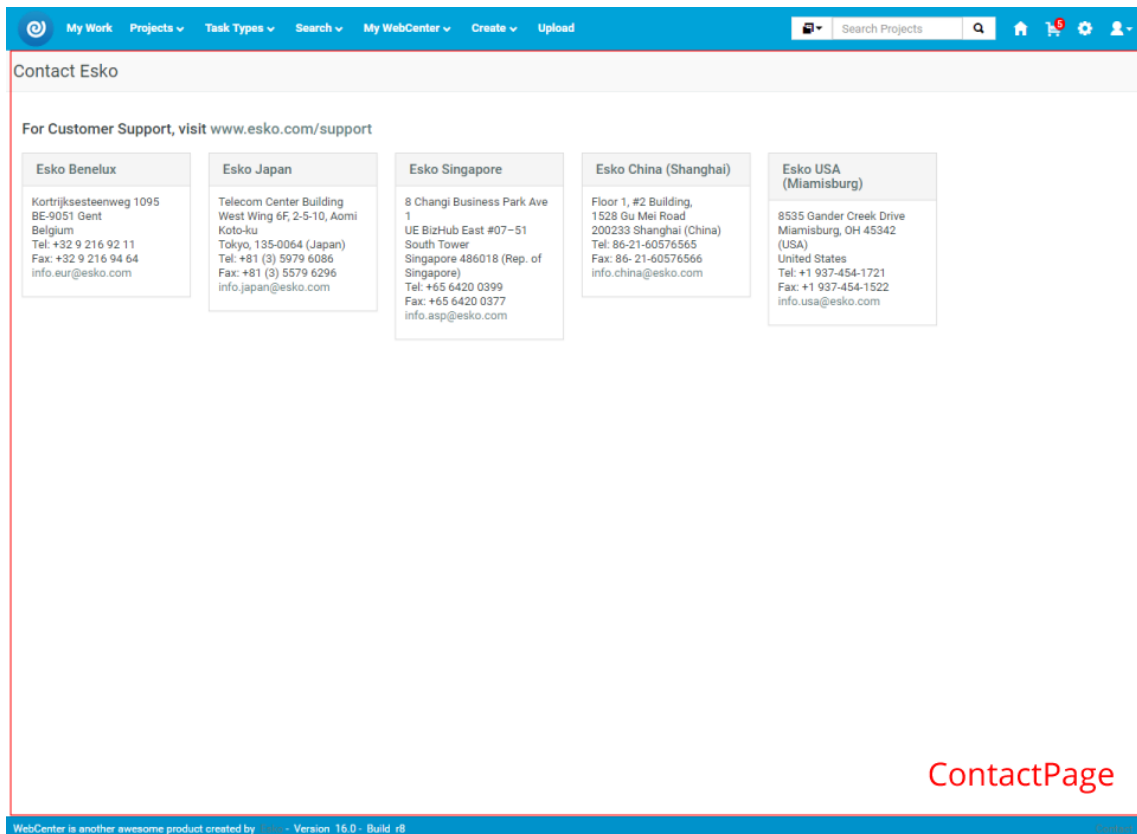




The screenshot displays the WebCenter application interface. At the top is a blue navigation bar with a home icon, a menu with items 'My Work', 'Projects', 'Task Types', 'Search', 'My WebCenter', 'Create', and 'Upload', a search bar labeled 'Search Projects', and icons for home, notifications (with a '5' badge), settings, and user profile. Below the navigation bar is a white header area with the text 'Welcome to WebCenter'. The main content area features six colored cards arranged in a 2x3 grid:

- Documentation** (teal): 'You can find a lot of information in the official WebCenter Documentation.' Includes buttons for 'Complete Documentation' and 'Customization Guide'.
- Learn** (dark red): 'Struggling with Static Text Customization? Try this free online course!' Includes a button for 'HTML & CSS Basics'.
- Encoding** (brown): 'You can use the following tool to properly encode your special characters.' Includes a button for 'XML Encoder'.
- Knowledge Base** (dark teal): 'Find interesting articles in the Knowledge Base.' Includes buttons for 'Search' and 'Customization Articles'.
- Validation** (red): 'Having problems creating a valid XHTML file? Use these validators to pinpoint the problem.' Includes buttons for 'XHTML Validator' and 'CSS Validator'.
- Movies** (dark brown): 'Watch some fun videos!' Includes a button for 'Esko YouTube'.

In the bottom right corner of the content area, the text 'HomeMarketingMsg' is displayed in red. At the very bottom of the page, a blue footer bar contains the text 'WebCenter is another awesome product created by Esko - Version 16.0 - Build #8' on the left and a 'Contact' link on the right.



6.5.2. 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, can either use inline styling in your HTML, or set styles in your CSS to `!important`. For example:

```
a {font-size:15px !important; color:#f00 !important;}
```

- To replace the pieces of static text, add the following tags into the `customizationConfig.xml` file:

```
<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.

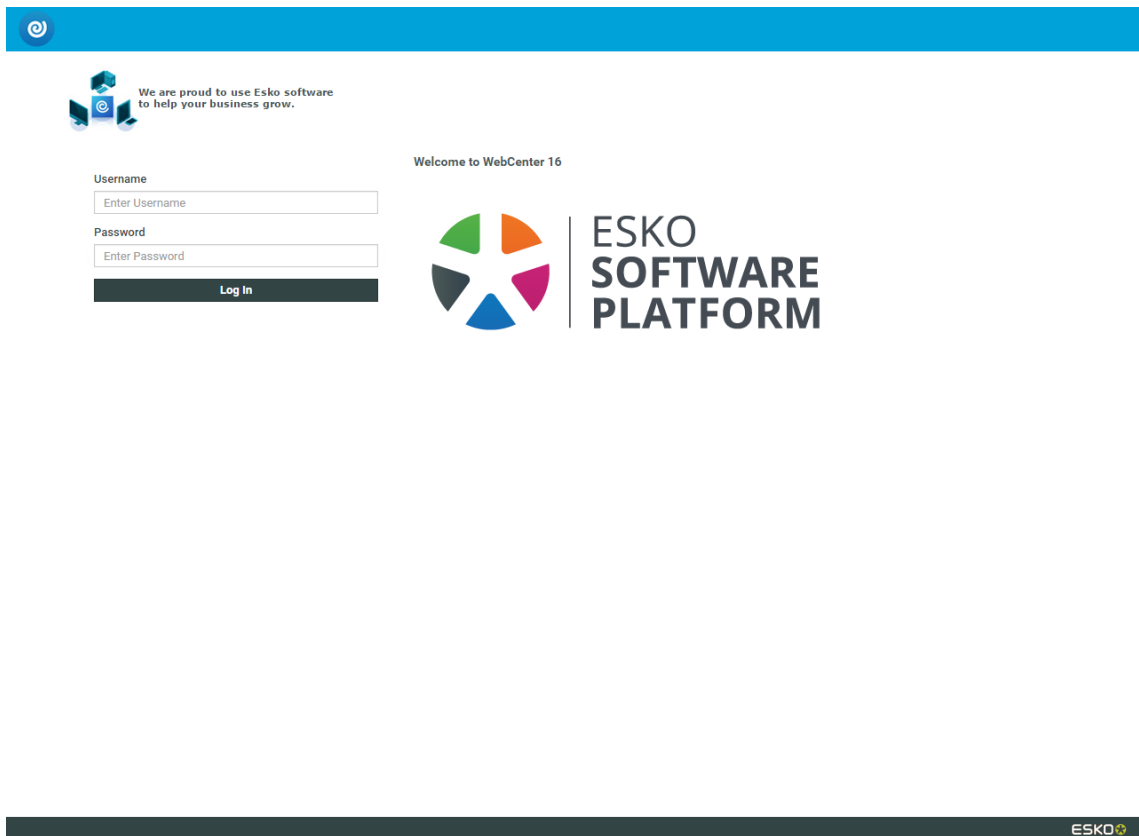
6.5.3. Example: Customizing Static Text

Let's look at the following HTML with which we want to replace the Login page Welcome message. 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.

```
<!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-us">
  <head>
    <title>Login Page Marketing Message</title>
    <style type="text/css">
      #welcome p {color:#4a5656;font-family:Verdana,Arial,Helvetica,sans-serif;font-
size:11px;font-weight:bold;}
      #welcomeimage {align:left;}
    </style>
  </head>
  <body>
    <table>
      <tr>
        <td id="welcomeimage">
          <a href="http://www.esko.com/webcenter">

          </a>
        </td>
        <td id="welcome" width="220px">
          <p>We are proud to use Esko software to help your business grow.</p>
        </td>
      </tr>
    </table>
  </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. Like in the example above, the path for images needs to be relative.



6.6. 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:

```
<Languages RelPath="languages">
  <Language key="en" zipencoding="Cp437"/>
</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.

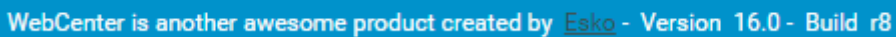
If you are not familiar with the default language files structure and usage, the easiest method is to copy the language file (from the default location `<WebCenter_Instance>\languages`) you

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 results look like the example below and it can contain one or a multitude of strings (you only need to add the ones you wish to modify):

```
<?xml version="1.0" encoding="UTF-8"?>
<language xml:lang="en-us">
  <application name="JSP_Strings">
    <lang_string key="WCAUTHOR" value="WebCenter is another awesome product created by"/>
  </application>
</language>
```

As you can see, we have changed the text in the footer lower left corner. The picture below shows the result of our customization.



6.7. Forgot Password

To add a **Forgot Password?** link to the WebCenter **Log In** page through the `customizationConfig.xml` file, do the following:

1. Open the `customizationConfig.xml` file in an XML editor.
See [Customizing the Look and Feel of WebCenter Using the customizationConfig.xml file](#) on page 15 to know where to find this file.
2. Add the following XML code between the `WebCenterCustomizationConfig` XML tags:

```
<ForgotPassword show="1"/>
```

3. Save the file and place it into the customization directory on your Web server instance.

When you click the **Forgot Password?** link, you are asked to supply user name or email address. You are then sent an email with a temporary password and a link to the **Log In** page.

After logging in, you are prompted to change the password.

**Note:**

The e-mail uses the e-mail customization framework to customize/translate the messages sent to the user.

You can find an example of the data that can be used in the e-mail in `C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Example/user`. This folder contains:

- `reset_password_example_body.xml`
- `reset_password_example_source.xml`
- `reset_password_example_subject.xml`

You should copy these files to `C:/Esko/Artios/WebCenter/ApplicationServer/EmailNotifications/Custom/user` before editing them.

For more information about customizing e-mails, see [Customizing the E-Mail Notifications](#) on page 54.

6.8. Fixed Navigation Bar

To fix the Navigation / Menu bar in the WebCenter application, do the following:

1. Go to **Admin > Preferences > Customizations**.
2. At the bottom of the Customization page, select **Use Fixed Navigation Bar** in the **Other** section.
3. **Save** the changes.

When a user scrolls down any page, the navigation bar is fixed on the top of the page.

7. 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,
- remove unwanted action buttons from tables,
- add custom action buttons.

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



Tip: To see what a menu looks like, hover on it in the WebCenter user interface. This shows the sub menus.

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

See [Menus](#) in the Administration Guide for details.

7.1. 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++).

However, if you are not comfortable using XML, you can also create a custom menu through the WebCenter user interface (see [Create a Custom Menu through the WebCenter User Interface](#) in the Administration Guide).

You will need to:

1. Determine the menu items a particular group of users would require for efficient functioning.
2. [Download the default 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 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 the menu with a name.

6. Upload your custom menu:
 - a) Click the **Menu Upload** button on the **Menus** page,
 - b) In 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.

7.1.1. Download the Default Menu

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

1. Download the default menu:
 - a) On the **Admin > Preferences > 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 Microsoft Edge,
 - **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 29 to familiarize yourself with the file.

7.1.2. Structure of the XML Menu File

Each menu and menu entry is enclosed in a **menuitem** tag.

```

- <menus>
  <!-- The Four Main Menus -->
  - <menuitem>
    <name>NON_ADMIN</name>
    <labeltag>HOME_TAG</labeltag>
    <url>homepage.jsp</url>
    <!-- My work menu -->
    - <menuitem>
      <name>MY_WORK</name>
      <labeltag>MY_WORK_TAG</labeltag>
      <url>mytodolist.jsp</url>
      - <menuitem>
        <name>MYTODOLIST</name>
        <labeltag>MY_WORK_TAG</labeltag>
        <url>mytodolist.jsp</url>
      </menuitem>
      - <menuitem>
        <name>MY_CHECKOUTS</name>
        <labeltag>MY_CHECKOUTS_TAG</labeltag>
        <url>mycheckouts.jsp</url>
      </menuitem>
    </menuitem>
    <!-- Projects menu -->
    - <menuitem>
      <name>PROJGMNT</name>
      <labeltag>PROJGMNTPROJS_TAG</labeltag>
      <url>myfavoriteproj.jsp</url>
      - <menuitem>
        <name>MY_FAVORITEPROJS</name>
        <labeltag>MY_FAVORITEPROJS_TAG</labeltag>
        <url>myfavoriteproj.jsp</url>
      </menuitem>
      - <menuitem>
        <name>MY_PROJECTS</name>
        <labeltag>MY_PROJECTS_TAG</labeltag>
        <url>myprojects.jsp</url>
        - <scope>
          <project_managers/>
        </scope>
      </menuitem>
      - <menuitem>
        <name>MY_PROJECTS_IN</name>
        <labeltag>MY_PROJECTS_IN_TAG</labeltag>
        <url>iamprojectmember.jsp</url>
      </menuitem>
    </menuitem>
  </menus>

```

You can expand and collapse menus and menu entries (use + to expand and - to collapse).

```

- <menuitem>
  <name>NON_ADMIN</name>
  <labeltag>HOME_TAG</labeltag>
  <url>homepage.jsp</url>
  <!-- My work menu -->
+ <menuitem></menuitem>
  <!-- Projects menu -->
+ <menuitem></menuitem>
  <!-- Tasks menu -->
+ <menuitem></menuitem>
  <!-- Search menu -->
+ <menuitem></menuitem>
  <!-- My webcenter menu -->
+ <menuitem></menuitem>
  <!-- Creation Menu -->
+ <menuitem></menuitem>
  <!-- Add Document Menu -->
+ <menuitem></menuitem>
</menuitem>

```



Tip: The comments between <!-- and --> indicate what the tags below correspond to.

Main Row Menus

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

```

- <menuitem>
  <name>NON_ADMIN</name>
  <labeltag>HOME_TAG</labeltag>
  <url>homepage.jsp</url>
  <!-- My work menu -->
+ <menuitem></menuitem>
  <!-- Projects menu -->
+ <menuitem></menuitem>
  <!-- Tasks menu -->
+ <menuitem></menuitem>
  <!-- Search menu -->
+ <menuitem></menuitem>
  <!-- My webcenter menu -->
+ <menuitem></menuitem>
  <!-- Creation Menu -->
+ <menuitem></menuitem>
  <!-- Add Document Menu -->
+ <menuitem></menuitem>
</menuitem>

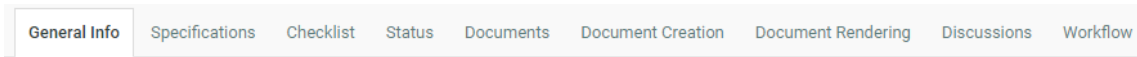
```



Note: It is not possible to edit the Admin menus.

Task Type Details

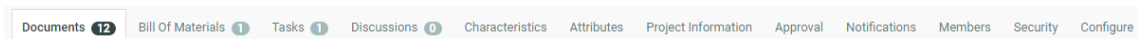
The menu items inside the "TASKTYPEDETAILS" `<menuitem>` (under `<!-- Task type details menu -->`) correspond to the tabbed menu you get when editing a task type (only for users who can create a task type).



You will probably not need to change this.

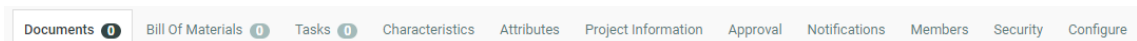
Project Details

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



Template Details

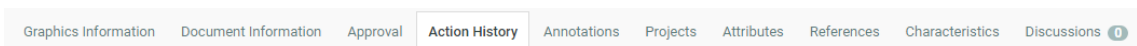
The menu items inside the "TEMPLATEDetails" `<menuitem>` correspond to the tabbed menu you get on the Template Details page (only for Admin users).



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.



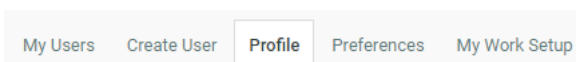
Task Type Details (Read Only Mode)

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).



7.1.3. 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>PROJGMNT</name>
  <labeltag>PROJGMNTPROJS_TAG</labeltag>
  <url>myfavoriteproj.jsp</url>
- <menuitem>
  <name>MY_FAVORITEPROJS</name>
  <labeltag>MY_FAVORITEPROJS_TAG</labeltag>
  <url>myfavoriteproj.jsp</url>
</menuitem>
- <menuitem>
  <name>MY_ALLPROJECTS</name>
  <labeltag>MY_ALLPROJECTS_TAG</labeltag>
  <url>myallprojects.jsp</url>
  <scope>
    <project_managers/>
  </scope>
</menuitem>

```




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

7.1.4. Add Custom Menu Items

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

<p><name></p>	<p>Choose a unique name for your menu.</p> <p>This name is only used to identify the menu in the XML file.</p>
<p><labeltag></p>	<p>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.</p> <p>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.</p> <p>If for example you are adding a menu linking to a(n existing) saved search showing rejected documents:</p>

	<ol style="list-style-type: none"> 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. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <p>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.</p> </div> <p>See Customize the Language Files on page 25 for more information about customizing the language files.</p>
<url>	<p>Enter the page which will be shown when clicking the new menu.</p> <p>Typically this will be a standard WebCenter page which you otherwise would get by more complex navigation.</p>
<scope>	<p>Define which type of users will see the new menu.</p> <ul style="list-style-type: none"> • Enter <project_managers/> if you only want Project Managers to see it. • Enter <admin_users/> if you only want Administrators to see it. • Enter <normal_users/> if you only want normal users to see it. • Enter <project_creators/> if you only want Project Creators to see it. <p>If you want three types of users to see the menu, enter them here. For example:</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <pre><scope> <admin_users/> <project_managers/> <project_creators/> </scope></pre> </div> <p>If you want all user types to see it, don't enter a scope at all.</p>



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:

```
<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.

```
<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>
```

7.2. Examples

7.2.1. 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**.

You can optimise the menu options by removing 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...

```
<menuItem> <name>PROJGMNT</name>
  ...
```

add:

```
<menuItem>
  <name>PROJECTS_NONPROJGMNT</name>
  <labeltag>PROJGMNTPROJS_TAG</labeltag>
  <url>iamprojectmember.jsp</url> <scope>
  <normal_users/> </scope>
</menuItem>
```



Note: The "PROJGMNTPROJS_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 by adding a `<scope>` element:

```
<menuitem> <name>PROJMGMT</name>
      <labeltag>PROJMGMTPROJS_TAG</labeltag>
      <url>myfavoriteproj.jsp</url> <scope>
      <admin_users/> <project_managers/> </scope> ...
```

3. Save and upload your custom XML file.

7.2.2. 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.

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 36.

If you have different project request types, [Create a Project from Different Templates](#) on page 37.

You can also [Show Projects Created from the Request Template\(s\)](#) on page 40 and [Add Your Requests to Your My Work Page](#) on page 41.

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").

```
<menuitem> <name>NEW_REQUEST</name>
      <labeltag>NEW_REQUEST_TAG</labeltag>
      <url>projmgmtnewproject.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 33 and [Customize the Language Files](#) on page 25).
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.

```
<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 33 and [Customize the Language Files](#) on page 25).
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.

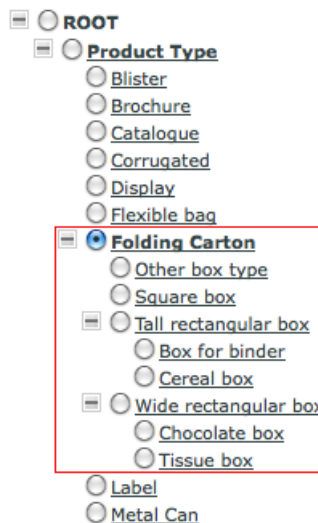
Create a Project and Run a CAD Standard

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.



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.

- 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.

```
<menuitem> <name>NEW_REQUEST</name>
      <labeltag>NEW_REQUEST_TAG</labeltag>
      <url>projmgmntnewproject.jsp?template=New
%20Box&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 37.

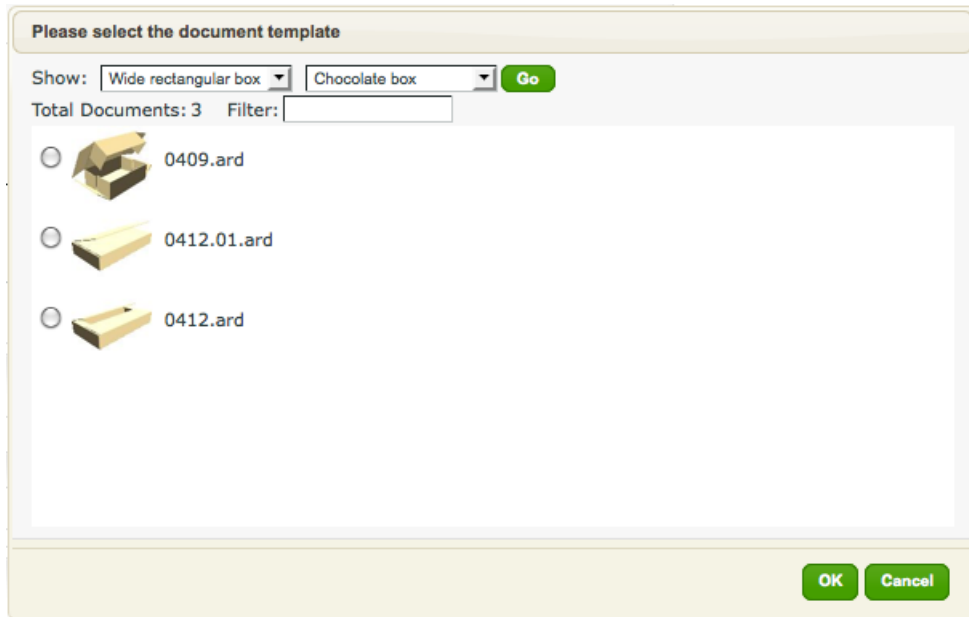
Then specify the project template and document template characteristics to use in the XML:

```
<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 33 and [Customize the Language Files](#) on page 25).
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 users 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):

```
<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.

Search for Projects

Fill in one or more fields. Prefer less fields? Try [Keyword Search](#).

		Search Results Options					Search
		View(List)	Column Order	Editable	Sort	Order	
<input type="checkbox"/> Use Advanced Search Characters ?		<input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1	Clear this Form
<input type="checkbox"/> Search Panel Design		<input checked="" type="checkbox"/>	2	<input type="checkbox"/>	<input type="checkbox"/>		Save this Search
<input type="checkbox"/> Enable SmartNames ?		<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		Show Result Layout Options
Project Name	<input type="text"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Description	<input type="text"/>	<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Project Status	Show All ▼ +I	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Created	<input type="text"/>	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Modified	<input type="text"/>	<input checked="" type="checkbox"/>	3	<input type="checkbox"/>	<input type="checkbox"/>		
Due	<input type="text"/>	<input checked="" type="checkbox"/>	4	<input type="checkbox"/>	<input type="checkbox"/>		
Customer/Location	<input type="text"/>	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Salesperson	<input type="text"/>	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Project Manager	<input type="text"/>	<input checked="" type="checkbox"/>	5	<input type="checkbox"/>	<input type="checkbox"/>		
Manager's Location	<input type="text"/>	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		
Add Role							
Attributes...							
Characteristics...							
Bill of Materials	Show All ▼	<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>		

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

Name *

Type

Saved Search

URL *

Scope

Role Groups All Users Specific Groups

Tip: Select one or more Role Groups to limit the Visibility of this section to the selected Groups







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 **MyWork > Edit My Work setup**).

My Work

▼ My Requests ↻

<input type="checkbox"/>	Name ↑	Description	Status
<input type="checkbox"/>	 Request0001		<input type="button" value="Active"/>
<input type="checkbox"/>	 Request0002		<input type="button" value="On Hold"/>
<input type="checkbox"/>	 Request0003		<input type="button" value="Active"/>
<input type="checkbox"/>	 Request0004		<input type="button" value="Completed"/>

Found: 4 items. Page 1 of 1

7.2.3. Removing the Keyword Search

To remove the ability to search using a keyword, add the following under the "NON_ADMIN" <menuitem>:

```
<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.

7.3. 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.

8. Integrating an External Web Connection into the User Interface

An external web connection is a connection to some data in an external system (usually a MIS system). The MIS system calculates the data (for example a bill's details) and WebCenter retrieves this data through the external web connection and displays it on a WebCenter page.

After configuring the external web connection (see [Configure External Web Connections](#) in the Administration Guide), you can choose to add a link to it in a WebCenter top-level menu, project menu or a My Work section.

8.1. Integrating an External Web Connection into a Top-Level Menu

To integrate an external web connection into a WebCenter top-level menu, you need to edit the `menu.xml` file (see [the Customizing the WebCenter Menus chapter](#) for some background information about menus and how to edit them).



Note: You can only have one link to an external web connection in the top level menu per menu file.

If the external system you are connecting to outputs XML, you will also need an XSL (eXtensible Stylesheet Language) file to transform this to HTML and display it in WebCenter. Your WebCenter installation comes with an example XSL file, that you can customize.

1. If the external system you are connecting to outputs XML, edit the example XSL file as necessary:

- a) Copy the example XSL file called `externalwebtransform.xml`, located in your `custom_example\externalwebconnection` folder, and paste it into `custom\externalwebconnection`.



Tip: Both the `custom_example\externalwebconnection` and the `custom\externalwebconnection` folders are located on your Web server instance, for example `C:\Artios\WebCenter\WebServer\tomcat\webapps\WebCenter_instance\custom_example\externalwebconnection`.

- b) Edit the `custom\externalwebconnection\externalwebtransform.xml` file as necessary. You can also rename it if desired.



Tip: See [the Customizing the E-Mail Notifications chapter](#) for principles and tips about editing an XSL file.

2. [Download the default menu](#) to use as a starting point.

3. In the `menu.xml` file, locate the part starting with `<!-- External Web Connection (1-level Menu)`.

4. Uncomment the following section:

```
<!-- <menuItem>
  <name>MYEXTERNALWEB_INFO</name>
  <labeltag>MYEXTERNALWEB_INFO_TAG</labeltag>
```

```
<url>myexternalwebinfo.jsp?ExternalWebConfigName=<Name of your external
web configuration>&XSLTransform=<Name of your xsl transformation in custom/
externalwebconnection directory>&CallFromWebserver=<0/1 depending on whether the
call to the external system needs to be made on the tomcat server (1), or the jboss
server (0)></url>
</menuitem> -->
```



Tip: To uncomment, remove the <!-- and --> signs that surround the section.

5. Edit this section as necessary:

- a) Replace <Name of your external web configuration> by the name of your external web connection.
- b) If the external system you are connecting to outputs XML, replace <Name of your xsl transformation in custom/externalwebconnection directory> by the name of the XSL file in your custom\externalwebconnection folder.

You can skip this step if you didn't rename the externalwebtransform.xml file.

- c) In the CallFromWebserver parameter, replace <0/1 depending on whether the call to the external system needs to be made on the tomcat server (1), or the jboss server (0)> by either:

- <0> if you want to connect to the external system from the WebCenter Application Server,
- <1> if you want to connect to the external system from the WebCenter Web Server.

If you don't specify anything, WebCenter will try to connect to the external system from the Application Server.

For example, if:

- your external web connection is called "ExternalWebConfig1",
- you haven't renamed the externalwebtransform.xml file,
- you want to connect from the Application Server,

then you should include the following section menu.xml file:

```
<menuitem>
  <name>MYEXTERNALWEB_INFO</name>
  <labeltag>MYEXTERNALWEB_INFO_TAG</labeltag>
  <url>myexternalwebinfo.jsp?ExternalWebConfigName=ExternalWebConfig1</url>
</menuitem>
```

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

7. 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**.

8. If desired, assign the menu to specific users (or companies).

9. 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.

When clicking the menu item, the system will send a request to the configured service, and show the result in a page containing the normal WebCenter navigation and:

- for a HTML connection: the returned HTML.



Note: The external system must make sure this HTML is sufficiently self-contained.

- for an XML connection: the HTML resulting from converting the returned XML with the XSL file you configured.

8.2. Integrating an External Web Connection into a Project Menu

To integrate an external web connection into a WebCenter project menu, you need to edit the `menu.xml` file (see [the Customizing the WebCenter Menus](#) for some background information about menus and how to edit them).



Note: You can only have one external web connection menu item per project.

If the external system you are connecting to outputs XML, you will also need an XSL (eXtensible Stylesheet Language) file to transform this to HTML and display it in WebCenter. Your WebCenter installation comes with an example XSL file, that you can customize.

1. If the external system you are connecting to outputs XML, edit the example XSL file as necessary:
 - a) Copy the example XSL file called `externalwebtransform.xml`, located in your `custom_example\externalwebconnection` folder, and paste it into `custom\externalwebconnection`.



Tip: Both the `custom_example\externalwebconnection` and the `custom\externalwebconnection` folders are located on your Web server instance, for example `C:\Artios\WebCenter\WebServer\tomcat\webapps\WebCenter_instance\custom_example\externalwebconnection`.

- b) Edit the `custom\externalwebconnection\externalwebtransform.xml` file as necessary. You can also rename it if desired.



Tip: See [the Customizing the E-Mail Notifications](#) for principles and tips about editing an XSL file.

2. [Download the default menu](#) to use as a starting point.
3. In the `menu.xml` file, locate the part starting with `<!-- External Web Connection (Project-level Menu)`.
4. Uncomment the following section:

```
<!-- <menuItem>
    <name>PROJDETAILSEXTERNALWEBINFO</name>
    <labeltag>PROJDETAILSEXTERNALWEBINFO_TAG</labeltag>
    <url>projdetailsexternalwebinfo.jsp?ExternalWebConfigName=<Name of your
External Web configuration>&XSLTransform=<Name of your xsl transformation in
custom/externalwebconnection directory>&CallFromWebserver=<0/1 depending on
whether the call to the External Web system needs to be made on the tomcat server
(1), or the jboss server (0)></url>
</menuItem> -->
```



Tip: To uncomment, remove the `<!--` and `-->` signs that surround the section.

5. Edit this section as necessary:

- a) Replace <Name of your external web configuration> by the name of your external web connection.
- b) If the external system you are connecting to outputs XML, replace <Name of your xsl transformation in custom/externalwebconnection directory> by the name of the XSL file in your custom\externalwebconnection folder.

You can skip this step if you didn't rename the externalwebtransform.xsl file.

- c) In the CallFromWebserver parameter, replace <0/1 depending on whether the call to the external system needs to be made on the tomcat server (1), or the jboss server (0)> by either:

- <0> if you want to connect to the external system from the WebCenter Application Server,
- <1> if you want to connect to the external system from the WebCenter Web Server.

If you don't specify anything, WebCenter will try to connect to the external system from the Application Server.

For example, if:

- your external web connection is called "ExternalWebConfig2",
- you have renamed the externalwebtransform.xsl file to ExternalWebProjecttransform.xsl,
- you want to connect from the Web Server,

then you should include the following section menu.xml file:

```
<menuitem>
  <name>PROJDETAILSEXTERNALWEBINFO</name>
  <labeltag>PROJDETAILSEXTERNALWEBINFO_TAG</labeltag>
  <url>projdetailsexternalwebinfo.jsp?
    ExternalWebConfigName=ExternalWebConfig2&amp;XSLTransform=ExternalWebProjecttransform.xsl&amp;CallFromWebServer=1
  </url>
</menuitem>
```

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

7. 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**.

8. Go to a WebCenter project, and check whether the menu works.

When clicking the menu item, the system will send a request to the configured service, and show the result in a page containing the normal WebCenter navigation and:

- for a HTML connection: the returned HTML.



Note: The external system must make sure this HTML is sufficiently self-contained.

- for an XML connection: the HTML resulting from converting the returned XML with the XSL file you configured.

8.3. Integrating an External Web Connection into a My Work Section

To integrate an external web connection into a **My Work** section, you need to create a new homepage section in the **Admin > Preferences > My Work** page.

If the external system you are connecting to outputs XML, you will also need an XSL (eXtensible Stylesheet Language) file to transform this to HTML and display it in WebCenter. Your WebCenter installation comes with an example XSL file, that you can customize.

1. If the external system you are connecting to outputs XML, edit the example XSL file as necessary:
 - a) Copy the example XSL file called `externalwebtransform.xml`, located in your `custom_example\externalwebconnection` folder, and paste it into `custom\externalwebconnection`.



Tip: Both the `custom_example\externalwebconnection` and the `custom\externalwebconnection` folders are located on your Web server instance, for example `C:\Artios\WebCenter\WebServer\tomcat\webapps\WebCenter_instance\custom_example\externalwebconnection`.

- b) Edit the `custom\externalwebconnection\externalwebtransform.xml` file as necessary. You can also rename it if desired.



Tip: See [the Customizing the E-Mail Notifications chapter](#) for principles and tips about editing an XSL file.

2. Click **Admin > Preferences > My Work**. The **My Work Sections** page appears.
3. Click **Create New My Work Section**.
4. Give your new section a **Name**.
5. In the **Type** option, select **Generic**.
6. In the **URL** field:
 - a) Paste the following:

```
http://<webserver_name>/<Instance name>/myexternalwebinfo.jsp?
ExternalWebConfigName=<Name of your External Web
configuration>&XSLTransform=<Name of your xsl transformation in custom/
externalwebconnection directory>&CallFromWebserver=<0/1>&homepageSection=1
```

- b) Replace `http://<webserver_name>/<Instance name>` by the address of your WebCenter server.
 - c) Replace `<Name of your external web configuration>` by the name of your external web connection.
 - d) If the external system you are connecting to outputs XML, replace `<Name of your xsl transformation in custom/externalwebconnection directory>` by the name of the XSL file in your `custom\externalwebconnection` folder.
You can skip this step if you didn't rename the `externalwebtransform.xml` file.
 - e) In the `CallFromWebserver` parameter, replace `<0/1>` depending on whether the call to the external system needs to be made on the tomcat server (1), or the jboss server (0) by either:

- <0> if you want to connect to the external system from the WebCenter Application Server,
- <1> if you want to connect to the external system from the WebCenter Web Server.

If you don't specify anything, WebCenter will try to connect to the external system from the Application Server.



Note: Make sure you keep the `homepageSection=1` parameter at the end!

For example, if:

- the address of your WebCenter server is `http://webcenter2.esko.com/WebCenter_Inst/`,
- your external web connection is called "ExternalWebConfig3",
- you have renamed the `externalwebtransform.xsl` file to `ExternalWebHomepage.xsl`,
- you want to connect from the Application Server,

then you should use the following as URL:

```
http://webcenter2.esko.com/WebCenter_Inst/myexternalwebinfo.jsp?
ExternalWebConfigName=ExternalWebConfig3&XSLTransform=ExternalWebHomepage.xsl
&homepageSection=1
```

7. In **Scope**, choose which type of users will be able to use this section in their **My Work** page:
 - **All Users**,
 - **Administrators**,
 - **Project Managers**.
8. You can use groups to further filter the users that will be able to see the section. In **Role Groups**, choose either:
 - **All Users** if you want all users of the type specified in **Scope** to be able to use the section,
 - **Specific Groups** if you want to restrict the section's visibility to specific groups (within that user type).

In this case, select the groups which will be able to use this section.
9. Click **Create**.
10. Log out and log in to WebCenter as one of the users to which you assigned the new My Work section, and check whether the link to the external web connection works.

When clicking the link, the system will send a request to the configured service, and show the result in a page containing the normal WebCenter navigation and:

- for a HTML connection: the returned HTML.



Note: The external system must make sure this HTML is sufficiently self-contained.

- for an XML connection: the HTML resulting from converting the returned XML with the XSL file you configured.

9. 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.

9.1. Create Image Label for an Attribute

You can create Project request forms or Order forms with graphical labels by setting image labels for 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.

Remove Grid Layout

NAPR_TASK_ATT1	NAPR_TEXTATT_2
vira_res_8155	MAKE PROJECT THUMBNAIL
napr_yes_no	napr_rich1
napr_docref_latest_version	NAPR_TEXTATT_3 [Button Click]
NAPR_ATT4	Text
NAPR_ATT6	

Attribute	NAPR_TASK_ATT1
Attribute Type:	Text
Label Text:	NAPR_TASK_ATT1
Label Image:	Danger.jpg
Position:	Left
Custom Unit Label:	
Span Columns:	1
Span Rows:	1
Style:	WebCenter Attribute
Style Rule:	No Rule
Visibility:	Editable
Restrict Values:	Restricted Set: NAPR_RES1
Rendering:	Dropdown
Allow Multiple Values:	<input type="checkbox"/>
Info Page:	Info HTML page (name+extension) <Web/instanceRoot>/custom/AttributesHelp/

Note: The image name is case sensitive.

You can see image labels appearing instead of attribute names in the forms.

Regulatory Input / Briefing

Packaging Symbols

Packaging Code	Barcode type	Barcode	Approved for Use?
*	*	*	*
<input type="text"/>	---Please Select---	<input type="text"/>	---Please Select---

Required Legal Disclaimers / Warnings

[Show Rich Text Controls](#)


Tip: In case you wish to display a check box for the **attribute**, make sure you use an **attribute** with a **restricted set** containing only the values Yes and No.

Note: You can either have a text label or an image label, but not both.

9.2. Create HTML Information 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 information. This is especially useful when technical fields need to be supported with extended HTML help.

There will be an information  icon that is added next to the attribute name or attribute image label. When the user clicks on this icon, a pop-up 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\AttributesHelp`).
3. Open the **Attribute category** and select the **attribute** for which you want to provide HTML help or information.
4. In the cell details, enter the correct file name (of the HTML file) in the **Info Page** field and save your **Attribute Category**.



Details for selected cell:

Attribute	w2p-Finishing
Attribute Type:	Text
Label:	Finishing
Label Image:	
Span Columns:	1 ▼
Style:	WebCenter Attribute ▼ Add Style
Visibility:	VisibilityRule ▼ Add Rule Edit Rule Delete Rule
Allow Multiple Values:	<input type="checkbox"/>
Info Page:	Info HTML page (name+extension) <WebInstanceRoot>/custom/AttributesHelp/ sheet-and-roll.html

If you have an HTML information page linked to an **attribute**, you can see an information icon next to the attribute in the form that you created. When you click on the information , you can see a pop-up window with the information to help the User.

▼ Order Information

Is this an Order? **Approval Required Before Print?**

Label Shape Quantity

Varnish Specialties

Finishing*


Add More Information?

Create Project


10.31.140.85/WebCenter_Inst/custom/AttributesHel... ⌵ ⌵ ✕

10.31.140.85/WebCenter_Inst/custom/AttributesHelp/sheet-and-roll.html

The picture below show the examples of labels printed on sheet and roll.



Labels on Sheet



Labels on Roll

10. 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.

10.1. E-Mail Notification Templates In Detail

Each of the notification message types consists of two XSLT template files

1. one for the message's body and
2. 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
```

```
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.

10.2. Start from the example

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 **Default** folder which contains an elaborate e-mail customization.



Note: This example has been tested for compatibility with the following e-mail clients: Outlook, 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.

10.2.1. 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:

```
<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:

```
<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:

```
<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:

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

By:

```
<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.

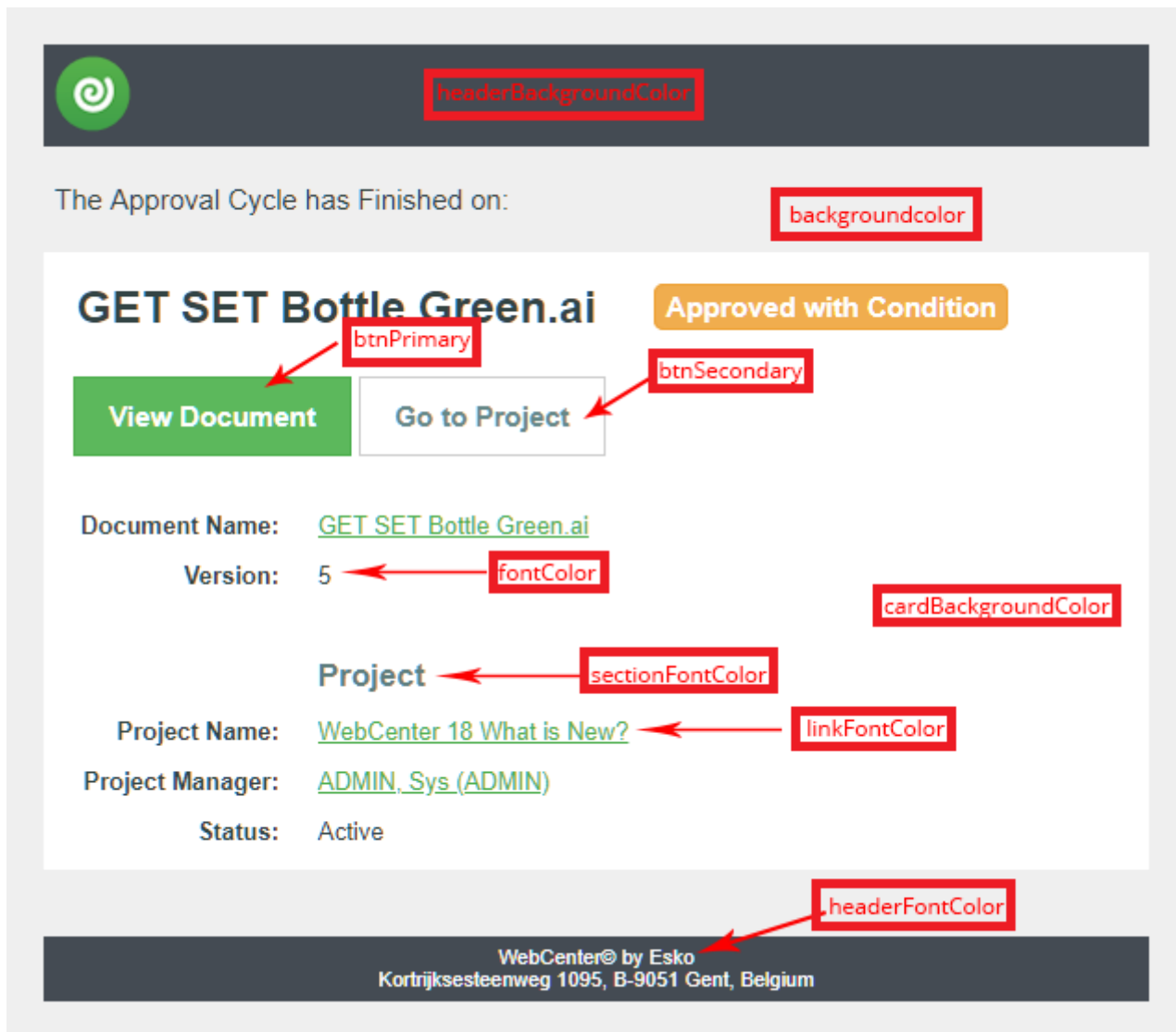
Customize.xsl

The `Customize.xsl` file contains three parts: **Styling**, **Content** and **Styles**.

You can customize fonts and colors of the notifications in this file.

Styling

The **styles** section contains all the elements that make up the fonts and the colors of the e-mail notifications. For example, you can set a background color, font, font-size, font color, etc..



Example for styling in a customize.xsl file is given below:

```
<xsl:variablename="fontFamily">Roboto, Arial, Helvetica, sans-serif</xsl:variable>
  <xsl:variablename="fontSize">13px</xsl:variable>
  <xsl:variablename="buttonFontSize">15px</xsl:variable>
  <!-- colors -->
  <xsl:variablename="fontColor">#334444</xsl:variable>
  <xsl:variablename="backgroundColor">#efefef</xsl:variable>
  <xsl:variablename="cardBackgroundColor">#ffffff</xsl:variable>
  <xsl:variablename="headerBackgroundColor">#babe13</xsl:variable>
  <xsl:variablename="headerFontColor">#ffffff</xsl:variable>
  <xsl:variablename="linkFontColor">#4cae4c</xsl:variable>
  <xsl:variablename="sectionFontColor">#5f7e7e</xsl:variable>
  <!-- Primary Button -->
  <xsl:variablename="btnPrimaryBackgroundColor">#5cb85c</xsl:variable>
  <xsl:variablename="btnPrimaryBorderColor">#4cae4c</xsl:variable>
  <xsl:variablename="btnPrimaryFontColor">#ffffff</xsl:variable>
  <!-- Secondary Button -->
```

```
<xsl:variablename="btnSecondaryBackgroundColor">#ffffff</xsl:variable>
<xsl:variablename="btnSecondaryBorderColor">#cccccc</xsl:variable>
<xsl:variablename="btnSecondaryFontColor">#668888</xsl:variable>
```

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.

Styles

The **styles** 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.

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 card
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 card

The **information card** 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.xml` file will now look something like this:

```
<!-- Section Header Style -->
  <xsl:variable name="sectionHeaderStyle"> font: 24px <xsl:value-of
  select="$fontFamily"/>; font-weight: bold; color: <xsl:value-of
  select="$sectionFontColor"/>; padding: 30px 0 5px 12px; </xsl:variable>
```

URL.xml

The `URL.xml` 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'"/>
```

By:

```
<xsl:variable name="internalURL" select="'http://my.intranet.esko.com'"/>
  <xsl:variable name="correctURL" select="'https://correct.esko.com'"/>
```



Note: Pay attention to the usage of both double and single quotes.

Customize Time Zone details in notifications

You can insert customized timezone information in the notifications emails in by modifying the `Shared.xml` file. You can use `Shared.xml` file's templates as they are quite generic and can be used in context of any type of any event-type notification customization. For example., Comments for specific logic: `dateTimeWithTimeZone` and `timeZone` templates allow for inserting the time zone as of the Application server. This ensures that Date and Time values indicate what Time zone they are related to.



Note: Time zone information is inserted by default in **Overdue reminder notification emails** and **Approval Started** emails. For other notifications, Time Stamp can be added via customization only.

10.2.2. 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.

10.3. 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.

10.4. 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 it 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:

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

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).

10.5. 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.

10.6. 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.

10.6.1. 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:

```
<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 the wheels element.

11. Customizing the Translation Service

By Default, the Auto Translation service supports Google Translate API. You can override this service by replacing "translator-service-1.0.jar" with a custom implemented jar file.

12. Customizing the Onboarding Hook

The onboarding hook allows you to enable a custom onboarding flow for WebCenter users, using a tool of your choice (i.e. intro.js, walk.me, userlane, userpilot, ...).

You can include a snippet of JavaScript and its needed dependencies on every page. To enable this feature, add your snippet to `WEBCENTER_INSTANCE/custom/onboarding/onboard.js`. If your JavaScript snippet needs external dependencies, you can add the URLs of these dependencies to `WEBCENTER_INSTANCE/custom/onboarding/dependencies.txt`

After installing the `onboard.js` and `dependencies.txt` files the web server must be restarted to enable the custom onboarding hook.