

Peregrine

BI Portal 5.0

Administrator's Guide

For Windows, AIX, and Solaris

Copyright © 2003 Peregrine Systems, Inc. or its subsidiaries. All rights reserved.

Information contained in this document is proprietary to Peregrine Systems, Incorporated, and may be used or disclosed only with written permission from Peregrine Systems, Inc. This book, or any part thereof, may not be reproduced without the prior written permission of Peregrine Systems, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Peregrine Systems®, AssetCenter®, and ServiceCenter® are registered trademarks of Peregrine Systems, Inc. or its subsidiaries. BI Portal™ is a trademark of Peregrine Systems, Inc. or its subsidiaries.

Microsoft, Windows, Windows NT, Windows 2000, SQL Server, and names of other Microsoft products referenced herein are trademarks or registered trademarks of Microsoft Corporation. Oracle is a registered trademark of Oracle Corporation. DB2 is a registered trademark of International Business Machines Corp.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). This product also contains software developed by: Sun Microsystems, Inc., Netscape Communications Corporation, and InstallShield Software Corporation

This product includes software developed by Business Objects, S.A. Portions (c) Copyright 1995 - 2003, Business Objects, S.A. All rights reserved.

This document and the related software described in this manual are supplied under license or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Systems, Inc. Contact Peregrine Systems, Inc., Customer Support to verify the date of the latest version of this document. The names of companies and individuals used in the sample database and in examples in the manuals are fictitious and are intended to illustrate the use of the software. Any resemblance to actual companies or individuals, whether past or present, is purely coincidental. If you need technical support for this product, or would like to request documentation for a product for which you are licensed, contact Peregrine Systems, Inc. Customer Support by email at support@peregrine.com. If you have comments or suggestions about this documentation, contact Peregrine Systems, Inc. Technical Publications by email at doc_comments@peregrine.com. This edition of the document applies to version 5.0 of the licensed program.

Peregrine Systems, Inc.
Worldwide Corporate Headquarters
3611 Valley Centre Drive San Diego, CA 92130
Tel 800.638.5231 or 858.481.5000
Fax 858.481.1751
www.peregrine.com



Contents

	About this Guide	7
	Using this Guide	8
	Related Documentation	8
	Documentation Conventions	9
	Contacting Peregrine Systems	9
	Customer Support	9
	Documentation Web site	10
	Education Services Web Site	10
Chapter 1	Peregrine OAA Architecture Overview	11
	Peregrine OAA Platform architecture	13
	Archway internal architecture	16
	Archway requests	17
Chapter 2	Customizing the Peregrine Portal	21
	Deploying the Classic theme variations	22
	Changing the default theme	23
	Changing the header graphic for all themes.	23
	Creating a custom theme	25
	Layers properties.	29
	Changing framesets.	30
Chapter 3	Using the Peregrine Portal	33
	Logging in to the BI Portal.	34
	Using the Activity menu.	35

	Personalizing the BI Portal	36
	Adding components	36
	Changing the layout	38
	Changing themes	40
	Displaying form information	41
Chapter 4	Using the OAA Administration Module	43
	Accessing the Peregrine Portal Admin module	44
	Using the Control Panel	46
	Viewing the Deployed Versions	47
	Viewing the Server Log	48
	Using the Settings page	48
	Setting parameters using the Admin module	49
	Verifying Script Status	51
	Displaying Message Queues	51
	Showing Queue Status	52
	Importing and exporting personalizations	53
	Viewing adapter transactions	53
	Using the IBM Websphere Portal	54
	Displaying form information	54
	Displaying form details	56
	User self-registration	57
	Changing passwords	58
	Logging and monitoring user sessions	58
	Understanding the usage.log file	58
Chapter 5	Security	61
	BI Portal security	62
	Document groups	62
	User capabilities	65
	Password encoding methods	68
	User registration	69
	Enabling the E-mail adapter	69
	Troubleshooting the MailAdapter connection	70
	Authenticating users	70
	Default security configuration	71

	Custom JAAS configuration	72
	JAAS LoginModule control flags	74
	JAAS configuration options	77
	Example: Defining an LDAP custom configuration	81
	Standard Sun Microsystems JAAS configuration	81
	Command line options	82
	Integrated Windows Authentication.	82
	Setting up Integrated Windows Authentication	83
	Testing the settings.	91
	Integrating with single sign-on tools.	91
	Testing access to BI Portal from a single sign-on tool	93
	Contact-based authentication	94
	Creating an alternate login page	99
	Creating a login Web page.	99
	Specifying an alternate authentication method	101
Chapter 6	RDS Universe Administration	103
	Understanding the structure of the rds universe.	104
	RDS elements	104
	RDS Object Hierarchy	104
	ServiceCenter Common Objects	106
	ServiceCenter Modules	106
	Report Writing Basics.	108
	Which ServiceCenter Module?	108
	Standard or OLAP?.	108
	Include Historical data?.	108
	Report Design Walkthrough	109
	Scheduling automatic data synchronization	113
Chapter 7	Troubleshooting	117
	Browser issues	117
	Navigation Issue	117
	Tomcat issues	118
	Index.	119



About this Guide

This *BI Portal Administrator's Guide* provides information about administration of BI Portal. The guide includes extensive information about Peregrine OAA, the software platform on which BI Portal is based, and specific information about BI Portal.

Using this Guide

This guide includes the following chapters:

This chapter	Describes
Chapter 1	Introduction to the BI Portal user interfaces; key concepts of BI Portal
Chapter 2	Standard reports included in BI Portal
Chapter 3	How to work with standard and custom reports in BI Portal
Chapter 4	How to create ad hoc queries and create custom reports in BI Portal

Prior to using this guide, you should read the following sections:

- *Related Documentation* on page 8
- *Documentation Conventions* on page 9
- *Contacting Peregrine Systems* on page 9

Related Documentation

In addition to this guide, the following documentation is available for the BI Portal product:

This manual...	Provides information on...
<i>BI Portal User's Guide</i>	standard reports and describes how to create and work with both standard and custom reports.
<i>BI Portal Installation Guide</i>	Installing and configuring the application and Web servers for BI Portal.
<i>BI Portal Release Notes</i>	Last-minute enhancements, known issues, and closed issues.
<i>OAA Platform Administration Guide</i>	Installing and maintaining systems that use the Peregrine OAA platform.
<i>WebIntelligence User's Guide</i>	Describes how to use WebIntelligence for building and running queries, reporting, and analysis.

Documentation Conventions

The following typographical conventions are used in this guide.

Text Formatting	Meaning
<i>italics</i>	Text that acts as a placeholder for information you will provide. Italics are also used for book titles and for emphasis.
sans serif font	Text that you type. Examples are filenames and URLs. This font is also used for samples of code and commands.
bold	Names of user interface elements. Examples are menu items and names (select Open from the File menu), button names (click Accept), and names of screens or dialogs (the Server Manager window).

Contacting Peregrine Systems

For further information and assistance with this release, you can download documentation or schedule training.

Customer Support

For further information and assistance, contact Peregrine Systems' Customer Support at the Peregrine CenterPoint Web site.

To contact customer support:

- 1 In a browser, navigate to <http://support.peregrine.com>
- 2 Log in with your user name and password.
- 3 Follow the directions on the site to find your answer. The first place to search is the KnowledgeBase, which contains informational articles about all categories of Peregrine products.
- 4 If the KnowledgeBase does not contain an article that addresses your concerns, you can search for information by product; search discussion forums; and search for product downloads.

Documentation Web site

For a complete listing of current BI Portal documentation, see the Documentation pages on the Peregrine Customer Support Web.

To view the document listing:

- 1 In a browser, navigate to <http://support.peregrine.com>.
- 2 Log in with your login user name and password.
- 3 Click either **Documentation** or **Release Notes** at the top of the page.
- 4 Click the BI Portal link.
- 5 Click a product version link to display a list of documents that are available for that version of BI Portal.
- 6 Documents may be available in multiple languages. Click the Download button to download the PDF file in the language you prefer.

You can view PDF files using Acrobat Reader, which is available on the Customer Support Web site and through Adobe at <http://www.adobe.com>.

Important: Release Notes for this product are continually updated after each release of the product. Ensure that you have the most current version of the Release Notes.

Education Services Web Site

Peregrine Systems offers classroom training anywhere in the world, as well as “at your desk” training via the Internet. For a complete listing of Peregrine’s training courses, refer to the following web site:

<http://www.peregrine.com/education>

You can also call Peregrine Education Services at +1 858.794.5009.

1 Peregrine OAA Architecture Overview

CHAPTER

Because BI Portal is based on the Peregrine Open Application Architecture, this guide includes extensive information about the platform.

Peregrine Open Application Architecture (OAA) platform is a software platform that enables the hosting of a variety of Web applications over a corporate intranet. The platform is Java based, encompassing the latest in Java technology including Java servlets, JAAS login authentication, and JSP pages that enable Web pages to display data dynamically.

Peregrine OAA Platform is the underlying architecture for many Peregrine products, including the Get-It suite of Employee Self-Service products which offers:

Get-It Product	Description
BI Portal	Web-based reporting tool for creating and executing queries against ServiceCenter 5.1 data; and for generating reports and graphs based on that data.
Get-Answers	Web-based, knowledge management application that enables you to capture and search on knowledge. Using Get-Answers, you can improve the quality and accuracy of the knowledge used by people in your company to get their jobs done and avoid calls to the help desk.

Get-It Product	Description
Get-Resources™	Web-based solution that integrates with AssetCenter Procurement, AssetCenter 4.x Portfolio, or ServiceCenter Request Management to enable employees to create requests for resources and to streamline the approval workflow of those requests throughout the organization.
Get-Services™	Web-based extension of ServiceCenter that enables users to report problems in the work environment by opening problem tickets in Get-Services and then storing them in the ServiceCenter back-end system, allowing users to view tickets from Get-Services and ServiceCenter. Modules include Administration, Service Desk, and Change Management (with ServiceCenter 5.0 and 5.1).

Peregrine OAA Platform provides a Web portal, Peregrine Portal, from which users can access their Web applications. The Peregrine Portal also provides access to the Admin module, from which all aspects of the Peregrine OAA Platform are monitored and maintained.

The base of Peregrine OAA Platform includes:

- Archway—a Java servlet that processes HTTP requests from a browser, sends the requests through an adapter to a back-end system, and returns XML data to be displayed in the browser.
- Core files—the Peregrine OAA Platform contains jsp and XML. The core consist mainly of low level Java utility classes used by the Portal Web applications built on the base OAA framework.
- Peregrine Portal—includes a login page and provides access to your Peregrine Web applications and to the Admin module for configuration of your application.
- Skins and style sheets—provide a choice for the appearance of the Web pages.

The Peregrine OAA Platform includes a number of optional components that are configured for use with Web applications as they are needed. These include:

- Adapters—enables connection to the back-end system database. The adapter required by your Web application is deployed during the installation.
- OAA Persistence (Get-Answers only)—provides a general purpose database that is used by certain Peregrine Web applications. OAA Persistence provides data persistence to a database.

- OAA Workflow (Get-Answers only)—enables workflow capabilities used by some Peregrine OAA Platform Web applications.
- Notification Services (Get-Answers only)—a centralized service for sending and receiving notifications through multiple communication devices and for tracking the status of these notifications.

Separate documentation for Notification Services is provided with the Web applications that use this feature.

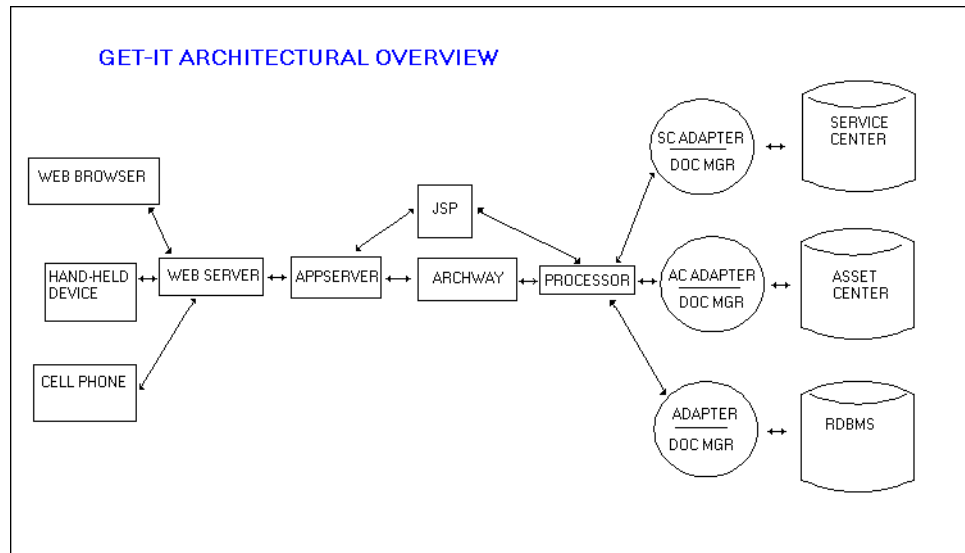
Peregrine OAA Platform architecture

Peregrine OAA Platform applications and interfaces use Web-based building blocks that include:

HTTP	A simple and widely supported protocol for sending client requests to a server. Variations such as HTTPS provide security as well.
XML	Extensible Markup Language. A documentation meta-language that allows you to format data, which can then be displayed through a Web browser. Unlike HTML, you create your own XML tags and define them any way you want.
Commercial web servers	The services provided by the Archway architecture can be served from any commercial Web server, including IIS and Apache.
Application servers	Peregrine OAA Platform supplies Apache Tomcat for an application server with the installation. JRun, WebSphere, and WebLogic are also supported.
Common clients	Applications can be deployed via Web browsers (IE, Netscape), handheld devices (Palm Pilot), or mobile phones (through HDML).

The application server processes data (JSP pages, XML, and so forth) that it receives from the database or client that is specifically related to the Peregrine Systems Web applications. The Web server converts the data into a form (HTML) that can be displayed in a Web browser.

The following diagram illustrates the architecture:



The Archway component listens to HTTP requests from clients, routes the requests to an appropriate server, and returns data or documents. The requests supported by Archway can vary, but they fundamentally consist of queries, data updates, or system events.

For example, a client can contact Archway and ask to query a database for a list of problem tickets. Another client could contact Archway and supply it with a new purchase request to be entered into the database.

All requests and responses are formatted using XML. For example, a problem ticket expressed in XML could appear as follows:

```

<problem>
  <number>PM5670</number>
  <contact> Joe Smith </contact>
  <description> My printer is out of paper </description>
</problem>
  
```

Clients that interact with Archway can do anything they need with the XML that is returned as a response. Very frequently, the client initiating the request is a user interface such as a Web browser. Such a client could easily display the XML documents returned by Archway. However, to be of better use, the XML documents are often displayed within a formatted HTML page. This is accomplished by using Java Server Pages (JSP).

JSP provides a syntax for creating HTML pages that is pre-processed by the Web server before being sent to the browser. During this processing, XML data obtained from Archway is merged into the HTML page.

Archway's architecture includes special support for automatically generating the HTML and JSP pages that make up a Web application.

Archway internal architecture

Archway is implemented as a Java servlet. The Java servlet is an application executed by a Web server that processes HTTP requests from client Web browsers and sends the request, by way of an adapter, to a database. It then retrieves the requested information from the database and returns it to the client. Archway requires both a Java environment and a Web server.

Each request is interpreted to determine its destination. Archway is able to communicate with a variety of back-end systems, including the AssetCenter or ServiceCenter products from Peregrine.

Requests can be handled in one of three ways:

- A request can be sent directly to an adapter that talks to a back-end server. For instance, a query request for opened tickets could be forwarded to an adapter capable of communicating with ServiceCenter.
- A request can be sent to a script interpreter hosted by Archway. This enables you to define your own application-specific services. Within a script, calls can be made back to Archway to access the back-end system with database operations and events.
- Finally, a request can be sent to a component known as a Document Manager. This component provides automated services for combining logical documents.

Archway communicates with back-end systems with the help of specialized adapters that support a predefined set of interfaces for performing connections, database operations, events, and authentication. All adapters use DLLs to communicate with each application.

Messages can be routed to a script interpreter hosted by Archway. The interpreter supports ECMAScript, a European standard based on the Core JavaScript language used by Netscape (JavaScript) and Microsoft Internet Explorer (JScript).

Messages can be routed to the Document Manager component. This component reads special schema definitions that describe application documents for logical entities such as a purchase request, problem ticket, or product catalog. The script interpreter uses these schemas to automatically generate database operations that query, insert, or update such documents.

Archway requests

Archway supports a variety of requests, all of which are based on two basic technologies: HTTP and XML. The HTTP protocol defines a simple way for clients to request data from a server. The requests are stateless and a client/server connection is maintained only during the duration of the request. All this brings several advantages to Archway, including the ability to support a large number of requests with the help of any of today's commercial Web servers.

Another important advantage is that any system capable of making HTTP requests can contact Archway. This includes Web browsers, of course. But in addition, all modern programming environments support HTTP. This makes it very simple to write new adapters that communicate with Peregrine servers without the need of specialized APIs.

You can test the output generated by your server-side onload scripts and schemas by using URL queries to the Archway servlet.

Archway will invoke the server script or schema as an administrative user and return the output as an XML document. Your browser will need an XML renderer to display the output of the XML message.

Note: Your browser may prompt you to save the XML output of the URL query to an external file.

URL Script Queries

Archway URL script queries use the following format:

```
http://server name/oa/servlet/archway?script name.function name
```

- For *server name*, enter the name of the Java-enabled Web server. If you are testing the script from the computer running the Web server, you can use the variable `localhost` as the server name.

The `/oa/servlet` mapping assumes that you are using the default URL mapping that BI Portal automatically defines for the Archway servlet. If you have defined another URL mapping, replace the servlet mapping with the appropriate mapping name.

- For *script name*, enter the name of the script you want to run.
- For *function name*, enter the name of the function used by the script.

Note: URL queries functionality can be removed by configuring the `WEB.xml` file. This is a recommended security setting.

URL Schema Queries

Archway URL schema queries use the following format:

```
http://server name/oa/servlet/archway?adapter name.Querydoc
&_document= schema name
```

- For *adapter name*, enter the name for the back-end database adapter the schema uses. The adapter listed here will use the ODBC connection that you have defined in the Admin module Settings page.
- For *schema name*, enter the name defined in the <document name="schema name"> element of the schema file.

The `/oa/servlet` mapping assumes that you are using the default URL mapping that BI Portal automatically defines for the Archway servlet. If you have defined another URL mapping, replace the servlet mapping with the appropriate mapping name.

URL SQL Queries

Archway URL SQL queries use the following format:

```
http://server name/oa/servlet/archway?adapter name.query&_table=
table name&field name=value&_[optional]=value
```

- For *adapter name*, enter the name for the back-end database adapter the schema uses. The adapter listed here will use the ODBC connection that you have defined in the Admin module Settings page.
- For *table name*, enter the SQL name of the table you want to query from the back-end database.
- For *field name*, enter the SQL name of the field you want to query from the back-end database.
- For *value*, enter the value you want to the field or optional parameter to have.
- For *_[optional]*, enter any optional parameters to limit your query. Examples include:

- `_return`. Returns the values only of the fields you list.
- `_count`. Specifies how many records you want returned with the query.

The `/oa/servlet` mapping assumes that you are using the default URL mapping that BI Portal automatically defines for the Archway servlet. If you have defined another URL mapping, replace the servlet mapping with the appropriate mapping name.

The following are sample URL SQL queries:

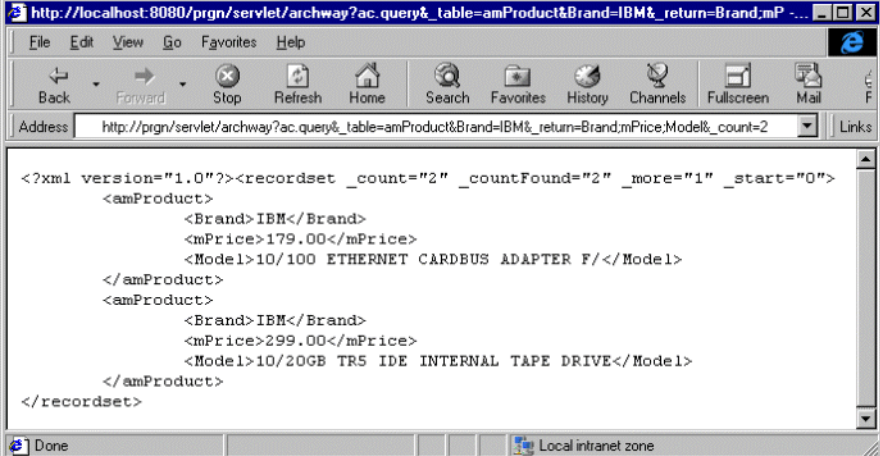
- `host name/oa/servlet/archway?sc.query&_table=probsummary&priority.code=1`

This sends a query request to ServiceCenter for all records in the `probsummary` table with a priority code of 1.

- `host name/oa/servlet/archway?ac.query&_table=amAsset&_return=Brand;mPrice;Model&_count=2`

This sends a query request to AssetCenter for the first two records in the `amProduct` table. Only the `Brand`, `mPrice`, and `Model` fields are returned for each record.

The screen below shows the XML results of a query for products from AssetCenter.



```
<?xml version="1.0"?><recordset _count="2" _countFound="2" _more="1" _start="0">
  <amProduct>
    <Brand>IBM</Brand>
    <mPrice>179.00</mPrice>
    <Model>10/100 ETHERNET CARDBUS ADAPTER F</Model>
  </amProduct>
  <amProduct>
    <Brand>IBM</Brand>
    <mPrice>299.00</mPrice>
    <Model>10/20GB TR5 IDE INTERNAL TAPE DRIVE</Model>
  </amProduct>
</recordset>
```


2 Customizing the Peregrine Portal

CHAPTER

Peregrine OAA provides a number of ways to customize the interface of an application built on the platform. You can make a quick change, such as replacing the logo with your company logo, or a more complex change such as rewriting the code that defines layer placement or frameset size.

This chapter includes advanced procedures for changing the ProductCoreAbbreviated interface. To use this information effectively, you should have knowledge of XML and the CSS2 specifications established by the W3C as outlined at www.w3.org.

Topics in this chapter include:

- *Deploying the Classic theme variations*
- *Changing the default theme*
- *Changing the header graphic for all themes*
- *Creating a custom theme*
- *Layers properties*
- *Changing framesets*

Deploying the Classic theme variations

The Classic theme is the default theme that applications built on Peregrine OAA use. It has a gray and teal design and is the theme shown in all the screenshots in the guide. This is the theme you will use to create a customized theme for your enterprise.

There are four variations of the Classic theme:

- *Accessible*, which makes the screen available to users who need high contrast colors or better accessibility support.
- *Baja*, which adds southwestern green and beige hues to the Classic design.
- *Quicksilver*, which adds silver and blue hues to the Classic design.
- *Sierra*, which adds teal hues to the Classic design.

These themes, as well as a number of other optional themes, are deployed with the application installation. Once you create your customized theme, Peregrine Systems recommends that you delete all other themes to prevent users from selecting one of them and overriding your custom theme. If you decide later that you want to manually deploy a theme that has been deleted, or if you did not deploy all themes during the installation, use the following procedure to deploy the themes. The additional themes are zip files located in the `C:\Program Files\Peregrine\oaa\packages` directory. You can identify the theme names from these zip file names.

To deploy an alternate Classic design:

- 1 Open a command prompt window, and change directories to your `oaa\packages` directory. The default path is:
`C:\Program Files\Peregrine\oaa\packages`

- 2 Type:

```
java -jar OAADeploy.jar <name of the theme>
```

Note: List each theme you want to deploy, separated by a space; for example,
`java -jar OAADeploy.jar bluestheme hightechtheme bajatheme.`

- 3 Press ENTER.
- 4 Stop and restart your application server.

The themes you deployed appear as options the next time you log in to BI Portal.

Changing the default theme

You can change the default theme that all users see when they log in to BI Portal. Out-of-the-box, the default theme is classic.

To change the default theme:

- 1 Open your Web browser and log in to the Admin module (`localhost/oa/admin.jsp`).
- 2 Click **Settings > Themes**. Change the following parameters:
 - a In the **Default skin/Theme** field, change the parameter to the name of the theme you want to use (for example, *Baja*).
 - b In the **Default stylesheet** field, change the parameter to the appropriate name for the CSS file (for example, *baja.css*).
 - c In the **Default XSL stylesheets** field, change the parameter to the name of the theme you want to use (for example, *Baja*).
- 3 Scroll to the bottom of the page, and then click **Save**.
- 4 When the Control Panel opens, click **Reset Server**.
- 5 Refresh your browser to see the new default theme.

Changing the header graphic for all themes

You can add your corporate logo to all themes in the ProductCoreAbbreviated from the Administration Settings page.

Warning: The administration setting discussed below overrides the image used by all themes. If you change this setting then you will see the same logo in all themes. If you want to use a different corporate logo for each theme, see *Creating a custom theme* on page 19.

To change the header graphic for all themes:

- 1 Create a custom header graphic.

Note: To fit within the default header frame, your customized header logo must be 514 pixels wide and 59 pixels high. If you want to change the header frame size, see *Changing framesets* on page 24.

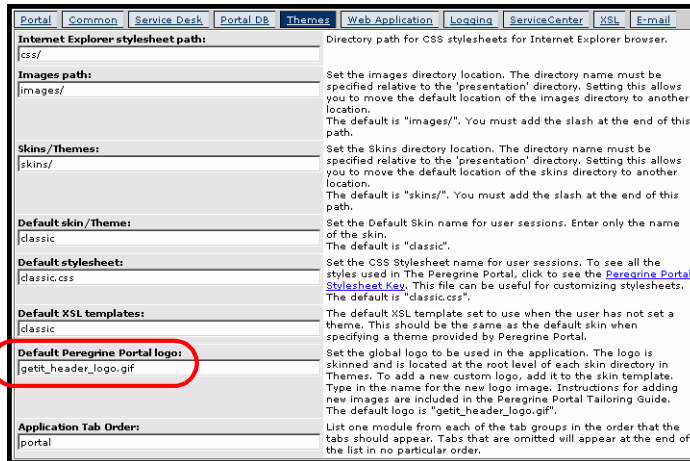


- 2 Save your custom header graphic to the following location:

C:\Program Files\Peregrine\Common\Tomcat4\\webapps\oaa\images\skins\classic

Note: The Classic theme is the default theme.

- 3 Log in to the BI Portal administration page (admin.jsp).
- 4 Click **Settings > Themes**.
- 5 In the **Default Peregrine Portal logo** field, enter the name of your custom header logo.



Type your new image name.

- 6 Scroll to the bottom of the page, and then click **Save**.
- 7 When the Control Panel opens, click **Reset Server**.
- 8 Refresh the browser to view your changes.

Creating a custom theme

You can create custom themes by copying and modifying the classic theme provided with BI Portal.

To create a custom theme:

- 1 Copy classic theme images, stylesheets, and XSL templates. These files are located at:
 - Images. `<application server>\oaa\images\skins\classic`
 - Stylesheets. `<application server>\oaa\css\classic`
 - XSL templates. `<application server>\oaa\WEB-INF\templates\classic`
- 2 Paste and then rename the folders for the classic theme to a new name. For example:
 - Images. `<application server>\oaa\images\skins\myTheme`
 - Stylesheets. `<application server>\oaa\css\myTheme`
 - XSL templates. `<application server>\oaa\WEB-INF\templates\myTheme`
- 3 Open and edit each image that you want to change in your new theme. Use the following image conventions.
 - Image file names must remain the same. BI Portal uses these image names to display theme elements.
 - Image height and width should remain the same unless you are also changing the size of the framesets to accommodate new image sizes.
- 4 Open and edit the `classic.css` file in your new theme.

The following table lists some of the more commonly modified styles.

Style Name	Style Description
<code>.ActionButton</code>	The style used on buttons throughout the Portal.
<code>.ActiveMenuLink</code>	Used when the mouse hovers over a menu link.
<code>.ActiveModuleMenu</code>	Designates the currently-selected page within the navigational subset.
<code>.CurrentModuleMenu</code>	Designates the currently-selected navigational subset.
<code>.FormTitle</code>	Used for the title of forms. Normally used to title DocExplorer window content.

Style Name	Style Description
.ListBoxEvenRow	A bolded version of TableEvenRow.
.ListBoxHeading	A bolded version of Table Heading.
.ListBoxOddRow	A bolded version of TableOddRow.
.MenuLink	Used within all module menus.
.ModuleMenu	Used for the left-hand navigational menu.
.ModuleMenuTitle	Designates the navigational subsets title.
.PageTitle	Used on the page title located directly below the logo and tabs.
.TableEvenRow	Used within the table heading with alternating background colors for ease of reading. Has a background color of white.
.TableHeading	Used for application headings for both search and results functions.
.TableOddRow	Used within the table heading with alternating background colors for ease of reading. Has a background color of light gray.
a.ListBoxEvenRow	Designates the style with a link attribute.
a.ListBoxOddRow	Designates the style with a link attribute.
a.TableEvenRow	Designates the style with a link attribute.
a.TableOddRow	Designates the style with a link attribute.

Tip: Modify the style sheets after you complete your overall theme design. Use your image editor's color picker to ensure that the your stylesheet colors match your image colors.

Note: You can see a detailed stylesheet key in the themes Administration section of the Portal. To access the stylesheet key, locate the Default stylesheet field on the Themes tab of the Admin Settings page and click the [Peregrine Portal Stylesheet Key](#) link.



- 5 Save your theme stylesheet with the same name as your new theme. For example, `<application server>\oaa\css\myTheme\myTheme.css`.
- 6 Open and edit the `layers_<xx>.jsp` file to change any layer descriptions. To change layers for Internet Explorer, open `layers_ie.jsp`. To change layers for Netscape open `layers_gecko.jsp` extension. For more information about editing layers see [Layers properties](#) on page 23.
- 7 Open and edit any XSL stylesheets you want to change.

Warning: Do not change these files unless you have knowledge of XSL and HTML transformation.

The XSL stylesheets determine how BI Portal displays form components in the main portal frame.

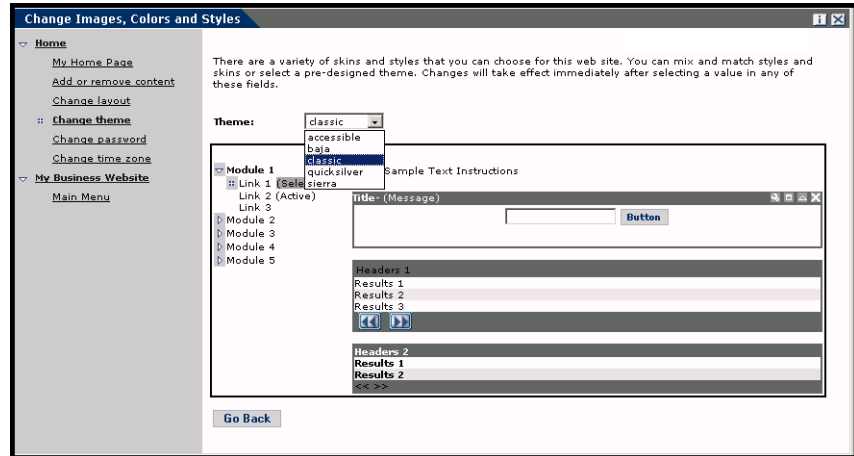
The following table lists the XSL stylesheets you can change.

To change	edit this XSL stylesheet
Attachment picker	attachments.xml
HTML form generation	basic-form.xml

To change	edit this XSL stylesheet
Action (button) properties	button.xsl
Template components	components.xsl
Debugging message properties	copy_nodes.xsl
Date-time picker properties	datetime.xsl
Text edit field properties	edit_fields.xsl
Entry table form component (see administration page for examples)	entrytable.xsl
Field section properties	fieldsection.xsl
Field table properties	fieldtable.xsl
HTML page generation	form.xsl
Frameset properties	frames.xsl
Images properties	image_fields.xsl
Label properties	labels.xsl
Link properties	link.xsl
Building of DocExplorer lists	list-builder.xsl
Lookup field properties	lookup_fields.xsl
Money text field properties	money_fields.xsl
Portal properties	portal.xsl
Radio checkbox properties	radio_checkbox_fields.xsl
Read-only text field properties	readonly_fields.xsl
Select text field properties	select_fields.xsl
Spinner properties	spinner_fields.xsl
SVG image properties	svg_cad.xsl
Table properties	table.xsl
Navigation tab properties	tabs.xsl

- 8 Stop and restart your application server.

You can view your new theme by selecting it from the *Change theme* page, available from the Peregrine Portal Home page.



Layers properties

The following sections describe the `layers_ie.jsp` and `layers_gecko.jsp` files. Each layer is defined by a separate `<div>` tag entry and includes an `id` attribute that names the layer. You can change layer properties as needed, but the following layers are required and should not be removed:

- **logo**

```
<div id="logo" style="position:absolute; left: 0px; top: 0px; width:
100%; height: 40px; z-index: 3;">

</div>
```

- **time**

```
<div id="time" style="position:absolute; right: 4px; top: 84px;
width: 100%; z-index: 13;" onmouseover="_pauseAlert()"
onmouseout="_startAlert()" class="userBarText">
</div>
```

- **toolbar**

```
<div id="toolbar" style="position:absolute; width: 50px; top: 59px;
right: 0px; z-index: 12;"></div>
```

- **user**

```

<div id="user" style="position:absolute; top: -4px; right: 0px;
z-index: 14;">
<table width="100%" border="0" cellpadding="0" cellspacing="0"
align="right">
<tr>
<td width="50%">&nbsp;&nbsp;&nbsp;</td>
<td nowrap width="3" align="right" valign="top">
">
</td>
<td nowrap align="right" valign="top" width="100%" background="<%=
Archway.getSkinImagePath("backgrounds/rt_tile.gif", user ) %>">
">
</td>
<td nowrap><font class="userBarText" size="1" face="Arial, Helvetica,
sans-serif"><%=userTitle%></font>&nbsp;&nbsp;&nbsp;</td>
</tr>
</table>
</div>

```

■ tabs

```

<div id="tabs" style="position:absolute; left: 0px; top: 60px; width:
100%; z-index: 11;" >
</div>

```

■ form titles

```

<div id="formTitles" style="position:absolute; left: 10px; top: 81px;
width: 200px; z-index: 16;">&nbsp;&nbsp;&nbsp;
</div>

```

Changing framesets

Important: You must have advanced knowledge of HTML, JSP, and framesets to modify these files. Keep all of the frames and do not change the names of any of the frames. Doing so will result in JavaScript errors.

There are two framesets to be modified for each browser. These files are in C:\Program Files\Peregrine\Common\Tomcat4\webapps\oaa\images\skins*<your theme>*.

The `frames_xx.jsp` files are for the pages that you access when logging in as an end-user (`login.jsp`). The `admin_frames_xx.jsp` files contain the configuration for the Admin module (accessed when you log in using `admin.jsp`).

To change framesets:

- 1 Stop your application server.
- 2 Open the browser-specific frameset file `frames_<xx>.jsp` in a text editor (where `<xx>` is `ie` for Internet Explorer and `gecko` for Netscape).
- 3 Modify the frameset properties.
- 4 Save the file.
- 5 Restart your application server.

You can now test your changes in your Web browser.

The following sections show the complete `_ie.jsp` files as examples of the frameset files.

`frames_ie.jsp`

```
<%@ include file="../../../jspheader_2.jsp" %>
<%@ include file="../../../message_special.jsp" %>

<frameset onload="setTopFrames()" onunload="closeChildWindows()"
border="0" framespacing="0" frameborder="NO" cols="*" rows="102,*">
  <frame scrolling="NO" marginwidth="0" marginheight="0"
src="oaa_header.jsp" name="getit_main_head">
    <frameset cols="185,10,*" rows="*" frameborder="no" border="0"
framespacing="0">
      <frame scrolling="AUTO" marginwidth="0" marginheight="0"
src="apphead.jsp" name="getit_header">
        <frame name="framesep" scrolling="no" marginheight="0"
marginwidth="0" src="framesep.jsp">
          <frameset rows="*,0">
            <frame scrolling="AUTO" marginwidth="6" marginheight="6"
src="e_login_main_start.jsp?<%= user.getADW(msg,"Params" ) %>"
name="getit_main">
              <frame noresize scrolling="NO" marginwidth="0"
marginheight="0" src="backchannel.htm" name="backchannel">
            </frameset>
          </frameset>
        </frameset>
      </frameset>
    </frameset>
```

`admin_frames_ie.jsp`

```
<%@ include file="../../../jspheader_2.jsp" %>
<%@ include file="../../../message_special.jsp" %>
```

```
<frameset onload="setTopFrames()" onunload="closeChildWindows()"
border="0" framespacing="0" frameborder="NO" cols="*" rows="102,*">
  <frame scrolling="NO" marginwidth="0" marginheight="0"
src="oaa_header.jsp" name="getit_main_head">
    <frameset cols="185,10,*" rows="*" frameborder="no" border="0"
framespacing="0">
      <frame scrolling="AUTO" marginwidth="0" marginheight="0"
src="apphead.jsp" name="getit_header">
        <frame name="framesep" scrolling="no" marginheight="0"
marginwidth="0" src="framesep.jsp">
          <frameset rows="*,0">
            <frame scrolling="AUTO" marginwidth="6" marginheight="6"
src="e_adminlogin_login_start.jsp?<%= user.getADW(msg, "Params") %>"
name="getit_main">
              <frame noresize scrolling="NO" marginwidth="0"
marginheight="0" src="backchannel.htm" name="backchannel">
            </frameset>
          </frameset>
        </frameset>
      </frameset>
```


3 Using the Peregrine Portal

CHAPTER

The BI Portal includes a Navigation menu, an Activity menu, and buttons that enable you to customize your Portal and to end your session.

Your installed Web applications determine the contents of the Navigation menu. However, if you log in as an administrator, all Navigation menus include an Administration tab that provides access to the Admin module.

The graphics in this chapter use the Classic stylesheet and are examples of a generic interface. Also, the Admin module displays only those features that BI Portal uses. For more advanced changes to the portal, see the chapter on *Customizing the Peregrine Systems Portal*.

Topics in this chapter include:

- *Logging in to the BI Portal* on page 34
- *Using the Activity menu* on page 35
- *Personalizing the BI Portal* on page 36

Logging in to the BI Portal

There are two login screens that provide access to the Peregrine Portal:

- A user login screen—<http://<server>/oaa/login.jsp>
- An administrator login screen—<http://<server>/oaa/admin.jsp>

Note: An alternative to this login method is the Integrated Windows Authentication. See the *Security* chapter of this guide.

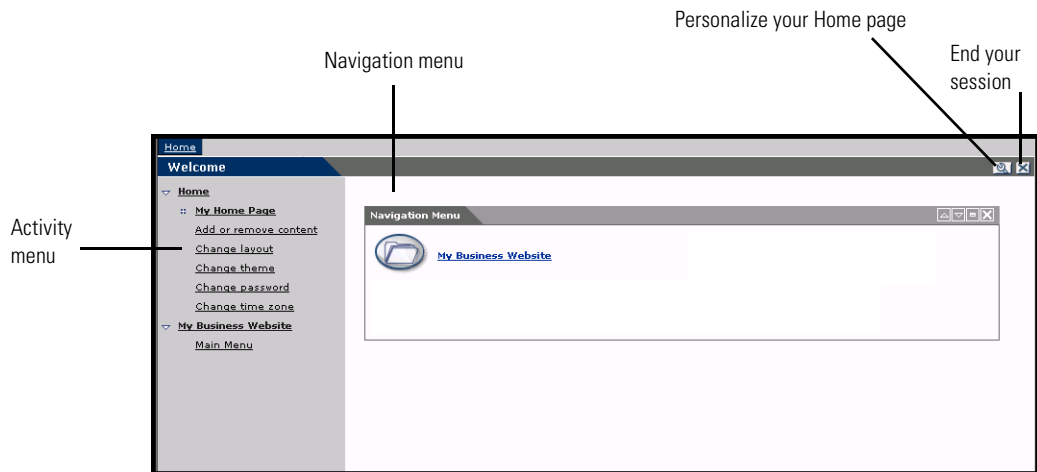
This chapter discusses the features available with a user login. For more information about the administrator login, see the chapter on *BI Portal Administration* in this guide.

The following is an example of the user login interface.



The screenshot shows the user login interface for the Peregrine Portal. The page has a blue header with the "Peregrine Portal" logo on the left and "powered by Peregrine SYSTEMS" on the right. Below the header is a "Login" tab and a "Welcome" message. The main content area contains a message: "Please enter your user name and password to enter the Peregrine Portal." Below this message are three input fields: "User Name:" with a text box and a blue arrow icon, "Password:" with a text box, and "Language:" with a dropdown menu set to "English". At the bottom of the form is a "Login" button.

The following graphic shows a Portal without any applications installed. The Navigation menu includes modules for your particular application. All applications have the Admin module.



Using the Activity menu

The Activity menu provides access to a number of tasks as you navigate through your Web application. The menu remains visible as you change screens.

The default Activity menu includes the following choices:

Use this option	When you want to
My Home Page	Return to the Peregrine Portal Home page.
Add or remove content	Access the same page as the Personalization button, allowing you to customize your Home page.
Change layout	Change the location of a component or remove it from the Peregrine Portal.
Change theme	Select from several options. Changes take effect immediately after selecting a value in any of these fields. Note: Select the accessible theme to access the alternate text-based interface.
Change time zone	Select the time zone.

Personalizing the BI Portal

By default, the Navigation menu is displayed on the Peregrine Portal. You can personalize the Peregrine Portal to add BI Portal utilities as well as personal tools such as a calendar, calculator, or the date and time. You can also change the layout of these components or minimize a component to hide the component details.

See the chapter on *Using the Personalization Interface* in this guide for more information on personalization.

Adding components

The following components are available:

Personal Utilities

This component	Provides
Calculator	A tool using standard arithmetic functions.
Calendar	A monthly calendar.
Theme Selector	A drop-down list to change themes.
Date and Time	A date and time display for the local time zone.

Peregrine Portal Web application components

This component	Provides
Navigation Menu	Quick links to the various modules that make up this application.
Document List	A display of a document search, list, or detail screen. Configure the component by choosing the document type you want to expose and the type of screen desired.
My Menu	A menu of links that can be configured dynamically. Links can point to arbitrary web sites, other menus, or document explorer screens.

Note: The Calendar and Calculator require Microsoft Internet Explorer 5.0+ or Netscape 6.1+.

Administration components

Only users with Admin capability have access to the Admin components.

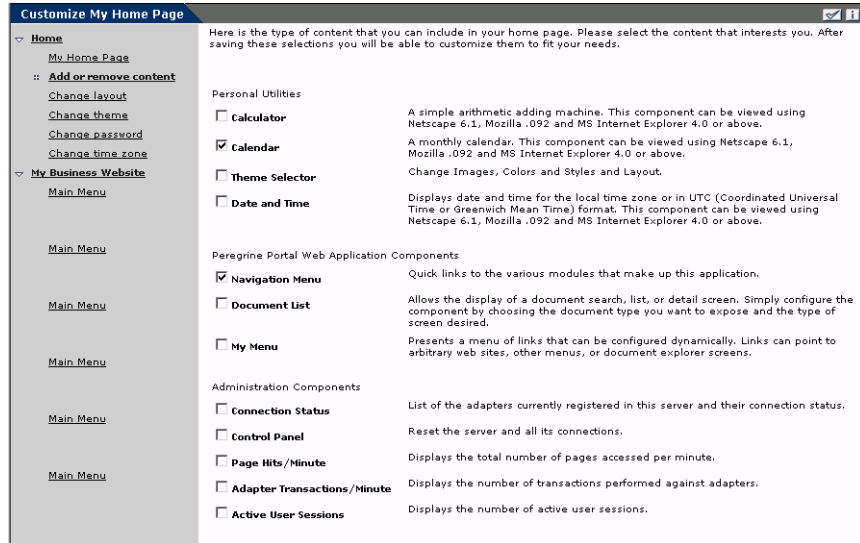
This component	Provides
Connection Status	A list of the adapters currently registered in this server and their connection status.
Control Panel	A button to reset the server and all its connections.
Page Hits / Minute	A list of the total number of pages accessed per minute.
Adapter Transactions / Minute	A list of the number of transactions performed against adapters.
Active User Sessions	A list containing the number of active user sessions.

To add Peregrine Portal components:

- 1 Click the **Personalize** (wrench) icon.

Note: You can also select the **Add or remove content** link from the Activity menu.

The **Customize My Home Page** opens containing a list of the available components.



- 2 Select the components you want to add to your Peregrine Portal.
- 3 When you complete your selections, scroll to the bottom of the page, and then click **Save**. To return to the Peregrine Portal without making any changes, click **Go Back**.

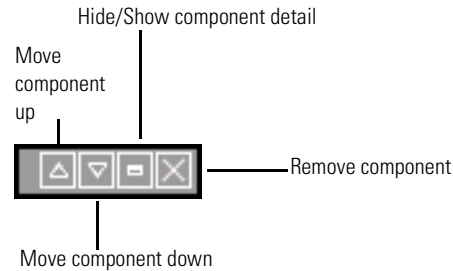
When you return to the Peregrine Portal, the new components are displayed. The following example shows the .

Changing the layout

The following sections contain procedures for changing the location of the components or removing them from the Peregrine Portal. The procedure you use is determined by the Web browser you are using.

Microsoft Internet Explorer

If you are using Microsoft Internet Explorer as your Web browser, use the buttons in the upper right corner of each component to move the component up or down, remove the component, or hide/show the component detail.



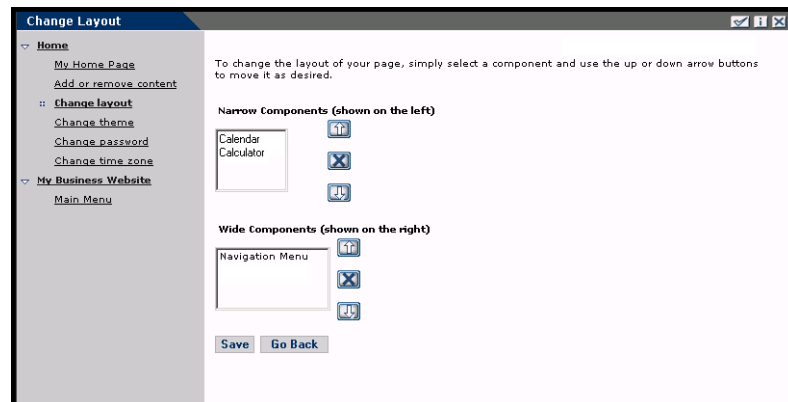
In the following screen, the Calendar is minimized.

Netscape Navigator

If you are using Netscape Navigator as your Web browser, use the following procedure to change the status of the components on the Peregrine Portal. You can move a component up or down, or remove the component.

- 1 From the Activity menu, select **Change layout**.

A **Change Layout** page opens where you select the components you want to change.



Components can be Narrow (for example, Calendar or Calculator) and are on the left side of the Peregrine Portal. Other components (for example, Navigation Menu) are Wide and are on the right side of the Peregrine Portal.

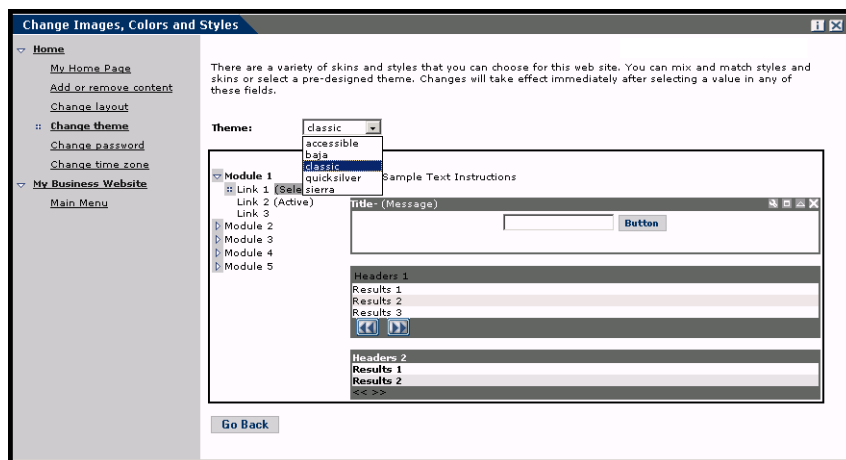
- 2 Select the component you want to modify, and then click the appropriate button to activate the change.
 - Up arrow moves the component up.
 - Down arrow moves the component down.
 - X removes the component from the Peregrine Portal.
- 3 Click **Save**.

Changing themes

You can choose from a number of themes to change the look of your Web pages. Out of the box, BI Portal provides five themes you can choose between. If you want to deploy additional themes, refer to *Customizing the Peregrine Portal*.

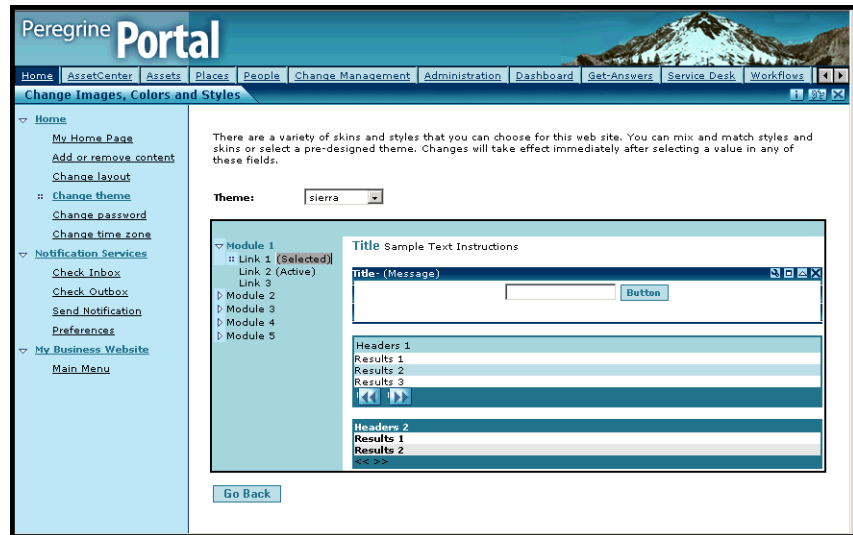
To change the theme:

- 1 From the Activity menu on the Portal Home page, select **Change theme**. The following page opens.



- 2 Choose from the drop-down list.

As soon as you make your selection, the page updates to reflect your selection. The following example shows the Sierra theme.



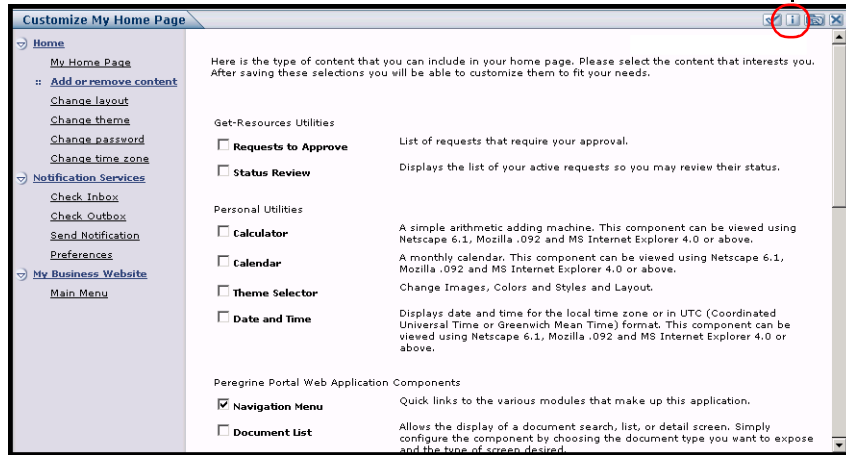
This new configuration remains through subsequent work sessions until changed.

Displaying form information

You can view information about the form you are using. Set this parameter from the Logging tab on the Settings page of the Admin module. See the BI Portal Administration chapter in this guide for more information.

When the **Show form info** parameter is set to Yes, a **Display Form Info** button appears on the upper right corner of forms.

The Display Form Info button shows information about the form you are using.



4 Using the OAA Administration Module

CHAPTER

This chapter includes instructions for administering your BI Portal system.

Topics in this chapter include:

- *Accessing the Peregrine Portal Admin module* on page 44
- *Using the Control Panel* on page 46
- *Viewing the Deployed Versions* on page 47
- *Viewing the Server Log* on page 48
- *Using the Settings page* on page 48
- *Verifying Script Status* on page 51
- *Displaying Message Queues* on page 51
- *Showing Queue Status* on page 52
- *Importing and exporting personalizations* on page 53
- *Viewing adapter transactions* on page 53
- *Using the IBM Websphere Portal* on page 54
- *Displaying form information* on page 54
- *User self-registration* on page 57
- *Changing passwords* on page 58
- *Logging and monitoring user sessions* on page 58

Accessing the Peregrine Portal Admin module

The Peregrine Portal administrator login page enables access to the Peregrine Portal Admin module. You use the Admin module to define the settings for your Peregrine system.

Note: After installing and building BI Portal, you must log in as a ServiceCenter user with **getit.admin** rights to access the Admin module and administer the BI Portal integration with ServiceCenter . For a list of access capability words and Adapter configuration instructions, see the section on BI Portal security in this guide.

A default administrator, System, gives you access to the Admin module without being connected to a back-end system. After you configure your user name on the Common tab, you can also access the Admin module from the Navigation menu.

Important: When you change parameters using the Admin module, a **local.xml** file is created in the `\<appsrvr>\WEB-INF` directory (where *appsrvr* is the path to your application server) to store these parameters. If you reinstall BI Portal, make a copy of this file and store it outside your BI Portal installation. Failure to do this will result in your parameter values being lost during the new installation.

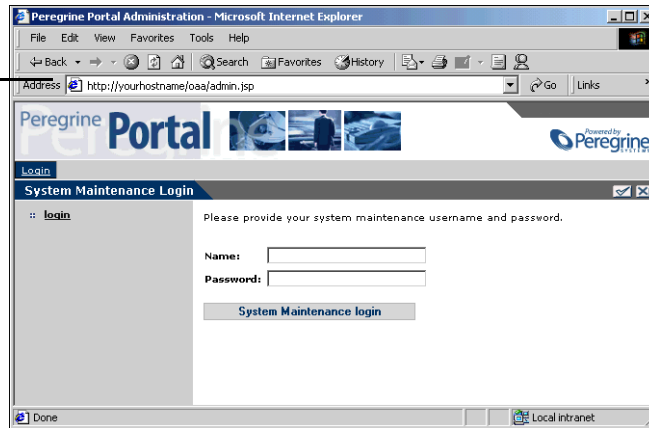
To access the Peregrine Portal administrator login page:

- 1 Verify that your application server (for example, Tomcat) is running.
- 2 In your Web browser Address field, type:
`<hostname>/oaa/admin.jsp`

3 Press Enter to open the Portal administrator login page.

Type your hostname to connect to your local server.

System is the default administrator name.



- 4 In the Name field, type System.
No password is required on initial login.
- 5 Click System Maintenance login to open the Control Panel page.

Administrators use the Admin module to define settings to the system.

Target	Adapter	Status
weblocation	com.peregrine.ooa.adapter.scSCAdapter	connected
mail	com.peregrine.ooa.adapter.mail.MailAdapter	connected
portalDB	com.peregrine.ooa.adapter.scSCAdapter	connected
sc	com.peregrine.ooa.adapter.scSCAdapter	connected

Server Name	Last Min.	5 Min. Avg.	20 Min. Avg.	Peak
localhost	1	1	1	2

Server Name	Last Min.	5 Min. Avg.	20 Min. Avg.	Peak
localhost	0	0	0	15

[Reset Server](#)

The activities available in the Admin module include:

Select this option	To do the following
Control Panel	view the status of connections to the back-end systems.
Deployed Versions	view the list of deployed applications with version numbers on this server.
Server Log	view activity on the BI Portal server.
Settings	view and change settings for the Peregrine Portal.
Show Script Status	view and verify which scripts are running. You can also start and stop scripts from this window.
Show Message Queues	view a list of all message queues.
Show Queue Status	view the current status of the queues: operational and unlocked, or suspended.
Import / Export	move Personalizations from a development to a production environment.
Adapter Transactions/Minute	view the transactions per minute for the back-end adapter.
IBM Websphere Portal Integration	view the installed OAA portal components in the IBM WPS environment

Using the Control Panel

Use the Control Panel page to check the status of the connections to the databases you are accessing with BI Portal and your Web applications. You can also reset the connection between the Archway servlet and the adapters to the back-end systems.

To reset the connection between the Archway servlet and back-end system:

- ▶ Click **Reset Server**.

A message at the top of the page indicates that the connections are reset.

Informational, warning, and error messages appear at the top of the page.

The Archway servlet and its Adapter connections have been reset successfully.

Here is a list of the adapters currently registered in this server. If necessary, you may also reset the server and all its connections.

Target	Adapter	Status
webication	com.peregrine.aaa.adapter.sc.SCAAdapter	connected
mail	com.peregrine.aaa.adapter.mail.MailAdapter	connected
portalDB	com.peregrine.aaa.adapter.sc.SCAAdapter	connected
sc	com.peregrine.aaa.adapter.sc.SCAAdapter	connected

Server Name	Last Min.	5 Min. Avg.	20 Min. Avg.	Peak
localhost	1	1	1	2

Server Name	Last Min.	5 Min. Avg.	20 Min. Avg.	Peak
localhost	1	0	0	15

[Reset Server](#)

Viewing the Deployed Versions

The Deployed Versions screen lists all of the packages that deploy during the installation, including the version number of each package.

To view the Deployed Versions list:

- 1 From the Activity menu, select **Deployed Versions**.
A list of the installed packages opens.

Current applications and their versions are available for viewing with the Deployed Versions option.

This is the list of deployed applications with version numbers on this server.

Applications	Versions
Peregrine Enterprise Portal Accessibility Theme	accessibletheme.4.1.0.5
OAA Archway Servlet	archway.4.1.0.27
Peregrine Enterprise Portal Baja Theme	bajatheme.4.1.0.4
Peregrine Enterprise Portal Classic Theme	classitheme.4.1.0.4
OAA Core Application	core.4.1.0.23
Get-Services Change Request	get-services-change-request.4.1.0.16
Get-Services Change	get-services-change.4.1.0.15
Get-Services	get-services.4.1.0.10
Get-IT Common Utilities	getitcommon.4.1.0.10
Mail Adapter	mailadapter.4.1.0.4
Peregrine Enterprise Portal	portal.4.1.0.28
Peregrine Enterprise Portal Quicksilver Theme	quicksilvertheme.4.1.0.4
ServiceCenter Adapter	scadapter.4.1.0.7
Peregrine Enterprise Portal Sierra Theme	sierratheme.4.1.0.5

[Print](#)

- 2 Click **Print** for a printout of this list.

Viewing the Server Log

The Server Log provides a history of server events. The default file name is `archway.log`.

To view the Server Log:

- 1 From the Activity menu, select **Server Log**.

A form opens with a drop-down list for you to select the log you want to view.

You can view the log file from your Web browser or download it to your preferred location.



- 2 Click the drop-down and select the log file you want to view.
- 3 Set the number of lines to view.
- 4 Do one of the following:
 - Click **View** to see the log file from your Web browser.
 - Click **Download** to initiate the File Download wizard that downloads the `archway.log` file to a location of your choice.

Using the Settings page

On the Activity menu, click **Settings** to open the current parameter settings. The Settings page is divided into tabs. The tabs that you see depend on the Web applications that you installed and the adapters that you use. The Common tab is available for all installations.

Settings for the Portal, PortalDB, Web Application tabs are set during the installation (refer to the *BI Portal Installation Guide*). You can access the Settings page at any time to change the installation settings. Set the E-mail tab only when users have access to self-registration (see *User self-registration* on page 57).

To view Settings:

- From the Activity menu, click **Settings**.

Each parameter on the tab has a description that guides you through the settings.

The tabs you see on the Settings page depend on the Web applications you installed.

The screenshot shows the 'Admin Settings' window with the 'Logging' tab selected. The left sidebar contains a tree view with 'Admin' expanded, showing sub-items like 'Control Panel', 'Deployed Versions', 'Server Log', 'Settings', 'Show Script Status', 'Show Message Queues', 'Show Queue Status', 'Import / Export', 'Adapter Transactions/Minute', and 'IBM WebSphere Portal Integration'. The main content area is titled 'Logging' and contains several sections:

- Log domains:** A text input field with a dropdown arrow. Description: 'Enter a semicolon-separated list of execution log traces you want to enable. Choices include:
 - dll - Adapter DLL loading and unloading
 - weblication - Web Application and personalization rendering
 - jvm - Java run-time environment management and status
 - locks - Script synchronization locks
 - security - Archway security trace
 - statistics - administration statistics
- Debug script:** Radio buttons for 'Yes' and 'No' (selected). Description: 'When enabled, information regarding ECMA Script execution is written to the log. Be sure to turn this off in a production system.'
- Show form info:** Radio buttons for 'Yes' and 'No' (selected). Description: 'When selected, form information is displayed in each screen to aid during Web Application development and customization.'
- Log file:** Text input field containing 'archway.log'. Description: 'Enter a full directory path to the file used for logging.'
- Logging Format:** Text input field containing '%d %-5p [%t] %x - %m%n'. Description: 'The logging format controls the printing pattern in a log file. The format is composed of literal text and conversion specifiers. The details of the specifiers can be found in the Apache [Log4j](#) documentation.'
- Log Level:** A dropdown menu currently set to 'Information'. Description: 'Controls the level of detail in the log file. Possible values are: all, debug, info, warn, error, fatal and off.'

Setting parameters using the Admin module

When you make changes using the Admin Settings page, a `local.xml` file is created in the `C:\<appsvr>\WEB-INF` directory. All changes to property settings are stored in this file. Restart Tomcat after making changes that are stored in `local.xml`.

Important: If you change parameters on the Admin Settings page and then need to reinstall BI Portal, it is important that you copy the `local.xml` file to a location other than your BI Portal installation, or all of your settings will be lost when you redeploy BI Portal. After the installation, move the copy back to the `WEB-INF` directory.

To define a parameter:

- 1 Locate the setting you want to change and type the new parameter.
Note: If you have previously changed a setting and want to return to the default setting, click the **Click for default** link displayed in the description area for the parameter you want to revert. This link appears only when a setting is different from the default.
- 2 Scroll to the bottom of the page, and then click **Save**.
Note: You must click **Save** on each page before making changes to another setting.

The Control Panel opens.

- 3 Click **Reset Server**.

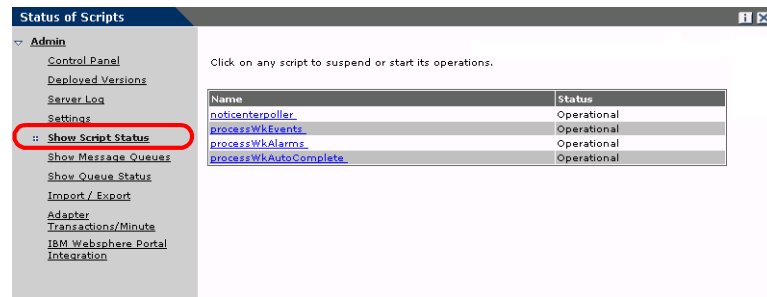
An information message at the top of the Control Panel indicates that the server has been reset.

Verifying Script Status

The Script Status page lists the name and status of any script that is currently running.

To verify the script status:

- 1 From the Administration Activity menu, click Show Script Status to display the Status of Scripts page that shows the name of each script.



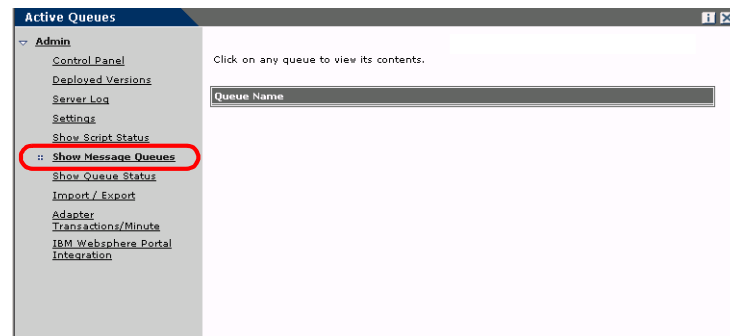
- 2 Click on the script to suspend it.

Displaying Message Queues

The Message Queues are displayed whenever a queue has data waiting to be transferred.

To display message queues:

- 1 From the Administration Activity menu, click Show Message Queues to display the Active Queues page.



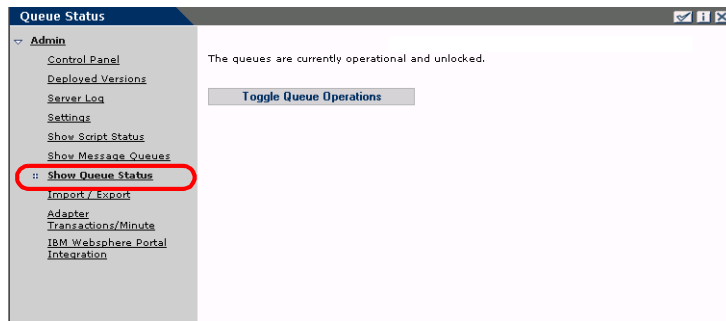
- 2 Click the queue name in the list to view the contents of a queue.

Showing Queue Status

Use the Show Queue Status option to verify or change the status of the message queues.

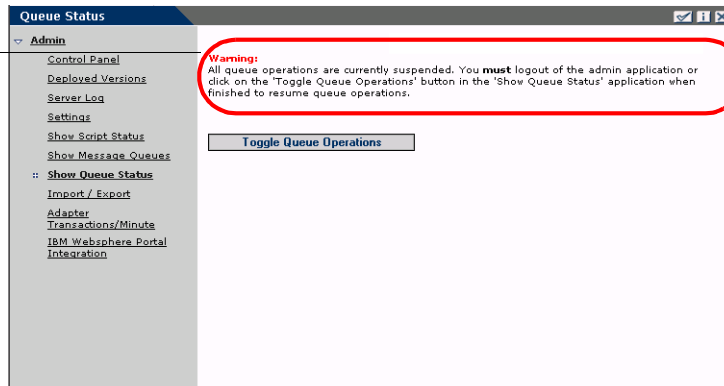
To show queue status:

- 1 From the Activity menu, click **Show Queue Status** to open the Queue Status page.



- 2 Click **Toggle Queue Operations** to change the status to suspended.

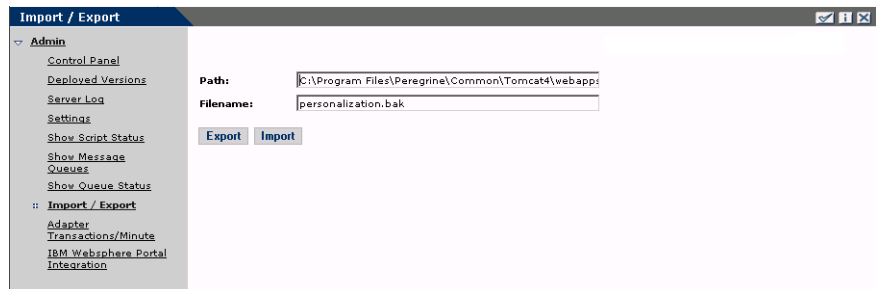
A warning message indicates that the Queue Status is suspended.



- 3 Click **Toggle Queue Operations** to return to the operational status.

Importing and exporting personalizations

You can move personalizations that you created in a development environment to a production environment. See *Using the Personalization Interface* chapter in this guide for detailed instructions on importing and exporting the personalizations. Select the **Import/Export** option from the Admin activity menu to access the page.

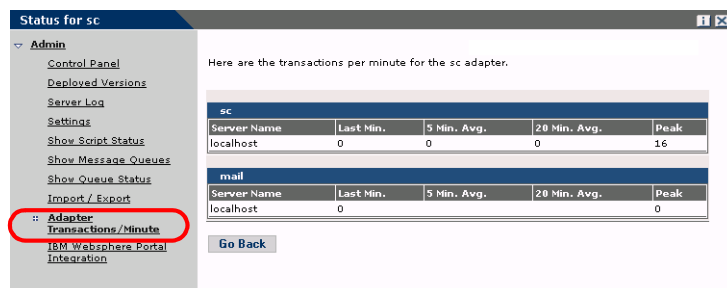


Viewing adapter transactions

You can track your adapter transactions by viewing the adapter Status page.

To view adapter transactions per minute:

- From the Activity menu, click **Adapter Transactions/Minute** to open the adapter Status page.



Using the IBM Websphere Portal

You can generate an IBM Websphere Portal Server web archive (war) file configured with references to installed OAA portal components.

To generate a war file:

- 1 From the Activity menu, click **IBM Websphere Portal Integration** to open the **Portal Integration** page.

IBM Websphere Portal Integration

An IBM Websphere Portal Server web archive configured with references to installed OAA portal components can be generated from this page. The websphere.war file found in the installed packages directory is copied and the portal.xml file within is replaced. Make sure the base URL is the correct URL for accessing pages on this server. Take the generated file and install it using the IBM WPS Portal Administration utility. Anytime new OAA applications are installed, this process should be repeated to expose any new portal components in the IBM WPS environment.

Source Path: Enter the complete source path on the server where the installed websphere.war file can be located.

Destination Path: Enter the destination path on the server where the generated websphere.war file will be created.

Base URL: Enter the base URL of this server.

Generate WAR File

- 2 Enter the following information:
 - source path
 - destination path
 - base URL
- 3 Click **Generate WAR File**.

Displaying form information

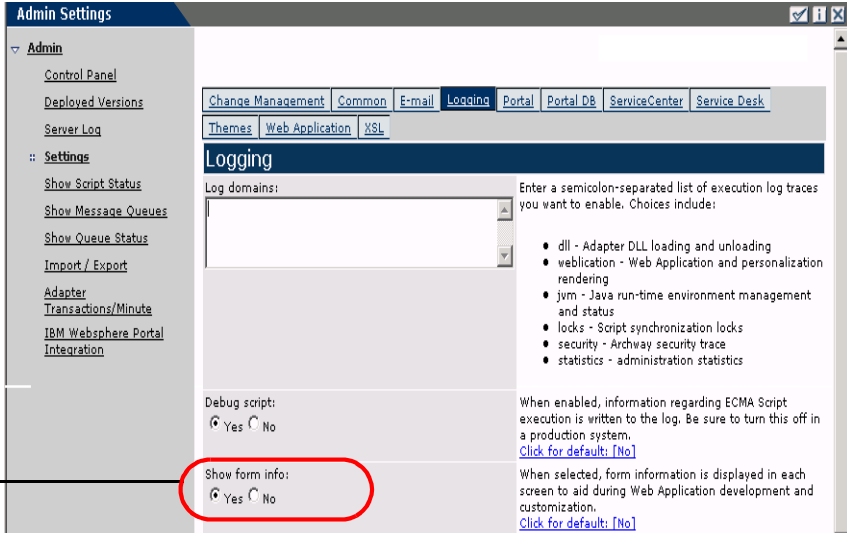
You can use the Admin module to configure Web application forms to display the location and file name of the current form.

To display form information:

- 1 From the Admin module, click **Settings**, then **Logging**.

2 Scroll to the Show form info field, and click Yes if necessary.

Set Show Form Info to Yes.



The screenshot shows the 'Admin Settings' window with the 'Logging' tab selected. The 'Show form info' field is circled in red, and a red arrow points from the text 'Set Show Form Info to Yes.' to it. The 'Show form info' field has the 'Yes' radio button selected.

3 Click Save.

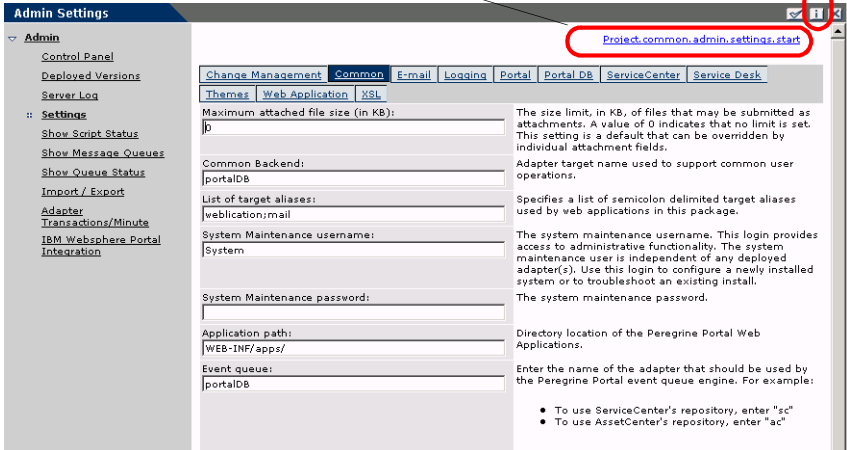
The Control Panel opens.

4 Click Reset Server.

The name of the form is at the top of each form.

The form name is at the top of the page.

Click the Display Form Info button to view the form composition.



The screenshot shows the 'Admin Settings' window with the 'Common' tab selected. The form name 'Project.common.admin.settings.start' is circled in red at the top of the page. A red arrow points from the text 'Click the Display Form Info button to view the form composition.' to a button in the top right corner.

Displaying form details

You can also display detailed information about the current form. Click the **Display Form Info** button at the top right of the form. A separate window opens.

View the contents in each tab for more information about the form.

```

Address http://hostname/oa/display_form_info.htm
Script Input Script Output User Session Log PreXSL Browser Source BackChannel Source Application Channel Source Tab Source
Menu Source Sync/Update Window Help
<?xml version="1.0" encoding="UTF-8"?>
<_doc>
  <_doExplorerView/>
  <_doExplorerModel>
    <target=portalDB/>
    <_doExplorerContext>AdminSettings</_doExplorerContext>
    <_doExplorerInstance/>
    <_doExplorerBackend>weblication</_doExplorerBackend>
    <_doExplorersSubType/>
    <_doExplorerAction>detail</_doExplorerAction>
    <_form>e_admin_settings_start.do</_form>
    <_module>common</_module>
    <_module>admin</_module>
    <_activity>settings</_activity>
    <_formname>start</_formname>
    <_return/>
    <_count>20</_count>
    <_tabs>
      <tab balloon="$$$ID$(common,configPortalLabel)" caption="$$$ID$(common,configPortalLabel)" url="e_admin_settings_start.do?target=portal"/>
      <tab balloon="$$$ID$(common,configCommonLabel)" caption="$$$ID$(common,configCommonLabel)" url="e_admin_settings_start.do?target=common"/>
      <tab balloon="$$$ID$(incidentmgt,configProblemTabLabel)" caption="$$$ID$(incidentmgt,configProblemTabLabel)" url="e_admin_settings_start.do?target=incidentmgt"/>
      <tab balloon="$$$ID$(common,configPortalDBLabel)" caption="$$$ID$(common,configPortalDBLabel)" selected="true" url="e_admin_settings_start.do?target=portalDB"/>
      <tab balloon="$$$ID$(common,configThemesLabel)" caption="$$$ID$(common,configThemesLabel)" url="e_admin_settings_start.do?target=themes"/>
      <tab balloon="$$$ID$(common,configWeblicationLabel)" caption="$$$ID$(common,configWeblicationLabel)" url="e_admin_settings_start.do?target=weblication"/>
      <tab balloon="$$$ID$(common,configLoggingLabel)" caption="$$$ID$(common,configLoggingLabel)" url="e_admin_settings_start.do?target=logging"/>
      <tab balloon="$$$ID$(common,configScLabel)" caption="$$$ID$(common,configScLabel)" url="e_admin_settings_start.do?target=sc"/>
      <tab balloon="XSL" caption="XSL" url="e_admin_settings_start.do?target=xsl"/>
      <tab balloon="$$$ID$(mailadapter,Label)" caption="$$$ID$(mailadapter,Label)" url="e_admin_settings_start.do?target=mail"/>
      <tab balloon="$$$ID$(changerequestmgt,configChangeTabLabel)" caption="$$$ID$(changerequestmgt,configChangeTabLabel)" url="e_admin_settings_start.do?target=changemgt"/>
    </_tabs>
    <_locale>en</_locale>
  </_doExplorerModel>
</_doc>

```

The form has the following tabs:

This tab	Contains
Script Input	the script that sends a request to the back-end system.
Script Output	the information returned by the script request to the back-end system.
User Session	details about the current user session, including browser type, back-end system version, and the access rights established for this user.
Log	a list of actions taken by the script to execute the form.
PreXSL	output from XSL before it gets rendered to the browser.
Browser Source	HTML source code for the current page.
BackChannel Source	HTML source code for frames where the data is stored.
Application Channel Source	HTML source code for the shared applications.

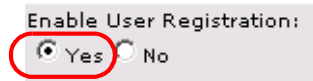
This tab	Contains
Tab Source	HTML source code for tabs.
Menu Source	HTML source code for menus.
Sync/Update Window	HTML source code to synchronize with the page and reload.
Help	Help for debugging the window.

User self-registration

With the Admin module, administrators can choose to have end users self-register for new accounts from the login screen if the user is not already in the ServiceCenter database. When the user registers, ServiceCenter creates an Operator record and a Contact record for the new user with basic user login rights. See the chapter on *Security* in this guide for more information on the registration process.

To enable users to self-register from the Login screen:

- 1 From the Admin module Settings page, click **Common**.
- 2 Scroll to **Enable User Registration**.



Click Yes to give users the ability to self-register for new accounts.

- 3 Click **Yes**.

Tip: When using an application with ServiceCenter 5.0 as the back-end system, the first name and last name are reversed in the ServiceCenter contact record from the format used in an OAA Platform application.

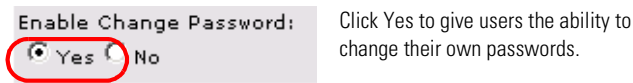
ServiceCenter 5.0 stores names in the format last name/first name. The OAA Platform stores names in the format first name/last name. As a temporary solution, you can change the way operator names are handled in ServiceCenter using the **Use Operator Full Name?** option in the Environment records for Incident and Service Managements. Refer to the *ServiceCenter 5.0 Application Administration Guide* for instructions.

Changing passwords

Using the Admin module, administrators can choose to have end users change their own passwords from the Home page.

To enable users to change passwords:

- 1 From the Admin module Settings page, click **Common**.
- 2 Scroll to **Enable Change Password**.



- 3 Click Yes.

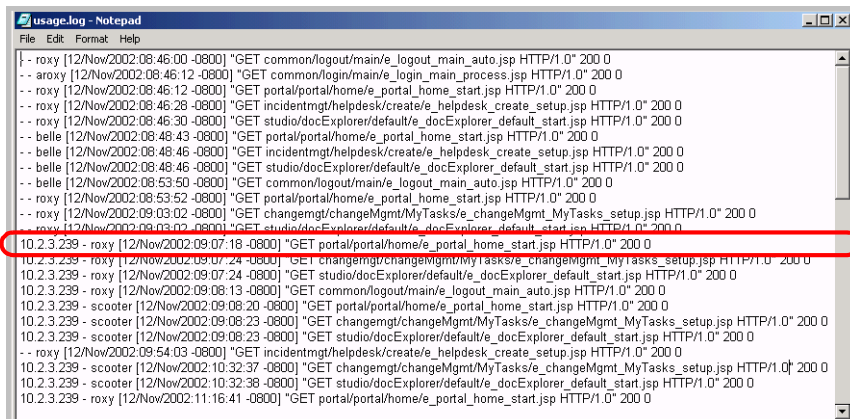
Logging and monitoring user sessions

The `usage.log` file has a record of user logins that is in the `bin` directory of your application server installation. With this file, you can determine which application is in use and how many users access an application during a day.

Understanding the `usage.log` file

The following line shows an excerpt from a `usage.log` file:

```
10.2.3.239 - roxy [12/Nov/2002:09:07:18 -0800] "GET
portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
```



```
usage.log - Notepad
File Edit Format Help
[12/Nov/2002:08:46:00 -0800] "GET common/logout/main/e_logout_main_auto.jsp HTTP/1.0" 200 0
-- aroxy [12/Nov/2002:08:46:12 -0800] "GET common/login/main/e_login_main_process.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:08:46:12 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:08:46:28 -0800] "GET incidentmgt/helpdesk/create/e_helpdesk_create_setup.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:08:46:30 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
-- belle [12/Nov/2002:08:48:43 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
-- belle [12/Nov/2002:08:48:46 -0800] "GET incidentmgt/helpdesk/create/e_helpdesk_create_setup.jsp HTTP/1.0" 200 0
-- belle [12/Nov/2002:08:48:46 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
-- belle [12/Nov/2002:08:53:50 -0800] "GET common/logout/main/e_logout_main_auto.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:08:53:52 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:09:03:02 -0800] "GET changemgt/changeMgmt/MyTasks/e_changeMgmt_MyTasks_setup.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:09:03:02 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
10.2.3.239 - roxy [12/Nov/2002:09:07:18 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
10.2.3.239 - roxy [12/Nov/2002:09:07:24 -0800] "GET changemgt/changeMgmt/MyTasks/e_changeMgmt_MyTasks_setup.jsp HTTP/1.0" 200 0
10.2.3.239 - roxy [12/Nov/2002:09:07:24 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
10.2.3.239 - roxy [12/Nov/2002:09:08:13 -0800] "GET common/logout/main/e_logout_main_auto.jsp HTTP/1.0" 200 0
10.2.3.239 - scooter [12/Nov/2002:09:08:20 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
10.2.3.239 - scooter [12/Nov/2002:09:08:23 -0800] "GET changemgt/changeMgmt/MyTasks/e_changeMgmt_MyTasks_setup.jsp HTTP/1.0" 200 0
10.2.3.239 - scooter [12/Nov/2002:09:08:23 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
-- roxy [12/Nov/2002:09:54:03 -0800] "GET incidentmgt/helpdesk/create/e_helpdesk_create_setup.jsp HTTP/1.0" 200 0
10.2.3.239 - scooter [12/Nov/2002:10:32:37 -0800] "GET changemgt/changeMgmt/MyTasks/e_changeMgmt_MyTasks_setup.jsp HTTP/1.0" 200 0
10.2.3.239 - scooter [12/Nov/2002:10:32:38 -0800] "GET studio/docExplorer/default/e_docExplorer_default_start.jsp HTTP/1.0" 200 0
10.2.3.239 - roxy [12/Nov/2002:11:16:41 -0800] "GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0" 200 0
```

Each login is on a line. Within one user session, each module logs only one line.

The following table shows the meaning of each element in the log entry:

Remote Host	Rfc931	User Login	Date	Request	Status	Bytes
10.2.3.239	-	roxy	[12/Nov/2002:09:07:18 -0800]	"GET portal/portal/home/e_portal_home_start.jsp HTTP/1.0"	200	0

This element	Contains
Remote Host	the remote host name or IP address if the DNS host name is not available or was not provided.
Rfc931	the remote login name of the user. This is always a dash because this information is not needed.
User Login	the user name authenticated to log in to the Peregrine Portal.
Date	the date and time of the request.
Request	the module accessed by the user. The name of the module is the first part of the GET parameter. In the previous above, the module accessed is <i>notificationsservices</i> , the location of the login script.
Status	the HTTP response code returned to the client. This value is always 200 to specify that it was a valid request.
Bytes	the number of bytes transferred. The number is always entered as 0, because this information is not needed.

5 Security

CHAPTER

This chapter describes the different security configuration options available in BI Portal. Topics in this chapter include:

- *BI Portal security* on page 62
- *Password encoding methods* on page 68
- *User registration* on page 69
- *Authenticating users* on page 70
- *Default security configuration* on page 71
- *Custom JAAS configuration* on page 72
- *Standard Sun Microsystems JAAS configuration* on page 81
- *Integrated Windows Authentication* on page 82
- *Integrating with single sign-on tools* on page 91
- *Contact-based authentication* on page 94
- *Creating an alternate login page* on page 99

BI Portal security

BI Portal keeps your information safe and secure using two separate security mechanisms: *document* security and *functional* security. Document groups control the documents that each user can see, and user capabilities control the functions that users can perform.

Document groups

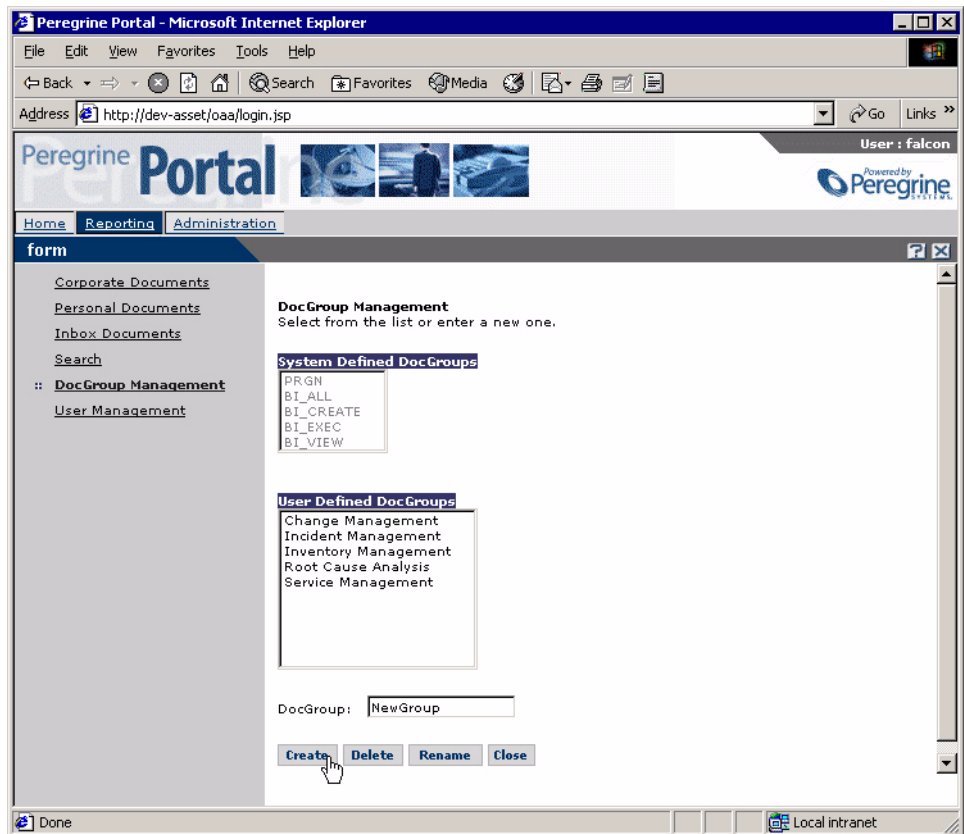
In BI Portal, each report is assigned to a *document group*. All the standard reports are assigned to pre-defined document groups. In addition, you can create additional document groups and assign reports to them. When you assign each user to one or more document groups, you control the reports that the user can execute and view.

The following table lists the reports and data that are available for querying and viewing by users assigned various groups:

This group...	Provides a view to these reports...
Change Management	All reports and data related to Change Management. See <i>Standard Reports for Change Management</i> on page 18 for more information.
Incident Management	All reports and data related to Incident Management. See <i>Standard reports for Incident Management</i> on page 19 for more information.
Inventory Management	All reports and data related to Inventory Management. See <i>Standard reports for Inventory Management</i> on page 21 for more information.
Service Management	All reports and data related to Service Management. See <i>Standard reports for Service Management</i> on page 21 for more information.

Creating document groups

You use the DocGroup Management page to create a user-defined document group:

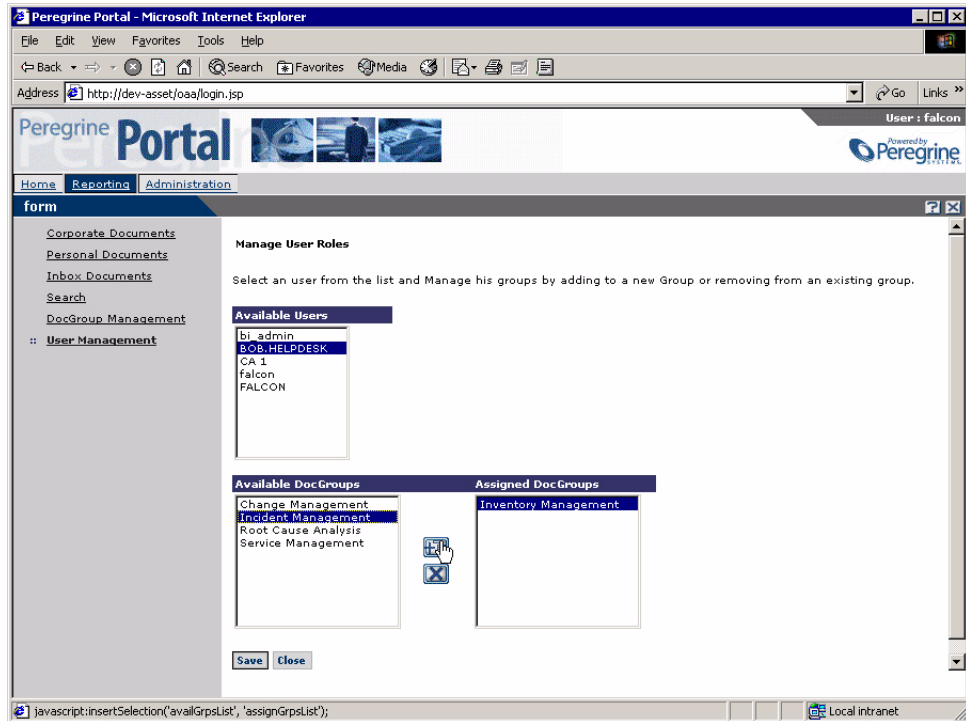


To create a document group:


- 1 Log into BI Portal as a user.
- 2 Click the **Reporting** tab.
- 3 Click **DocGroup Management** on the Navigator.
- 4 Type the name of the new document group in the DocGroup field.
- 5 Click inside the User Defined DocGroups list.
- 6 Click **Create** to add the new document group to the User Defined DocGroups list.


Assigning users to document groups

You use the Manage User Roles page to assign each user to as many document groups as you like:



To assign a user to a document group:

- 1 Log into BI Portal as a user.
- 2 Click the **Reporting** tab.
- 3 Click **User Management** in the Navigator.
- 4 Click a user in the Available Users list to highlight the user's name.
- 5 In the Available DocGroups list double-click a document group to move it to the Assigned DocGroups list. Or, click a document group in the Available DocGroups list and click the Add button  to move the document group to the Assigned DocGroups list.

- 6 To un-assign a user from a document group, double-click the document group in the Assigned DocGroups list to move it to the Available DocGroups list. Or, click the document group in the Assigned DocGroups list and click the Remove button  to move the document group to the Available DocGroups list.
- 7 Click Save to commit the document group assignments.

User capabilities

In BI Portal, you control access that users have to various reporting functions by assigning user capabilities to them.

The following table summarizes the functions that each capability allows:

This capability...	Access level	Allows the user to...
BI_Access	6 (Lowest)	<ul style="list-style-type: none"> ■ Gain access to the WebIntelligence Reporting module <p>Note: Each user needs BI_Access capability simply to access the WebIntelligence Reporting module. In addition, each user need one of the following capabilities in order to perform querying and reporting functions.</p>
BI_View	5	<ul style="list-style-type: none"> ■ Manage personal documents and categories ■ Read corporate and inbox documents ■ Run and refresh documents ■ Use and refresh list of values ■ Work in drill mode ■ Schedule documents ■ Send documents to users within and outside of the user's own group
BI_Exec	4	Perform the same functions as BI_View. However, BI_Exec capability does not have any data security control.

This capability...	Access level	Allows the user to...
BI_Create	3	<p>Perform the same functions as a user assigned BI_View capability; and:</p> <ul style="list-style-type: none">■ Download Zero Administration BusinessObjects■ Create and edit documents■ Format the toolbar■ Perform a transparent drill outside the cube■ View SQL <hr/> <p>Warning: Users who have BI_Create user capability have full access to all data in the rds universe when creating reports and ad hoc queries in the WebIntelligence Reporting module.</p> <hr/>

This capability...	Access level	Allows the user to...
BI_All	2	<p>Perform the same functions as a user assigned BI_Create capability, and:</p> <ul style="list-style-type: none"> ■ Publish corporate documents <hr/> <p>Warning: Users who have BI_All user capability have full access to all data in the rds universe when creating reports and ad hoc queries in the WebIntelligence Reporting module.</p> <hr/>
BI_Admin	1 (Highest)	<p>Perform the same functions as a user assigned BI_All capability; and:</p> <ul style="list-style-type: none"> ■ Change passwords ■ Change list display and default Web site ■ Change, view, and edit technology options ■ Open the DocGroup Management form ■ Open the User Management form <hr/> <p>Warning: Users who have BI_Admin user capability have full access to all data in the rds universe when creating reports and ad hoc queries in the WebIntelligence Reporting module.</p> <hr/>

If a user is assigned multiple capabilities, the lowest-level capability overrides other, higher-level capabilities. Therefore, assign only one capability that is appropriate to the functions that the user needs to perform.

Password encoding methods

By default, BI Portal does not encode passwords sent over the network. BI Portal sends plain text passwords to the authenticating back-end databases and stores plain text passwords in a browser cookie if the user selects to **enable automatic login**. If you want to secure your BI Portal passwords, you have three options:

- Enable Secure Sockets Layer (SSL) on your Web server
- Configure BI Portal to use a directory service such as LDAP
- Enable your Web server to use Windows NT Challenge/Response

In order to use SSL, you need to acquire a digital certificate. If your Web server has a certificate, then your BI Portal login URL must include the **https** protocol indicator. After the user browser has made a secure connection to the Web server, all data transferred is encrypted. Refer to your Web server documentation for information on configuring SSL.

BI Portal also supports authentication via a directory service such as LDAP. When you authenticate to a directory service, BI Portal passes SHA hash encoding passwords to the service. For instructions configuring a directory service see *Custom JAAS configuration* on page 72.

BI Portal also supports Integrated Windows Authentication. When this form of authentication is used, passwords are not actually exchanged between the browser and Web server, and the authentication process is kept secure. However, Integrated Windows Authentication is only supported by Internet Explorer browsers on Windows systems. For instructions configuring Integrated Windows Authentication see *Integrated Windows Authentication* on page 82.

User registration

All BI Portal users need a login account. If a user is attempting to log in for the first time, the user is prompted for certain default information as shown in the following page. The first four fields are required, as indicated by the arrows to the right of each field.

When the user clicks **Register**, the information is stored in the appropriate database.

Basic registration information and login scripts are stored in the `.../oaa/apps/common/jscript/` directory. Login scripts are in the `login.js` file. If you want to make changes to the registration process, such as changing the way a user's password is defined, you can change the scripts in this directory.

Enabling the E-mail adapter

If users have the ability to self-register, you must make sure that the E-mail tab from the BI Portal Admin module Settings page contains the MailAdapter name.

The MailAdapter is an implementation of JavaMail API 1.2 and supports the following mail protocols:

- POP3 for inbound mail

- IMAP for inbound mail
- SMTP for outbound mail

The MailAdapter also supports MIME type attachments in outbound e-mail.

Set the following parameters, as needed, on the E-mail tab of the Admin module Settings page.

Portal	Common	Portal DB	Web Application	Logging	XSL	E-mail
Inbound mail host: The full name or IP address of the machine hosting the inbound mail server. If this field is empty, then the status of the mail adapter will indicate the status of the outbound mail server connection.						
mailhost						
Inbound mail protocol: The protocol used by the inbound mail server, which is either imap or pop3.						
imap						
Inbound mail user ID: The user ID used to access the inbound mail server.						
Inbound mail password: The user password used to access the inbound mail server.						
Mail sender address: This address is used as the default sender address in outbound email messages.						
Legal domains: Enter a semicolon-separated list of mail domains that the Peregrine Portal may correspond with. Only users with an email address in these domains are allowed to complete online self-registration.						
peregrine.com;apxydev.com;getmarketaccess.com						
Anonymous user: Anonymous user name used when an unknown user attempts to communicate with the mail adapter						
falcon						
Anonymous password: Anonymous user password for the mail adapter						
Outbound mail host: The full name or IP address of the machine hosting the outbound mail server. Click for default: mailhost 						
condor.peregrine.com						
Outbound mail user ID: The user ID used to access the outbound mail server.						
Outbound mail password: The user password used to access the outbound mail server.						
Adapter: Full class path for adapter associated with this target.						
com.peregrine.oaa.adapter.mail.MailAdapter						
<input type="button" value="Save"/>						

Type the name of your MailAdapter in the Adapter field.

Troubleshooting the MailAdapter connection

You can check the status of the MailAdapter connection on the Control Panel. If the adapter shows as *disconnected*, check that the settings on the E-mail tab of the Settings page are correct. If you are still unable to connect, contact your system administrator for verification of the parameter values.

Authenticating users

You can configure the Peregrine OAA Platform to use one of five security authentication options:

- Use the default configuration to authenticate users against Peregrine adapters. See *Default security configuration* on page 71.

- Use a custom configuration to authenticate users against user-defined adapters such as LDAP or JDBC compliant databases. See *Custom JAAS configuration* on page 72.
- Use a standard JAAS configuration to authenticate users against the Sun Microsystem's standard Java Authentication and Authorization Service (JAAS). See *Standard Sun Microsystems JAAS configuration* on page 81.
- Use Integrated Windows authentication to authenticate users and pass the information to the Web application. See *Integrated Windows Authentication* on page 82.
- Use an alternate login page and authenticate users against any of the other login options. See *Creating an alternate login page* on page 99.

Once a user is authenticated, the modules to which the user has access are defined by the back-end system. If you are using ServiceCenter for the back-end system, the user must have the appropriate capability words set in the Operator record in ServiceCenter in order to see the corresponding module in the web application.

Default security configuration

The default configuration authenticates users against a set of pre-configured JAAS login modules. By default, one JAAS login module is configured for each registered Peregrine adapter. For example, if you are using both AssetCenter and ServiceCenter, then BI Portal creates login modules for *both* the ACAdapter and the SCAdapter.

These login modules are *only* used to authenticate users. User access rights are derived from user profile records in the back-end systems (for example, ServiceCenter or AssetCenter). User access rights determine which modules the user can access and what tasks they can perform within those modules. For example, one user can open tickets only, while another has rights to approve tickets as well.

You do not have to do any additional configuration to use the default security configuration. BI Portal automatically generates login modules for each Peregrine adapter installed on the system.

The default login module settings are:

- loginModule=com.peregrine.OAA.security.OAALoginModule

- control flag=OPTIONAL
- options=<none>

Custom JAAS configuration

A custom JAAS configuration authenticates users against a set of JAAS LoginModules you define in a `local.xml` file. This file contains the settings to use for each JAAS LoginModule. A `<jaas_config>` entry in `local.xml` has the following format.

```
<jaas_config>

  <jaasConfiguration>CustomConfig</jaasConfiguration>
  <CustomConfig>adapter1;adapter2</CustomConfig>

  <adapter1>
    <loginModule>Java class of login module</loginModule>
    <controlFlag>authentication behavior</controlFlag>
    <options>semicolon separated list of options</options>
  </adapter1>

  <adapter2>
    <loginModule>Java class of login module</loginModule>
    <controlFlag>authentication behavior</controlFlag>
    <options>semicolon separated list of options</options>
  </adapter2>

</jaas_config>
```

The following table describes how to use the XML tags and assign appropriate values.

Important: XML is case sensitive.

Use these XML tags	To do this
<code><jaas_config> </jaas_config></code>	Define a custom JAAS configuration. All JAAS configuration settings must be between these two tags.
<code><jaasConfiguration> </jaasConfiguration></code>	Define the name of your custom JAAS LoginModule. The value of this tag determines the tag name to use for the next tag. For example, if you create a custom configuration with the value <code>CustomConfig</code> , then you must use the tags <code><CustomConfig></code> and <code></CustomConfig></code> to define the list of adapters used.
<code><CustomConfig> </CustomConfig></code> <i>This is a user definable tag</i>	Define the list of <i>all</i> adapters that you want to use for authentication. Use semicolons between entries to specify multiple adapters. If the adapter name you list does not match a registered AdapterPool, then BI Portal assumes the name is the logical name of a non-OAA LoginModule. BI Portal attempts to authenticate users against each adapter you list. The values listed in this tag determine the tags names to use for each adapter. For example, if you create two adapters <code>adapter1</code> and <code>adapter2</code> , then you must use the tags <code><Adapter1></code> , <code></Adapter1></code> , <code><Adapter2></code> , and <code></Adapter2></code> to define your adapters.
<code><adapter1> </adapter1> <adapter2> </adapter2></code> <i>These are user definable tags</i>	Define the JAAS LoginModule settings for each adapter. Each adapter <i>must</i> have both <code><loginModule></code> and <code><controlFlag></code> tags defined for it.

Use these XML tags	To do this
<code><loginModule> </loginModule></code>	<p>Define the fully qualified class name of the JAAS LoginModule.</p> <p>This is <i>required</i> only when authenticating against non-OAA LoginModules (adapters). The default value is <code>com.peregrine.oaa.archway.security.OAALoginModule</code>.</p> <p>This is <i>optional</i> only when authenticating against Peregrine back-ends.</p>
<code><controlFlag> </controlFlag></code> This tag is <i>optional</i> .	<p>Define the authentication behavior of this LoginModule. The default value is REQUIRED.</p> <p>See <i>JAAS LoginModule control flags</i> on page 74 for a description of available options.</p>
<code><options> </options></code>	<p>Define the list of authentication options. Use semicolons between entries to specify multiple options. This is an <i>optional</i> setting for each JAAS LoginModule you use. See <i>JAAS configuration options</i> on page 77 for a description of available options.</p>

JAAS LoginModule control flags

The following table lists the possible settings for the `<controlFlag>` tag. A JAAS LoginModule can have one of four behaviors:

Control flag	Authentication behavior
REQUIRED	If the user cannot be authenticated against the adapter, the login fails. Whether it succeeds or fails, authentication continues to the next LoginModule in the list. This is the default behavior.
REQUISITE	If the user cannot be authenticated against the adapter, the login fails. If it succeeds, authentication continues to the next LoginModule in the list.

Control flag	Authentication behavior
SUFFICIENT	Authentication can proceed even if this LoginModule fails. If it succeeds, authentication does not continue to the next LoginModule in the list. If it fails, authentication continues to the next LoginModule in the list.
OPTIONAL	Authentication can proceed even if this LoginModule fails. Whether it succeeds or fails, authentication continues to the next LoginModule in the list.

Note: ControlFlag settings are case insensitive.

The overall authentication succeeds only if all Required and Requisite LoginModules succeed. If a Sufficient LoginModule is configured and succeeds, then only the Required and Requisite LoginModules prior to that Sufficient LoginModule need to have succeeded for the overall authentication to succeed. If no Required or Requisite LoginModules are configured for an application, then at least one Sufficient or Optional LoginModule must succeed.

By default, the controlFlag setting of all BI Portal Web applications LoginModules is Optional. For most enterprises, this is the desired configuration.

The following table shows some sample scenarios and how the login process works.

Module Name	Status	Scenario 1	Scenario 2	Scenario 3
LoginModule1	required	pass	pass	fail
LoginModule2	sufficient	fail	fail	fail
LoginModule3	requisite	pass	pass	pass
LoginModule4	optional	pass	fail	fail
Final Authentication		pass	pass	fail

In Scenario 1, authentication succeeds even though LoginModule2 fails. This is because the Required loginModule takes precedence over the sufficient loginModule.

In Scenario 2, authentication succeeds because the loginModules that failed are only Sufficient and Optional.

Scenario 3 authentication fails because a loginModule with a status of Required failed.

JAAS configuration options

The following tables list the possible settings for the <options> tag.

Standard JAAS Options

The following table lists the standard JAAS options available for all adapters.

Option	Use	Description
debug=true	optional	Instructs a LoginModule to output debugging information. The OAALoginModule logs debugging information to stdout and not to archway.log .
tryFirstPass=true	optional	The first LoginModule in the list saves the password entered and this password is used by subsequent LoginModules. If authentication fails, the LoginModules prompt for a new password and repeats the authentication process.
useFirstPass=true	optional	The first LoginModule in the list saves the password entered and this password is used by subsequent LoginModules. If authentication fails, LoginModules do not prompt for a new password.
storePass=true	optional	Stores the password for the user being authenticated.
clearPass=true	optional	Clears the password for the user being authenticated.

Peregrine JndiLoginModule options

The following table lists the options available to custom JAAS LoginModules using the Peregrine JndiLoginModule.

Note: The Peregrine JAAS LoginModule `com.peregrine.oaa.security.JndiLoginModule` is modeled after Sun's `JndiLoginModule`. The main difference is that an RFC 2307 (NIS over LDAP) compliant schema is not required. User must have “uid” and “userPassword” properties defined.

Option	Use	Description
<code>user.provider.url</code>	required	<p>Use this option to provide the URL to the starting point in your directory service where you want to search for users.</p> <p>For example, <code>ldap://server/dc=peregrine,dc=com</code></p> <p>Note: This option corresponds to the Java constant <code>Context.PROVIDER_URL</code>.</p>
<code>security.principal</code>	optional	<p>Use this option to specify which directory service user you want to use to authenticate non-anonymous queries of your directory service. Use the DN of the directory service user. For example, <code>uid=user,dc=peregrine,dc=com</code></p> <p>Tip: To prevent user passwords from being visible to users, you should only set this option if you are using a directory server such as IPlanet where user passwords are SHA hashed by default.</p> <p>Note: This option corresponds to the Java constant <code>Context.SECURITY_PRINCIPAL</code>.</p>

Option	Use	Description
security.credentials	optional	<p>Use this option to define the password for the <code>security.principal</code> user. This option should only be used in conjunction with the <code>security.principal</code> option.</p> <hr/> <p>Important: If you are using a simple security authentication protocol, then this password may be passed as plain text.</p> <hr/> <p>Tip: To safeguard this password, either enable SSL (set the <code>security.protocol=ssl</code> option) or use an <code>security.authentication</code> that protects passwords.</p> <p>Note: This option corresponds to the Java constant <code>Context.SECURITY_CREDENTIALS</code>.</p>
security.protocol	optional	<p>Use this option to enable or disable an SSL connection between the <code>JndiLoginModule</code> and your directory server. This option has two possible values:</p> <ul style="list-style-type: none"> ■ <code>simple</code> (Default setting) ■ <code>ssl</code> <p>Note: This option corresponds to the Java constant <code>Context.SECURITY_PROTOCOL</code></p>
security.authentication	optional	<p>Use this option to enable or disable anonymous binding to your directory service. Typically, this option has one of two values:</p> <ul style="list-style-type: none"> ■ <code>none</code> (Default setting) ■ <code>simple</code> <p>Note: If you do not specify a value for <code>security.principal</code> then <code>security.authentication</code> defaults to a value of <code>none</code>. Likewise, if you set <code>security.authentication</code> to <code>simple</code> but <code>security.credentials</code> is omitted or has zero length, then <code>security.authentication</code> resets to <code>none</code>.</p> <p>Note: This option corresponds to the Java constant <code>Context.SECURITY_AUTHENTICATION</code>.</p>

Option	Use	Description
<code>user.search.scope</code>	optional	<p>Use this option to specify the number of levels to descend when searching for the user being authenticated by <code>user.provider.url</code>. This value must be an integer. The default value is 1.</p> <p>Note: This option corresponds to the Java constant <code>SearchControls.ONELEVEL_SCOPE</code>.</p>
<code>group.provider.url</code>	optional	<p>Use this option to provide the URL to the starting point in your directory service where you want to search for groups.</p> <p>For example, <code>ldap://server/dc=peregrine,dc=com</code></p> <p>Note: This option corresponds to the Java constant <code>Context.PROVIDER_URL</code>.</p>
<code>group.search.scope</code>	optional	<p>Use this option to specify the number of levels to descend when searching for a group. This option should only be used with <code>group.provider.url</code>. This value must be an integer. The default value is 1.</p> <p>Note: This option corresponds to the Java constant <code>SearchControls.ONELEVEL_SCOPE</code>.</p>
<code>group.search.objectClass</code>	optional	<p>Use this option to specify the name of the LDAP group objectClass. Valid values are:</p> <ul style="list-style-type: none"> ■ <code>groupOfNames</code> (Default value) ■ <code>groupOfUniqueNames</code>. ■ <code>groupOfUrls</code> <p>Note: Either <code>groupOfNames</code> or <code>groupOfUniqueNames</code> can be used to define static groups in LDAP, but they may not be used together.</p> <p>If you choose the <code>groupOfUrls</code> option, then you are configuring dynamic groups. No additional configuration settings are required to recognize dynamic groups.</p>
<code>storeIdentity=true</code>	optional	<p>Use this option to store a reference to the User being authenticated.</p>
<code>clearIdentity=true</code>	optional	<p>Use this option to clear a reference to the User being authenticated.</p>

Example: Defining an LDAP custom configuration

The following XML code is an example of how to define a loginModule to authenticate users against an LDAP directory service.

Note: LDAP is not an adapter and does not imply any other functionality.

```
<jaas_config>
  <jaasConfiguration>myConfig</jaasConfiguration>
    <myConfig>ldap</myConfig>

    <ldap>
      <loginModule>
        com.peregrine.oaa.security.JndiLoginModule
      </loginModule>
      <options>
        user.provider.url=ldap://server/dc=peregrine,dc=com;
        group.provider.url=ldap://server/dc=peregrine,dc=com
      </options>
    </ldap>
</jaas_config>
```

Standard Sun Microsystems JAAS configuration

The standard JAAS configuration option authenticates users against the Sun Microsystems formatted JAAS configuration. To enable the standard JAAS configuration, you must edit the local.xml file and add the following lines:

```
<jaas_config>
  <useStandardJAASConfiguration>true</useStandardJAASConfiguration>
</jaas_config>
```

If you choose to use the standard JAAS configuration, then you must also do one of the following two things:

- Specify the appropriate JAAS command line options when the container is started
- or–
- Configure the java.security file in \$JAVA_HOME/jre/lib/security for JAAS.

Command line options

The command line properties required for use of the standard file-based configuration are as follows:

```
java -classpath <list of jars> \  
-Djava.security.manager \  
-Djava.security.policy==java2.policy \  
-Djava.security.auth.policy==jaas.policy \  
-Djava.security.auth.login.config==jaas.config \  
<MyMainClass>
```

For <list of jars>, enter the list of jars used by your JAAS-enabled Java application.

For <MyMainClass>, enter the fully qualified class name of the Java main program class.

Integrated Windows Authentication

Windows Integrated Authentication (known as NT/Challenge Response in previous versions of Windows) is one of the ways Windows facilitates the authentication of users on a Web server. The process consists of a secure handshake between Internet Explorer (IE) and the Internet Information Server (IIS) Web server. The handshake lets the Web server know exactly who the user is, based on how they logged in to their workstation. This allows the Web server to restrict access to files or applications based on who the user is. Applications running on the Web server can use this information to identify users without requiring them to log in.

BI Portal uses Integrated Windows Authentication as follows:

- The user logs in to a Windows XP/2000/NT workstation.
- The user starts the IE browser and navigates to the `login.asp` page.
- IE automatically sends user authentication information to IIS. The user's password is not transferred, but the Integrated Windows Authentication handshake between IE and IIS is enough for the server to recognize the user.
- The Web application login automatically detects the user by using the Integrated Windows Authentication/IIS server data.
- The user is logged in without requiring that a name and password be entered.

During this process, the back-end database authenticates and impersonates the Windows user with each of its adapters.

The following circumstances are exceptions to the normal Integrated Windows Authentication login process:

- The Windows user has already registered with a back-end database adapter. When this occurs, BI Portal asks the user to register and enter profile information. BI Portal then lets the user log in and stores this information for future login attempts.
- The Windows user name is not already registered as an Administrator in the back-end system. When this occurs, the web application does not proceed with automatic login. The user is presented with another login screen and is asked to verify their password. This step is an added security measure to prevent a user from accidentally logging in with administrative rights.

Setting up Integrated Windows Authentication

This section describes how to configure BI Portal to use IIS for Integrated Windows Authentication while using Apache as the primary Web server. You can also follow these instructions if you use IIS as your primary Web server.

It is a seven-step process:

- Step 1** Verify that all users have an Operator record in the appropriate back-end database. See *Creating an Operator record* on page 84.
- Step 2** Install and configure BI Portal with Apache and Tomcat. See *Preparing to configure Integrated Windows Authentication* on page 84.

Note: Tomcat/Apache is the preferred configuration for BI Portal.
- Step 3** Set Web server properties for the `login.asp` file. See *Setting Web server properties for the login.asp file* on page 84.
- Step 4** Set Web server properties for the `e_login_main_start.asp` file. See *Setting Web server properties for the e_login_main_start.asp file* on page 86.
- Step 5** Set Web server properties for the `loginverify.asp` file. See *Setting Web server properties for the loginverify.asp file* on page 89.

- Step 6** Set the **Require Windows NT Challenge/Response Authentication** parameter, and optionally the **Default User Login Name** and **Default Login User Password** parameters from the BI Portal administration page. See *Setting the Admin parameters* on page 90.
- Step 7** Optionally, define the **LogoutURL** from the BI Portal administration page. This step is necessary when BI Portal and IIS reside on different servers. See *Setting up the LogoutURL* on page 91.

The following procedures illustrate how to setup Integrated Windows Authentication using Windows 2000 as an example. If you are using Windows XP, the overall procedure is the same. The IIS Management Console is called Internet Information Services.

Creating an Operator record

All users must have a back-end database Operator record. Contact your AssetCenter or ServiceCenter administrator to verify that users have Operator records. Create an Operator record as needed.

Preparing to configure Integrated Windows Authentication

Note: If you are not using the preferred Tomcat/Apache configuration, skip this section.

- 1 Install and configure BI Portal with Apache and Tomcat, and verify that you can log in through `login.jsp`.
- 2 On a server running IIS, create a virtual directory named `oaa`.
This virtual directory must have read access and permission to run scripts.
- 3 From the BI Portal deployment directory, copy the following files to the `oaa` virtual directory on the IIS server:
 - `login.asp`
 - `loginverify.asp`
 - `e_login_main_start.asp`

The default BI Portal deployment directory is:

`C:\Program Files\Peregrine\Common\Tomcat4\webapps\oaa`

Setting Web server properties for the login.asp file

Note: If you are using IIS for your Web server, go directly to step 3.

- 1 On the IIS server, edit `login.asp` using a text editor.

Edit <FORM... action...> and change it from login.jsp to the absolute URL of login.jsp on the Apache server.

For example, change from:

```
<FORM name="f" action="login.jsp" method="post">
```

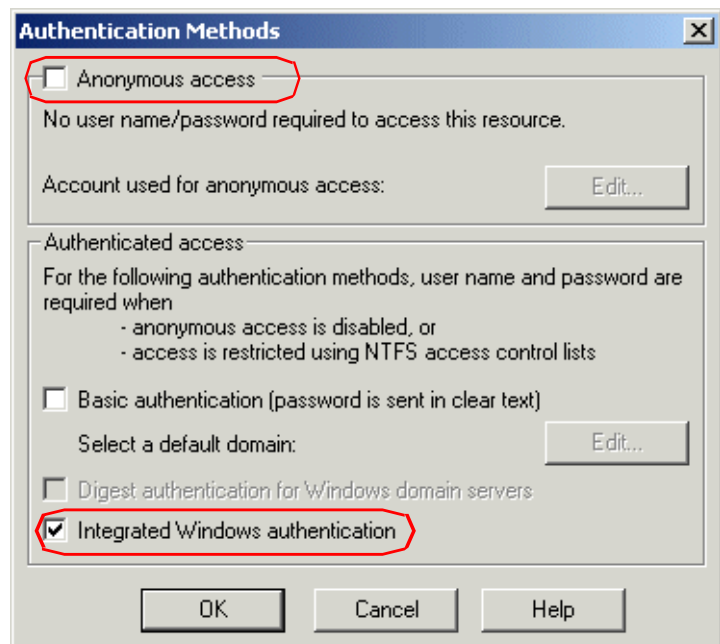
to:

```
<FORM name="f" action="
"http://<apacheserver.mycompany.com>/oaa/login.jsp" method="post">
```

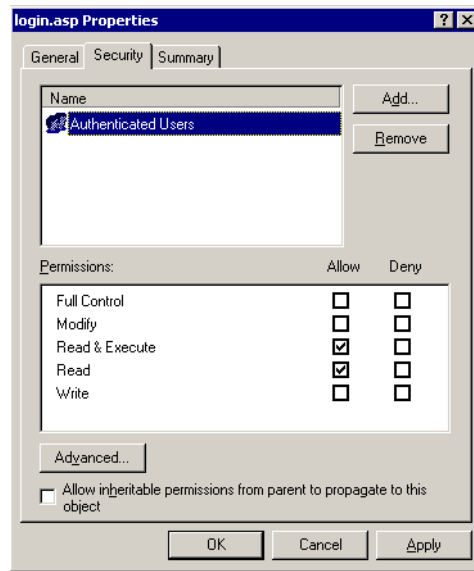
- 2 Open the IIS Management Console (**Start>Programs>Administrative Tools>Internet Information Services**).
- 3 Click on the oaa virtual directory.
- 4 Right-click on login.asp and select **Properties**.
- 5 Select the **File Security** tab.
- 6 Click **Edit** in the **Anonymous Access and Authentication Control** section and set the permissions as follows:
 - a Disable **Anonymous access**.
 - b Require **Integrated Windows authentication**.

Clear the Anonymous access check box.

Select the Integrated Windows authentication check box.



- 7 Click OK on all windows displayed until you return to the Microsoft Management Console.
- 8 From Windows Explorer, update the following properties to **login.asp**.
 - a Add the **Authenticated Users** group to the list of authorized users.
 - b Grant the following **Permissions** to the Authenticated Users group:
 - **Read & Execute** – Allow
 - **Read** – Allow



Setting Web server properties for the e_login_main_start.asp file

Note: If you are using IIS for your Web server, go directly to step 3.

- 1 On the IIS server, edit e_login_main_start.asp using a text editor. Edit <FORM... action...> and change it from e_login_main_start.do to the absolute URL of e_login_main_start.do on the Apache server.

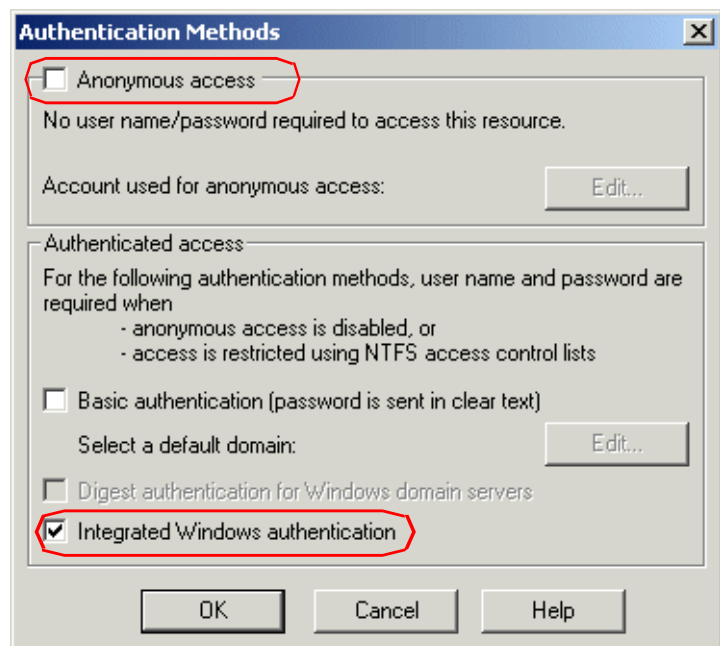
For example, change from:

```
<FORM name="f" action="e_login_main_start.do" method="post">
to:
```

```
<FORM name="f" action="http://<apacheserver.mycompany.com>
/0aa/e_login_main_start.do" method="post">
```

- 2 Open the IIS Management Console (Start>Programs>Administrative Tools>Internet Information Services).
- 3 Click on the oaa virtual directory.
- 4 Right-click on e_login_main_start.asp and select **Properties**.
- 5 Select the **File Security** tab.
- 6 Click **Edit** in the **Anonymous Access and Authentication Control** section and set the permissions as follows:
 - a Disable **Anonymous** access.
 - b Require **Integrated Windows** authentication.

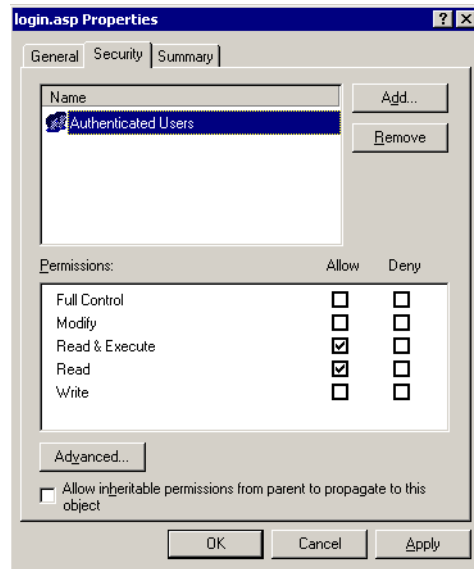
Clear the Anonymous access check box.



Select the Integrated Windows authentication check box.

- 7 Click **OK** on all windows displayed until you return to the Microsoft Management Console.
- 8 From Windows Explorer, update the following properties to e_login_main_start.asp.
 - a Add the **Authenticated Users** group to the list of authorized users.
 - b Grant the following **Permissions** to the **Authenticated Users** group:
 - **Read & Execute** – Allow

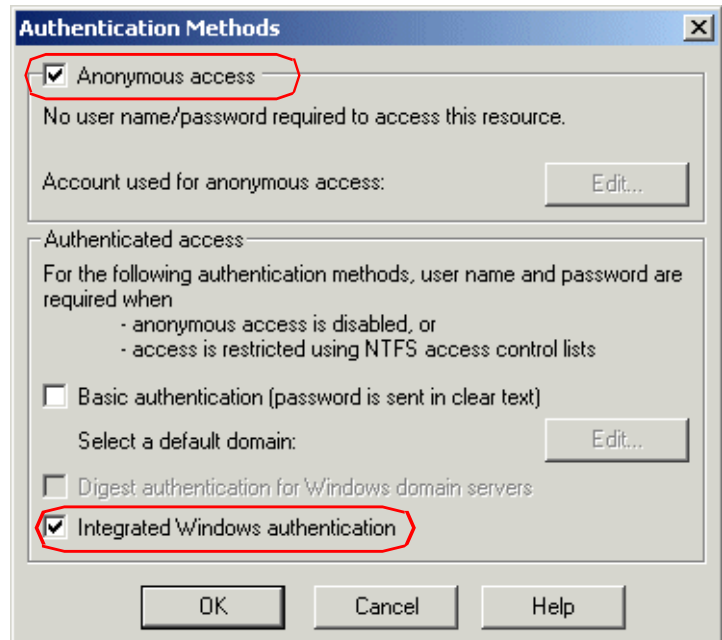
■ Read – Allow



Setting Web server properties for the loginverify.asp file

- 1 Open the IIS Management Console (Start>Programs>Administrative Tools>Internet Information Services).
- 2 Click on the oaa virtual directory.
- 3 Right-click on loginverify.asp and select **Properties**.
- 4 Select the **File Security** tab.
- 5 Click **Edit** in the **Anonymous Access and Authentication Control** section.

Select the Anonymous access check box.



Select the Integrated Windows authentication check box.

- 6 Verify that **Anonymous access** and **Integrated Windows authentication** have a check.
- 7 Click **OK** on all windows displayed until you return to the Microsoft Management Console.
- 8 Close the Management Console.

Setting the Admin parameters

You must set the **Require Windows NT Challenge/Response Authentication** parameter to Yes if you want only users who have a Windows account to log in. Users without Windows authentication can still have login capabilities by assigning a Default Login User Name.

Warning: The default login user has whatever capabilities you assign in the ServiceCenter or AssetCenter back-end. When you enable this feature, anyone can log in. Assign minimal user rights to this user.

To set Windows NT Challenge/Response Authentication:

- 1 Open a Web browser.
- 2 Enter the following URL: `http://<webserver>/<oaa>/admin.jsp` in the browser address field (where `<webserver>` is the name of your Web server and `<oaa>` is the name of the virtual directory created during installation).
- 3 Login using the administrator name and password.
- 4 From the Administration Home page, click **Settings**.

Select the Yes option in Require Windows NT Challenge/Response Authentication to allow only Windows users to log in.

The screenshot shows the 'Admin Settings' window with a sidebar on the left containing navigation links: Admin, Control Panel, Deployed Versions, Server Log, Settings (selected), Show Script Status, Show Message Queue, Show Queue Status, Import / Export, Adapter, Transactions/Minute, IBM Websphere Portal Integration. The main content area is a table with two columns: a parameter name and its description. The 'Require Windows NT Challenge/Response Authentication' row is highlighted with a red circle, and its radio button is set to 'Yes'. Other visible parameters include Logout URL, Server URL, Message URL Prefix, Help URL Prefix, Loginverify.asp URL prefix, Destination URL, Default Login User Name (Harko), and Default Login User Password.

Logout URL:	Destination URL displayed when a user logs off of system.
Server URL:	Link back to server. Replace localhost with the server name. Applications may append at run time the targeted jsp page.
Message URL Prefix:	Defines the http domain used to build URL references with coded messages.
Help URL Prefix:	Defines the URL Prefix used to access help forms. When set, a help URL is generated in the form {HelpURLPrefix}?form={module}_{activity}_{form}&language={userlocale}, where {HelpURLPrefix} is a configuration property. If {HelpURLPrefix} is undefined, then the URL is generated in the form help/{userlocale}/e_{module}_{activity}_{form}.html.
Loginverify.asp URL prefix:	Enter URL prefix for loginverify.asp if IIS is on a different server than OAA. For example: http://iisserver.mycompany.com/oaaf
Require Windows NT Challenge/Response Authentication:	Set to true to allow only users who are preauthenticated by Windows to log in. You must configure NT Challenge/Response authentication as described in the setup guide before enabling this option. Set this together with the Logout URL option. Click for default.
Default Login User Name:	Optional default user name, having minimal privileges, to use to log into back-end systems when the initial NT Challenge/Response login attempt fails. Note: Enabling this setting could pose a security threat; assign a default user with very minimal access rights to backend systems. Click for default.
Default Login User Password:	The password for the Default Login User Name. It is not necessary to provide a password for the Default Login User Name unless the user name has been assigned an actual password to the backend systems.

- 5 From the **Common** tab, set the **Require Windows NT Challenge/Response Authentication** parameter to Yes.
- 6 To allow users without Windows authentication to login, assign a Default Login User Name, and optionally a password.
- 7 Click **Save**, then click **Reset Server**.

Setting up the LogoutURL

Note: This step is necessary when BI Portal and IIS reside on different servers.

- 1 From the Administration home page (see *To set Windows NT Challenge/Response Authentication*: on page 90), click **Settings**.
- 2 From the **Common** tab, set the **LogoutURL** setting to the URL you want users to go to if Integrated Windows Authentication fails or is not possible due to the user's current browser.
- 3 Click **Save**, then click **Reset Server**.

Testing the settings

Log in to your Peregrine Web application to make sure the access permissions are set correctly. The Integrated Windows Authentication settings are activated when you log in through a special login page named `login.asp`. Accessing your applications through the standard `login.jsp` page results in the users needing to log on as usual.

To test the settings:

- 1 Open a Web browser.
- 2 Enter the following URL: `http://<webserver>/<oaa>/login.asp` in the browser address field (where `<webserver>` is the name of your Web server and `<oaa>` is the name of the virtual directory created during installation).
- 3 Verify that access to BI Portal is what you expected based on the settings you chose for the `login.asp` and `loginverify.asp` files.

Integrating with single sign-on tools

You can integrate BI Portal with a single sign-on tool such as SiteMinder to eliminate displaying the BI Portal login screen. When you integrate with a single sign-on tool, BI Portal users browse to a special URL that obtains their user information from the sign-on tool and then automatically logs them in if the sign-on tool validates them. The following steps are for integrating BI Portal with a third-party single sign-on tool. If you want to use Integrated Windows Authentication as your single sign-on tool, refer to *Integrated Windows Authentication* on page 82.

To integrate with a single sign-on tool:

- 1 Choose or create one user record for each single sign-on user you want to access BI Portal. Each user record must have a password and a list of capability words or user rights.

Important: The back-end database user record is required to determine what portions of the BI Portal interface the user can access.

- 2 Open a text editor such as NotePad.
- 3 Create a new JSP file to be the target of your automatic login URL.

You can use the following code as a template:

<p>Add JSP code here to obtain the user name of the person that the single sign-on tool has authenticated _____</p>	<pre><%@ include file="jspheader.jsp" %> <% // Add JSP code that obtains proper user name from // the third party single-sign on tool // ... // Replace "user" with the user name obtained above String sUser = "user"; // Turn on OAA pre-authentication user.setPreAuthenticated(true); %></pre>
<p>Replace the value _____ "user" with the user name obtained from your single sign-on tool</p>	<pre><HTML> <BODY> <FORM name="f" action="login.jsp" method="post"> <INPUT type="hidden" name="loginuser" value="<%=sUser%>" /> </FORM> </BODY> </HTML> <SCRIPT LANGUAGE="JavaScript"> self.document.forms[0].submit() </SCRIPT></pre>

- 4 Add any necessary JSP code to query your single sign-on tool for the name of the user who has been pre-authenticated.

Typically, these tools use HTTP headers to submit this information. See your single sign-on tool API documentation for details.

- 5 Save the file as `autologin.jsp` in your application server's presentation folder. For example:

C:\Program Files\Peregrine\Common\Tomcat4\webapps\oaa\autologin.jsp

Note: The name you choose for the JSP file will be the file name required in the URL.

Testing access to BI Portal from a single sign-on tool

You can use the following steps to test access to BI Portal from your single sign-on tool.

To test your single sign-on settings:

- 1 Login to your single sign-on tool.
- 2 Open a browser and go to the following URL:

`http://<server_name>/oaa/autologin.jsp`

If you configured the login settings correctly you will be authenticated and redirected automatically to the BI Portal home page.

Note: If you saved the automatic login page with a different file name, then use that file name instead of `autologin.jsp`.

Contact-based authentication

The following method describes an alternate authentication scheme that automatically verifies Windows users as ServiceCenter contacts.

You can configure BI Portal to automatically log in specific groups of authenticated Windows users as one or more pre-defined Operators in ServiceCenter. Each group of Windows users has its own login page.

- Step 1** Choose or create one Operator record in ServiceCenter for each group of Windows users you want to authenticate. See *Creating an Operator record in ServiceCenter* on page 94.
- Step 2** From the Windows domain server, add a Windows group for each Operator that you defined in step 1. Refer to your Windows documentation for more information on adding groups. See *Adding groups* on page 95.
- Step 3** Create a login ASP file for each Operator defined in step 1. See *Configuring the login ASP file* on page 95.
- Step 4** Configure each login ASP file to be exclusive to each Windows group defined in step 2. See *Setting properties for the login ASP file* on page 96.
- Step 5** Edit local.xml in <application server>\oaa\WEB-INF to define the passwords for each Operator defined in step 2. See *Editing the local.xml file* on page 98.

Creating an Operator record in ServiceCenter

Choose or create one Operator record for each group of users or role you want to access BI Portal. Each Operator must have a password and a list of capability words. For example, you can define one Operator with default access (**scdefault**) and one Operator with manager access (**scmgr**). Refer to your ServiceCenter documentation for more information on adding Operator records.

The following procedures describe how to use **scdefault** and **scmgr** as the Operators.

Using Operator records in ServiceCenter:

- 1** Create two Operator records: **scdefault** and **scmgr**.
Refer to your ServiceCenter documentation for information on adding Operator records.

- 2 Add the BI Portal capability words you want users assigned to this Operator to have. For example:

Operator	Capability words
scdefault	getit.service getit.personalization.default
scmgr	getit.service getit.employee getit.itmanager getit.personalization.default

Note: Each Operator will use its own login page.

In this example, users who log in to `logindefault.asp` have the capabilities of the `scdefault` Operator in ServiceCenter. Users who log in to `loginmgr.asp` have the capabilities of the `scmgr` Operator in ServiceCenter.

- 3 Assign a password to each Operator.

Note: The password must match the password defined in *Editing the local.xml file* on page 98.

Adding groups

You must have an equivalent Windows group for each Operator that you want to authenticate. For example:

Operator	Suggested group
scdefault	Authenticated Users (default Windows group)
scmgr	Managers (created on domain server)

Refer to your Windows documentation for adding groups to Windows.

Configuring the login ASP file

You must configure or create a separate login ASP file for each Operator you define (see *Creating an Operator record in ServiceCenter* on page 94). Each file needs a unique name.

Two sample login ASP files, `logindefault.asp` and `loginmgr.asp` are in the BI Portal deployment directory: `<application server>\oaa`

To configure the login ASP file:

- 1 Create a unique login file for each Operator.
For example, create `logindefault.asp` for `scdefault` and create `loginmgr.asp` for `scmgr`.
 - a Copy `logindefault.asp` from the deployment folder:
<application server>\oaa
 - b Paste the file in the same folder and rename the copied file.

Note: Whatever file name you choose becomes part of the URL users enter to access BI Portal. For example, if the file name is `mylogin.asp`, the URL is: `http://yourhostname/oaa/mylogin.asp`.
- 2 Edit the value of the OPERATOR form input to match the Operator you defined in *Creating an Operator record in ServiceCenter* on page 94.

```

...
<FORM name="f" action="login.jsp" method="post">
  <INPUT type="hidden" name="AUTH_TYPE" value="<%=sType%>" />
  <INPUT type="hidden" name="AUTH_USER" value="<%=sUser%>" />
  <INPUT type="hidden" name="AUTH_KEY" value="<%=sKey%>" />
  <INPUT type="hidden" name="OPERATOR" value="scdefault" />
</FORM>
...

```

The value of the OPERATOR must match the Operator name.

- 3 Save and close the file.

Setting properties for the login ASP file

You must configure each login ASP file to be exclusive to each Windows group. This requires changing the authentication method in IIS and setting the file security properties in Windows.

To change the authentication method in IIS:

- 1 Open the IIS Management Console (Start > Programs > Administrative Tools > Internet Information Services).
- 2 Navigate to the `oaa` virtual directory.
- 3 For each Operator, navigate to the ASP file that you created in *Configuring the login ASP file* on page 95.
For example, navigate to `logindefault.asp` for `scdefault`; navigate to `loginmgr.asp` for `scmgr`.

- 4 Right-click on the file and select **Properties**.
- 5 Select the **File Security** tab.
- 6 Click **Edit** in the **Anonymous Access and Authentication Control** section and set the permissions as follows:
 - a Disable **Anonymous** access.
 - b Require **Integrated Windows** authentication.

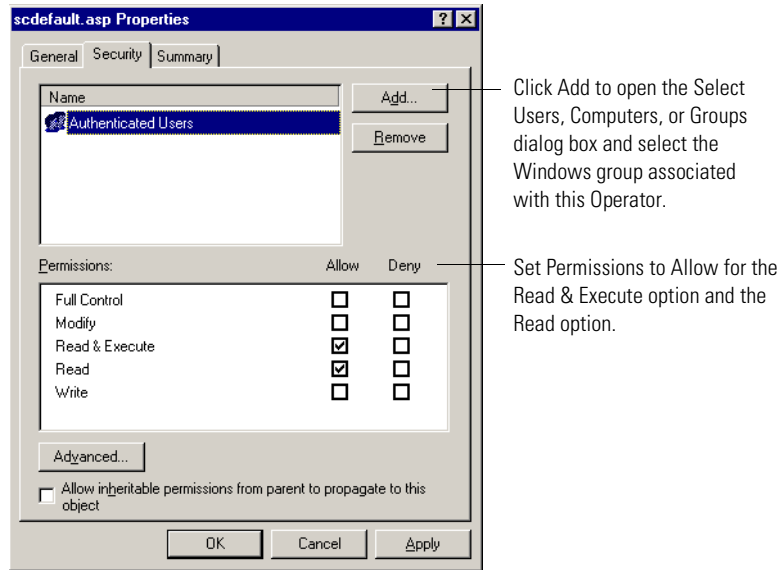


- 7 Click **OK** on all windows displayed until you return to the Microsoft Management Console.

To set the file security properties in Windows:

- 1 Open Windows Explorer.
- 2 Browse to your deployment folder: <application server>\oaa
- 3 Update the following login ASP properties.
 - a Right-click on your login ASP file; for example, scdefault.asp, and click **Properties**.

- b Add the user group associated with this Operator; for example, **Authenticated Users**.



- c Grant the following **Permissions** to the **Authenticated Users** group:
- Read & Execute – Allow
 - Read – Allow
- d Click **OK**.
- 4 Repeat step 3 for each login ASP file.

Editing the local.xml file

You must identify the password for each Operator that you defined in the local.xml file. This file is located at:

<application server>\oaa\WEB-INF\local.xml.

To edit the local.xml file:

- 1 Using a text editor, edit local.xml.
The default location is:
C:\Program Files\Peregrine\Common\Tomcat4\webapps\oaa\WEB-INF.
- 2 Add an XML entry for each Operator.

The tag has the format: <[operator name]password>

For example, for `scmgr` and `scdefault`, add the following inside the <settings> ... </settings> tags:

```
<scmgrPassword>password1</scmgrPassword>
<scdefaultPassword>password2</scdefaultPassword>
```

Important: The password must match the Operator password in ServiceCenter.

- 3 Restart your application server for your changes to take effect.

Creating an alternate login page

If you do not want to use the default Peregrine OAA login page, you can create your own login page that authenticates users and redirects them to the proper start page. Creating an alternate login page requires two basic steps:

- Step 1** Create a login Web page with the necessary authentication parameters. See the following section, *Creating a login Web page*.
- Step 2** Edit the `archway.xml` to specify the HTTP authentication method you want to use. See *Specifying an alternate authentication method* on page 101.

Creating a login Web page

Your custom login web page can be any HTML form that prompts for the following required parameters:

- Username
- Password

In addition, you can include optional login parameters such as:

- Display Language and Locale
- Time format
- Theme

A sample HTML login form, `login_sample.htm` is in the OAA deployment folder of your application server:

```
<application server>\WEB-INF\oaa\
```

Customize this sample HTML form using the following guidelines:

- Whatever custom login file you create becomes part of your login URL. For example, if you create a custom page called `my_login.htm`, then the login URL is `http://<server>:<port>/oaa/my_login.htm`
- You must specify the `basicauth` servlet in the form action. For example, `action="http://<server>:<port>/oaa/servlet/basicauth"`
- Users who have the `getit.portal` capability word see the `e_portal_home_start.do` home page if successfully authenticated
- Users who do *not* have the `getit.portal` capability word see the `e_home_main_start.do` home page if successfully authenticated
- Users who fail to be successfully authenticated see the page specified in the `_failURL` value
- The `basicauth` servlet does not encrypt usernames and passwords during login. You must enable HTTPS if you are concerned about password security on your intranet.
- If no URL is specified in the form action, then authenticated users are redirected to `http://<server>:<port>/oaa/login.jsp` where they see either `e_portal_home_start.do` or `e_home_main_start.do` depending on their login capability words.
- There are no specific Administration page settings needed to set up a custom login page. You must define all login parameters in your custom login page.
- If you want to display a specific OAA page at login, you can specify the page in the form action URL. For example, the value `action="http://<server>:<port>/oaa/servlet/basicauth/e_home_main_start.do"` displays the non-portal version of the Peregrine OAA home page.
- The following login parameters are available:

Login parameters	Description
<code>loginuser</code>	This is a required login parameter specifying the user name. You must specify a form input for this parameter.
<code>loginpass</code>	This is a required login parameter specifying the login password. You must specify a form input for this parameter.
<code>_locale</code>	This is an optional login parameter specifying the user's locale and regional display settings.

Login parameters	Description
<code>_timezone</code>	This is an optional login parameter specifying the user's timezone.
<code>_theme</code>	This is an optional login parameter specifying which theme should be displayed in the Peregrine OAA Portal

Specifying an alternate authentication method

By default, Peregrine OAA uses HTTP basic authentication provided by the `HttpBasicAuthenticationManager` class. If you create a custom login page, you need to specify the alternate authentication method in the `archway.xml` file.

To specify an alternate HTTP authentication method:

- 1 Stop your application server.
- 2 Using a text editor, open the `archway.xml` file located at:

```
<application server>\webapps\oaa\WEB-INF\default.
```

- 3 Edit the line containing:

```
<httpauthclass ...>HttpBasicAuthenticationManager</httpauthclass>
```

- 4 Change the value `HttpBasicAuthenticationManager` to `HttpAlternateAuthenticationManager`.
- 5 Save the file.
- 6 Modify the `web.xml` file.

You will need to enable the `AuthController` servlet to establish a proxy for HTTP basic authentication.

- a Using a text editor, open the `web.xml` file located at:

```
<application server>\webapps\oaa\WEB-INF.
```

b Add the following lines at the end of the last `<servlet>` definition:

```
<servlet>
  <servlet-name>AuthController</servlet-name>
  <display-name>AuthController</display-name>
  <description>A controller (decorator) servlet that can be used to enable
  configurable auth protection of any resource.</description>

  <servlet-class>com.peregrine.oaa.archway.AuthControllerServlet
</servlet-class>
  <load-on-startup>2</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>AuthController</servlet-name>
  <url-pattern>/servlet/basicauth/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>AuthController</servlet-name>
  <url-pattern>/servlet/auth/*</url-pattern>
</servlet-mapping>
```

c Save the file.

7 Restart your application server.

Warning: Changing the HTTP authentication setting to the Alternate Authentication Manager exposes queries (including login names and passwords) in the URL. If you want to protect URL queries, then you must restrict access to this information through your Web server.

6 RDS Universe Administration

CHAPTER

This chapter describes the structure of the rds universe in BI Portal, including its hierarchy and the types of objects it contains. The chapter includes exercises for designing and building reports.

Understanding the structure of the rds universe

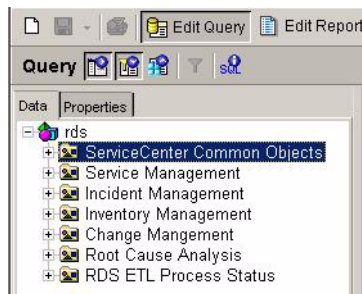
RDS elements

There are four basic elements that you will see as you navigate the RDS structure, each indicated by the icon to the right of the object name:

- Folders — Folders are general grouping of data fields. They don't map to anything in the database. They simply allow us to organize fields into meaningful groups. Folders are indicated by a file folder icon.
- Dimensions — Dimensions are categorical data (names, locations, descriptions). This data is typically used to create report sections or show detail. Dimensions are indicated by a blue cube icon.
- Measures — Measures are numeric data (counts, summaries, finances, elapsed times). These can be used in the same manner as dimensions, or can be used in roll-up or summary operations. Measures are indicated by a red sphere.
- Filters — Filters are pre-made common data filters, used to quickly construct limitations on the data displayed in the report. For example, in Service Management, there is a filter to show only Open Calls. Filters are indicated by a gold funnel.

RDS Object Hierarchy

Now that you understand the basic building blocks of the RDS, let's discuss how we set up the data. Let's start from the top.

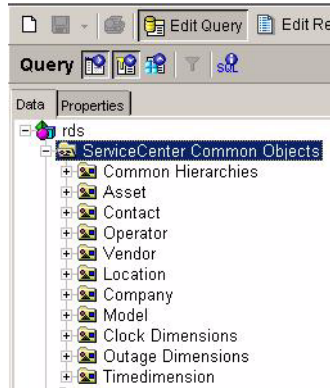


At the root of the hierarchy are seven folders:

- ServiceCenter Common Objects - This folder contains fields from tables that are referenced from every ServiceCenter module, such as Contacts, Assets, and Locations. Most of the objects here can be found on the Support tab of the main menu on an out of the box ServiceCenter system. Many of your reports will entail data from one SC module, supported by data from the Common Objects area. We'll discuss this at length later in the chapter.
- Service Management - This data maps directly to the ServiceCenter call tracking system and it's supporting tables.
- Incident Management - This data maps directly to the ServiceCenter ticket tracking system and it's supporting tables.
- Inventory Management - This data contains advanced Asset handling data, including outages.
- Change Management - This data maps to ServiceCenter's SCR and Task areas.
- Root Cause Analysis - This data maps to ServiceCenter's Root Cause codes.
- RDS ETL Process Status - This data contains information about the mapping of data between ServiceCenter and the RDS. Here you will find information regarding when the data was last updated, relevant keys, and historical updates. This folder is provided primarily for troubleshooting and logging, and will rarely be accessed for day-to-day reporting.

ServiceCenter Common Objects

Much of the Common Objects area is self-explanatory. The Asset, Contact, Operator, Vendor, Location, Company, and Model folders all map directly to their equivalent objects in ServiceCenter. However, there are additional objects that require some explanation.



- **Common Hierarchies** - There are four supporting tables in ServiceCenter which are self-referential (i.e., some records are parents of other record in the same table). The Hierarchy folder maps to special tables that only exist in the RDS. Here, we have taken the records from Asset, Location, Department, and Contact and displayed them levels which more clearly display parent-child relationships. This is largely to facilitate OLAP reporting, where you may want to drill down on these fields. By default, 5 levels are defined, but you may customize this to suit your needs.
- **Clock Dimensions** - Clocks are running timers in ServiceCenter. If you have configured a clock, it can keep track of the status of a record, turning on and off as certain events take place. For example, a clock can track how long an Incident is open, turning off on holidays and during any time when the incident has a status of Waiting On Customer.
- **Outage Dimensions** - Outages are occasions when an Asset is down, due to repair or error. Outage data is used to determine compliance with service contracts and overall support staff performance.
- **Timedimension** - This special generic Date-Time field can be used for OLAP drill-downs.

ServiceCenter Modules

Currently, the RDS supports five ServiceCenter modules. These are:

- Service Management
- Incident Management
- Inventory Management
- Change Management
- Root Cause Analysis.

The ServiceCenter module folders all conform to a basic standard hierarchy. Under each main folder, you will find 3 subfolders: Dimensions, Measures, and Filters (See RDS Elements earlier in this chapter).

- Dimensions - Under the Dimensions folder are several subfolders.

The first contains the common details. These are fields that are typically most important to report designers, such as IDs, status, priority, open and close details, etc.

There are also subfolders grouped by area, such as relevant asset information, contact information, parts and labor.

And, there is a folder for Related Records, which maps for associated records from other modules (incidents tied to calls, for example).

- Measures - Under the measures folder are the most common numeric data.

There is a subfolder containing Aggregations. These are pre-calculated sums and averages, provided to add efficiency for reports only concerned with totals.

- Filters - Here are pre-made filters. Most are self-explanatory, but there are two special types of filters you should be aware of.

First, you will see a series of No Match filters. These exist to help you find bad data in the system. For example, a filter of No Match Contact under Service Management Filters will show you all calls where the Contact Name contains a value not found in the Contacts table.

Second, you will see Active filters. Active is not the same as Open. The RDS contains more than the current data from your ServiceCenter system. It also contains historical records. When you change a key field in a record, the RDS makes a duplicate of that record and marks one as Current and one as Historical. This way you can track changes in the life of your data. So, for any report in which you want to use the current data only, you will want to add the appropriate Active filter, otherwise you'll get multiple copies of the same record. We'll discuss this more in Report Writing Basics below.

Report Writing Basics

Before you start designing a report, you want to give some thought as to the type of report you wish to create. To begin, ask yourself three questions:

- Which ServiceCenter Module do you want to report on?
- Do you want a Standard or OLAP report?
- Do you want to include Historical data?

Which ServiceCenter Module?

This should be a fairly straightforward question. If you want information regarding calls, start in Service Management. If you want to track tickets, start in Incident Management. If you are looking for information regarding SCRs and tasks, start in Change Management. For assets, outages, and installations, start in Inventory Management. And for Cause Codes and resolutions, start in Root Cause Analysis.

Standard or OLAP?

A Standard report is a typical display of data. Data is shown on single record or group of records, and usually displays information in detail. Examples of a standard report would be to display the full details, including notes, of an Incident, or to show a list of all computers sorted by network and domain. This is the most common report you will design.

OLAP reports are less about detail than about grouping and drilling. High-level information is cross-referenced by one or more dimensions, and the end user can drill on any dimension to show the information in greater granularity. An example of an OLAP report would be a cross tab table showing the number of calls made by company and category, where the user could drill down on company into department and contact, and on category into subcategory and product type.

Include Historical data?

When a record in ServiceCenter is updated such that a key field is changed (contact name, category, etc.), the RDS creates a new record, rather than overwriting the old one. This allows us to track changes and do reports on these updates. Old records are marked as Obsolete to distinguish them from the active data.

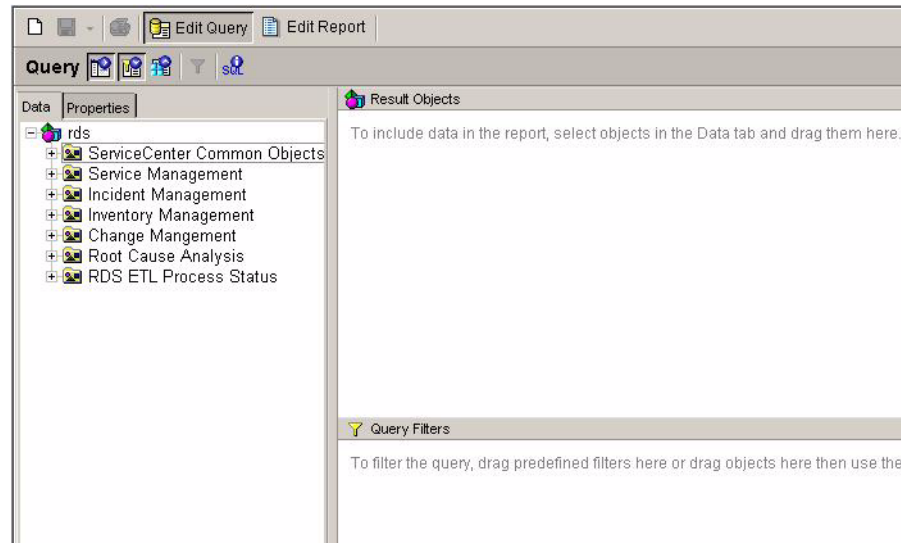
You will need to decide whether you want to exclude this historical data from your report.

Report Design Walkthrough

Lets walk through the basic steps in report design. For this example, we're going to create a report from Incident Management, standard format, excluding historical data. Our report will be Closed Incidents by Operator.

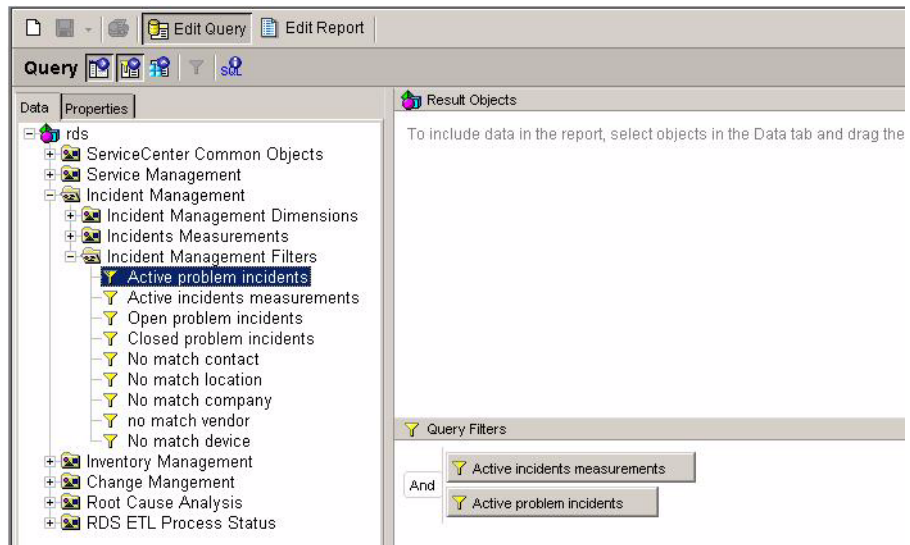
Note: Before you build reports, read the `rds.pdf` document that is included in the Documentation directory on the BI Portal installation CD. The `rds.pdf` document describes all the relationships among data in the rds physical schema.

- 1 Login to BI Portal.
- 2 Click the Reporting link.
- 3 Click the Create icon.
- 4 Select the RDS universe. The Report Designer tool opens in a new window.
- 5 If you are prompted to download the component, click Yes.



- 6 There are three sections on the screen. On the left is the Universe Hierarchy, from which you select the data to add to the report. On the right top is the Result Objects pane. You drag the fields you want to appear on the report or to be used in formulas to the Results Objects pane. On the right bottom is the Query Filters pane. You either drag a pre-made filter or build a custom one to the Query Filters pane.
- 7 Click the plus sign beside the **Incident Management** folder to expand it.
- 8 Because we are excluding historical data in our report, we need to add the Active filters to our query. Open the **Incident Management Filters** folder and drag the two Active filters into the Query Filters pane.

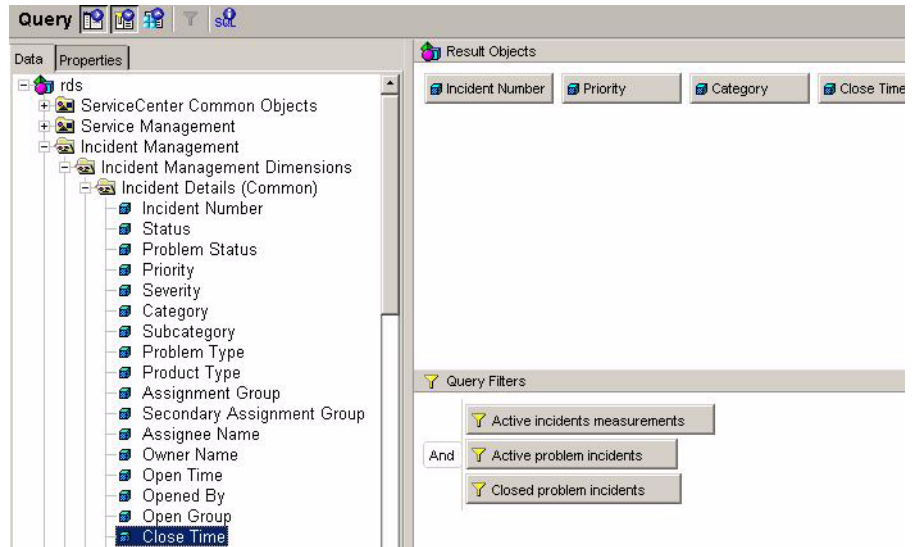
Note: It's a good habit to immediately add the relevant Active filter to the report if you don't want historical data. This will help prevent inaccuracies.



Notice two things. First, there are two filters. The Active Problem Incidents filter filters Dimension data to only include current data. The Active Incidents Measurements filter filters Measure data (in the RDS, Dimensions and Measures are kept in separate tables).

Second, when you add the second filter, an AND join is created between the two, meaning the criteria for BOTH filter must be met. If you wanted to change this to an OR, click in the word AND to change it. For this report, leave it as an AND.

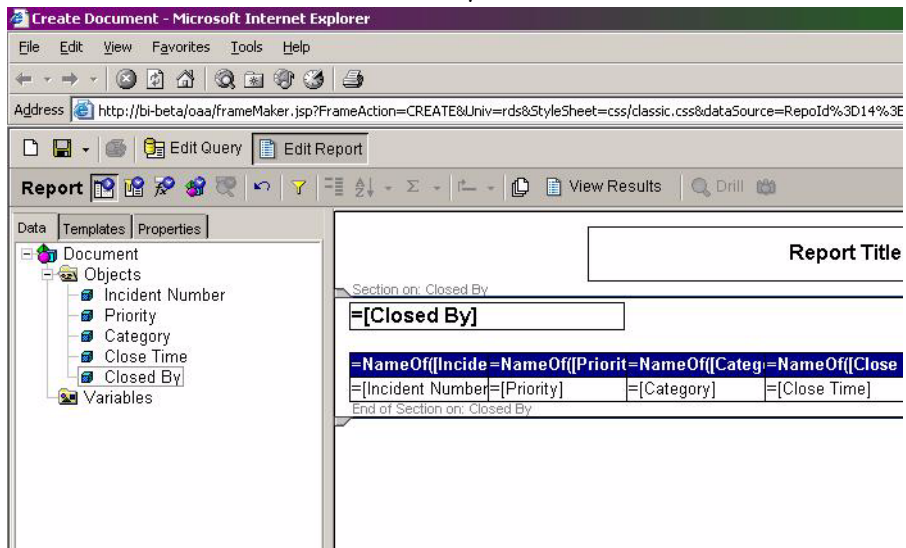
- 9 Because we are concerned only with closed incidents, drag the **Closed Problem Incidents** filter over as well. Or, you could manually create the filter by dragging the **Close Time** dimension from the Incident Details folder to the Query Filters pane and entering the filter is **not null** from the drop-down to retrieve all closed incidents.
- 10 Now that we have filtered the records we want, we get the data for display. Expand the folder **Incident Management Dimensions > Incident Details (Common)**.



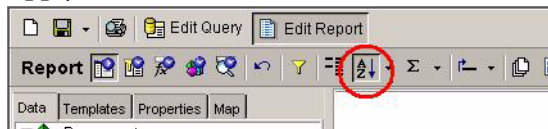
- 11 Drag the objects you want to display to the Result Objects pane. For this report, add **Incident Number**, **Priority**, **Category**, **Close Time**, and **Closed By**.

Note: If this were an OLAP report, you would use the values from the **Incident Close Time Dimension** folder, instead of **Close Time**. This would allow you to drill on the date from Year to Quarter to Month and so on.
- 12 Click **Run Query** in the top-right corner of the screen. This starts a database query and returns a basic report.
- 13 To group data for each operator, we need to create a section. A section is an area of the report that repeats for every unique value of the field used to create it. Click the **View Structure** button to see the design of the report.

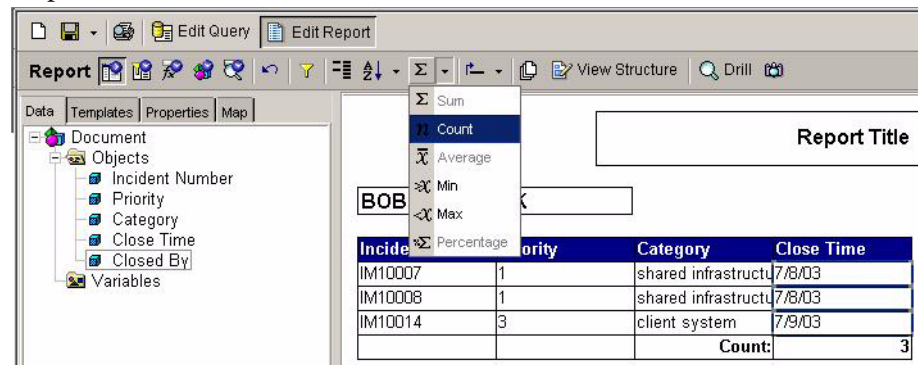
- 14 Expand the Objects folder on the left. Drag the Closed By field to the top of the report. (Make sure to drop it on empty white space, not on the table or title). A section is created automatically.



- 15 Since the Closed By information is now in the section heading, you don't need it in the table. Right-click on the **Closed By** column of the table and select **Delete Column**.
- 16 Click **View Results** to see your report as users will see it.
- 17 To sort by priority, click on the data in the **Priority** column and click the **Apply/Remove Sort** button.



- 18 To get a count of Incidents per operator, click on the data of any column, click the **Insert Sum** pull-down, and select **Count** from the summary operations list.



- 19 Double click on the **Report Title** box to edit the title. Click the green check icon to validate and save the change.
- 20 Click the **Save** pull-down and select **Save as Personal Document** or **Save as Corporate Document** to save your report.

Scheduling automatic data synchronization

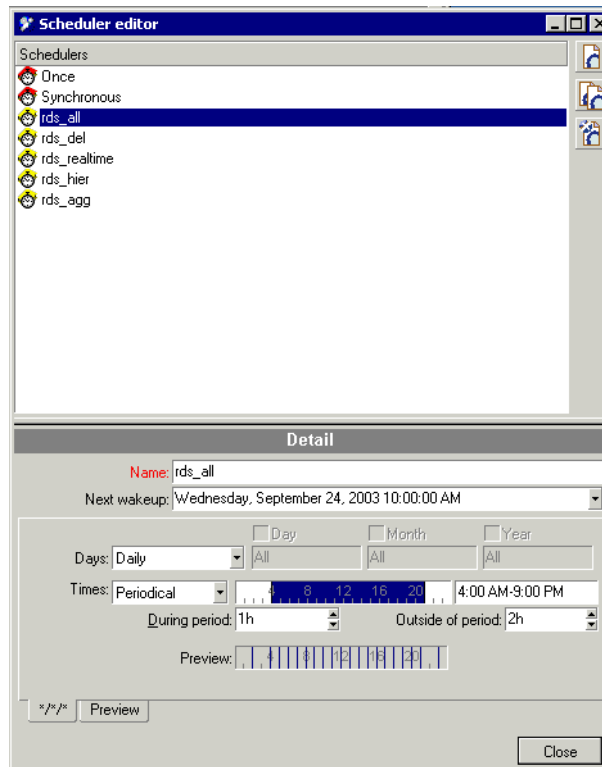
RDS has set of pre-defined Connect-It scenario schedulers to run different synchronization tasks. Most of these RDS default schedulers are set up to run every hour between 4:00 AM to 9:00 PM, and every two hours outside of that time period.

Because some of the synchronization tasks require more system resources than others, they should not be re-configured to occur more frequently than the default time intervals.

To schedule synchronization, you use the Connect -It Scheduler Editor.

To open the Connect-It Scheduler Editor:

- 1 Click **Start -> Programs -> Peregrine -> Connect-It -> Service Console**.
- 2 Click **Schedulers** to open the Connect-It Scheduler Editor:



The following schedulers define synchronization schedules:

Scheduler:	Description:
rds_all	Synchronizes new and updated records at the default intervals of one hour (1h) within the defined period and 2 hours (2h) outside of that period.
rds_del	Synchronizes deleted records at the default intervals of one hour (1h) within the defined period and two hours (2h) outside of that period. Because ServiceCenter physically deletes records from its database, and RDS is currently designed to use a timestamp instead of database triggers, this setting for synchronization tasks require intensive scanning and comparison of tables to determine those records that should be marked as logically deleted. Because of intense usage of system resources, these tasks are programmatically controlled to run no more frequently than once an hour; schedule time of less than one hour will be ignored.

Scheduler:	Description:
rds_realtime	Synchronizes new and updated important records, such as operator accounts, at default intervals of two minutes (2min) within the defined period and two hours (2h) outside of that period.
rds_hier	Synchronizes the hierarchies defined within the same ServiceCenter objects, including Asset, Location, Department and Contact. RDS has defined separate tables to track these hierarchies. These hierarchy tables are rebuilt during each synchronization interval. Default synchronization intervals are one hour (1h) within the defined period and two hours (2h) outside of that period.
rds_all	Recalculates accumulated totals and averages for selected ServiceCenter objects, including Service Requests, Incidents, Inventory, and Change Requests. RDS has defined separate tables to track these aggregations. These aggregation tables are repopulated during each synchronization interval. Default synchronization intervals are one hour (1h) within the defined period and two hours (2h) outside of that period.

For more information about using the Connect-It Scheduler Editor, see the Connect-It documentation.

7 Troubleshooting

CHAPTER

This section offers solutions when trying to resolve administration problems.

This chapter covers the following topics:

- *Browser issues* on page 117
- *Tomcat issues* on page 118

Browser issues

The following problems can result from the Internet browser you use to view BI Portal.

Navigation Issue

When logged in to BI Portal, using the browser Back, Forward, and Refresh buttons can cause unexpected behavior of BI Portal forms.

Solution Do not use the browser navigation or Refresh buttons with BI Portal forms displayed.

Issue When using the Microsoft Internet Explorer 5.5 browser, the following can occur:

- Icons fail to display in dataset results.
- You cannot personalize Collections and Subdocuments.

- JavaScript errors appear during login (apparent only if the option to display JavaScript errors is turned on for the browser).

Solution Upgrade to Internet Explorer 6.

Issue After changing a theme using the Change Themes page, clicking the Go Back button does not return you to the Home page.

Solution On the Activity menu in the sidebar, click My Home Page.

Issue Using the Back button intermittently produces a page expired error message. This error most often appears when you attempt to return to a list screen from a detail screen.

Solution Create a new search to regenerate your list. BI Portal does not cache what is on the screen.

Tomcat issues

The following problems involve issues with Tomcat as the application server.

Issue Tomcat fails to launch after a new version of the JDK is installed.

Solution When installing a new JDK, you must copy the JAR files from C:\Program Files\Peregrine\oaa\external (or to the installation location you specified) to the new JDK jre\lib\ext directory.

Issue Tomcat and Apache do not automatically start after a UNIX upgrade.

Solution Restart OAA by executing the command:
`/usr/local/peregrine/bin/oaactl restart`

Index

A

- Activity menu 35
- adapter transactions, viewing 53
- Admin module
 - changing Settings 50
 - Control Panel 46
 - displaying message queues 51
 - generating web archive files 54
 - importing and exporting personalizations 53
 - message queues 51
 - script status 51
 - Server Log 48
 - Settings page 48
 - showing queue status 52
 - verifying script status 51
 - viewing adapter transactions 53
- Archway architecture
 - building blocks 13
 - clients 15
 - diagram 14
 - requests 17
 - XML 15
- authentication
 - overriding the login script 99
 - users 71

C

- changing passwords 58
- changing the Peregrine Portal layout 38
- changing themes 40
- components

- adding Portal 36
 - creating new 35
- Control Panel 46
- CSS files, editing 25
- customer support 9

D

- deploying themes 22
- document groups 62
 - assigning users to 64
 - creating 63

E

- exporting personalized pages 53

F

- form details 56
- form details, displaying 56
- Form Info, displaying 54
- form information, displaying 41
- framesets, changing 30

G

- getit.admin user rights 44
- groups
 - document 62

H

- header graphic, changing 23

- I**
- IBM Websphere portal 54
- importing personalized pages 53
- Info button 56
- Integrated Windows Authentication
 - configuring 82
 - security 68
- J**
- JAAS
 - authentication 71
 - login modules 72
- L**
- layers, changing 27
- layout, changing
 - MSIE 39
 - Netscape Navigator 39
- LDAP 68
- Lightweight Directory Access Protocol 68
- local.xml file 44, 49
- log, form details 56
- logging user sessions 58
- login authentication 71
- login modules, JAAS 72
- login script, overriding 99
- login.asp 91
- M**
- message queues 51
- message queues, displaying 51
- monitoring user sessions 58
- moving personalized pages 53
- O**
- overriding the login script 99
- P**
- parameters, defining 50
- password, changing 58
- passwords
 - protecting 68
- Peregrine Portal
 - adding components 36
 - personalizing 36
- Peregrine Portal, tailoring 21
- Peregrine Systems customer support 9
- personalized pages
 - moving 53
- personalizing
 - portal 36–41
- personalizing the Peregrine Portal 36
- Portal Components, creating new 35
- preXSL, form details 56
- Q**
- queue status, displaying 52
- R**
- resetting the server 46
- S**
- schemas
 - testing from a URL 18–??
- script input, form details 56
- script output, form details 56
- script status 51
- script status, verifying 51
- scripts
 - testing from a URL 17
- Secure Sockets Layer 68
- security
 - alternate login authentication 99
 - user authentication 71
 - Windows Integrated Authentication 82
- self-registration 57
- server log 48
- Settings page 50
- SSL 68
- T**
- tailoring themes 21
 - changing framesets 30
 - changing layers 27
 - changing stylesheets 25
 - changing the header graphic 23
 - deploying themes 22
- technical support 9
- themes
 - deploying 22

- tailoring 21
- themes, changing 40
- themes, creating 25

U

- URL
 - querying scripts and schemas from 17
- user registration 57
- user rights
 - getit.admin 44
- user session 56
- user sessions, logging 58
- user.log file 58

W

- web archive (war) files 54
- Websphere portal 54



September 30, 2003