

KINTANA™

# Customizing the Kintana Interface

**Version 5.0.0**

Publication Number: CustomizeKintana-0603A

Kintana, Inc. and all its licensors retain all ownership rights to the software programs and related documentation offered by Kintana. Use of Kintana's software is governed by the license agreement accompanying such Kintana software. The Kintana software code is a confidential trade secret of Kintana and you may not attempt to decipher or decompile Kintana software or knowingly allow others to do so. Information necessary to achieve the interoperability of the Kintana software with other programs may be obtained from Kintana upon request. The Kintana software and its documentation may not be sublicensed and may not be transferred without the prior written consent of Kintana.

Your right to copy Kintana software and this documentation is limited by copyright law. Making unauthorized copies, adaptations, or compilation works (except for archival purposes or as an essential step in the utilization of the program in conjunction with certain equipment) is prohibited and constitutes a punishable violation of the law.

THIS DOCUMENTATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. IN NO EVENT SHALL KINTANA BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OR DATA, INTERRUPTION OF BUSINESS, OR FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, ARISING FROM ANY ERROR IN THIS DOCUMENTATION.

Kintana may revise this documentation from time to time without notice.

Copyright © 1997, 1998, 1999, 2000, 2001, 2002, 2003 Kintana, Incorporated. All rights reserved.

Kintana, Kintana Deliver, Kintana Create, Kintana Drive, Kintana Dashboard, Kintana Accelerator, Kintana Demand Management (DM), Kintana Portfolio Management (PFM), Kintana Program Management Office (PMO), Kintana Enterprise Change Management (ECM), Object\*Migrator, GL\*Migrator and the Kintana logo are trademarks of Kintana, Incorporated. All other products or brand names mentioned in this document are the property of their respective owners.

Kintana Version 5.0.0

© Kintana, Incorporated 1997 - 2003

All rights reserved.

Printed in USA

**Kintana, Inc.**

1314 Chesapeake Terrace, Sunnyvale, California 94089

Telephone: (408) 543-4400

Fax: (408) 752-8460

<http://www.kintana.com>

# Contents

<b>Chapter 1</b>	
<b>Introduction .....</b>	<b>3</b>
Who should read this guide.....	3
<b>Chapter 2</b>	
<b>Key Concepts.....</b>	<b>5</b>
Kintana HTML Customization Levels.....	5
Level 1 HTML Configuration.....	5
Level 2 HTML Configuration.....	6
Level 3 HTML Configuration.....	6
Web Architecture Overview .....	6
Directory Structure.....	7
<b>Chapter 3</b>	
<b>Altering Kintana’s HTML Layout.....</b>	<b>9</b>
Replacing the Kintana Logo .....	9
Changing the Presentation Layer.....	10
Global Style Sheets.....	11
Common Header File (Header.jsp).....	11
JSP Definitions .....	12
Methods of Customization .....	13
Method 1 - Creating a New Presentation Layer .....	13
Method 2 - Directly Changing the ‘knta’ Home Directory Files.....	14
Method 3 - Creating New JSP and Related Files .....	14
Guidelines for Modifying JSP Pages .....	15
Changing the Appearance of a Particular Screen.....	16
Example: Modifications to the New Request Screen.....	16
Reordering Sections of the Create New Request Page.....	16
Adding HTML Layout Customizations .....	17
<b>Chapter 4</b>	
<b>Customizing the Menu.....</b>	<b>19</b>
Menu Structure Overview .....	19

---

<menu seq="#">.....	21
<menu> </menu>.....	21
<menu_item> </menu_item> .....	22
<b>Creating a New Menu .....</b>	<b>23</b>
<b>Modifying the Menus.....</b>	<b>24</b>
<b>Controlling Menu Access.....</b>	<b>24</b>
<b>Notes on Menu xml files .....</b>	<b>27</b>
<b>Configuring the URLs in the Menu .....</b>	<b>27</b>
Request Create Dispatcher .....	27
Report Create Dispatcher .....	27
Smart URL Formats .....	28
<b>Chapter 5</b>	
<b>Customizing Dashboard Templates .....</b>	<b>29</b>
<b>Template Structure Overview .....</b>	<b>29</b>
Portlet Structure in Template .....	30
<portletType> .....	31
<portletTitle> .....	31
<section> and <rowInSection>.....	31
<maxRows> .....	31
<personalizationValue> .....	31
<b>Creating a New Template .....</b>	<b>33</b>
<b>Modifying Templates .....</b>	<b>33</b>
<b>Enabling User Access to Templates .....</b>	<b>34</b>

# Chapter 1 Introduction

The Kintana 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 Kintana interface.

The following topics are discussed in this document:

- *Key Concepts*
- *Altering Kintana's HTML Layout*
- *Customizing the Menu*
- *Customizing Dashboard Templates*
- *Customizing Page Functionality*

## Who should read this guide

This document provides details for customizing the Kintana interface. This business application guide is used primarily by:

- Advanced users or administrators responsible for the look and feel of the Kintana application
- Advanced users or administrators responsible for implementing corporate branding specifications
- Advanced users or administrators who wish to modify Kintana's interface and navigation to better fit with their processes



# Chapter 2 Key Concepts

The following key concepts and definitions are used when customizing the Kintana interface.

- [Kintana HTML Customization Levels](#)
- [Web Architecture Overview](#)

## Kintana HTML Customization Levels

Because all customizations do not have the same level of complexity or upgrade exposure, Kintana recognizes several different levels of advanced customization techniques. These Kintana Customization Levels are based on Kintana's assessments of:

- How many users employ a configuration technique.
- The complexity of the configuration technique.
- The risk level involved when using the technique that any customizations performed will have to be re-implemented.

The Kintana HTML Configuration Levels 1, 2 and 3 are defined for reference in the following paragraphs.

### Level 1 HTML Configuration

- Kintana expects many customers to use this configuration technique.
- Level 1 configuration techniques are primarily documented in this guide. Additional documentation (including examples) may be available on-line.
- The upgrade path for this configuration technique is low-impact. Future versions of Kintana will attempt to maintain backward compatibility with

this configuration technique, or provide automated upgrade paths. If this is not possible, manual re-implementation can be performed as easily as typical Kintana configuration changes.

## Level 2 HTML Configuration

- Kintana expects some customers to use this configuration technique. They will often do so with the assistance of Kintana consultants.
- Customers should check on-line for current documentation. Documentation and support may assume proficiency with relevant technology, such as HTML, JSP, and Java Script.
- Customers using this configuration technique may need to re-implement this configuration in future versions of Kintana. Kintana will attempt to provide backward compatibility or automated upgrade paths. Customers using these techniques are highly encouraged to inform Kintana support when doing so. This information will be taken into account when developing future versions of the product.

## Level 3 HTML Configuration

- Kintana expects that only customers with the most demanding requirements will use this configuration technique. They will usually do so with the assistance of Kintana consultants.
- Customers should check on-line or contact Kintana Support for current documentation.
- Customers using this configuration technique should be prepared to re-implement these configurations in future versions of Kintana. Customers using these techniques should inform Kintana support when doing so. This information will be taken into account when developing future versions of the product.

## Web Architecture Overview

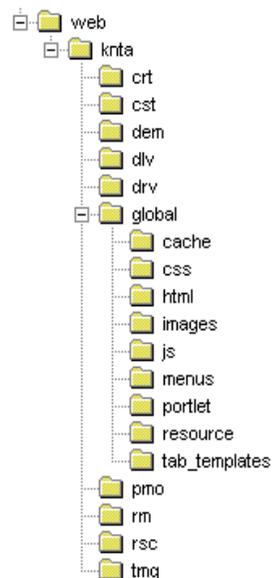
The Kintana Web Architecture is powered by a high-performance JSP (Java Server Page) engine. A JSP is simply a file containing familiar HTML elements that also has special tags and sometimes limited use of Java code. A great deal of information and training is available for the JSP user.



Throughout this document, a JSP is often referred to as a page.

## Directory Structure

All JSP and related files are located in the ‘web’ directory under KINTANA\_HOME on the Kintana server. The structure is shown in *Figure 2-1*:



*Figure 2-1 JSP Directory Structure*

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/kintana/web/knta/global/Home.jsp`

Home.jsp will be located on the file system under Kintana\_Home/web/knta/global.

The 'knta' directory represents the presentation key. Directories can be added under 'web' to contain copies of files in the 'knta' file system. This is possible because all URL links have been made relative from the presentation key directory. See "[Methods of Customization](#)" on page 13 for more information.

# Chapter 3

## Altering Kintana's HTML Layout

Kintana's 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. These customizations can vary from a *Level 1 HTML Configuration* to a *Level 2 HTML Configuration*. Advanced administrators can customize the HTML layout using any of the following procedures:

- *Replacing the Kintana Logo*
- *Changing the Presentation Layer*
- *Changing the Appearance of a Particular Screen*

### Replacing the Kintana Logo

To replace the Kintana logo in the header, you must replace two images. `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.

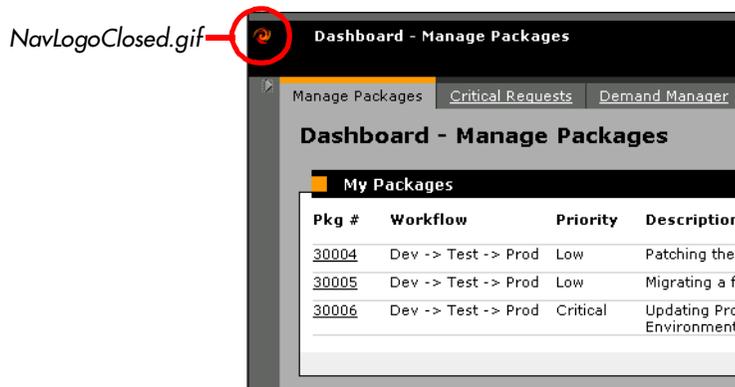
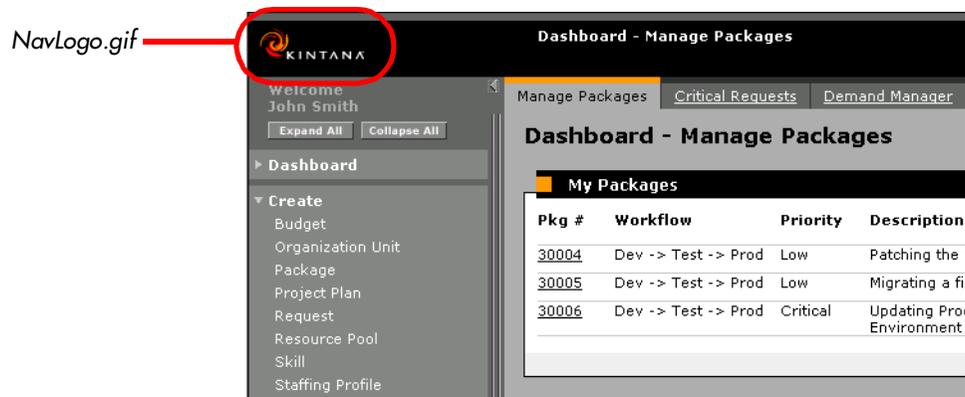
- `global\images\NavLogo.gif`

This image has a width of 90 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.

You should use the same dimensions as the Kintana logo.



## 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.



Changing the Presentation Layer, by modifying any of the global style sheets or common header and footer JSP files, is considered a *Level 1 HTML Configuration*.

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 are listed below:

- `KINTANA_HOME/web/knta/global/css/KintanaGlobalStyleSheet.css`  
The global style sheet used for the Kintana interface. The same style sheet can be used for both Internet Explorer and Netscape Navigator.
- `KINTANA_HOME/web/knta/global/Header.jsp`  
Determines the look of the navigation menu and header for most pages.
- `KINTANA_HOME/web/knta/global/Footer.jsp`  
Determines the look of the footer for most pages.
- `KINTANA_HOME/web/knta/global/SimpleHeader.jsp`  
A simple version of the normal header JSP used for certain pages that do not need a navigation menu.
- `KINTANA_HOME/web/knta/global/SimpleFooter.jsp`  
A simple version of the normal footer JSP used for certain pages.

## Global Style Sheets

Global style sheets are standard HTML cascading style sheets. HTML documents returned by the Kintana 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. Customers who have experience editing and debugging cascading style sheets should feel free to modify the style class definitions contained within this file.

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

- `KINTANA_HOME/web/knta/global/html/GlobalStyleSheet.html`

## Common Header File (Header.jsp)

The common header file affects the Kintana interface layout. Header.jsp divides the page into three main sections:

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

```
-----  
|           | Navigation Path   |  
| Logo     |                       |  
-----  
|           |                       |  
|           |                       |  
| Menu     | Page Content                 |  
|           |                       |  
|           |                       |  
-----  
|           | Footer                         |  
-----
```

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, there are some flags that a page can use to control whether the Menu or Navigation Path should be displayed.

To not display the menu on the page, set displayMenuFlag to false.

To open a page in a Window (request.getParameter("IS\_WINDOW")=="Y"), then both the Menu and Navigation Path will not appear.

Every page that includes the Header.jsp must also include Footer.jsp

A page can choose to whether or not to be part of the Navigation Path by passing some additional parameters to the Header.jsp.

## JSP Definitions

You can modify the presentation layer by changing your JSPs. JSPs are dynamic HTML documents which are interpreted by the Kintana JSP engine prior to being presented to an Internet 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 Kintana application server, or pulls information stored in the user's session.
- JSP 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

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 of Kintana require separate presentations and customizations.

To create a new presentation layer:

1. Copy the 'knta' directory to a new directory under the 'KINTANA\_HOME/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/kintana/web/custom/crt/RequestSearch.jsp>

Placing the files on the file system makes them accessible. 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 Kintana, 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

You can 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, Kintana recommends that you apply your standard development processes for documentation and source control to the processes of 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 Kintana, the entire ‘knta’ directory structure will be deleted and refreshed with the latest JSP files. You will need to reapply any changes you have made to these files.

## Method 3 - Creating New JSP and Related Files

Create 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, Kintana recommends you choose a naming convention that can 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 Kintana, the entire 'knta' directory structure will be deleted and refreshed with the latest JSP files.

## Guidelines for Modifying JSP Pages

The following are Kintana's best practice recommendations for editing the global JSP files:

- Never edit the live JSP files directly on your production system! Changes to these files are immediately reflected in Kintana. Any error in modifying this file can prevent access to Kintana for all users.
- Be methodical as you make changes to the HTML file. Save your changes frequently. Check them by reloading the relevant screen in your browser. You may need to reconfigure your browser to prevent page caching.
- Test your JSP changes in each browser that is used among your user community. Kintana has attempted 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 your 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 in the previous section, Kintana recommends that you apply your standard development processes for documentation and source control to the processes of 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. The technique of including common files or similar layout styles is used extensively. This is the reason multiple JSP files are used to implement a screen. By customizing these JSPs, a screen's appearance could be changed or extended.



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 the included files referenced within 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 in the following ways:

- *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 Kintana. Please search for additional documentation on Contori (<http://customer.contori.com>).

#### *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, you can move more critical information to the top of the page. To do this, you must 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 Kintana 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

Kintana's menus are delivered as xml files on the Kintana file system. Users can customize menus by creating new files or altering existing xml files. This chapter defines the main elements of the menu files and provides some instructions for modifying them.

The following topics are discussed:

- [\*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 Kintana HTML interface can be customized. The menu structure is defined using a collection of xml files. One file defines the “core” menus including Dashboard, Create, Search, Reports, Administration, etc. Other menu groups on the menu bar are defined in additional xml files. For example, Resource, Cost, or other special menus delivered with Kintana Solutions such as Demand or PMO.

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

web\knta\global\menus\

A sample menu is shown in [Figure 4-1](#) with the Demand and Team Manager expanded. The elements of the menu are highlighted in the figure.

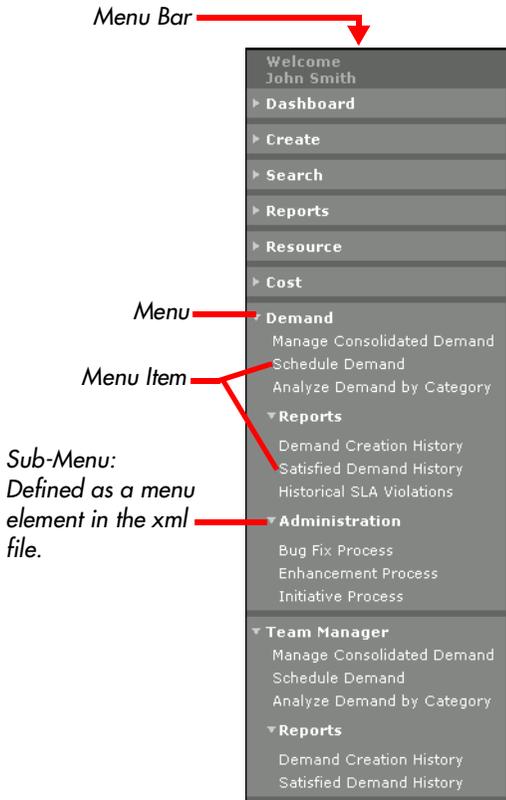


Figure 4-1 Kintana Menu Bar (Navigation bar)

[Figure 4-2](#) highlights the menu bar element within the xml menu file.

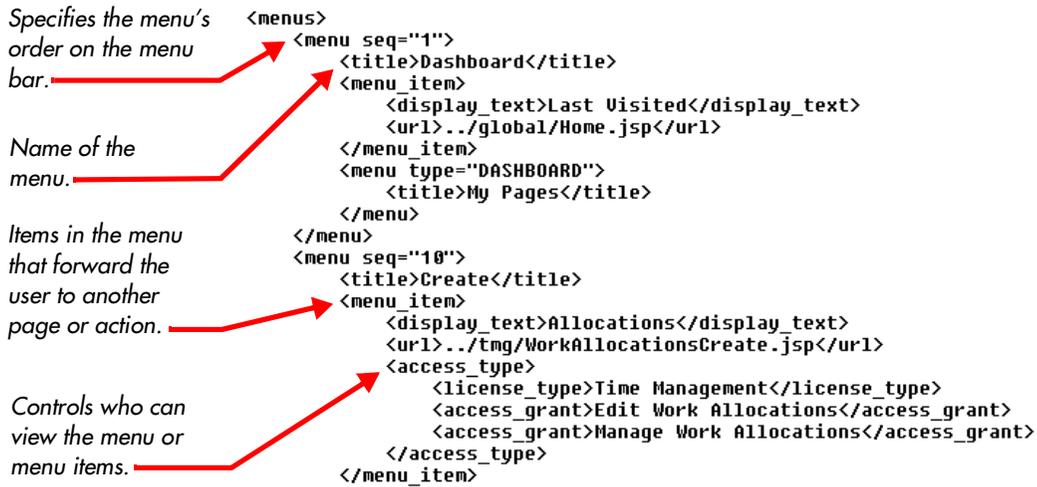


Figure 4-2 Section of the core\_menu.xml file

**<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 Create menu is 10, and the next menu sequence in the file (Search) has a sequence of 20. If you create a custom menu with a menu sequence of 15, it will appear between the Create and Search menus.

**<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 following structure, 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>`

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. See “[Configuring the URLs in the Menus](#)” on page 27 for more information on tools for configuring the urls.

## Creating a New Menu

You can create a new menu to appear on the Kintana navigation bar by creating an xml file that uses the structure outlined above. Consider copying and editing an existing menu to ensure that you are using the correct format.

To create a new menu:

1. Create an xml file that adheres to the structure outlined in “[Menu Structure Overview](#)” on page 19.
2. Place the file in the following directory.

```
web\knta\global\menus\
```

3. Stop and re-start the Kintana server for the changes to be recognized. Ask your Kintana system administrator for assistance with this activity.

Menus located in the `menus` directory will be recognized by Kintana 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

Menu items that link directly to Reports should open the report in another window using the `OPENWINDOID` call so the user is not presented with a navigational dead end. For example: to open the DEM - Demand Creation History Report in a “windoid:”

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

Kintana also recommends that you open menu items that link to outside systems in a new window.



### Note

Custom menus will not be deleted during a product upgrade, but Kintana does not guarantee that your custom menus will be compatible with future versions of Kintana software.

## Modifying the Menus

You can customize menus that were delivered with Kintana 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, or changing which users can access the menu or menu items.

Adhering to the structure outlined in “[Menu Structure Overview](#)” on page 19 will ensure a successful menu update.

You must stop and re-start the Kintana server for the changes to be recognized. Ask your Kintana system administrator for assistance with this activity.



Note

Modifications to an existing Kintana (system) menu may be overwritten during an upgrade. Consider copying and renaming the menu’s file name if you would like to make modifications to an existing template.

Custom menus created at your site will not be deleted during a product upgrade, but Kintana does not guarantee that your custom menus will be compatible with future versions of Kintana software.

## Controlling Menu Access

You can control which users can view specific menus or menu items by modifying the `<access_type>` tag in the xml file. For example, you can 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 are:

- `<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>`

You can control which users can view menus, sub-menus, and menu items. This is controlled by including the `<access_type>` `</access_type>` tag within the menu or menu item that you would like to restrict.

The location of the `<access_type>` tag determines which element is controlled: menu, sub-menu, or menu item:

- If the `<access_type>` section is placed anywhere directly within the `<menu> </menu>` section (i.e. not in the `<menu_item> </menu_item>` section), it will restrict access to that menu.
- 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 Kintana Create 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>
```

- 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 Kintana Create 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(' &amp;screen=WF&amp;pkname=DEM+-+Bug+Request+Workflow');</url>
  </menu_item>
  <menu_item>
    <display_text>Enhancement Process</display_text>
    <url>javascript:openWorkbench(' &amp;screen=WF&amp;pkname=DEM+-+Enhancement+Request+Process');</url>
  </menu_item>
  <menu_item>
    <display_text>Initiative Process</display_text>
    <url>javascript:openWorkbench(' &amp;screen=WF&amp;pkname=DEM+-+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 Kintana License
Kintana Create Workbench	Kintana Create Power License
Kintana Create	Kintana Create Standard License
Kintana Drive Workbench	Kintana Drive Power License
Kintana Drive	Kintana Drive Standard License
Kintana Deliver Workbench	Kintana Deliver Power License
Kintana Deliver	Kintana Deliver Standard License
Kintana Dashboard Workbench	Kintana Dashboard Power License
Kintana Dashboard	Kintana Dashboard Standard License
Demand Management	Demand Management Standard License
PMO	PMO Standard License
Time Management	Time Management Standard License

## Notes on Menu xml files

The following additional details may help you when modifying the menu.

- The URLs can be expressed as absolute paths or relative to the <global> directory.
- You can not use spaces 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

You can use the following tools when configuring the URLs within the <menu\_item> tags. The dispatchers all you to refer to Request Types by name for creation purposes.

### 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 Kintana Workbench.

# Chapter 5

## Customizing Dashboard Templates

Users can import templates into their Kintana Dashboard. These templates are added as a new page (tab) on your Dashboard. Each template can consist of one or more Dashboard pages. Once imported, the user can personalize the pages to suit his specific business needs; for example, add, delete or personalize portlets.

You can create and edit these templates. The template's structure and data are defined in an xml file located in the following directory:

```
web\knta\global\tab_templates\
```

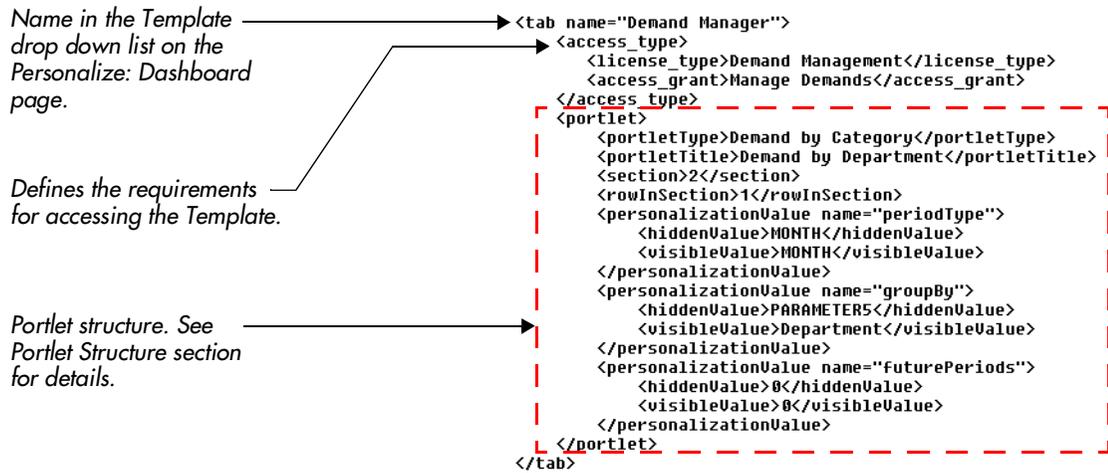
Each template is defined in its own xml file.

The following sections discuss the template structure and configuration notes. Pieces of the Demand Management Solution template (Demand\_Manager.xml) are used for illustrative purposes.

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

### Template Structure Overview

The following figure highlights key points in the template structure. More details are discussed following the graphic.



Note

You can 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

You can specify multiple portlets to include in your template. Each portlet definition must be contained within the `<portlet>` `</portlet>` tags. Other parameters of the portlet structure and definition are discussed below.

```

<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 Kintana portlet. This can be any available system or custom Kintana portlet.

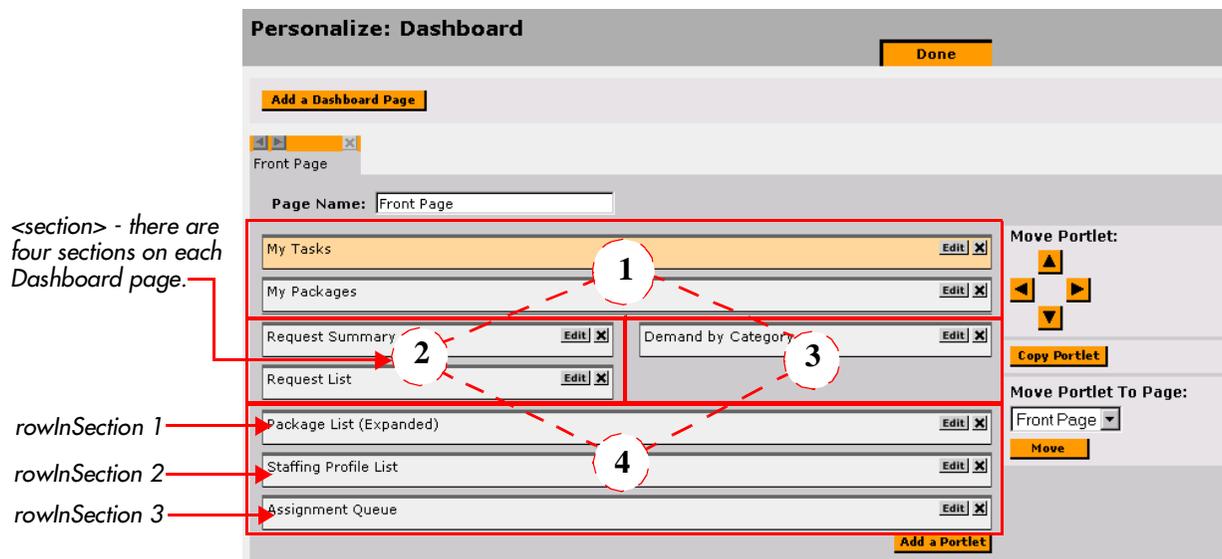
**<portletTitle>**

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 4 main areas on a Dashboard page. The <rowInSection> tag refers to the portlets placement within those sections.

The following figure illustrates these two terms.



**<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. The easiest way to ensure proper configuration is to use the following procedure.

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, you should enter the value that you would normally enter in the <hiddenValue> element.

For example,

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

## Creating a New Template

You can create a new Dashboard template by using the template structure outlined above. Consider copying and editing an existing template to ensure that you are using the correct format.

To create a new template:

1. Create an xml file that adheres to the structure outlined in *“Template Structure Overview”* on page 29.
2. Place the file in the following directory.

```
web\knta\global\tab_templates\
```

As soon as the template is placed in the tab\_templates directory, it will be instantly recognized by Kintana. 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 Kintana does not guarantee that your custom templates will be compatible with future versions of Kintana software.

## Modifying Templates

You can customize templates that were delivered with Kintana 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.

Adhering to the structure outlined in *“Template Structure Overview”* on page 29 will ensure a successful template update.



Note

Modifications to an existing Kintana (system) template may be overwritten during an upgrade. Consider copying and renaming the template name if you would like to make modifications to an existing template.

Custom templates will not be deleted during a product upgrade, but Kintana does not guarantee that your custom templates will be compatible with future versions of Kintana software.

## Enabling User Access to Templates

You can control which users can add the template to their Dashboard by modifying the `<access_type>` tag in the xml file. For example, you can 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 are:

- `<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>Kintana User</security_group>
</access_type>
```

This code segment shows that only users with:

- Demand Management license and
- Membership in the “Kintana User” Security Group and
- The Manage Demand Access Grant or
- Manage Requests Access Grants

can import this template onto their Dashboards. 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.

You can 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 he will be able to access the template.



Note

Portlet security (configured in the Portlet Workbench) is respected. If a user has access to a template, but is restricted from viewing 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.*

License Type	Corresponding Kintana License
Kintana Create Workbench	Kintana Create Power License
Kintana Create	Kintana Create Standard License
Kintana Drive Workbench	Kintana Drive Power License
Kintana Drive	Kintana Drive Standard License
Kintana Deliver Workbench	Kintana Deliver Power License
Kintana Deliver	Kintana Deliver Standard License
Kintana Dashboard Workbench	Kintana Dashboard Power License
Kintana Dashboard	Kintana Dashboard Standard License
Demand Management	Demand Management Standard License
PMO	PMO Standard License
Time Management	Time Management Standard License



# Index

## A

access\_grant 24, 34  
 access\_type 24, 34  
 Architecture  
   directory structure 7  
   web 6

## C

can\_create\_request\_type 24  
 Changing a Particular Screen  
   16  
   adding layout customizations example 17  
   example 16  
   reordering sections example 16  
 Common Header File 11  
 Creating a New Presentation Layer 13  
 Creating New JSP and Related Files 14  
 Customization Levels 5  
 Customizing Kintana 3

## D

Dashboard Templates 29  
 Directly Changing Home Directory Files 14

## G

Global Style Sheet 11

## H

Header.jsp 11  
 HTML Layout  
   customizing 9

## J

JSP Definitions 12  
 JSP Modification Guidelines 15

## K

Kintana Customization  
   changing particular screen 16  
   changing particular screen example 16  
   JSP modification guidelines 15  
   methods 13  
 Kintana Logo  
   replacing 9

## L

Level 1 Customization 5  
 Level 2 Kintana Customiza-

tion 6

Level 3 Kintana Customization 6  
 License Types 26  
 license\_type 24, 34

## M

maxRows 31  
 Menu  
   creating new 23  
   notes on xml file 27  
 menu  
   element in xml file 21  
 menu seq 21  
 menu\_item 22  
 Menus  
   configuring urls 27  
   controlling user access 24  
   customizing 19  
   modifying 24  
   overview 19  
 Methods of Customization 13  
   creating new JSP and related files 14  
   directly changing home dir files 14  
   new presentation layer 13

## P

personalizationValue 31  
 portletTitle 31

portletType 31  
Presentation Layer 10

## R

Report Dispatcher 27  
Request Dispatcher 27  
rowInSection 31

## S

section 31  
security\_group 24, 34  
Smart URL Formats 28

## T

Template  
    controlling user access 34  
    creating 33  
    customizing 29  
    modifying 33  
    overview 29  
    portlet structure and definition 30

## U

URL 27

## W

Web Architecture Overview 6  
    directory structure 7