

# Mercury™ IT Governance Center

## Customizing the Standard Interface

Version 5.5.0

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: U.S. Patent Nos. 5,701,139; 5,657,438; 5,511,185; 5,870,559; 5,958,008; 5,974,572; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332; 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912 and 6,694,288. Other patents pending. All rights reserved.

Mercury, Mercury Interactive, the Mercury Interactive logo, LoadRunner, LoadRunner Test-Center, QuickTest Professional, SiteScope, SiteSeer, TestDirector, Topaz and WinRunner are trademarks or registered trademarks of Mercury Interactive Corporation or its subsidiaries, in the United States and/or other countries. The absence of a trademark from this list does not constitute a waiver of Mercury Interactive's intellectual property rights concerning that trademark.

All other company, brand and product names are registered trademarks or trademarks of their respective holders. Mercury Interactive Corporation disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury Interactive Corporation  
1325 Borregas Avenue  
Sunnyvale, CA 94089 USA  
Tel: (408) 822-5200  
Fax: (408) 822-5300

© 2004 Mercury Interactive Corporation. All rights reserved.

If you have any comments or suggestions regarding this document, please send them via e-mail to [documentation@merc-int.com](mailto:documentation@merc-int.com).

---

# Table of Contents

<b>Chapter 1</b>	
<b>Introduction</b> .....	<b>1</b>
<b>About This Document</b> .....	<b>1</b>
<b>Intended Audience</b> .....	<b>2</b>
<b>Document Conventions</b> .....	<b>2</b>
<b>Additional Resources</b> .....	<b>3</b>
Related Documentation .....	3
Customer Support .....	3
Education Services .....	4
<b>Chapter 2</b>	
<b>JSP Directory Structure</b> .....	<b>5</b>
<b>Chapter 3</b>	
<b>Altering the HTML Layout</b> .....	<b>7</b>
<b>Replacing the Logo</b> .....	<b>7</b>
<b>Changing the Presentation Layer</b> .....	<b>8</b>
Global Style Sheets.....	9
Common Header File (Header.jsp).....	10
JSP Definitions .....	11
<b>Methods of Customization</b> .....	<b>11</b>
Method 1 - Creating a New Presentation Layer .....	12
Method 2 - Directly Changing the knta Home Directory Files .....	12
Method 3 - Creating New JSP and Related Files .....	13
<b>Guidelines for Modifying JSP Pages</b> .....	<b>13</b>
<b>Changing the Appearance of a Particular Screen</b> .....	<b>14</b>
<b>Example: Modifications to the New Request Screen</b> .....	<b>14</b>
Reordering Sections of the Create New Request Page .....	15
Adding HTML Layout Customizations .....	15
<b>Chapter 4</b>	
<b>Customizing the Menu</b> .....	<b>17</b>
<b>Menu Structure Overview</b> .....	<b>17</b>

<menu seq="#">.....	19
<menu> </menu>.....	19
<menu_item> </menu_item> .....	20
<b>Creating a New Menu .....</b>	<b>20</b>
<b>Modifying the Menus.....</b>	<b>21</b>
<b>Controlling Menu Access.....</b>	<b>22</b>
<b>Notes on Menu xml files .....</b>	<b>24</b>
<b>Configuring the URLs in the Menus .....</b>	<b>24</b>
Request Create Dispatcher .....	25
Report Create Dispatcher .....	25
Smart URL Formats .....	25
 <b>Chapter 5</b>	
<b>Customizing Dashboard Templates .....</b>	<b>27</b>
<b>Template Structure Overview .....</b>	<b>28</b>
Portlet Structure in Template .....	28
<PortletType> .....	29
<PortletTitle> .....	29
<section> and <rowInSection>.....	29
<maxRows> .....	30
<personalizationValue> .....	30
<b>Creating a New Template .....</b>	<b>31</b>
<b>Modifying Templates .....</b>	<b>32</b>
<b>Enabling User Access to Templates .....</b>	<b>32</b>

# Chapter 1 Introduction

The standard interface can be configured to meet business processes and standards. This document details the directory structure and methods of customization for changing the presentation of the standard interface.

## About This Document

This document provides information on customizing the presentation of the standard interface. Each chapter or appendix covers specific topics on the customization process:

<i>JSP Directory Structure</i>	Describes the directory structure of JSP and related files.
<i>Altering the HTML Layout</i>	Discusses ways to customize the HTML layout.
<i>Customizing the Menu</i>	Defines the main elements of the menu files and provides instructions for modifying them.
<i>Customizing Dashboard Templates</i>	Discusses the template structure and configuration notes.

## Intended Audience

This document provides details for customizing the standard interface and is used primarily by:

- Advanced users or administrators responsible for the look and feel of the Mercury IT Governance Center
- Advanced users or administrators responsible for implementing corporate branding specifications
- Advanced users or administrators who wish to modify the interface and navigation to better fit with their processes

## Document Conventions

*Table 1-1* lists the types of conventions used in this document.

*Table 1-1. Document conventions*




Convention	Description	Example
<b>Button, menu, tabs</b>	Names of interface components that can be clicked (such as buttons, menus, and tabs) are shown in bold.	<b>Apply</b> button
Fields, Windows, Pages	Names of windows, fields, and pages are shown as displayed.	New Request window
Code	Code input and output are shown as displayed.	CauchoConfigFile C:/ITG_Home/conf/ resin.conf
<i>Link</i>	Linked URLs, filenames, and cross references are shown as blue italicized text.	<a href="http://www.merc-int.com">www.merc-int.com</a>
<i>Variable</i>	Variables are shown as italicized text.	<i>ITG_Home</i> /bin directory
Note	Used to identify note boxes that contain additional information.	
Caution	Used to identify caution boxes that contain important information. Follow the instructions in all caution boxes, failure to do so may result in loss of data.	

Table 1-1. Document conventions

Convention	Description	Example
Example	Used to identify example boxes that contain examples of related procedure.	

## Additional Resources

Mercury Interactive provides the following additional resources to help you successfully customize the standard interface:

- [Related Documentation](#)
- [Customer Support](#)
- [Education Services](#)

## Related Documentation

The Library includes additional documents related to the topics discussed in this guide. Access the Library through the Mercury ITG Center online help.

*Security Model Guide and Reference*

This document presents an overview of the data security model and provides instructions for controlling access to different entities.

*Using the Dashboard*

This document provides details for defining and configuring the Dashboard and custom Portlets.

## Customer Support

Customer support for Mercury IT Governance Center and additional product information can be accessed from the Mercury Interactive Support Web site at <http://support.mercuryinteractive.com>.

## Education Services

Mercury Interactive provides a complete training curriculum to help you achieve optimal results using the Mercury IT Governance Center. For more information, visit the Education Services Web site at <http://www.merc-training.com/main/ITG>.



# Chapter 2

## JSP Directory Structure

The Mercury IT Governance (ITG) Center Web Architecture is powered by a high-performance JSP (Java Server Page) engine. A JSP is simply a file containing familiar HTML elements that also has special tags and limited use of Java code. A great deal of information and training is available for JSP users.

Note

Throughout this document, a JSP is often referred to as a page and the Mercury ITG Center home directory is referred to using the *ITG\_Home* variable.

All JSP and related files are located in:

```
ITG_Home/server/kintana/deploy/itg.war/web
```

As shown in this path, the `server` subdirectory contains a directory named `kintana`, which contains folders that hold various types of files used by the Mercury ITG Server. In a Single Server configuration, the `server` subdirectory will only contain the `kintana` directory, which is the default name used for the Mercury ITG Server instance. In a server cluster configuration, each server has its own directory within *ITG\_Home*/server. The name of the directory will match the name of the Mercury ITG Server instance as configured by the `KINTANA_SERVER_NAME` server parameter in the `server.conf` file.

Within the *ITG\_Home*/server/kintana/deploy/itg.war/web directory are several subdirectories. One such directory is the `knta` directory, which represents the presentation key. Directories can be added under `web` to contain copies of files in the `knta` file system because all URL links have been made

relative from the presentation key directory. For more information, see *“Methods of Customization”* on page 11.

The `knta` directory also contains various subdirectories. The `global` directory is reserved for files that affect JSPs system wide. The other directories—such as `crt`, `dlv`, and `drv`—contain product-specific files. The file hierarchy for these directories is similar to the `global` directory under each of these folders. Language-specific files are stored in Java resource bundles in the `resource` directory. The `resource` directory excludes images and Java Script.

The directory structure maps directly to a URL page. Consider the following URL as an example:

```
http://host.machine.com:port/itg/web/knta/global/Home.jsp
```

In this example, `Home.jsp` will be located on the file system under:

```
ITG_Home/server/kintana/deploy/itg.war/web/knta/global
```

# Chapter 3

## Altering the HTML Layout

The HTML layout can be configured on a number of levels: the presentation layer can be changed on a global level, a particular screen can be altered, or a new screen with new functionality can be added.

This chapter discusses ways to customize the HTML layout and covers the following topics:

- *Replacing the Logo*
- *Changing the Presentation Layer*
- *Methods of Customization*
- *Guidelines for Modifying JSP Pages*
- *Changing the Appearance of a Particular Screen*

### Replacing the Logo

To replace the Mercury logo in the header, replace the `NavLogo.gif` and `NavLogoClosed.gif` image files. `NavLogo.gif` corresponds to the image that is displayed when the menu bar is displayed. `NavLogoClosed.gif` corresponds to the image when the menu bar is hidden.

- `global/images/NavLogo.gif`

This image has a width of 165 pixels and a height of 40 pixels.

- `global/images/NavLogoClosed.gif`

This image has a width of 20 pixels and a height of 40 pixels.

These images are shown in *Figure 3-1*. It is recommended that the same dimensions as the Mercury logo be used.

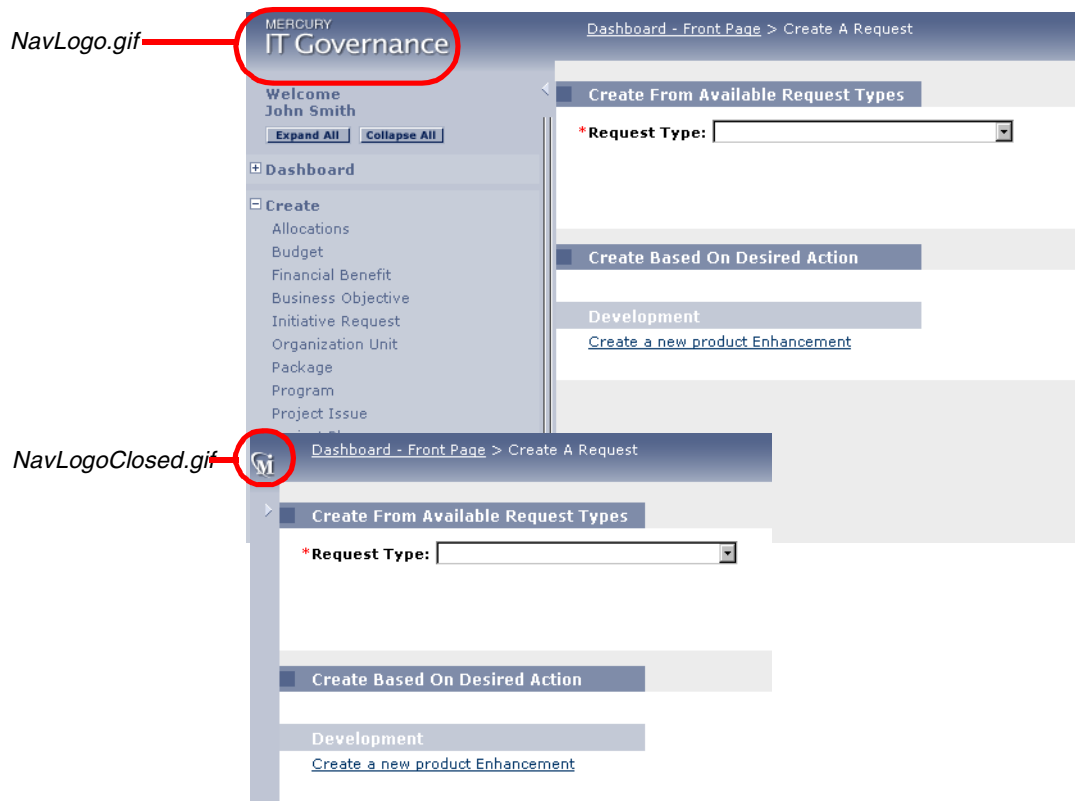


Figure 3-1 Logo images

## Changing the Presentation Layer

The presentation layer governs the general look and feel for all pages by providing common attributes and a way to include these elements for different layout styles.

Most of the appearance and layout of the presentation layer is determined by style sheets and common header and footer JSP files. By modifying these files, it is easy to brand the interface to match corporate Web standards. Custom branding includes such changes as logos, images, header, titles, and footers. The most widely used files include:

- `ITG_Home/server/kintana/deploy/itg.war/web/knta/global/css/GlobalStyleSheet.css`

The global style sheet used for the standard interface. The same style sheet can be used for both Internet Explorer and Netscape Navigator.

- `ITG_Home/server/kintana/deploy/itg.war/web/knta/global/Header.jsp`  
Determines the look of the navigation menu and header for most pages.
- `ITG_Home/server/kintana/deploy/itg.war/web/knta/global/Footer.jsp`  
Determines the look of the footer for most pages.
- `ITG_Home/server/kintana/deploy/itg.war/web/knta/global/SimpleHeader.jsp`  
A simple version of the normal header JSP used for certain pages that do not need a navigation menu.
- `ITG_Home/server/kintana/deploy/itg.war/web/knta/global/SimpleFooter.jsp`  
A simple version of the normal footer JSP used for certain pages.

This section describes the procedures for changing the presentation layer by modifying the following style sheets:

- [Global Style Sheets](#)
- [Common Header File \(Header.jsp\)](#)
- [JSP Definitions](#)

## Global Style Sheets

Global style sheets are standard HTML cascading style sheets. HTML documents returned by the Mercury ITG Server will associate CSS class names with various HTML elements, such as table data cells (HTML `<TD>` tags). The global style sheet can be used to supplement or override the properties of these HTML elements.

Style classes are currently used to centralize the definition of fonts, point sizes, background and foreground colors. Users who have experience editing and debugging cascading style sheets may modify the style class definitions contained within this file.

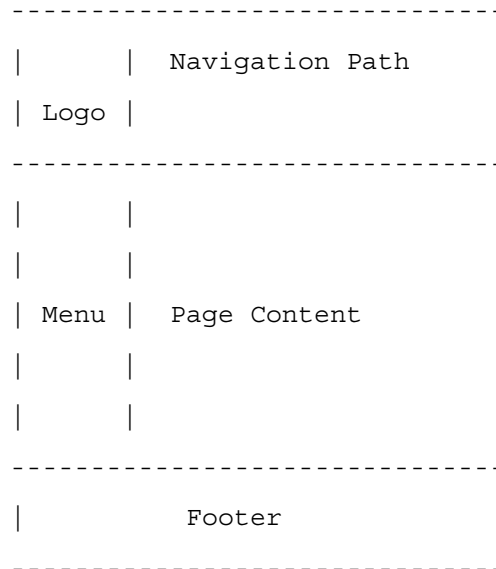
All styles can be viewed by opening the following files in the appropriate browser:

```
ITG_Home/server/kintana/deploy/itg.war/web/knta/global/html/GlobalStyleSheet.html
```

## Common Header File (Header.jsp)

The common header file, named `Header.jsp`, affects the standard interface layout. `Header.jsp` divides the page into three main sections:

- Navigation Path
- Side-Bar Menu
- Page Content.



Every time `Header.jsp` gets called, these three components are generated. By default, any pages that include the `Header.jsp` will get the Navigation Path and Menu. However, it is possible to control whether the Menu or Navigation Path should be displayed by setting the following values:

- To not display the menu on the page, set `displayMenuFlag` to **False**.
- To open a page in a Windoid, set the request parameter “`IS_WINDOID`” to **Y** as shown in the following example:

```
<jsp:include page = "../global/Header.jsp">  
<param name = "IS_WINDOID" value ="Y"/>  
</jsp:include>
```

It is possible to select whether a page will be part of the Navigation Path by passing some additional parameters to the `Header.jsp`.



Every page that contains `Header.jsp` must also include `Footer.jsp`.

## JSP Definitions

It is possible to modify the presentation layer by changing your JSPs. JSPs are dynamic HTML documents which are interpreted by the Mercury ITG JSP engine prior to being presented to a Web browser. It is important to note the following points:

- JSPs contain tag-like entries like:

```
<knta:I18NGetBundle bundleName="KEXP_Resources" />
```

These tags are not standard HTML tags. The tags act as placeholders where the JSP Engine should insert information known at the time of the user's request on the server side.

- JSPs often contain Java code that sends and retrieves data from the Mercury ITG Server, or pulls information stored in the user's session.
- JSPs may also contain client-side Java Script. JSPs often use dynamically provided information in client-side Java Script (`<SCRIPT language='Javascript'>` tags).
- JSP files should be edited in a plain text editor or a Web authoring tool that can understand JSP and Java Script tags.

## Methods of Customization

This section describes the different methods used to customize JSPs. JSP customizations can be made using any of the following methods:

- [\*Method 1 - Creating a New Presentation Layer\*](#)
- [\*Method 2 - Directly Changing the knta Home Directory Files\*](#)
- [\*Method 3 - Creating New JSP and Related Files\*](#)

## Method 1 - Creating a New Presentation Layer

This method is most appropriate when a significant amount of customization is required, or when various groups using the same instance require separate presentations and customizations.

### To create a new presentation layer:

1. Copy the `knta` directory to a new directory under the `ITG_Home/server/kintana/deploy/itg.war/web` directory.

Recall that `knta` is the default presentation key. The directory name is used in the URL to access the JSP files. The directory name should be short and not contain any special characters.

2. Refer your users to the URL which uses the new presentation key.

The URL to access a Request Search JSP with presentation key `custom` would be:

```
http://host.machine.com:port/itg/web/custom/crt/  
RequestSearch.jsp
```

Placing the files on the file system makes them accessible. You will need to restart the server for JSP changes to take effect. Alternatively, you can set the `JSP_RECOMPILE_ENABLED` parameter in the `server.conf` file to recompile JSP changes. No other system registration or configuration is required. As noted earlier, all JSP file references are relative to the presentation key.

During upgrade to future versions of Mercury ITG Center, custom directories under `web` will be preserved. However, depending on the changes in the release, the old JSP pages under the `custom` directory will have to be re-implemented to gain new and maintain existing functionality.



Note

URLs used in Notifications will, by default, reference the `knta` directory. That can be changed by manually adding URLs using a combination of system Tokens and hard-coding the new presentation key.

## Method 2 - Directly Changing the `knta` Home Directory Files

It is possible to make changes to files directly in the `knta` home directory. This technique is appropriate for making small changes from the default



implementation. If this technique is used, it is recommended that the standard development processes for documentation and source control be applied to the processes for changing any JSP and related files. There are many JSP files on the file system and it will be harder to keep track of what was changed.

Note

During upgrade to future versions of Mercury ITG Center, the entire `knta` directory structure will be deleted and refreshed with the latest JSP files. Any changes made must be reapplied to these files.

## Method 3 - Creating New JSP and Related Files

Create the new JSP and related files within the `knta` home directory (rather than modifying existing ones). This requires more work to make sure all JSP files reference the correct pages, but it is straight-forward. When using this technique, it is recommended that a naming convention be used to easily identify custom files. The creation of sub-directories is a safe method with the slight overhead of updating relative references in the JSP files.

Example

A JSP file may contain several lines similar to the following:

```
<jsp:include page="../../global/Header.jsp" />
```

If using a sub-directory in the original location of the file, the line would need to change to:

```
<jsp:include page="../../../global/Header.jsp" />
```

Note

During upgrade to future versions of Mercury ITG Center, the entire `knta` directory structure will be deleted and refreshed with the latest JSP files.

## Guidelines for Modifying JSP Pages

The best practice recommendations for editing the global JSP files include:

- Never edit the live JSP files directly on the production system. Changes to these files are immediately reflected in the system. Any error in modifying this file can prevent access to the system for all users.
- Be methodical when making changes to the HTML file. Save changes frequently. Check them by reloading the relevant screen in the browser. It may be necessary to reconfigure the browser to prevent page caching.
- Test the JSP changes in each browser that is used among the user community. Mercury ITG Center attempts to use HTML tags and techniques that are portable and perform acceptably with most current browsers, but it is highly advisable to reconfirm the portability of the changes carefully.
- Java code changes should be avoided unless there is an explicit open API to use. All other Java changes are not supported.
- As mentioned previously, it is recommended that the standard development processes for documentation and source control be applied to the processes for changing any JSP and related files.

## Changing the Appearance of a Particular Screen

Most JSP screens derive their appearance and layout from one or more JSP pages. By customizing these JSPs, a screen's appearance could be changed or extended.



Example

The Request search screen could be customized to include instructions on how to effectively search for Requests in a specific implementation. The Request creation screen could also be customized to offer a cross-link to the Help desk's support hours and service level standards for response times.

JSP files are named descriptively and should be easy to identify. Upon examining the major JSP files, examine any referenced files to understand what other pages may need modification.

### Example: Modifications to the New Request Screen

The New Request page can be altered by editing the respective JSP: `web/knta/crt/RequestCreate.jsp` file. This page can be altered using the following methods:

- [Reordering Sections of the Create New Request Page](#)
- [Adding HTML Layout Customizations](#)



The following sections should be viewed as customization examples. The techniques used below translate to other screens within Mercury ITG Center.

### *Reordering Sections of the Create New Request Page*

The Request JSP is represented by a large section of HTML containing JSP elements. The flexibility exists to rearrange sections of the Create New Request page. For instance, to move more critical information to the top of the page, rearrange the associated HTML elements.

### *Adding HTML Layout Customizations*

The flexibility exists to modify the appearance of the Create New Request page by adding customized HTML to the `RequestCreate.jsp` file. This allows the look and feel of the standard interface to be customized to match corporate standards or clarify on-screen instructions. This should be used in conjunction with global header and footer files that include global style sheets.



## Chapter

## 4

# Customizing the Menu

The standard interface menus are delivered as xml files on the Mercury ITG Center file system. Menus can be customized by creating new files or altering existing xml files.

This chapter defines the main elements of the menu files and provides instructions for modifying them. This chapter covers the following topics:

- [\*Menu Structure Overview\*](#)
- [\*Creating a New Menu\*](#)
- [\*Modifying the Menus\*](#)
- [\*Controlling Menu Access\*](#)
- [\*Notes on Menu xml files\*](#)
- [\*Configuring the URLs in the Menus\*](#)

## Menu Structure Overview

The menu in the standard interface can be customized. The menu structure is defined using a collection of xml files. One file defines the core menus, such as Dashboard, Demand Management, Search, Reports, and Administration. Other menu groups on the menu bar are defined in additional xml files. For example, Resource, Cost, or other special menus delivered with Mercury ITG products such as PMO.

All xml files used to define the menu structure are located in the following directory:

*ITG\_Home/conf/menus/*

A sample menu is shown in *Figure 4-1* with the **Demand** and **Team Manager** menus expanded. The elements of the menu are highlighted in the figure.

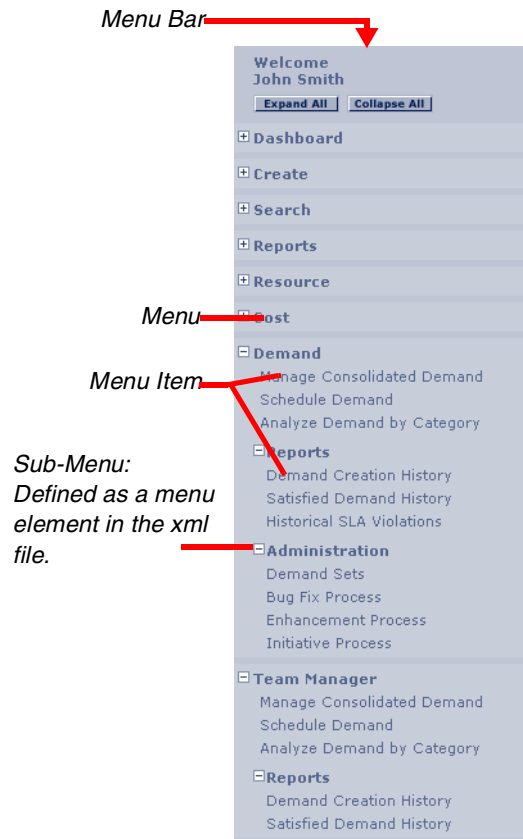


Figure 4-1 Menu Bar (Navigation bar)

*Figure 4-2* shows the menu bar element within the xml menu file.

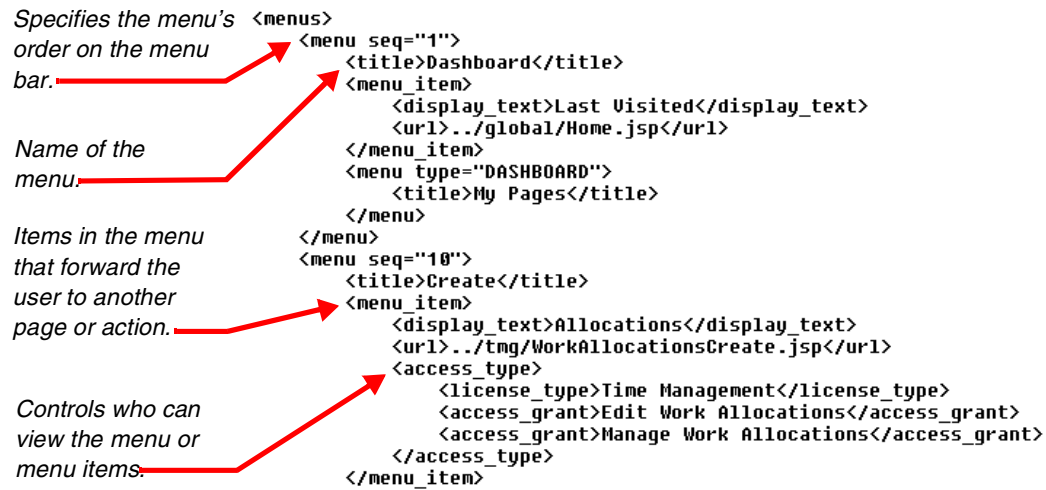


Figure 4-2 Section of the core\_menu.xml file

## <menu seq="#">

<menu seq="#"> specifies the order that the menu will appear on the menu bar. The core\_menu.xml file leaves sequential room for menu insertions between the standard menu items. For example, the menu sequence number for the **Dashboard** menu is 1, and the next menu sequence in the file (**Create**) has a sequence of 10. If a custom menu is created with a menu sequence of 5, it will appear between the **Dashboard** and **Create** menus.

## <menu> </menu>

<menu> </menu> encompasses the definition of the menu. This includes the menu\_title and the menu\_item definitions. The <menu> tag can be nested to create sub-menus. For example, in the structure shown in [Figure 4-3](#), the **Demand** menu contains an **Administration** sub-menu.

```
<menus seq="100">
  <title>Demand</title>
  <access_type>
    <license_type>Demand Management</license_type>
    <access_grant>Manage Demands</access_grant>
  </access_type>
  <menu_item>
    <display_text>Manage Consolidated Demand</display_text>
    <url>../dem/ConsolidatedPictureOfDemand.jsp</url>
  </menu_item>
  <menu_item>
    <display_text>Schedule Demand</display_text>
    <url>../dem/ScheduleDemand.jsp</url>
  </menu_item>
  <menu_item>
    <display_text>Analyze Demand by Category</display_text>
    <url>../dem/DemandByCategory.jsp</url>
  </menu_item>
  <menu>
    <access_type>
      <license_type>Kintana Create</license_type>
      <access_grant>Manage Requests</access_grant>
      <access_grant>Manage Packages</access_grant>
    </access_type>
    <menu_title>Administration</menu_title>
    <menu_item>
      <display_text>Bug Fix Process</display_text>
      <url>javascript:openWorkbench('&screen=WF&pkname=DEM+-+Bug+Request+Workflow');</url>
    </menu_item>
    <menu_item>
      <display_text>Enhancement Process</display_text>
      <url>javascript:openWorkbench('&screen=WF&pkname=DEM+-+Enhancement+Request+Process');</url>
    </menu_item>
    <menu_item>
      <display_text>Initiative Process</display_text>
      <url>javascript:openWorkbench('&screen=WF&pkname=DEM+-+Project+Initiative+Process');</url>
    </menu_item>
  </menu>
</menus>
```

Figure 4-3 Sample nested menu

`<menu_item>` `</menu_item>`

`<menu_item>` `</menu_item>` defines a single item in a menu. This includes the visible text as well as the URL to the linked page or action.

- `<display_text>`: text that appears in the expanded menu
- `<url>`: includes a link to the action or page that the menu points to. These URLs can utilize special Request and Report Dispatchers, or Smart URL functionality. For more information on tools for configuring the URLs, see [“Configuring the URLs in the Menus”](#) on page 24.

## Creating a New Menu

To create a new menu to appear on the navigation bar, create an xml file that uses the structure outlined in [“Menu Structure Overview”](#) on page 17. Consider copying and editing an existing menu to ensure that the correct format is being used.

**To create a new menu:**



1. Create an xml file that adheres to the structure outlined in “[Menu Structure Overview](#)” on page 17.
2. Place the file in the following directory.

```
ITG_Home/conf/menus/
```

3. Stop and re-start the Mercury ITG Server for the changes to take effect.

Ask the application administrator for assistance with this activity.

Menus located in the `menus` directory will be recognized by Mercury ITG Center and inserted into the menu bar according to its sequence. Users with appropriate access (defined in the menu's `<access_type>` tags) will be able to view the menu and menu items.



Tip

To avoid a navigational dead end, menu items that link directly to Reports should open the report in another window using the `openWindooid` call. For example, to open the DEM - Demand Creation History Report in a windoid, use the `openWindooid` call as shown:

```
<url>javascript:openWindooid(' ../global/ReportCreateDispatcher  
.jsp?REPORT_TYPE_NAME=DEM+-  
+Demand+Creation+History+Report ' )</url>
```

It is also recommended that menu items that link to outside systems be opened in a new window.



Note

Custom menus will not be deleted during a product upgrade, but Mercury Interactive does not guarantee that the custom menus will be compatible with future versions of Mercury ITG Center applications.

## Modifying the Menus

It is possible to customize menus that were delivered with Mercury ITG Center, as well as any custom menus that were developed at your site. This can include such activities as:

- Adding sub-menus to the menu
- Changing which menu items are included on the menu

- Changing which users can access the menu or menu items

Adhering to the structure outlined in “*Menu Structure Overview*” on page 17 will ensure a successful menu update.

For changes to take effect, stop and re-start the Mercury ITG Server. Ask the application administrator for assistance with this activity.



Note

Modifications to an existing (system) menu may be overwritten during an upgrade. Consider copying and renaming the menu’s file name before making modifications to an existing template.

Custom menus created at your site will not be deleted during a product upgrade, but Mercury Interactive does not guarantee that the custom menus will be compatible with future versions of Mercury ITG Center applications.

## Controlling Menu Access

To control which users can view specific menus or menu items, modify the `<access_type>` tag in the xml file. For example, select the License and Access Grants required to view a menu by using the `<license_type>` and `<access_grant>` tags within the `<access_type>` section of the template. The current list of supported access types include:

- `<license_type> License Type Name </license_type>`
- `<access_grant> Access Grant Name </access_grant>`
- `<security_group> Security Group Name </security_group>`
- `<can_create_request_type> Request Type Name </can_create_request_type>`

To control which users can view menus, sub-menus, and menu items, include the `<access_type> </access_type>` tag within the menu or menu item that will be restricted.

The location of the `<access_type>` tag determines which element is controlled:

- Menu — If the `<access_type>` section is placed anywhere directly within the `<menu> </menu>` section (such as not in the `<menu_item> </menu_item>` section), it will restrict access to that menu.

- **Menu item** — If the `<access_type>` section is placed anywhere within a `<menu_item>` `</menu_item>` section, it will restrict access to that menu item. In the following example, only users with a Demand Management Standard Licence, and the Manage Request or the Manage Package Access Grants, would see this menu item.

```

<menu seq="100">
  <title>Demand</title>
  <access_type>
    <license_type>Demand Management</license_type>
    <access_grant>Manage Demands</access_grant>
  </access_type>
  <menu_item>
    <display_text>Manage Consolidated Demand</display_text>
    <url>../dem/ConsolidatedPictureOfDemand.jsp</url>
  </menu_item>
  <menu_item>
    <display_text>Schedule Demand</display_text>
    <url>../dem/ScheduleDemand.jsp</url>
    <access_type>
      <license_type>Kintana Create</license_type>
      <access_grant>Manage Requests</access_grant>
      <access_grant>Manage Packages</access_grant>
    </access_type>
  </menu_item>
  <menu_item>
    <display_text>Analyze Demand by Category</display_text>
    <url>../dem/DemandByCategory.jsp</url>
  </menu_item>
  <menu>
    <menu_title>Reports</menu_title>
    <menu_item>
      <display_text>Demand Creation History</display_text>

```

- **Sub-menu** — If the `<access_type>` section is placed within a `<menu>` `</menu>` section that defines a sub-menu, it will restrict access to that sub-menu. In the following example, only users with a Demand Management Standard Licence, and the Manage Request or the Manage Package Access Grants, would see the Administration sub-menu.

```

</url>
</menu_item>
</menu>
<menu>
  <access_type>
    <license_type>Kintana Create</license_type>
    <access_grant>Manage Requests</access_grant>
    <access_grant>Manage Packages</access_grant>
  </access_type>
  <menu_title>Administration</menu_title>
  <menu_item>
    <display_text>Bug Fix Process</display_text>
    <url>javascript:openWorkbench(' &screen=WF&pkname=DEH--+Bug+Request+Workflow');</url>
  </menu_item>
  <menu_item>
    <display_text>Enhancement Process</display_text>
    <url>javascript:openWorkbench(' &screen=WF&pkname=DEH--+Enhancement+Request+Process');</url>
  </menu_item>
  <menu_item>
    <display_text>Initiative Process</display_text>
    <url>javascript:openWorkbench(' &screen=WF&pkname=DEH--+Project+Initiative+Process');</url>
  </menu_item>
</menu>
</menus>

```

*Table 4-1* lists the license types that can be used to control access within the `<license_type>` tags.

Table 4-1. License types.

License Type	Corresponding License
Demand Management Workbench	Demand Management Power License
Demand Management	Demand Management Standard License
Project Management Workbench	Project Management Power License
Project Management	Project Management Standard License
Change Management Workbench	Change Management Power License
Change Management	Change Management Standard License
Portfolio Management	Portfolio Management Standard License
Program Management	Program Management Standard License
Time Management	Time Management Power License
Administrator	Administrator License

## Notes on Menu xml files

This section contains additional details for modifying the menu:

- The URLs can be expressed as absolute paths or relative to the `<global>` directory.
- Spaces cannot be used in xml files. Use the plus sign (+) as a substitute for spaces. For example:

`DEM++Application+Enhancement = DEM - Application Enhancement.`

## Configuring the URLs in the Menus

This section describes the following tools that can be used when configuring the URLs within the `<menu_item>` tags:

- [\*Request Create Dispatcher\*](#)
- [\*Report Create Dispatcher\*](#)

- *Smart URL Formats*

## Request Create Dispatcher

Use the following URL format to access the Request Creation HTML page:

```
<url>../crt/RequestCreateDispatcher.jsp?REQUEST_TYPE_NAME=DEM+-  
+Operations+Request</url>
```

This example will open the Request Creation page for the DEM - Operations Request.

## Report Create Dispatcher

Use the following URL format to access the Report Submission html page:

```
<url>ReportCreateDispatcher.jsp?REPORT_TYPE_NAME=DEM+-  
+Demand+Creation+History+Report</url>
```

This example will open the Report Submission page for the DEM - Demand Creation History Report.

## Smart URL Formats

Use the following URL format to access a configuration screen:

```
<url>javascript:openWorkbench (' &screen=WF&pkname=DEM+-  
+Enhancement+Request+Process' );</url>
```

This example will open the DEM - Enhancement Request Process Workflow in the Workbench.



**Chapter**  
**5****Customizing Dashboard Templates**

Users can import templates into their Dashboard. These templates are added as a new page (tab) on the Dashboard. Each template can consist of one or more Dashboard pages. Once imported, the user can personalize the pages to suit their specific business needs (such as adding, deleting or personalizing Portlets).

It is possible to create and edit these templates. The template's structure and data are defined in an xml file located in the following directory:

*ITG\_Home/conf/tab\_templates/*

Each template is defined in its own xml file.

This chapter discusses the template structure and configuration notes and covers the following topics:

- [\*Template Structure Overview\*](#)
- [\*Creating a New Template\*](#)
- [\*Modifying Templates\*](#)
- [\*Enabling User Access to Templates\*](#)

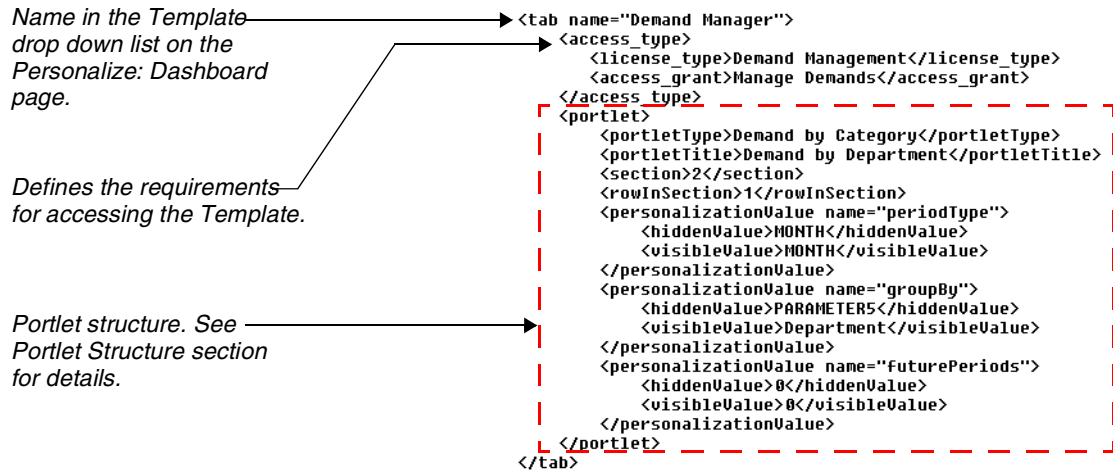


Note

For illustrative purposes, portions of the Demand Management product template (`Demand_Manager.xml`) is used as an example in this chapter.

## Template Structure Overview

The following figure highlights key points in the template structure. More details are discussed in *Portlet Structure in Template*.



Note

It is possible to include multiple Dashboard pages (tabs) in each template. Simply include the additional template definitions within a new set of `<tab name> </tab>` tags.

## Portlet Structure in Template

To specify multiple Portlets to include in the template, each Portlet definition must be contained within the `<Portlet> </Portlet>` tags. This section also contains descriptions of the following parameters of the Portlet structure and definition:

- `<PortletType>`
- `<PortletTitle>`
- `<section>` and `<rowInSection>`
- `<maxRows>`
- `<personalizationValue>`



```
<portlet>
  <portletType>Demand by Category</portletType>
  <portletTitle>Demand by Business Initiative</portletTitle>
  <section>3</section>
  <rowInSection>1</rowInSection>
  <maxRows>10</maxRows>
  <personalizationValue name="periodType">
    <hiddenValue>MONTH</hiddenValue>
    <visibleValue>MONTH</visibleValue>
  </personalizationValue>
  <personalizationValue name="groupBy">
    <hiddenValue>BUSINESS_INITIATIVE</hiddenValue>
    <visibleValue>Business Initiative</visibleValue>
  </personalizationValue>
  <personalizationValue name="futurePeriods">
    <hiddenValue>0</hiddenValue>
    <visibleValue>0</visibleValue>
  </personalizationValue>
</portlet>
```

### *<PortletType>*

Specifies the name of the Portlet. This can be any available system or custom Portlet.

### *<PortletTitle>*

Specifies the default title of the Portlet. The title can be later personalized by the end user.

### *<section> and <rowInSection>*

Specifies the location of the Portlet on the Dashboard page. The `<section>` tag refers to one of the four main areas on a Dashboard page. The `<rowInSection>` page refers to the Portlets placement within those sections.

*Figure 5-1* illustrates these two terms.

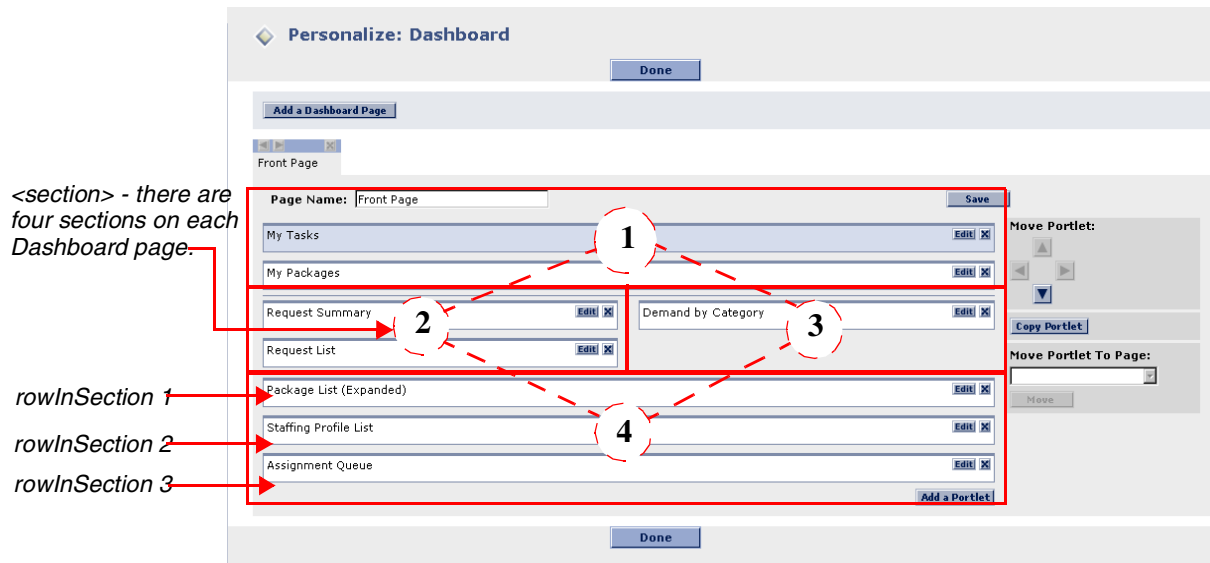


Figure 5-1 *<section>* and *<rowInSection>*

### *<maxRows>*

Specifies the default number of maximum rows displayed in a Portlet. This setting is ignored for Portlets featuring graphs.

### *<personalizationValue>*

Each Portlet type potentially has a unique set of personalization options. Additionally, some personalization values may be required for specific Portlet types.

#### **To retrieve personalization values:**

1. Configure a Portlet and figure out the Portlet\_id (hover over the edit Portlet button, the id is in the URL).
2. Run this SQL:

```
SELECT pv.PARAMETER_NAME,
       pv.PARAMETER_VISIBLE_VALUE,
       pv.PARAMETER_HIDDEN_VALUE
FROM KNTA_PERSONALIZATION_VALUES pv,
     knta_Portlets p
where 1=1
      and p.personalization_value_set_id =
         pv.personalization_value_set_id
      and p.Portlet_id = <Portlet id>
```

This will give you the personalization values needed to construct Portlet xml for the template.



Tip

The following tokenized values can be used as personalization values:

- [USER\_ID]
- [USERNAME]
- [FIRST\_NAME]
- [LAST\_NAME]



Note

Exception for non-Demand Management Portlets.

If the personalization value is for a drop-down list, specify the personalization value by only including the `<visibleValue>` element. Within the `<visibleValue>` element, enter the value that is typically used in the `<hiddenValue>` element.

For example,

```
<personalizationValue name="groupBy">
  <visibleValue>DEPARTMENT</visibleValue>
</personalizationValue>
```

## Creating a New Template

It is possible to create a new Dashboard template by using the template structure outlined in *“Template Structure Overview”* on page 28. Consider copying and editing an existing template to ensure that the correct format is being used.

**To create a new template:**

1. Create an xml file that adheres to the structure outlined in *“Template Structure Overview”* on page 28.
2. Place the file in the following directory.

```
ITG_Home/conf/tab_templates/
```

As soon as the template is placed in the `tab_templates` directory, it will be instantly recognized by Mercury ITG Center. Users with appropriate access (defined in the template's `<access_type>` tab) will be able to import the template onto their Dashboard.



Note

Custom templates will not be deleted during a product upgrade, but Mercury Interactive does not guarantee that the custom templates will be compatible with future versions of Mercury ITG Center applications.

## Modifying Templates

It is possible to customize templates that were delivered with Mercury ITG Center, as well as any custom templates that were developed at your site. This can include such activities as changing which Portlets are included on the template, or changing which users can access the Portlet.

To ensure a successful template update, follow the structure outlined in [“Template Structure Overview”](#) on page 28.



Note

Modifications to an existing system template may be overwritten during an upgrade. Consider copying and renaming the template name before making modifications to an existing template.

Custom templates will not be deleted during a product upgrade, but Mercury Interactive does not guarantee that the custom templates will be compatible with future versions of Mercury ITG Center applications.

## Enabling User Access to Templates

To control which users can add the template to their Dashboard, modify the `<access_type>` tag in the xml file. For example, select the License and Access Grants required to access a template by using the `<license_type>` and `<access_grant>` tags within the `<access_type>` section of the template. The current list of supported access types include:

- `<license_type>` License Type Name `</license_type>`
- `<access_grant>` Access Grant Name `</access_grant>`
- `<security_group>` Security Group Name `</security_group>`

In the example shown below, the tags have been added to the template:

```
<access_type>
<license_type>Demand Management</license_type>
<access_grant>Manage Demands</access_grant>
<access_grant>Manage Requests</access_grant>
<security_group>ITG User</security_group>
</access_type>
```

This code segment shows that only users who meet the following conditions may import this template onto their Dashboards:

- Demand Management license
- Membership in a Security Group
- Manage Demand Access Grant or Manage Requests Access Grant

All other users will be restricted from importing the template. Notice in the above example that the user is required to have one of each access types in order to access the template. Also notice that the user is only required to have one of the Access Grants to satisfy that access requirement. If more than one of a single access type are specified, the user needs only one to satisfy the requirement. Similarly, if two `<license_types>` were specified, the user would only need one of the licenses to access the template.

It is possible to extend this model and introduce multiple `<access_type>` tags within a single template. If the user meets the requirements of any one of the `access_types`, then access to the template will be granted.

Note

Portlet security (configured in the Portlet Workbench) is respected. If a user has access to a template—but is restricted from accessing a Portlet—the page would appear, but would not display the restricted Portlets.

*Table 5-1* lists the license types that can be used to control template access within the `<license_type>` tags.

*Table 5-1. License types.*

<b>License Type</b>	<b>Corresponding License</b>
Demand Management Workbench	Demand Management Power License
Demand Management	Demand Management Standard License
Project Management Workbench	Project Management Power License
Project Management	Project Management Standard License
Change Management Workbench	Change Management Power License
Change Management	Change Management Standard License
Portfolio Management	Portfolio Management Power License
Program Management	Program Management Standard License
Time Management	Time Management Standard License
Administrator	Administrator License

# Index

## A

access\_grant 22, 33  
access\_type 22, 32

## C

can\_create\_request\_type 22  
Changing a Particular Screen  
  14  
  adding layout customizations example 15  
  example 14  
  reordering sections example 15  
Common Header File 10  
Creating a New Presentation Layer 12  
Creating New JSP and Related Files 13  
Customization  
  changing particular screen 14  
  changing particular screen example 14  
  JSP modification guidelines 13  
  methods 11

## D

Dashboard Templates 27  
Directly Changing Home Directory Files 12

## G

Global Style Sheet 9

## H

Header.jsp 10  
HTML Layout  
  customizing 7

## J

JSP  
  definitions 11  
  modification 13  
  guidelines

## L

License Types 22, 24, 33  
Logo  
  replacing 7

## M

maxRows 30  
menu seq 19  
menu\_item 20  
Menus  
  configuring urls 24  
  controlling user access 22  
  creating new 20

  customizing 17  
  element in xml file 19  
  modifying 21  
  notes on xml file 24  
  overview 17

Methods of Customization 11  
  creating new JSP and related files 13  
  directly changing home dir files 12  
  new presentation layer 12

## P

personalizationValue 30  
portletTitle 29  
portletType 29  
Presentation Layer 8

## R

Report Dispatcher 25  
Request Dispatcher 25  
rowInSection 29

## S

section 29  
security\_group 22, 33  
Smart URL Formats 25

## T

### Template

- controlling user access 32
- creating 31
- customizing 27
- modifying 32
- overview 28
- portlet structure and definition 28

## U

- URL 24