Oracle® Fusion Middleware

Administrator's Guide for Oracle WebCenter Ensemble

10*g* Release 3 (10.3.0.1.0)

E14114-01

May 2009

Describes how to perform administration tasks for Oracle WebCenter Ensemble.

ORACLE

Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Ensemble, 10*g* Release 3 (10.3.0.1.0)

E14114-01

Contents

8 Pagelets

9 Audit

Preface

This book describes how to administer and use Oracle WebCenter Ensemble.

Audience

This guide is written for users responsible for administering, configuring, and auditing Oracle WebCenter Ensemble and Oracle WebCenter Ensemble resources.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at `http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For more information, see the following documents in the Oracle WebCenter Ensemble 10*g* Release 3 (10.3.0.1.0) documentation set:

- *Oracle WebCenter Ensemble Release Notes*
- *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Ensemble*
- *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Ensemble*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction to Oracle WebCenter Ensemble

This chapter provides an introduction to the features, functionality, and architecture of Oracle WebCenter Ensemble, and is divided into the following sections:

- Section 1.1, "About Oracle WebCenter Ensemble," describes the features and functionality of Oracle WebCenter Ensemble.
- Section 1.2, "Oracle WebCenter Ensemble Architecture," describes the component architecture of Oracle WebCenter Ensemble.

1.1 About Oracle WebCenter Ensemble

This section describes the features and functionality of Oracle WebCenter Ensemble.

1.1.1 The Ensemble Console

The Ensemble Console is a browser-based administration tool used to create and manage the various objects in your Oracle WebCenter Ensemble deployment. From the Ensemble Console you can register web applications and pagelets with the Ensemble Proxy, configure authentication, manage the user experience, and configure auditing.

For more details on the Ensemble Console, see Chapter 2, "The Ensemble Console."

1.1.2 Resources

Resources are web applications registered with the Ensemble Proxy. Registering a resource allows the Ensemble Proxy to map internal applications to external URLs, manage authentication, and transform applications using Oracle WebCenter Ensemble adaptive tags and the Oracle WebCenter Interaction Development Kit (IDK).

For details on proxy resources, see Chapter 3, "Proxy Resources."

Authentication of users is managed using two mechanisms: proxy authentication and credential mapping.

Proxy authentication is how a user authenticates with the proxy in order to access proxied resources. Multiple authentication sources can be configured as login resources with Oracle WebCenter Ensemble. The relative strength of authentication sources is described using authentication levels. A user authenticated at an authentication level can access resources configured for that authentication level, plus any resources with lower authentication levels. Authentication sources for proxy authentication can be based on the Oracle WebCenter Interaction portal database or third-party SSO providers such as CA SiteMinder or Oracle COREid, or a customer's LDAP or Active Directory server.

For details on proxy authentication, see Chapter 4, "Proxy Authentication."

For details on how authentication sources are associated with users accessing resources, see Chapter 7, "Experience Definitions."

Credential mapping is how Oracle WebCenter Ensemble authenticates with the proxied application. Credentials can be static and defined within the resource configuration, can be stored in the user's profile in Oracle WebCenter Interaction, or can be stored in the Credential Vault after the user has logged into the proxied application once.

For details on credential mapping, see Chapter 5, "Credential Mapping."

In addition to authentication, access to resources is controlled by policies and rules. Policies determine who can access a resource and under what conditions. Conditions for access can include factors such as time of day, an IP address within a configured range, the user's browser, and other criteria.

For details on policies and rules, see Chapter 6, "Policies and Rules."

1.1.3 Pagelets

A pagelet is a sub-component of a web page, accessed through the Ensemble Proxy, and able to be injected into any proxied application. Any application on an Oracle WebCenter Ensemble resource that returns text can be registered as a pagelet.

You can pass data to pagelets using pagelet parameters or the pagelet payload. The former passes name-value pairs to the pagelet application, while the latter is any text, including XML. Parameters can even be configured to be sent using transport types that mimic Oracle WebCenter Interaction parameter transport, which allows you to port Oracle WebCenter Interaction portlets to be Oracle WebCenter Ensemble pagelets.

For details on pagelets, see Chapter 8, "Pagelets."

1.1.4 Audit and Oracle WebCenter Analytics

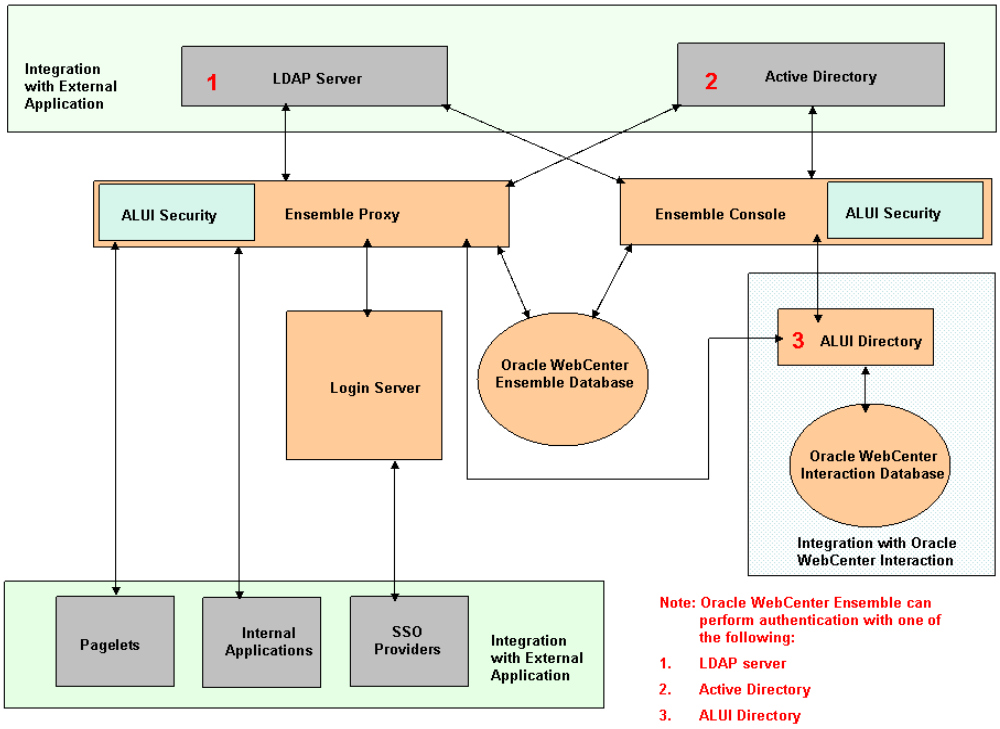
You can access usage information for Oracle WebCenter Ensemble proxied applications and the Ensemble Console by using the Oracle WebCenter Ensemble audit system and the Oracle WebCenter Ensemble integration with Oracle WebCenter Analytics.

For details on Oracle WebCenter Ensemble audit features, see Chapter 9, "Audit."

For details on using Oracle WebCenter Ensemble with Oracle WebCenter Analytics, see Chapter 10, "Oracle WebCenter Analytics."

1.2 Oracle WebCenter Ensemble Architecture

The following diagram illustrates the architecture of an Oracle WebCenter Ensemble deployment, integrated with Oracle WebCenter Interaction as well as with an external application.



- The Ensemble Proxy provides users with external access to internal resources including login resources, internal applications, and pagelets.
- The Ensemble Console allows administrators to configure resources, pagelets, access, and auditing features of Oracle WebCenter Ensemble.
- The Login Server interfaces with the security and directory service and SSO providers to provide authentication services to the Ensemble Proxy.
- ALUI Directory uses the Oracle WebCenter Interaction portal database to authenticate users in both the Ensemble Console and the Ensemble Proxy via the Login Server.

2

The Ensemble Console

This chapter provides a high-level description of the Ensemble Console and Ensemble Console security and resource ownership. It is divided into the following sections:

- Section 2.1, "About the Ensemble Console," provides an overview of what the Ensemble Console does and where to find more information on configuration of Oracle WebCenter Ensemble objects.
- Section 2.2, "Launching the Ensemble Console," describes how to access the Ensemble Console.
- Section 2.3, "Ensemble Console Roles," describes how Ensemble Console roles allow you to control how users access the Ensemble Console.

2.1 About the Ensemble Console

The Ensemble Console is a browser-based administration tool used to create and manage the objects in your Oracle WebCenter Ensemble deployment. From the Ensemble Console:

- Developers register resources and pagelets and configure credential mapping for remote applications. For details, see:
  - Chapter 3, "Proxy Resources"
  - Chapter 5, "Credential Mapping"
  - Chapter 8, "Pagelets"
- IT configures proxy authentication and manages experience definitions and interstitial pages. For details, see:
  - Chapter 4, "Proxy Authentication"
  - Chapter 7, "Experience Definitions"
- Users edit policy sets and manage resources that they own. For details, see:
  - Chapter 6, "Policies and Rules"
- Auditors enable and disable auditing for remote applications. For details, see:
  - Chapter 9, "Audit"

2.2 Launching the Ensemble Console

To launch the Ensemble Console, perform one of the following:

- To launch the Ensemble Console via a supported web browser, type the following URL into your browser's address bar: http://*host*:20070/ensembleadminui/

  Replace *host* with the name of the server on which you installed the Ensemble Console.

- To launch the Ensemble Console from the Windows Start menu, click Start > All Programs > Oracle > Ensemble > Ensemble Admin UI (20070)

2.3 Ensemble Console Roles

Ensemble Console roles control which parts of the Ensemble Console users can access and what actions they can perform. The following table summarizes the Ensemble Console roles, the tabs each role can access, and the actions each role can perform:

Table 2–1 Oracle WebCenter Ensemble Roles

Role	Accessible Tabs	Available Actions
Administrators	■ All	■ All actions
Managers	■ Applications ■ Policies ■ Experiences ■ Proxy Authentication	■ Any action on accessible tabs.
Resource Owners	■ Applications	■ Edit resources the user owns. ■ Edit pagelets associated with resources the user owns. ■ Create pagelets associated with resources the user owns.
Policy Set Owners	■ Policies	■ Edit policy sets the user owns. ■ Create policy rules in the rule library.
Auditors	■ Audit	■ Enable or disable auditing on any resource.

2.3.1 Configuring Administrators, Managers, and Auditors

You configure *Administrators*, *Managers*, and *Auditors* by adding or deleting users or groups in each role.

To add users to the Administrators, Managers, or Auditors roles:

1. Launch the Ensemble Console.
2. Click the **ADMINISTRATION** tab.
3. Click the **Administrators**, **Managers**, or **Auditors** sub-tab.
4. To display the user and group picker, click **Add**.
5. Select one or more users or groups.
6. Click **Add selected items**.
7. Click **OK**.
8. Click **Save**.

To remove users from these roles:

1. Launch the Ensemble Console.

2. Click the **ADMINISTRATION** tab.
3. Click the **Administrators**, **Managers**, or **Auditors** sub-tab.
4. Select one or more users or groups to remove.
5. Click **Remove**.
6. Confirm that you want to delete these users or groups by clicking **OK**.
7. Click **Save**.

2.3.2 Configuring Resource and Policy Set Owners

The *Resource Owners* and *Policy Set Owners* roles are granted to a user when the user is made owner of a resource or policy set. To change the owner of a resource or policy set:

1. Launch the Ensemble Console.
2. Click the **ADMINISTRATION** tab.
3. Click the **Resource Owners** or **Policy Set Owners** sub-tab.
4. Click the name of the resource or policy set you want to edit.
5. To display the user picker, next to the **New Owner** box, click **Select**.
6. Select the user whom you want to make owner of the resource or policy.
7. Click **OK**.
8. To replace all instances of the **Current Owner** with the **New Owner**, select the check-box next to **Replace all ownership instances assigned to this user**.
9. Click **Save**.

3

Proxy Resources

This chapter describes how to configure Oracle WebCenter Ensemble resources. It is divided into the following sections:

- Section 3.1, "About Oracle WebCenter Ensemble Resources," describes what an Oracle WebCenter Ensemble resource is.
- Section 3.2, "Registering a Resource," describes how to create a basic Oracle WebCenter Ensemble resource.
- Section 3.3, "Advanced Resource Configuration," describes additional configurations of Oracle WebCenter Ensemble resources, including URL rewriting, resource roles, the Oracle WebCenter Interaction login token, and authentication.
- Section 3.4, "Migrating Resources," describes how to export and import Oracle WebCenter Ensemble resources and their associated pagelets.
- Section 3.5, "Working with Web Injectors," describes how to create web injectors, configure injection patters, and apply web injectors to resources.

3.1 About Oracle WebCenter Ensemble Resources

Oracle WebCenter Ensemble resources are web applications registered in Oracle WebCenter Ensemble. A registered resource maps an internal URL, accessible by Oracle WebCenter Ensemble, to an external URL, accessible by end users. Any web application can be registered as a resource.

Registering a web application as an Oracle WebCenter Ensemble resource allows Oracle WebCenter Ensemble to do the following:

- Proxy internal web applications to external addresses.
- Manage authentication, both at the proxy level (Oracle WebCenter Ensemble controls access to resources using Oracle WebCenter Ensemble policies and roles) and at the resource level (Oracle WebCenter Ensemble provides credentials to proxied web applications).
- Transform proxied web applications, including URL-rewriting and the use of Oracle WebCenter Ensemble and Oracle WebCenter Interaction adaptive tags.
- Customize the user experience through custom login, logout, interstitial, and error pages.

3.2 Registering a Resource

You register a resource in Oracle WebCenter Ensemble using the Ensemble Console. The simplest Oracle WebCenter Ensemble resource has three configured properties:

- Name.
- Internal URL prefix, which is the URL of the application to be proxied.
- External URL prefix, which is the URL that end users will use to access the application.

Once configured, all URLs starting with the Internal URL prefix are accessible via the External URL prefix. For example, if the Internal URL prefix is:

```
http://internalServer/foo
```

and the External URL prefix is:

```
http://externalServer/bar
```

the external path:

```
http://externalServer/bar/index.jsp
```

will map to:

```
http://internalServer/foo/index.jsp
```

and

```
http://externalServer/bar/baz/index.jsp
```

will map to:

```
http://internalServer/foo/baz/index.jsp
```

To register a simple resource in Oracle WebCenter Ensemble:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. To create a new resource, click **Create new**.
5. On the General page, in the **Name** box, type the name of the resource.
6. On the Connections page, in the **Internal URL prefix** box, type the URL to the internal web application to be proxied. For example, `http://internalServer/foo/`.
7. In the **External URL prefixes** box, type the URL to be used to access the resource. This URL must be on the Ensemble Proxy server. You may specify a fully-qualified URL or a path relative to the Ensemble Proxy server. For example, `http://externalServer/bar/` or just `/bar/`.

> **Note:** A fully-qualified external URL prefix must include the same port used by the Ensemble Proxy server unless the Use Proxy Port option is checked, in which case any specified port will be changed to an asterisk and Oracle WebCenter Ensemble will assume that the proxy's port is in the external URL.

8. Click **Save**.

3.3 Advanced Resource Configuration

This section describes advanced configuration options for Oracle WebCenter Ensemble resources. It is divided into the following sub-sections:

- Section 3.3.1, "URL Rewriting and DNS"
- Section 3.3.2, "Roles"
- Section 3.3.3, "Proxy Authentication"
- Section 3.3.4, "Credential Mapping"
- Section 3.3.5, "The AquaLogic Interaction Login Token"

3.3.1 URL Rewriting and DNS

When you enable URL rewriting, the Ensemble Proxy rewrites URLs in the proxied application that begin with the internal URL prefix so that they point to the external URL prefix. Oracle WebCenter Ensemble enables URL rewriting by default.

You should disable URL rewriting when the internal URL prefix and external URL prefix are identical. When this occurs, the user's DNS must resolve the URL to the Ensemble Proxy server, and the Ensemble Proxy server's DNS must resolve the URL to the internal resource. Because DNS only resolves IP and not port, both servers must listen to the same port.

To disable URL rewriting:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the resource you want to edit.
5. On the General page, uncheck the box next to **Enable URL Rewriting**.
6. Click **Save**.

3.3.2 Roles

You can configure Oracle WebCenter Ensemble to send role information to proxied applications. You define the roles available for Oracle WebCenter Ensemble to send to the proxied application within the resource configuration. Policies determine which of these roles Oracle WebCenter Ensemble sends for a given user.

For details on policies and how they map to roles, see Chapter 6, "Policies and Rules."

Oracle WebCenter Ensemble sends roles in the HTTP header and are accessed by the proxied application using the Oracle WebCenter Interaction Development Kit (IDK) proxy API. For details on using the Oracle WebCenter Interaction Development Kit (IDK) proxy API, see the Oracle WebCenter Interaction Development Kit (IDK) documentation on the Oracle Technology Network at `http://download.oracle.com/docs/cd/E13158_01/alui/idk/docs103/index.html`.

To configure roles to send to a proxied application:

1. Launch the Ensemble Console.

2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the resource you want to edit.
5. On the Roles page, type the names of the role or roles. Click **Add** to create additional roles.
6. Click **Save**.

The roles entered on the Roles page are the values that Oracle WebCenter Ensemble can send to the proxied application, based on what policy or policies are associated with the user.

3.3.3 Proxy Authentication

Proxy Authentication describes how users log into Oracle WebCenter Ensemble resources. Oracle WebCenter Ensemble can facilitate authentication using a variety of methods, including basic authentication, HTML form-based authentication, and integration with third-party SSO products.

For details on Proxy Authentication, see Chapter 4, "Proxy Authentication."

3.3.4 Credential Mapping

Credential mapping allows Oracle WebCenter Ensemble to automatically supply credentials to proxied applications. The credentials can be a static set used for all users, credentials specific to the user and stored in the user's Oracle WebCenter Interaction user profile, or credentials used once by the user and captured and stored by Oracle WebCenter Ensemble in the Credential Vault. The Credential Vault allows users to authenticate once and then be logged in automatically by Oracle WebCenter Ensemble in future accesses to the proxied resource.

For details on credential mapping, see Chapter 5, "Credential Mapping."

3.3.5 The AquaLogic Interaction Login Token

The AquaLogic Interaction login token allows the Oracle WebCenter Ensemble resource to access the AquaLogic Interaction IPortletContext object. By default, the AquaLogic Interaction login token is not passed to the proxied resource.

To pass the login token to the proxied resource:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the resource you want to edit.
5. On the CSP page, select **Send ALI login token**.
6. Click **Save**.

3.4 Migrating Resources

Ensemble Migration allows you to export and import Oracle WebCenter Ensemble resources and their associated pagelets. The exported resource configuration is stored in an XML file, which you can then import into another installation of Oracle WebCenter Ensemble.

3.4.1 Exporting Resources with Migration

To export a resource and its associated objects:

1. Launch the Ensemble Console.
2. Click the **Administration** tab.
3. Click the **Migration** sub-tab.
4. Click the **Export** tab.
5. Launch the resource picker by clicking **Add Resource**.
6. Select the resources you want to export. Click **Add selected items**.
7. Confirm the resources to be exported and click **OK**.
8. Click **Export**.

   The File Download dialog box appears.
9. Click **Save**.

   The Save As dialog box appears.
10. Navigate to the location on the local machine to which you want to save the file.
11. If desired, rename the file to something other than the default of ensemble_migration.xml.
12. Click **Save**.

The migration file is saved to the local machine. You can now import the migration file to any Oracle WebCenter Ensemble installation. now located on the local machine in the chosen location, and can be imported to any Ensemble installation.

3.4.2 Importing Resources With Migration

To import a resource from an existing migration package:

1. Launch the Ensemble Console.
2. Click the **Administration** tab.
3. Click the **Migration** sub-tab.
4. Click the **Import** tab.
5. Type the full path to the migration package in the **Import this file** box.
6. Choose how you want to deal with duplicate resources by selecting **Rename imported resource** or **Overwrite existing resource**.
7. Click **Prepare Import**.
8. Depending on the content of the migration package, you might be prompted to update values for specific object properties. Follow the instructions on the screen.
9. Click **Import**.

On success, the Ensemble Console displays a list of imported objects and any autofixes that were completed during the import process.

3.5 Working with Web Injectors

A web injector inserts content into a specified location in the proxied resource page. The content may be any text, including HTML, CSS, JavaScript, and pagelet declarations.

To use a web injector with a resource, you must first create the web injector and then, from the resource configuration, apply the web injector to that resource.

3.5.1 Creating Web Injectors

To create a new web injector:

1. Click the **Applications** tab.
2. Click the **Web Injectors** sub-tab.
3. Under the web injectors view, click **Create New**.
4. On the **General** tab, type a name and description.
5. On the **Details** tab, configure one or more *injection patterns*. For details, see Section 3.5.2, "Configuring Injection Patterns."

3.5.2 Configuring Injection Patterns

An Injection Pattern describes what the content is and where it is to be injected. An Injection Pattern can either be based on an existing web injector, or defined as a new pattern:

- To base the Injection Pattern on an existing web injector, click Add Existing Injector.
- To create a new pattern, click Define New Pattern.

The following table describes the configuration options for a new Injection Pattern:

Table 3–1 Injection Pattern Configuration Options

Field	Description
What to Inject	Type the content to be injected into the **What to Inject** box. Content may be any text, including HTML, CSS, JavaScript, and pagelet declarations. To automatically populate the **What to Inject** box with a pagelet declaration, select a pagelet from the **Insert pagelet declaration for** drop-down list and click **Insert**.
Where to inject	Define where in the resource's output the injection occurs by selecting one of the following: ■ **Top of the page** puts the content first in the page. Do not use this option to inject pagelet declarations. ■ **Bottom of the page** puts the content last in the page. ■ The **Before/After/In place of** drop-down list puts the content into the page relative to what you type in the **specific text** box. To make the text identify an entire start tag, select **Enclosing tag**. This will match the tag regardless of what attributes are present, if any. For example, if your text is *div* and **Enclosing tag** is selected, * * will match as will * *. To match the text regardless of case, select **Ignore case**. To match the text in pagelet output, select **Apply to underlying pagelets**. If **Apply to underlying pagelets** is not selected, the content of any pagelets in the page will not be considered when Ensemble determines where to make injections. **Note:** Even when **Apply to underlying pagelets** is selected, the pagelet declaration is a candidate for matching. If you have selected **In place of** and your text matches part of the pagelet declaration, your injected content will replace that part of the pagelet declaration. This might cause the pagelet to not be displayed or yield other undesired results.
Apply only to Content Types	To restrict the injector to specific kinds of content, type a comma separated list of MIME types in the **Apply only to Content Types** box. For example, *text/html* restricts the injector to HTML content, while text/css only restricts the injector to CSS content.

3.5.3 Applying Web Injectors to Resources

To apply a web injector to a resource:

1. Launch the Ensemble Console.
2. Click the **Applications** tab.
3. Click the **Resources** sub-tab.
4. Click the name of the resource to which you want to apply a web injector.
5. Click the **Web Injectors** tab.
6. From the drop-down list, select the name of the web injector to apply to the resource.
7. If desired, make the web injector applicable to a subset of the resource by typing a URL pattern into the **URL Filter** box. If the box is empty or contains only a /, the web injector is applied to the entire resource. If anything else is typed into the box, only URLs within the resource that begin with what is in the box will have the web injector applied.
8. To apply more than one web injector to the resource, click **Add** and repeat steps 6 and 7 for each web injector.

4

Proxy Authentication

This chapter describes resource access control using Oracle WebCenter Ensemble proxy authentication. It is divided into the following sections:

- Section 4.1, "Authentication Levels," describes how Oracle WebCenter Ensemble uses authentication levels to control access to resources.
- Section 4.2, "Configuring Authentication Levels," describes how to configure authentication level associated with each authenticator.
- Section 4.3, "SSO Integration," describes how to configure Oracle WebCenter Ensemble to use SiteMinder, Oracle COREid, and SPNEGO SSO products.

4.1 Authentication Levels

There are two factors that control access to a resource: authentication levels and policies. Before any policy is evaluated for a given resource, the user must first be authenticated with an *authenticator* that has an authentication level equal to or greater than the authentication level of the resource. If an authentication level does not have an authenticator associated with it, the next higher authenticator is used to authenticate.

Authentication levels range from 0 to 10. An authenticator cannot be assigned level 0; authentication level 0 is reserved for anonymous access. For details on anonymous access, see Section 6.2.4, "Configuring Anonymous Access."

An authenticator is a method for authentication. HTML form-based authentication and third-party SSO providers are examples of authenticators.

When a user attempts to access a resource without credentials appropriate for the resource's authentication level, the following happens:

1. Oracle WebCenter Ensemble evaluates experience rules to determine which experience definition is appropriate for the user.
2. Oracle WebCenter Ensemble passes the authenticator associated with the experience definition to the authentication stack.
3. If the authenticator is equal to or greater than the resource's authentication level, Oracle WebCenter Ensemble uses the authenticator associated with the experience definition to authenticate the user.

   If the authenticator is lower than the resource's authentication level, Oracle WebCenter Ensemble uses the authenticator associated with the resource's authentication level.

4. Once the user is authenticated, Oracle WebCenter Ensemble evaluates the policies for the resource. If one or more policies evaluate to true, the user is granted access to the resource.

For details on experience rules and experience definitions, see Chapter 7, "Experience Definitions."

For details on policies, see Chapter 6, "Policies and Rules."

4.2 Configuring Authentication Levels

You configure the authentication level that is associated with an authenticator in the Ensemble Console. You configure each authenticator with a numerical level between 1 and 10. Two authenticators cannot have the same authentication level.

To configure authentication levels:

1. Launch the Ensemble Console.
2. Click the **PROXY AUTHENTICATION** tab.
3. Select the authentication level from the **Level** drop-down next to the authenticator you are configuring.

Note: Changing authentication levels for authenticators will not change authentication levels associated with policy sets. The authentication level will remain the same and the authenticator will change.

4.3 SSO Integration

This section describes how to configure Oracle WebCenter Ensemble to authenticate users with the Oracle WebCenter Interaction portal, or one of the supported third-party SSO systems: Siteminder, COREid, or Active Directory via SPNEGO. The following subsections describe each configuration in detail:

- Section 4.3.1, "Integrating with the Oracle WebCenter Interaction Portal"
- Section 4.3.2, "Integrating with Computer Associates SiteMinder"
- Section 4.3.3, "Integrating with Oracle COREid"
- Section 4.3.4, "Integrating with Microsoft Active Directory via SPNEGO"

In addition, configuring Oracle WebCenter Ensemble to log users out of an SSO system is described in Section 4.3.6, "SSO Logout."

Note: For all SSO integrations, the user name used to authenticate to the SSO software must also exist as an Oracle WebCenter Ensemble user name. To add users to Oracle WebCenter Ensemble, add users to your Oracle WebCenter Interaction installation or the LDAP server that contains your Oracle WebCenter Interaction users.

4.3.1 Integrating with the Oracle WebCenter Interaction Portal

This section provides details about configuring Oracle WebCenter Ensemble to automatically log users in to the Oracle WebCenter Interaction portal.

To integrate with the Oracle WebCenter Interaction portal:

1. Deploy the **portal.war** file, using an instance of Apache Tomcat.

   Note: Your version of Apache Tomcat must be *prior* to 5.5.25 for proper SSO integration between Oracle WebCenter Ensemble and Oracle WebCenter Interaction. This integration will fail if you use Apache Tomcat 5.5.25 or above.

   The portal.war file is located in: ***install_dir*\ensembleproxy*version*\integration\alisso**

   Deploy the portal.war file on the portal server. The location to which you deploy the portal.war file is the *Portal Cookie Replication URL*.

   Note: Although you can use various brands of web servers to host the Oracle WebCenter Interaction portal, you must use Apache Tomcat to host the portal.war file.

2. Enable the Remember Me cookie features of Oracle WebCenter Interaction by performing the following:

   Note: Before performing these steps, determine the security impact that enabling Remember Me cookie features might have on your portal environment.

   a. On the machine on which Oracle WebCenter Interaction is installed, navigate to ***install_dir*\settings\portal\portalconfig.xml**

   b. Ensure that the **AllowAutoConnect** node is set to **1**.

   c. Restart Oracle WebCenter Interaction.

3. Navigate to **Configuration Manager** > **Ensemble** > **SSO Login**, and enable portal cookie replication. Additionally, configure the following settings:

   - **Portal Cookie Replication URL:** The location to which you deployed the portal.war file in the previous step. Oracle WebCenter Ensemble redirects to this URL to verify that a portal cookie exists.
   - **Timeout:** The frequency -- in milliseconds -- at which Oracle WebCenter Ensemble checks for portal cookies.
   - **Access Level:** The access level that is assigned to the user when Oracle WebCenter Ensemble detects a valid portal session cookie. Valid values are between 1and 10. For information on authentication levels, see Section 4.1, "Authentication Levels."

4. Navigate to **Oracle WebCenter Configuration Manager** > **Ensemble** > **ALUI Security Login Tokens**, and ensure that the following settings are correctly configured:

   a. The default token type should be set to **ALUI**.

   b. The message authentication code seed value should match the login token root value. You can find the login token root value in the **ptserverconfig** table of the Oracle WebCenter Interaction database.

5. Navigate to **Oracle WebCenter Configuration Manager** > **Ensemble** > **ALUI Directory**.
   a. Ensure that the authentication provider is set to **ALUI**.
   b. Ensure that all values for the **Connection Information**, **Users**, and **Groups** sections are correctly configured.

4.3.2 Integrating with Computer Associates SiteMinder

Configuring Oracle WebCenter Ensemble to authenticate users with SiteMinder involves protecting a special Oracle WebCenter Ensemble resource, **sso.aspx**, with SiteMinder. Oracle WebCenter Ensemble uses this resource to authenticate a user with SiteMinder when the user attempts to access any resource with an authentication level that requires SiteMinder.

The process flow is as follows:

1. The user attempts to access a resource proxied by Oracle WebCenter Ensemble.
2. Oracle WebCenter Ensemble determines the user needs to authenticate with SiteMinder.
3. Oracle WebCenter Ensemble redirects the user to sso.aspx. Since sso.aspx is protected by SiteMinder, the user is asked to authenticate to SiteMinder.
4. On successful authentication, the user accesses sso.aspx, which redirects the user to Oracle WebCenter Ensemble marked as authenticated.
5. Oracle WebCenter Ensemble redirects the user to the resource he initially attempted to access.

The redirects between sso.aspx and Oracle WebCenter Ensemble are transparent to the user. The user experiences attempting to access the resource, being authenticated by SiteMinder, and then accessing the resource.

4.3.2.1 Configuring Oracle WebCenter Ensemble and SiteMinder

To configure Oracle WebCenter Ensemble for use with SiteMinder, first install **sso.aspx** and configure SiteMinder to protect it:

1. Create a virtual directory on IIS and protect it with SiteMinder.
2. Copy **sso.aspx** and **sso.aspx.cs** to the virtual directory you created. There are versions of these files for .NET v1.1 and .NET v2.0. In a default installation, the files are located under the **NET v1.1 aspx** or **NET v2.0 aspx** directory in: ***install_dir*\loginserver\2.0\webapp\loginserver\ssointegration\siteminder**
3. Verify that the files are installed and SiteMinder is correctly configured. Attempt to access sso.aspx via IIS. You are prompted to log into SiteMinder and then are presented with a page of header information. (The result from sso.aspx is not intended to be human-readable.)

Once you have correctly **sso.aspx,** you must configure Oracle WebCenter Ensemble to access sso.aspx via IIS. To configure Oracle WebCenter Ensemble:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the **CA SiteMinder sample login resource**.

5. On the **Connections** page, edit the **Internal URL prefix** to point to the location of **sso.aspx.** For example: http://siteminder.company.com:80/ensembleIntegration/

   Do not include the file name sso.aspx.

6. Restart the BEA ALI Security Service, the BEA AL Ensemble Administrative UI, and the BEA AL Ensemble Proxy.

7. Verify that the login resource is correctly configured. Create a resource and policy to protect it with your SiteMinder authentication level and configure your experience rules to request SiteMinder authentication when a user accesses the resource.

   - For details on creating resources, see Chapter 3, "Proxy Resources."
   - For details on configuring policies, see Chapter 6, "Policies and Rules."
   - For details on configuring authentication levels, see Section 4.2, "Configuring Authentication Levels."
   - For details on configuring experience rules, see Chapter 7, "Experience Definitions."

4.3.3 Integrating with Oracle COREid

Configuring Oracle WebCenter Ensemble to authenticate users with COREid involves protecting a special Oracle WebCenter Ensemble resource, **sso.aspx**, with COREid. Oracle WebCenter Ensemble uses this resource to authenticate a user with COREid when the user attempts to access any resource with an authentication level that requires COREid.

The process flow is as follows:

1. The user attempts to access a resource proxied by Oracle WebCenter Ensemble.
2. Oracle WebCenter Ensemble determines that the user needs to authenticate with COREid.
3. Oracle WebCenter Ensemble redirects the user to sso.aspx. Since sso.aspx is protected by COREid, the user is asked to authenticate to COREid.
4. On successful authentication, the user accesses sso.aspx, which redirects the user to Oracle WebCenter Ensemble marked as authenticated.
5. Oracle WebCenter Ensemble redirects the user to the resource he initially attempted to access.

The redirects between sso.aspx and Oracle WebCenter Ensemble are transparent to the user. The user experiences attempting to access the resource, being authenticated by COREid, and then accessing the resource.

4.3.3.1 Configuring Oracle WebCenter Ensemble and COREid

To configure Oracle WebCenter Ensemble for use with COREid, first install **sso.aspx** and configure COREid to protect it:

1. Create a virtual directory on IIS and protect it with COREid.
2. Copy **sso.aspx** and **sso.aspx.cs** to the virtual directory you created. There are versions of these files for .NET v1.1 and .NET v2.0. In a default installation, the files are located under the **NET v1.1 aspx** or **NET v2.0 aspx** directory in: ***install_dir*\loginserver\2.0\webapp\loginserver\ssointegration\coreid**

3. Verify that the files are installed and that COREid is correctly configured. Attempt to access sso.aspx via IIS. You are prompted to log into COREid and then are presented with a page of header information. (The result from sso.aspx is not intended to be human-readable.)

Once you have correctly installed **sso.aspx**, you must configure Oracle WebCenter Ensemble to access sso.aspx via IIS. To configure Oracle WebCenter Ensemble:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the **Oracle COREid sample login resource**.
5. On the **Connections** page, edit the **Internal URL prefix** to point to the location of **sso.aspx**. For example: http://coreid.company.com:80/ensembleIntegration/

   Do not include the file name sso.aspx.
6. Restart the BEA ALI Security Service, the BEA AL Ensemble Administrative UI, and the BEA AL Ensemble Proxy.
7. Verify that the login resource is correctly configured. Create a resource and policy to protect it with your COREid authentication level and configure your experience rules to request COREid authentication when the user accesses the resource.
   - For details on creating resources, see Chapter 3, "Proxy Resources."
   - For details on configuring policies, see Chapter 6, "Policies and Rules."
   - For details on configuring authentication levels, see Section 4.2, "Configuring Authentication Levels."
   - For details on configuring experience rules, see Chapter 7, "Experience Definitions."

4.3.4 Integrating with Microsoft Active Directory via SPNEGO

Configuring Oracle WebCenter Ensemble for SPNEGO authentication is a complex process involving configuration of the Active Directory server in addition to the creation of Oracle WebCenter Ensemble configuration files and Oracle WebCenter Ensemble configuration within the Ensemble Console.

For instructions on configuring credential mapping with SPNEGO authentication, see Section 5.2.3, "Configuring Credential Mapping with SPNEGO Authentication."

To complete the Oracle WebCenter Ensemble / SPNEGO integration, complete the instructions of each of the following sub-sections in the order provided:

1. Section 4.3.4.1, "Configuring Microsoft Active Directory"
2. Section 4.3.4.2, "Configuring the Oracle WebCenter Ensemble Server"
3. Section 4.3.4.3, "Verifying the Oracle WebCenter Ensemble / SPNEGO Integration"

4.3.4.1 Configuring Microsoft Active Directory

Oracle WebCenter Ensemble requires an Active Directory account with which to query the Active Directory. To configure this account:

1. Create a new Active Directory user. Record the OU because you will need it when configuring Kerberos on the Oracle WebCenter Ensemble server. For example, assume the user is in:

```
CN=Users,DC=ensemble,DC=mydomain,DC=com
```

Oracle WebCenter Ensemble will need to use the *ensemble.mydomain.com* realm.

2. Verify that the user account is Kerberos enabled:
   - Turn on **Use DES encryption types for this account**.
   - Verify that **Do not require Kerberos pre-authentication** is not selected.
3. Enable Oracle WebCenter Ensemble to access Active Directory as a service by using the Windows utility **setspn** to create an SPN for Oracle WebCenter Ensemble. For example, type:

   ```
   setspn -a HTTP/ensembleserver.mydomain.com ensembleuser
   ```

   Replace *ensembleserver.mydomain.com* with the fully qualified domain name of your Oracle WebCenter Ensemble server and *ensembleuser* with the user you just created in Active Directory.
4. Create a keytab file for the SPN you created using **ktab**. This file will be used on the Oracle WebCenter Ensemble server to authenticate Oracle WebCenter Ensemble to the Active Directory server. For example, type:

   ```
   ktab -k mykeytab -a HTTP/ensembleserver.fakedomain.com
   ```

   This will create a keytab file, *mykeytab*.
5. Put a backup copy of the keytab file in a secure location. Then copy the keytab file to the Oracle WebCenter Ensemble server.

4.3.4.2 Configuring the Oracle WebCenter Ensemble Server

To configure the Oracle WebCenter Ensemble server to access Active Directory:

1. Copy the keytab file you created in Section 4.3.4.1, "Configuring Microsoft Active Directory," to a location on your Oracle WebCenter Ensemble server. For example: *C:\SPNEGO\mykeytab*
2. Create a new text file named *jaas.conf*. For example: *C:\SPNEGO\jaas.conf*
3. Copy the following into **jaas.conf**:

   ```
   com.sun.security.jgss.krb5.initiate {
   com.sun.security.auth.module.Krb5LoginModule required debug=true
   principal="host/ensembleserver.mydomain.com" useKeyTab=true
   keyTab="c:\\SPNEGO\\mykeytab" storeKey=true;
   };
   com.sun.security.jgss.krb5.accept {
   com.sun.security.auth.module.Krb5LoginModule required debug=true
   principal="host/ensembleserver.mydomain.com" useKeyTab=true
   keyTab="c:\\SPNEGO\\mykeytab" storeKey=true;
   };
   ```

   Replace *host/ensembleserver.mydomain.com* with your SPN and *c:\\SPNEGO\\mykeytab* is your keytab file.

   Note: Use *host/* instead of *HTTP/* for the SPN.

4. Configure the Ensemble Proxy server **wrapper.conf** to refer to your jaas.conf. By default, the Ensemble Proxy server wrapper.conf is located at: ***install_dir*\ensembleproxy\2.0\settings\config**

Add the following lines to wrapper.conf, replacing *C:\SPNEGO\jaas.conf* with the location of your jaas.conf. You must add the lines near the top of the wrapper.conf, in the section titled *Additional -D Java Properties*. You must number the **wrapper.java.additional.#** properties consecutively in ascending order, starting with wrapper.java.additional.19. The wrapper.java.additional.19 property will already exist. Add the following lines:

```
wrapper.java.additional.21=-Djava.security.auth.login.config=C:\SPNEGO\jaas.con
f
wrapper.java.additional.22=-Djavax.security.auth.useSubjectCredsOnly=false
wrapper.java.additional.23=-Dsun.security.krb5.debug=true
```

5. Create a **krb5.ini** file in your winnt directory. For example: *C:\winnt\krb5.ini*
6. Copy the following into the krb5.ini file you created:

```
[libdefaults]
udp_preference_limit = 1
default_realm = ENSEMBLE.MYDOMAIN.COM
default_tkt_enctypes = des-cbc-crc
default_tgs_enctypes = des-cbc-crc
ticket_lifetime = 600

[realms]
ENSEMBLE.MYDOMAIN.COM = {
kdc = ADSERVER.MYDOMAIN.COM
admin_server = ADSERVER.MYDOMAIN.COM
default_domain = ENSEMBLE.MYDOMAIN.COM
}

[domain_realm]
. ENSEMBLE.MYDOMAIN.COM = ENSEMBLE.MYDOMAIN.COM
ENSEMBLE.MYDOMAIN.COM = ENSEMBLE.MYDOMAIN.COM

[appdefaults]
autologin = true
forward = true
forwardable = true
encrypt = true
```

7. Edit the krb5.ini file so that:
   - ENSEMBLE.MYDOMAIN.COM is the realm (OU) of the server user account you created on your Active Directory server.
   - ADSERVER.MYDOMAIN.COM is the fully qualified domain name of your Active Directory server.
8. Retrieve the **shared secret key** from the Oracle WebCenter Interaction portal database. Open the PTSERVERCONFIG table. The shared secret key is VALUE where SETTINGID=65. For example:

```
select VALUE from PTSERVERCONFIG where SETTINGID=65;
```

9. Add the shared secret key to the Oracle WebCenter Ensemble configuration.xml. On the Oracle WebCenter Ensemble server, configuration.xml is located by default at: ***install_dir*\settings\runner\configuration.xml**

   In configuration.xml, ensure the value of the following setting is your shared secret key:

```
<setting name="runnersso:ssologin:sharedSecretKey">
```

```
    <value xsi:type="xsd:string">[Your shared secret key]</value>
</setting>
```

10. In Oracle WebCenter Configuration Manager, browse to **ENSEMBLE** > **ALUI Security Login Tokens** and ensure that you have correctly configured this component's settings.

    Correct configuration includes setting the default token type to **ALUI** and providing a message authentication code seed value for ALUI login tokens.

11. Restart the BEA AL Ensemble Administrative UI, and the BEA AL Ensemble Proxy.

4.3.4.3 Verifying the Oracle WebCenter Ensemble / SPNEGO Integration

Verify the login resource is correctly configured. Create a resource and policy to protect it with your SPNEGO authentication level and configure your experience rules to request SPNEGO authentication when the user accesses the resource.

- For details on creating resources, see Chapter 3, "Proxy Resources."
- For details on configuring policies, see Chapter 6, "Policies and Rules."
- For details on configuring authentication levels, see Section 4.2, "Configuring Authentication Levels."
- For details on configuring experience rules, see Chapter 7, "Experience Definitions."

> **Note:** For SPNEGO authentication to work from the client side, the user must be logged into Windows into the appropriate Active Directory domain. In addition, Internet Explorer must be configured so that the Oracle WebCenter Ensemble server is in the *Local Intranet* zone and integrated Windows authentication must be enabled.

4.3.5 Integrating with Oracle Virtual Directory

This section provides details about configuring Oracle WebCenter Ensemble to access multiple LDAP servers via Oracle Virtual Directory.

To integrate Oracle WebCenter Ensemble with Oracle Virtual Directory:

1. Navigate to **Oracle WebCenter Configuration Manager** > **Ensemble** > **ALUI Directory**.
2. Ensure that the authentication provider is set to **LDAP**.
3. Ensure that the values for the settings in the ALUI Directory component are correctly configured to point to the Oracle Virtual Directory server.

> **Note:** In Oracle WebCenter Configuration Manager, the value for the **User Authentication Name** setting must match the user authentication name attribute that is assigned to all users, including the Principal user. The value for the **Principal** setting can be the distinguished name of any administrative account in Oracle Virtual Directory, and must be searchable by the authentication name attribute. If these settings are not configured correctly, the Ensemble Admin UI will not start properly, and users will not be able to log in.

4.3.6 SSO Logout

A user may be accessing multiple resources under a single SSO authentication. When a user logs out of an Oracle WebCenter Ensemble proxied resource, Oracle WebCenter Ensemble can prompt the user to log out of only that application or all applications.

For Oracle WebCenter Ensemble to capture logout attempts, you must configure one or more internal logout patterns for each resource.

To configure SSO log out patterns:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the name of the resource you want to edit.
5. On the SSO Log Out Settings page, type the regular expression pattern that matches your log out page into the **Internal log out URL patterns** box.
6. To add more patterns, click **Add**.
7. To delete patterns, click **Delete**.
8. When you are done configuring SSO Log Out Settings, click **Save**.

5 Credential Mapping

This chapter describes how to configure credential mapping for Oracle WebCenter Ensemble resources. It is divided into the following topics:

- Section 5.1, "About Credential Mapping," describes what credential mapping is and how it can be used.
- Section 5.2, "Configuring Credential Mapping," provides details on how to configure credential mapping.

5.1 About Credential Mapping

Credential mapping allows Oracle WebCenter Ensemble to supply credentials to proxied applications automatically. The credentials used by Oracle WebCenter Ensemble to log in to the application can come from:

- The Credential Vault. When the user logs into the proxied resource, her credentials are stored in the Credential Vault. Subsequent access to that resource is authenticated using the stored credentials.
- The user's Oracle WebCenter Interaction or LDAP profile. Credentials for specific applications can be stored in the user's profile and used by Oracle WebCenter Ensemble to automatically log the user into proxied applications.
- Static credentials. The Oracle WebCenter Ensemble resource can be configured with static credentials that are used for every user with access to the resource.

5.2 Configuring Credential Mapping

Oracle WebCenter Ensemble can automatically log in to resources through HTML forms and basic authentication.

The following sections describe how to configure credential mapping for authentication:

- Section 5.2.1, "Configuring Credential Mapping for HTML Forms," describes how to configure a resource to log in automatically to a resource that prompts for authentication with an HTML form.
- Section 5.2.2, "Configuring Credential Mapping with Basic Authentication," describes how to configure a resource to log in automatically to a resource that prompts for authentication with basic authentication.
- Section 5.2.4, "Authentication Field Sources," describes the static, user profile, and credential vault authentication field sources.

5.2.1 Configuring Credential Mapping for HTML Forms

This section describes how to configure credential mapping for a resource that prompts for authentication with an HTML form.

To configure a resource for HTML form credential mapping:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the name of the resource you want to edit.
5. On the Credential Mapping page, next to **Status**, select **Enabled**.
6. Next to **Login Method**, select **HTML Form**.
7. Create a new login form mapping by clicking **New Form Configuration**.
8. The login page can be identified by an URL or a regular expression:
   - If the login form is located at a static URL, select **An URL** and type the URL into the box.
   - If the login form is dynamic, select **A Regular Expression** and type the regular expression pattern into the box.
9. Map one or more field values to authentication field sources.
   a. Type the name of the HTML form input in the **Field Name** box.
   b. For details on how to configure the Source and Mapped Value properties, see Section 5.2.4, "Authentication Field Sources."
   c. To automatically detect and populate field mappings, click **Detect Form Fields**.
   d. To add additional field mappings, click **Add**.
   e. To delete field mappings, click the delete icon.
10. Set the login form action.
    - If the login form action is a static URL, select **An URL** and type the URL into the box.
    - If the login form is dynamic, select **The url returned by the above regular expression** and type the regular expression pattern into the box.
11. To submit the login form data as an HTTP POST, select **Submit action as post**. Otherwise, login form data will be submitted as an HTTP GET.

5.2.2 Configuring Credential Mapping with Basic Authentication

This section describes how to configure credential mapping for a resource that prompts for authentication with basic authentication.

To configure a resource for basic authentication credential mapping:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the name of the resource you want to edit.

5. On the Credential Mapping page, next to **Status**, select **Enabled**.
6. Next to **Login Method**, select **Basic**.
7. Enter values for **Basic Auth Username** and **Basic Auth Password**. For details on how to configure the Credential Source and Credential Value properties, see Section 5.2.4, "Authentication Field Sources."

5.2.3 Configuring Credential Mapping with SPNEGO Authentication

This section describes how to configure credential mapping for a resource that prompts for SPNEGO authentication.

For instructions on integrating with Microsoft Active Directory via SPNEGO, see Section 4.3.4, "Integrating with Microsoft Active Directory via SPNEGO."

To configure a resource for SPNEGO authentication credential mapping:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Resources** sub-tab.
4. Click the name of the resource you want to edit.
5. On the Credential Mapping page select **Send Spnego Token**.

   The Spnego token enables Spnego authentication with the resource.
6. Click **Save**.

5.2.4 Authentication Field Sources

Authentication field sources map values to login fields. The following table describes each of the authentication field source values in the **Source** drop-down:

Table 5–1 Authentication Field Sources

Source	Description
Static	Use the static source when the authentication field is the same for all users accessing the resource. Type the static value in the **Mapped Value** box.
Masked Static	The masked static source is like the static source, except that the value typed into the **Mapped Value** box is obscured in the Ensemble Console UI. Use this source to protect the values of passwords and other sensitive fields.
User Profile	The user profile source uses properties from the user's Oracle WebCenter Interaction profile to supply credential data for authentication. For each form field with a user profile source, select the profile property from the picker.
Ensemble authentication credentials	Ensemble authentication credentials are the credentials a user has with Ensemble. In the **Credential Value** field, supply either the text *username* or password to specify whether the field should map to the user's username or password values.
Credential Vault	With the Credential Vault source, Oracle WebCenter Ensemble prompts the user for credentials the first time she accesses the resource. The supplied credentials are stored in the credential vault, and each subsequent access to that resource is authenticated with the stored credentials.

6 Policies and Rules

This chapter describes how to use policies and rules to control access to Oracle WebCenter Ensemble resources. It is divided into the following sections:

- Section 6.1, "About Policies and Rules," provides an overview of how policies and rules control access to an Oracle WebCenter Ensemble resource.
- Section 6.2, "Policies," describes policies in depth, including how to create policies and configure policy sets.
- Section 6.3, "Rules," describes rules in depth, including what kinds of rules can be created and how to create them.

6.1 About Policies and Rules

Each non-login resource has an associated *policy set*. A policy set is a collection of *policies* that control access to a resource. Each policy grants access to a resource based on two criteria:

- Users and Groups. A user must be among the users or groups configured in the policy.
- Rules. A set of rules, one or all of which must evaluate to true for the user to have access.

For details on creating and configuring policies, see Section 6.2, "Policies."

Rules describe a set of criteria that must be met. If the criteria are met, the rule evaluates to true. For example, a rule could restrict access to business hours or evaluate to true when the user's client is a specific browser. For details on creating and configuring rules, see Section 6.3, "Rules."

In addition to controlling access to a resource, policies associate a *role* with the user. Role information is sent to the proxied application, allowing the application to determine the correct access level for the user. Since more than one policy can be granted for a given user on a given resource, more than one role can be associated with a user. Roles are created with the resource configuration. For details on configuring roles, see Section 3.3.2, "Roles."

6.2 Policies

When you create a resource, Oracle WebCenter Ensemble creates a default policy for that resource. Policy sets map to resources 1:1. The name of the policy set is the same as the name of the resource and cannot be changed.

When Oracle WebCenter Ensemble creates the policy, it creates a default policy for that policy. The default policy grants the Administrator user access to the resource. You can edit or delete this policy, and you can add new policies.

6.2.1 Creating a New Policy

To create a new policy:

1. Launch the Ensemble Console.
2. Click the **POLICIES** tab.
3. Click the **Policy Sets** sub-tab.
4. Click the name of the policy set associated with the resource you are configuring.
5. On the Policies page, click **Add policy**.

6.2.2 Configuring a Policy

A policy consists of four properties:

- A name.
- The resource role the policy maps to. Roles are configured in the resource configuration. For details, see Section 3.3.2, "Roles."
- One or more rules that describe the conditions for access.
- Zero or more users or groups that are allowed access by this policy.

At minimum, a policy must have a name, a mapped resource role, and an associated rule.

To configure a policy:

1. Launch the Ensemble Console.
2. Click the **POLICIES** tab.
3. Click the **Policy Sets** sub-tab.
4. Click the name of the policy set associated with the resource you are configuring.
5. On the Policies page, expand the policy you want to configure by clicking the expand icon.
6. Type a **Name** for the policy.
7. Associate a role with the policy. In the **Maps to Resource Role** drop-down list, select a role.
8. Associate one or more rules with the policy:
    a. Click **Add Rule**.
    b. Select the rule or rules you want to add.
    c. Click **Add selected items**.
    d. Click **OK**.
    e. Select **ANY** or **ALL**. When ANY is selected, and one or more rule evaluates to true, the policy will evaluate to true (provided any users and groups restrictions are satisfied). When ALL is selected, all rules must evaluate to true.
9. Restrict the policy to specific users or groups (optional).

a. Click **Add User or Group**.

b. Select the users or groups you want to add.

c. Click **Add selected items**.

d. Click **OK**.

To delete users, groups, or rules, highlight the item to be deleted and click **Delete**.

6.2.3 Authentication Levels

Authentication levels determine the minimum credential level required to access a resource. Oracle WebCenter Ensemble checks the authentication level of a policy set before it evaluates any policies. If the user is not logged in, or is logged in with credentials lower than the set authentication level, he is challenged with the authentication method.

For details on authentication, see Chapter 4, "Proxy Authentication."

6.2.4 Configuring Anonymous Access

Anonymous access allows user to access a resource without providing credentials. This is useful for resources such as login resources, where the user is not expected to be authenticated prior to accessing the resource.

To configure anonymous access:

1. Launch the Ensemble Console.
2. Click the **POLICIES** tab.
3. Click the **Policy Sets** sub-tab.
4. Click the name of the policy set associated with the resource you want to configure for anonymous access.
5. Set the authentication level to Anonymous. In the **Minimum Credential Level** drop-down, select *0 (Anonymous)*.
6. When prompted, create an anonymous policy. Select a resource role from the drop-down and click **Create anonymous policy**.
7. Click **Save**.

A new policy, *Anonymous policy*, is created. This policy always evaluates to true for any user.

6.2.5 Granting Access to Users Who Are Currently Logged in to Oracle WebCenter Interaction

To allow a user who has already logged into the Oracle WebCenter Interaction portal to be granted access to the resource without authenticating with Oracle WebCenter Ensemble, perform the following:

1. Launch the Ensemble Console.
2. Click the *POLICIES* tab.
3. Click the **Policy Sets** sub-tab.
4. Click the name of the policy set associated with the resource you want to configure.
5. Check the box next to **Allow Portal Login Token**.

6. Click **Save**.

6.3 Rules

Rules are defined by one or more *rule types*. A rule type is a single condition that evaluates to true or false. The rule is configured so that either any or all of the rule types must evaluate to true for the rule to evaluate to true. The following table describes the available rule types:

Table 6–1 Rule Types

Rule Type	Description
Client IP	Evaluates to true if this value matches the user's IP. You can configure the Client IP rule to match a range of IP addresses by using regular expressions.
Date	You can set the Date rule to be equal to, greater than, less than, greater than or equal to, or less than or equal to a given date. You can combine two Date rule types to provide access over a range of dates.
User	Evaluates to true if this value is the current user.
Secure connection	Evaluates to true of the connection is secure (HTTPS).
Time	You can set the Time rule to be equal to, greater than, less than, greater than or equal to, or less than or equal to a given time. You can combine two Time rule types to provide access over a period of time.
Browser	Evaluates to true if this value matches the user's browser type.
Group membership	Evaluates to true if this value is a group of which the user is a member.
Non-secure connection	Evaluates to true of the connection is not secure (HTTP).
Day of Week	Evaluates to true if this value is equal to the current day of the week.
Locale	Evaluates to true if this value matches the user's locale.
User property	Evaluates to true if this value matches the user's property value.
Always true	Always evaluates to true.
Always false	Always evaluates to false.

6.3.1 Creating and Editing Rules

You create rules in the rule library. To create a new rule:

1. Launch the Ensemble Console.
2. Click the **POLICIES** tab.
3. Click the **Rule Library** sub-tab.
4. To create a new rule, click **Create new**.
5. On the General page, in the **Name** box, type the name of the rule.
6. Type a **Description** of the rule.
7. On the Definition page, click **Add**.
8. Either select the rule type to create or click on an existing rule.

   Existing rules can be added as rule types. This allows compound rules to be formed. For example, a rule might evaluate to true if any of three users is accessing the resource from a secure connection. A rule type is created that

evaluates to true for *any* of the three uses. That rule type is added to a rule type where it *and* the Secure connection rule type must evaluate to true.

9. Add the rule type by clicking **OK**.
10. Click **Add** to add another rule type or finish creating the rule by clicking **Save**.

6.3.2 Published Rules

You can configure a rule to be published or not published. You are able to add a published rule to a policy. You are able use an unpublished rule only as a rule type for other rules.

To publish a rule, from the rule's General page, select **Is published**. To unpublish the rule, clear the check box next to **Is published**.

Note: If the rule is currently being used in a policy, it cannot be unpublished.

---

7 Experience Definitions

This chapter describes how to configure Oracle WebCenter Ensemble user experiences. It is divided into the following sections:

- Section 7.1, "About Experience Definitions," provides an overview of how experience definitions control aspects of the user experience.
- Section 7.2, "Configuring Experience Definitions," describes how to create and configure experience definitions.
- Section 7.3, "Configuring Experience Rules," describes how to create and configure experience rules.
- Section 7.4, "Login Resources and Interstitial Pages," describes how login resources and interstitial pages are used in the login and logout process.

7.1 About Experience Definitions

An *experience definition* describes the following aspects of the user experience:

- The authenticator used to authenticate the user.
- The login, logout, error, and other interstitial pages displayed to the user.

Oracle WebCenter Ensemble employs a set of *experience rules* to determine which experience definition to associate with a user. Each experience rule evaluates to true if its set of conditions is satisfied. Experience rules are evaluated in order, and the first rule to evaluate to true determines the experience definition that is associated with the user.

7.2 Configuring Experience Definitions

To configure an experience definition:

1. Launch the Ensemble Console.
2. Click the **EXPERIENCES** tab.
3. Click the **Definitions** sub-tab.
4. Click the experience definition you want to edit, or to create a new resource, click **Create new**.
5. On the General page, type a **Name** and **Description**for the experience definition.
6. On the Log In Settings page, configure the login resource and interstitial pages. For details, see Section 7.4, "Login Resources and Interstitial Pages."

7. On the Authentication Settings page, select an **Authentication method** from the drop-down.

> **Caution:** Oracle WebCenter Ensemble uses the authentication method set in the experience definition if it meets or exceeds the authentication level required by the resource being accessed. If the resource requires a greater authentication level, Oracle WebCenter Ensemble uses the authentication method appropriate for that authentication level.

8. Click **Save**.

7.3 Configuring Experience Rules

The experience definition that Oracle WebCenter Ensemble chooses depends on a set of experience rules that Oracle WebCenter Ensemble evaluates in a specified order. You configure experience rules by first adding or editing rules in the Rule Library. You then set the precedence of rules in the Rule Order.

Rules are defined by one or more *rule types*. A rule type is a single condition that can be evaluated as true or false. You can configure the rule so that any or all of the rule types must evaluate to true for the rule to evaluate to true. The following table describes the available rule types:

Table 7–1 Rule Types

Rule Type	Description
Client IP	Evaluates to true if this value matches the user's IP. You can configure the Client IP rule to match a range of IP addresses by using regular expressions.
Date	You can set the Date rule to be equal to, greater than, less than, greater than or equal to, or less than or equal to a given date. You can combine two Date rule types to provide access over a range of dates.
User	Evaluates to true if this value is the current user.
Secure connection	Evaluates to true of the connection is secure (HTTPS).
Time	You can set the Time rule to be equal to, greater than, less than, greater than or equal to, or less than or equal to a given time. You can combine two Time rule types to provide access over a period of time.
Browser	Evaluates to true if this value matches the user's browser type.
Group membership	Evaluates to true if this value is a group of which the user is a member.
Non-secure connection	Evaluates to true of the connection is not secure (HTTP).
Day of Week	Evaluates to true if this value is equal to the current day of the week.
Locale	Evaluates to true if this value matches the user's locale.
User property	Evaluates to true if this value matches the user's property value.
Always true	Always evaluates to true.
Always false	Always evaluates to false.

7.3.1 Creating and Editing Rules in the Rule Library

1. Launch the Ensemble Console.

2. Click the **EXPERIENCES** tab.
3. Click the **Rule Library** sub-tab.
4. Click **Create new**.
5. On the General page, in the **Name** box, type the name of the rule.
6. Type a **Description** of the rule.
7. On the Definition page, click **Add**.
8. Either select the rule type to create or click on an existing rule.

   You can add existing rules as rule types. This allows compound rules to be formed. For example, you might create a rule that evaluates to true if any of three users is accessing the resource from a secure connection. To do this, you create rule type that evaluates to true for *any* of the three users. You then add that rule type to a rule type where it *and* the Secure connection rule type must evaluate to true.
9. Add the rule type by clicking **OK**.
10. Click Add to add another rule type, or finish creating the rule by clicking **Save**.

7.3.2 Published Rules

You can configure a rule to be published or not published. You are able to add a published rule to a policy. You are able use an unpublished rule only as a rule type for other rules.

To publish a rule, from the rule's General page, select **Is published**. To unpublish the rule, clear the check box next to **Is published**.

Note: If the rule is currently being used in the Rule Order, it cannot be unpublished.

7.3.3 Rule Order

The Rule Order sub-tab associates experience rules with experience definitions and provides the order in which Oracle WebCenter Ensemble evaluates the rules. When determining the experience definition, Oracle WebCenter Ensemble first checks the first (lowest numbered) rule in the Rule Order. If the experience rule evaluates to true, Oracle WebCenter Ensemble associates the experience definition with the user. If the experience rule evaluates to false, the next rule in the order is checked, and so on, until an experience rule evaluates to true and Oracle WebCenter Ensemble can associate an experience definition with the user.

To change the order of rules, adjust the numbers in the **Order** column.

To create a new rule in the Rule Order:

1. Launch the Ensemble Console.
2. Click the **EXPERIENCES** tab.
3. Click the **Rule Order** sub-tab.
4. Click **Add Rule**.
5. On the General page, in the **Name** box, type the name of the rule.
6. Type a **Description** of the rule.

7. On the Settings page, next to **Condition**, click **Select**.
8. Select the rule. Click **OK**.
9. Next to **Experience definition**, click **Select**.
10. Select the experience definition to be assigned if this rule is selected. Click **OK**.
11. Click **Save**.

7.4 Login Resources and Interstitial Pages

The experience definition associated with a user determines, in part, the specifics of the user's login and logout experience. On the **Log In Settings** page of the experience definition configuration you can supply the login resource and login, logout, error, and interstitial pages.

> **Note:** Only the login page is required. The logout, error, and interstitial pages are not required, although an error page is recommended.

The login resource is a proxied application server used to host the various pages associated with the experience definition.

To create a login resource, create a resource and select **Is Login resource**on the General page.

For details on creating a resource, see Chapter 3, "Proxy Resources."

The following table describes the various pages that you can associate with an experience definition.

Table 7–2 Login, Logout, and Interstitial Page Settings

Setting	Definition
Pre-log in page	Oracle WebCenter Ensemble displays this page to the user prior to attempting to authenticate the user.
Login page	This page provides the form for login when the authenticator is HTML form-based login. Oracle WebCenter Ensemble displays this page after the Pre-log in page and before the Post-log in page.
Post-log in page	Oracle WebCenter Ensemble displays this page to the user after successful authentication and before the user accesses the resource.
Error page	Oracle WebCenter Ensemble displays this page if there is an error in the login process.
Post-log out page	Oracle WebCenter Ensemble displays this page after the user logs out of the resource.

For details on customizing the login, logout, error, and other interstitial pages, see Section 11.1, "Custom Login Resources."

Caution: Oracle WebCenter Ensemble uses the login, logout, error, and interstitial page settings in the experience definition regardless of the final authenticator used to access the resource. Oracle WebCenter Ensemble uses an authenticator other than the authenticator configured with the experience definition if the resource being access requires a higher authentication level. If the required authenticator uses a login page and there is no login page configured in the experience definition, the user is presented with a blank page and is unable to authenticate.

8

Pagelets

This chapter describes how to use Oracle WebCenter Ensemble to create and deploy pagelets. It is organized into the following sections:

- Section 8.1, "About Pagelets," describes what pagelets are.
- Section 8.2, "Registering a Pagelet," describes how you register a pagelet with Oracle WebCenter Ensemble.
- Section 8.3, "Using Lightweight Clipping," describes how to form a pagelet by *clipping* a portion of a larger web page in a proxied application.
- Section 8.4, "Adding Pagelets to Web Pages," describes how to add pagelets to applications that are proxied and not proxied by Oracle WebCenter Ensemble.
- Section 8.5, "Configuring Pagelet Parameters and Transport Type," describes how to pass parameters to your pagelets.
- Section 8.6, "Configuring Metadata Fields," describes how to configure metadata fields to store additional information about a pagelet.
- Section 8.7, "Configuring Pagelet Consumers," describes how to restrict the resources that can use a pagelet.
- Section 8.8, "Accessing Pagelet Discovery for Developers," describes the Oracle WebCenter Ensemble pagelet discovery UI.
- Section 8.9, "Exposing Oracle WebCenter Analytics Pagelets through Oracle WebCenter Ensemble," describes how to import the Oracle WebCenter Analytics resource migration file and add Oracle WebCenter Analytics and Oracle WebCenter Interaction image server files to Oracle WebCenter Ensemble in order to expose Oracle WebCenter Analytics reports as pagelets through Oracle WebCenter Ensemble.
- Section 8.10, "Exposing BEA AquaLogic Pathways Pagelets through Oracle WebCenter Ensemble," describes how to import the BEA AquaLogic Pathways resource migration file in order to expose BEA AquaLogic Pathways pagelets through Oracle WebCenter Ensemble.

8.1 About Pagelets

A *pagelet* is a fragment of HTML that describes a self-contained, reusable UI element. With a portal system, a portlet is a self-contained UI element that can be used in the portal. A pagelet is like a portlet that you can easily insert into any web page proxied by Oracle WebCenter Ensemble.

When writing pagelets, developers can make use of the Oracle WebCenter Interaction Development Kit (IDK) proxy API, the client Scripting Framework and Oracle

WebCenter Ensemble Adaptive Tags. Pagelets are hosted on a resource and are consumed by other resources via the pagelet Adaptive Tag.

8.2 Registering a Pagelet

A pagelet is an application hosted on an Oracle WebCenter Ensemble resource. Before registering a pagelet in Oracle WebCenter Ensemble, you must create a resource and configure it to point to the application server where the pagelets are hosted. For details on creating resources, see Chapter 3, "Proxy Resources."

To register a new pagelet:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. To create a new pagelet, click **Create new**.
5. On the General page, type the **Name** of the pagelet.
6. Select a parent resource. Next to **Parent resource**, click select. From the **Select a resource** picker, select the resource your pagelet is hosted on and click **OK**.
7. Type the **Library name** to associate this pagelet with. This can be any text value. The library setting is a user-defined way of grouping pagelets together within the pagelet documentation. This setting is optional.
8. Type a **Description**.
9. By default, a pagelet is included in the generated pagelet documentation. Clear the **Publish documentation** check box if the pagelet should not be published.
10. Select the **Add inline refresh to all URLs** check box if you want Oracle WebCenter Ensemble to automatically perform inline refresh.
11. Type the **Refresh Interval** -- in milliseconds -- at which Oracle WebCenter Ensemble refreshes the pagelet.
12. Click **Save**.
13. On the Location page, perform the following:
    - Type the **URL suffix**. The Internal URL prefix (taken from the parent resource) appended to the URL suffix forms the URL to the pagelet.
    - Choose whether the pagelet returns the whole contents of the source, or a clipped region of the source. For details on using clipping, Section 8.3, "Using Lightweight Clipping."

Metadata can be used to store additional information about a pagelet. For details, see Section 8.6, "Configuring Metadata Fields."

Once you save the pagelet, sample code for inserting the pagelet into a web page is available on the General page of the pagelet configuration. For details on adding pagelets to web pages, see Section 8.4, "Adding Pagelets to Web Pages."

Oracle WebCenter Ensemble can restrict the resources that can insert each pagelet into its web pages. For details, see Section 8.7, "Configuring Pagelet Consumers."

Parameters can be configured to be passed to the pagelet. For details, see Section 8.5, "Configuring Pagelet Parameters and Transport Type."

8.2.1 Example: Creating and Accessing Pagelets

This section provides examples for creating a pagelet in Oracle WebCenter Ensemble, and accessing a pagelet via Oracle WebCenter Ensemble.

8.2.1.1 Example: Creating a Pagelet in the Ensemble Console

This procedure provides an example of creating a pagelet in the Ensemble Console.

1. Click the **APPLICATIONS** tab.
2. Click the **Resources** sub-tab.
3. Click **Create new**.
4. On the **General** tab, type a name for the resource.
5. On the **Location** tab, provide an internal URL prefix and external URL prefix.

   For the internal URL prefix, specify the parent URL of the pagelet. For example, if the pagelet is at http://www.foo.org/bar/pagelet1.html, specify `http://www.foo.org/bar/` as the Internal URL.

   For the external URL prefix, choose the URL that you will use to call the pagelet.
6. Save the resource.

   You just created the pagelet's producer or *parent* resource.
7. Click the **APPLICATIONS** tab.
8. Click the **Pagelets** sub-tab.
9. Click **Create new**.
10. On the **General** tab, type a name for the pagelet.
11. Select the parent resource that you just created.
12. Type a library name for the pagelet.

    The library is name that groups pagelets together; it does not exist outside of this context. You can create a unique library name for each pagelet, or share one library name across multiple pagelets. The library name serves only as a logical grouping.
13. Note the **Sample code** section.

    This example will return to this section when using the pagelet; the contents of the Sample code section are automatically generated when saving the pagelet.
14. Click the **Location** tab.

    The internal URL prefix is automatically populated because it is inherited from the parent resource.
15. Add the URL suffix that completes the internal URL of the pagelet.

    In the above example, the Internal URL prefix is http://www.foo.org/bar/. For the URL suffix, you would add `pagelet1.html`.

    Note: Unlike resources, pagelets are actual file names rather than directory names.
16. Click **Save**.

8.2.1.2 Example: Accessing a Pagelet via Oracle WebCenter Ensemble

This procedure provides an example of accessing a pagelet via Oracle WebCenter Ensemble.

To access a pagelet via Oracle WebCenter Ensemble:

1. Create a consumer resource.

   This consumer resource will call the pagelet that you created in Section 8.2.1.1, "Example: Creating a Pagelet in the Ensemble Console." Pagelets are not accessed directly; they are accessed in the context of resources that *consume* the pagelets.

   a. Click the **APPLICATIONS** tab.

   b. Click the **Resources** sub-tab.

   c. Click **Create new**.

   d. On the **General** tab, type a name for the resource.

   e. On the **Connections** tab, type an internal URL prefix.

      The internal URL prefix should specify the location of the HTML file that you will create. This HTML file will call the pagelet.

   f. Provide an external URL prefix; visiting this URL will access the consumer page. The consumer page will access the pagelet that you created in Section 8.2.1.1, "Example: Creating a Pagelet in the Ensemble Console."

   g. Click **Save**.

2. Create the HTML file that will access the pagelet.

   a. Click the Pagelets sub-tab.

   b. Click the name of the pagelet that you created in Section 8.2.1.1, "Example: Creating a Pagelet in the Ensemble Console."

   c. Click the **General** tab.

   d. Note the **Sample code** section.

      This section contains the HTML that calls the pagelet when the page containing it is proxied through Oracle WebCenter Ensemble. It is automatically generated for each pagelet, and contains the library name and pagelet name.

   e. Create a web page with some content.

      This content can be as simple as the following:

      ```
      <html><head></head><body></body></html>
      ```

   f. Copy the sample code that you noted in the previous step, and paste it into the body of the web page.

   g. Declare the XML namespace by entering the following in front of the sample code:

      ```
      <div xmlns:pt='http://www.plumtree.com/xmlschemas/ptui/'>
      ```

      After the sample code, type the following:

      ```
      </div>
      ```

      Alternatively, you can simply insert the following into the tag specified in the sample code:

```
xmlns:pt='http://www.plumtree.com/xmlschemas/ptui/
```

3. The following is an example HTML file:

```
<html>
<head>
</head>
<body>
<p>
This is a pagelet test page.
</p><p>
<div xmlns:pt='http://www.plumtree.com/xmlschemas/ptui/'>
<pt:ensemble.inject pt:name="Pagelet Library:Pagelet One">
</pt:ensemble.inject>
</div>
</p>
</body>
```

4. Save the file in the location specified in the consumer resource's internal URL.

> **Note:** If you save the file as *index.html*, you will not need to access it by filename when clicking the external URL. If you do not save this file as index.html, you will need to access it by *Internal_URL*/filename.html.

5. Access the consuming resource's external URL.

   Oracle WebCenter Ensemble: 1) retrieves the HTML from the file that was created at the internal URL location, 2) inserts the pagelet into the HTML file, and 3) presents the entire page to the user's browser.

8.3 Using Lightweight Clipping

Lightweight clipping allows you to form a pagelet by *clipping* a portion of a larger web page in a proxied application. For example, in a news web page there is a box listing the latest headlines. By identifying the containing HTML for that box, you can clip only the headlines and serve that subset of the news web page as an Oracle WebCenter Ensemble pagelet.

In addition to clipping a portion of the body of the web page, the pagelet's <*head*> element can also be included. This allows CSS, JavaScript, or other declarations that occur in the <*head*> element to be included with the clipped body portion.

To use lightweight clipping:

1. Click the **APPLICATIONS** tab.
2. Click the **Pagelets** sub-tab.
3. Open the Pagelet Editor:
   - To create a new pagelet, click **Create New** under the pagelets view.
   - To edit an existing pagelet, navigate to the pagelet you want to edit. Then click the pagelet name.
4. Click **Location**.
5. Choose from one of the following choices for lightweight clipping:
   - Select **Clip nothing** and all content will be returned with this pagelet.

- Select **Clip a region using Intelligent Matcher** and choose the section to be clipped. Next, click **Select Intelligent Matcher Clip Region**. In the pop-up display, use the mouse to move the red box. Then click when selection you want clipped is outlined. To include the <*head*> element from the pagelet, select **Also include contents of pagelet's <head> tag**.
- Select **Clip a region using tag attributes** and specify HTML tag attributes that describe the section to be clipped. Next, type the name of the tag in the **Tag** type box. For example, *div*. Then, type an attribute name and value in the **Attribute Name** and **Attribute Value** boxes. If more attributes are required, click **Add**. Alternately, click **Select Clip Region** and choose the section to be clipped. To include the <*head*> element from the pagelet, select **Also include contents of pagelet's <head> tag**.

Note: If you are clipping an element where there are multiple identifiable instances (for example, a *divs* with a *class* attribute that is the same for a number of *divs* on the same page,) only one div will be returned. Using **Clip a region using tag attributes** will return the first occurrence of an element that matches the attribute value. To clip an occurrence other than the first occurrence, you must use **Clip a region using Intelligent Matcher**.

8.4 Adding Pagelets to Web Pages

Oracle WebCenter Ensemble allows you to inject pagelets into web pages that are proxied, as well as not proxied by Oracle WebCenter Ensemble. You can also use Oracle WebCenter Ensemble as a portlet provider for a portal such as Oracle WebCenter Interaction or Oracle WebLogic Portal.

8.4.1 Overview of HTML Changes to Injected Markup

This section describes the changes that Oracle WebCenter Ensemble makes to HTML code when injecting pagelets into consuming web pages. Note that Oracle WebCenter Ensemble does not make changes to:

- HTML that is injected using lightweight clipping.
- HTML that is wrapped in an IFrame.

Oracle WebCenter Ensemblee creates fragments out of pagelet HTML, separating the code into <HEAD> and <BODY> sections that can be injected into the necessary code locations of the consuming web page.

8.4.1.1 Example 1: Injecting Markup that Includes Adaptive Tags

In this example, Oracle WebCenter Ensemble injects a pagelet's <HEAD> contents into the <HEAD> section of the consuming page.

Representation of a consumer page's code before injection:

```
<HTML>
<HEAD>
Consumer page script
</HEAD>
<BODY>
<pt:ensemble.inject ...>
</BODY>
</HTML>
```

Representation of a pagelet's code before injection:

```
<HTML>
<HEAD>
Pagelet script
</HEAD>
<BODY>
Pagelet body
</BODY>
</HTML>
```

Representation of the consumer web page's code after injection of the pagelet:

```
<HTML>
<HEAD>
Consumer page script
Pagelet script
</HEAD>
<BODY>
<DIV pageletcontainer>
Pagelet body
</DIV>
</BODY>
</HTML>
```

8.4.1.2 Example 2: Injecting Markup that Includes JavaScript

In this example, Oracle WebCenter Ensemble does not render the <HEAD> contents of the consumer page, because the consumer page is already rendered. Oracle WebCenter Ensemble adds the <HEAD> contents above the <DIV> pagelet container in the pagelet body.

Representation of a consumer page's code before injection:

```
<HTML>
<HEAD>
Consumer page script
</HEAD>
<BODY>
<script>
injectpagelet(...);
</script>
</BODY>
</HTML>
```

Representation of a pagelet's code before injection:

```
<HTML>
<HEAD>
Pagelet script
</HEAD>
<BODY>
Pagelet body
</BODY>
</HTML>
```

Representation of the consumer web page's code after injection of the pagelet:

```
<HTML>
<HEAD>
Consumer page script
</HEAD>
<BODY>
Pagelet script
<DIV pageletcontainer>
```

```
Pagelet body
</DIV>
</BODY>
</HTML>
```

8.4.1.3 Example 3: Injecting Markup that Includes JavaScript and an IFrame

In this example, Oracle WebCenter Ensemble does not need to separate the <HEAD> and <BODY> sections, because the pagelet is being injected into an IFrame, whose code is separated from the surrounding web page. For this reason, Oracle WebCenter Ensemble returns the pagelet as it appears on the server.

Representation of a consumer page's code before injection:

```
<HTML>
<HEAD>
Consumer page script
</HEAD>
<BODY>
<script>
injectpagelet(..., 'iframe', ...);
</script>
</BODY>
</HTML>
```

Representation of a pagelet's code before injection:

```
<HTML>
<HEAD>
Pagelet script
</HEAD>
<BODY>
Pagelet body
</BODY>
</HTML>
```

Representation of the consumer web page's code after injection of a pagelet:

```
<HTML>
<HEAD>
Consumer page script
</HEAD>
<BODY>
Pagelet script
<DIV pageletcontainer>
<IFRAME SRC=...>

<HTML>
<HEAD>
Pagelet script
</HEAD>
<BODY>
Pagelet body
</BODY>
</HTML>

</IFRAME>
</DIV>
</BODY>
</HTML>
```

8.4.2 Adding Pagelets to Proxied Web Pages

This section describes how to add a pagelet to any web page that is proxied by Oracle WebCenter Ensemble. Sample code for adding a pagelet to a web page is provided on the General page of the pagelet configuration. The basic format of the code you use to add pagelets to proxied web pages is:

```
<pt:ensemble.inject pt:name="library:pagelet" />
```

Library is the library name and pagelet is the pagelet name, as entered in the Oracle WebCenter Ensemble pagelet configuration.

Any data passed to the pagelet is also included in the above. For details on configuring pagelet parameters, see Section 8.5, "Configuring Pagelet Parameters and Transport Type."

Note: You must define the namespace prefix 'pt' in the web page as `xmlns:pt='http://www.plumtree.com/xmlschemas/ptui/`

8.4.3 Adding Pagelets to Non-Proxied Web Pages

This section describes how to add pagelets to non-proxied web pages. To do so, you must add special HTML code to the location of the web page where you want to add the pagelet. During runtime, this HTML code calls the *injectpagelet* javascript function, which adds Oracle WebCenter Ensemble pagelets as widgets into the parent page. Oracle WebCenter Ensemble assigns a unique name to each pagelet instance. If the pagelet requires authentication, a log in screen appears before the pagelet is rendered.

Add the following HTML code to the `<HEAD>` section of the web page to which you want to add the pagelet:

```
<script type="text/javascript" src="http://proxy_name:port_
number/inject/v2/csapi">
</script>
```

This HTML code adds a javascript API that provides access to the `injectpagelet` javascript function:

```
function injectpagelet(<library>, <name>, <injectmethod>, <payload>, <arguments>)
{
...
}
```

Parameters of the injectpagelet function are the following:

- `<library>`: The name of the library that includes the pagelet that you want to add to the web page. The value for this parameter must be a string, which can contain spaces. For example: *library name*.
- `<name>`: The name of the pagelet that you want to add to the web page. The value for this parameter must be a string, which can contain spaces. For example: *pagelet name*.
- `<injectmethod>`: Specifies whether an iFrame is or is not used to wrap the pagelet. iFrames are most often used with browsers that have security constraints when making HTTP calls in javascript. Possible values are:
  - `" "`: This is the default value, and appears as a space character surrounded by quotation marks. Using this value results in an iFrame *not* wrapping the pagelet.

Note: Do not delete the quotation marks from this value. If you delete the quotation marks, the java compiler will mistakenly compile the value of the payload parameter as the value of the injectmethod parameter.

  - `iframe`: Using this value results in an iFrame wrapping the pagelet. You can add the following options to specify the appearance of the iFrame: `width`, `height`, `frameborder`, `align`, `longdesc`, `marginheight`, `marginwidth`, `scrolling`, `stylem class`

- `<payload>`: The XML payload to send with the pagelet request.
- `<arguments>`: The pagelet arguments to send with the pagelet request. These arguments should be in the following format: `param1=value1&param2=value2&param3=value3`.

8.4.4 Managing Pagelet Error Messages

The values that you supply for the `pt:onhttperror` parameter of the `pt:ensemble.inject` tag control Oracle WebCenter Ensemble's pagelet request error handling behavior.

In the following example, the `pt:onhttperror`'s value is set to `comment`. Oracle WebCenter Ensemble does not display an error to the end user, but instead adds an HTML comment to the resource page that includes error details:

```
<pt:ensemble.inject pt:name="library:pagelet" pt:onhttperror="comment">
```

In the following example, the `pt:onhttperror`'s value is set to `inline`. Oracle WebCenter Ensemble renders the body of the error response (for example, an error message or login form) in the page:

```
<pt:ensemble.inject pt:name="library:pagelet" pt:onhttperror="inline">
```

In the following example, the `pt:onhttperror`'s value is set to `fullpage`. Oracle WebCenter Ensemble returns the error response to the browser, instead of sending the resource response. Oracle WebCenter Ensemble handles the first error that it encounters while transforming a resource page, thus alleviating any chance of subsequent errors.

```
<pt:ensemble.inject pt:name="library:pagelet" pt:onhttperror="fullpage">
```

8.4.5 Using Oracle WebCenter Ensemble as a Portlet Provider for a Portal

You can use Oracle WebCenter Ensemble to add a non-proxied pagelet to a portal such as Oracle WebCenter Interaction or Oracle WebLogic Portal. Non-proxied pagelets appear as portlets in a portal. To use Oracle WebCenter Ensemble as a portlet provider for a portal, register the following URL as a portlet location in your portal:

http://*host*:*port*/inject/v2/portlet/*libraryname*/*pageletname*?<instanceid=*instance_ID_number*>&content-type=html&<payload=*xmlpayload*>&<*param1=value1*>&<*param2=value2*>

This URL contains the call that returns the pagelet to the portal. The following query string arguments define how the pagelet should be returned:

- `<instanceid>`: The instance ID of the pagelet.

- `<content-type>`: Specifies the type of content that is returned to the portal.Values can be the following:
  - `javascript`: Returns injectable code to the portal.
  - `html`: Returns the pagelet markup with its associated PTPortlet object.
  - `iframe`: Returns an IFrame that points back to the inject api, filling the IFrame with the pagelet content, instead of directly inline with the page. The IFrame can be styled by providing a set of query string parameters.
- `<payload>` and `<pagelet>` parameters: If you want the content of the pagelet to change dynamically, configure the parameters in your portal, then add these parameters to the consumer pagelets. If you want the content to remain static, you can hard code these values.

8.5 Configuring Pagelet Parameters and Transport Type

This section describes how to use pagelet parameters to pass data to pagelets. It is divided into the following sections:

- Section 8.5.1, "Passing Data with Pagelet Parameters," describes how to use *pagelet parameters* to pass data to the pagelet.
- Section 8.5.2, "Passing Data with the Pagelet Payload," describes how to use the *pagelet payload* to pass data to the pagelet.
- Section 8.5.3, "Configuring Pagelet Parameter Transport Type," describes how the pagelet parameter transport type allows you to port Oracle WebCenter Interaction portlets to work as pagelets within Oracle WebCenter Ensemble.

8.5.1 Passing Data with Pagelet Parameters

Parameters are name/value pairs that provide information to the pagelet code. For example, in a discussions pagelet, parameters could specify which discussions or the number of discussion threads to display. Parameters are viewable in the pagelet documentation.

You configure the pagelet parameters that can be passed to the pagelet in the pagelet configuration in the Ensemble Console. You include pagelet parameter values in the pagelet injection code that is added to a web page.

This section is divided into the following sub-sections:

- Section 8.5.1.1, "Configuring Parameters in the Ensemble Console"
- Section 8.5.1.2, "Setting Parameter Values in Pagelet Injection Code"

8.5.1.1 Configuring Parameters in the Ensemble Console

To configure parameters for a pagelet in the Ensemble Console:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. Click the name of the pagelet you want to configure.
5. On the Parameters page, type a **Name**, **Description**, and **Type** for the parameter.

> **Note:** The **Description** and **Type** fields are used for pagelet documentation and are optional. Pagelet documentation is automatically created and can be viewed in the Pagelet Discovery UI. For details on the Pagelet Discovery UI, see Section 8.8, "Accessing Pagelet Discovery for Developers."

6. To make the parameter mandatory, select the check-box under **Mandatory**.
7. To add the parameter, click **Add**.
8. Click **Save**.

To delete a parameter, select the checkbox to the left of the parameter and click **Delete**.

The **Pagelet Parameter Transport Type** setting is provided for porting Oracle WebCenter Interaction portlets to Oracle WebCenter Ensemble pagelets. For details, see Section 8.5.3, "Configuring Pagelet Parameter Transport Type."

8.5.1.2 Setting Parameter Values in Pagelet Injection Code

You set pagelet parameter values in the pagelet injection code using the parameter names configured in the Ensemble Console. For example:

```
<pt:ensemble.inject pt:name="library : pagelet"
param1="foo"
param2="bar"
/>
```

In this example, Oracle WebCenter Ensemble passes the pagelet *library*:*pagelet* two parameters: *param1* with a value of *foo*, and *param2* with a value of *bar*.

You can send a resource's query string parameters as pagelet parameters. For example:

```
<pt:ensemble.inject pt:name="library:pagelet"
pt:forwardparams="true"
param1="foo"
param2="bar"
/>
```

In the example above, any query string parameters that are in the request to the resource are sent as pagelet parameters in the request to the pagelet. Pagelet parameters defined in the `pt:ensemble.inject` tag override identically-named parameters in the resource request.

8.5.2 Passing Data with the Pagelet Payload

Any text data can be passed to the pagelet by including it within the `<pt:ensemble.inject>` tag. For example:

```
<pt:ensemble.inject pt:name="library:pagelet">
This is the payload.
</pt:ensemble.inject>
```

In this example, Oracle WebCenter Ensemble passes the text *This is the payload* to the pagelet as the pagelet payload.

The pagelet retrieves the payload through the Oracle WebCenter Interaction Development Kit (IDK) proxy API. In addition to extracting the payload as raw text, the Oracle WebCenter Interaction Development Kit (IDK) proxy API provides methods to extract an XML payload as an XML document.

For more information on the Oracle WebCenter Interaction Development Kit (IDK) proxy API, see the following documentation:

- Oracle WebCenter Interaction Development Kit (IDK) proxy API tutorials, located in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Ensemble* on the Oracle Technology Network at `http://www.oracle.com/technology/index.html`.

Oracle WebCenter Ensemble allows you to configure a payload schema URL to point to an XML schema that can validate an XML payload. Oracle WebCenter Ensemble only supplies the URL to the pagelet; it is up to the pagelet to use the schema to validate the XML payload.

To configure the payload schema URL:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. Click the name of the pagelet you want to configure.
5. On the Parameters page, type the schema URL in the **Payload schema URL** text box.

8.5.3 Configuring Pagelet Parameter Transport Type

Pagelet parameter transport type allows you to port Oracle WebCenter Interaction portlets to work as pagelets within Oracle WebCenter Ensemble. Oracle WebCenter Interaction portlets may require Administrator, CommunityPortlet, or Community level preference settings.

For details on portlet settings and preferences, see the developer documentation for Oracle WebCenter Interaction Portlet Settings and Preferences in the *Oracle Fusion Middleware Web Service Developer's Guide for Oracle WebCenter Interaction* on the Oracle Technology Network at `http://www.oracle.com/technology/index.html`.

To supply these preferences from Oracle WebCenter Ensemble:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. Click the name of the pagelet you want to configure.
5. On the Parameters page, select the appropriate **Pagelet Parameter Transport Type** from the drop-down.
   - To supply Administrator preferences, select **Global - Admin Prefs**.
   - To supply Community preferences, select **Realm - Community Prefs**.
   - To supply CommunityPortlet preferences, select **Pagelet Realm - Community Pagelet Prefs**.
6. Add parameters, using the same name as the preferences in the portlet. For details on adding parameters, see Section 8.5.1.1, "Configuring Parameters in the Ensemble Console."
7. Define the parameters in your pagelet injection code. For details on setting parameter values, see Section 8.5.1.2, "Setting Parameter Values in Pagelet Injection Code."

8.6 Configuring Metadata Fields

Metadata can be used to store additional information about a pagelet. Metadata fields are viewable in the pagelet documentation.

To configure metadata fields:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. Click the name of the pagelet you want to configure.
5. On the Metadata page, perform the following:
   - In the **Name** box, type the name of the metadata field.
   - In the **Value** box, type the mapped value.

To create additional metadata fields, click **Add**.

To delete metadata fields, select the metadata field and click **Delete**.

8.7 Configuring Pagelet Consumers

By default, pagelets can be consumed by any Oracle WebCenter Ensemble proxied application. To restrict which resources can consume a pagelet:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelets** sub-tab.
4. Click the name of the pagelet you want to configure.
5. Restrict all resources from consuming the pagelet. On the Consumers page, clear the **All consumers allowed** check box.
6. Add resources able to consume the pagelet by clicking **Add**.
7. Select one or more resources to add.
8. Click **Add selected items**.
9. Click **OK**.

To remove a resource from the list of consumers, select the resource to remove and click **Delete**.

8.8 Accessing Pagelet Discovery for Developers

Pagelets configured in Oracle WebCenter Ensemble are automatically documented in the Oracle WebCenter Ensemble Pagelet Discovery UI. To access the pagelet discovery UI:

1. Launch the Ensemble Console.
2. Click the **APPLICATIONS** tab.
3. Click the **Pagelet Docs** sub-tab.

8.9 Exposing Oracle WebCenter Analytics Pagelets through Oracle WebCenter Ensemble

To expose Oracle WebCenter Analytics reports as pagelets through Oracle WebCenter Ensemble, you perform the following:

1. Import the Oracle WebCenter Analytics resource migration file that is installed with Oracle WebCenter Ensemble. For details, see Section 8.9.1, "Importing the Oracle WebCenter Analytics Migration File."
2. Add Oracle WebCenter Analytics and Oracle WebCenter Interaction image server files to Oracle WebCenter Ensemble's image server directory. For details, see Section 8.9.2, "Adding Oracle WebCenter Analytics and Oracle WebCenter Interaction Image Server Files to Oracle WebCenter Ensemble."

For instructions on configuring Oracle WebCenter Ensemble to send event data to Oracle WebCenter Analytics, see Section 10.1, "Configuring Oracle WebCenter Ensemble to Send Event Data to Oracle WebCenter Analytics."

8.9.1 Importing the Oracle WebCenter Analytics Migration File

To import the Oracle WebCenter Analytics Migration file:

1. Launch the Ensemble Console.
2. Click the **Administration** tab.
3. Click the **Migration** sub-tab.
4. Click the **Import** tab.
5. Type the full path to the Oracle WebCenter Analytics resource migration file in the **Import this file** box:

   install_dir/ensembleadminui/*version_number*/migration/analytics_ensemble_migration.xml
6. Choose how you want to deal with duplicate resources by selecting **Rename imported resource** or **Overwrite existing resource**.
7. Click **Prepare Import**.
8. Enter the URL for the Analytics Console following into the **Import Preparation** box. For example: http://*machine_name*:11944/analytics/
9. Click **Import**.

   On success, the Ensemble Console displays a list of imported objects and any autofixes that were completed during the import process.
10. Go to Oracle WebCenter Configuration Manager.
11. Copy the settings found in Ensemble > **ALUI Security Database** to **Analytics** > **ALUI Security Database**, overwriting the original **Analytics ALUI Security Database** settings.
12. Restart the Analytics service.
13. Check the following URL, which should now successfully display Analytics portlets being proxied through Oracle WebCenter Ensemble: http://*proxy_machine_hostname*:*proxy_port*/analytics/ui/console.jsf

8.9.2 Adding Oracle WebCenter Analytics and Oracle WebCenter Interaction Image Server Files to Oracle WebCenter Ensemble

Oracle WebCenter Analytics reports use Cascading Style Sheets (CSS). These style sheets are delivered with Oracle WebCenter Analytics, in the Oracle WebCenter Analytics image server directory. You must add Oracle WebCenter Analytics image server files to the Oracle WebCenter Ensemble image server directory for Oracle WebCenter Analytics reports to correctly appear.

Additionally -- because Oracle WebCenter Analytics reports include some Oracle WebCenter Interaction images -- you must add Oracle WebCenter InteractionOracle WebCenter Interaction image server files to Oracle WebCenter Ensemble's image server directory for Oracle WebCenter Analytics reports to correctly appear.

To add Oracle WebCenter Analytics and Oracle WebCenter Interaction image server files to Oracle WebCenter Ensemble:

1. Copy the **ptanalytics** folder, located in the following location on the Oracle WebCenter Analytics server:

   install_dir/ptimages/imageserver/plumtree/ptanalytics

2. Paste the ptanalytics folder into the **plumtree** folder, located in the following path on the machine on which you installed Oracle WebCenter Ensemble's Administrative UI and Login Server component:

   install_dir/ensembleproxy/*version*/webapp/ensemblestatic/imageserver/plumtree/...

3. Copy the **common** folder, located in the following location on the Oracle WebCenter Interaction server:

   install_dir/ptimages/imageserver/plumtree/common

4. Paste the common folder into the **plumtree** folder, located in the following path on the machine on which you installed Oracle WebCenter Ensemble's Administrative UI and Login Server component:

   install_dir/ensembleproxy/*version*/webapp/ensemblestatic/imageserver/plumtree/...

5. Copy the **portal** folder, located in the following location on the Oracle WebCenter Interaction server:

   install_dir/ptimages/imageserver/plumtree/portal

6. Paste the portal folder into the **plumtree** folder, located in the following path on the machine on which you installed Oracle WebCenter Ensemble's Administrative UI and Login Server component:

   install_dir/ensembleproxy/*version*/webapp/ensemblestatic/imageserver/plumtree/...

7. Restart Oracle WebCenter Ensemble.

8.10 Exposing BEA AquaLogic Pathways Pagelets through Oracle WebCenter Ensemble

To expose BEA AquaLogic Pathways pagelets through Oracle WebCenter Ensemble, you import the BEA AquaLogic Pathways resource migration file that is installed with Oracle WebCenter Ensemble, then edit the resource migration file to point to the parent URL of the BEA AquaLogic Pathways pagelets.

To expose BEA AquaLogic Pathways pagelets through Oracle WebCenter Ensemble:

1. Launch the Ensemble Console.
2. Click the **Administration** tab.
3. Click the **Migration** sub-tab.
4. Click the **Import** tab.
5. Type the full path to the BEA AquaLogic Pathways resource migration file in the **Import this file** box:

   install_dir/ensembleadminui/*version_number*/migration/pathways_ensemble_migration.xml
6. Choose how you want to deal with duplicate resources by selecting **Rename imported resource** or **Overwrite existing resource**.
7. Click **Prepare Import**.
8. Enter the parent URL for the BEA AquaLogic Pathways pagelets into the **Import Preparation** box. For example: http://*machine_name*:8081/graffiti-webui/
9. Click **Import**.

On success, the Ensemble Console displays a list of imported objects and any autofixes that were completed during the import process.

9 Audit

This chapter describes how to configure and use the auditing functionality of Oracle WebCenter Ensemble. Auditing provides information about the creation, modification, and deletion of Oracle WebCenter Ensemble resources and policies from within the Ensemble Console, along with usage information for resources proxied by Oracle WebCenter Ensemble.

This chapter is divided into the following sections:

- Section 9.1, "Enabling Audit," describes how audit is enabled for Oracle WebCenter Ensemble resources and policies.
- Section 9.2, "Generating Audit Reports," describes how to use the Oracle WebCenter Ensemble database to generate audit reports.

9.1 Enabling Audit

Auditing data is automatically recorded when Oracle WebCenter Ensemble resources and policies are created, modified, or deleted.

You can enable and disable auditing of usage for each proxied resource. When you create a resource, its audit status defaults to disabled.

To change the audit status of a resource:

1. Launch the Ensemble Console.
2. Click the **AUDIT** tab.
3. Click the name of the resource you want to configure.
4. To enable or disable auditing, next to **Audit Status**, select Enabled or Disabled.
5. Click **Save**.

9.2 Generating Audit Reports

Audit information is stored in the Oracle WebCenter Ensemble database. You can generate audit reports using SQL queries. The following sections provide sample SQL scripts and describe the data returned by the queries:

- Section 9.2.1, "Auditing Access to Proxied Resources"
- Section 9.2.2, "Auditing Creation, Modification, and Deletion of Oracle WebCenter Ensemble Resources"
- Section 9.2.3, "Auditing Creation, Modification, and Deletion of Oracle WebCenter Ensemble Policies"

9.2.1 Auditing Access to Proxied Resources

Audit information regarding access to proxied resources is stored in the **ACCESSAUDITRECORDS** table in the Oracle WebCenter Ensemble database.

9.2.1.1 Example SQL Queries

The following query displays all accesses to any resource by a specific *username*. Replace *owner* with the database owner of the ACCESSAUDITRECORDS table.

```
select ID, CREATE_DATE, USERNAME, USERTYPE, SERVICENAME, RESOURCE_ID,
RESOURCENAME, ACCESSSUCCESS, ACCESSURL, ACCESSPRIMAUTHENTICATIONMETHOD,
ACCESSRESAUTHENTICATIONMETHOD
from owner.ACCESSAUDITRECORDS
where USERNAME='username';
```

The following query displays all accesses by a specific *username* to a specific *resource*. Replace *owner* with the database owner of the ACCESSAUDITRECORDS table.

```
select ID, CREATE_DATE, USERNAME, USERTYPE, SERVICENAME, RESOURCE_ID,
RESOURCENAME, ACCESSSUCCESS, ACCESSURL, ACCESSPRIMAUTHENTICATIONMETHOD,
ACCESSRESAUTHENTICATIONMETHOD
from owner.ACCESSAUDITRECORDS
where USERNAME='username' and RESOURCENAME='resource';
```

You should create custom queries to meet your reporting needs.

9.2.1.2 Schema Description

The following table describes the ACCESSAUDITRECORDS schema.

Table 9–1 ACCESSAUDITRECORDS

Column	Description
ID	A unique identifier for the audit record in this table.
CREATE_DATE	The date and time the audit event was created.
USERNAME	The name of the user accessing the resource.
USERTYPE	A comma-delimited list of roles. Roles are assigned by the experience definition associated with the user when the resource is accessed.
SERVICENAME	The fully-qualified domain name of the Ensemble Proxy server.
ACCESSSUCCESS	1 if the resource is successfully accessed. 0 if access to the resource fails. **Note:** Access to the resource fails if the HTTP request from the proxy service to the proxied resource fails, or if the user does not have a role required to access the resource. If the user does not authenticate with Oracle WebCenter Ensemble, no audit event is generated.
ACCESSURL	The URL the user used to access the resource.
ACCESSIPADDRESS	The user's IP address.

Table 9–1 (Cont.) ACCESSAUDITRECORDS

Column	Description
ACCESSPRIMAUTHENTICATIONMETHOD	The authentication method used to authenticate the user. Possible values include: ■ **Basic authentication**: [class com.plumtree.runner.authentication.integrated.BasicAuthIntegratedAuthenticator] ■ **Form authentication**: INTERACTIVE ■ **Oracle COREid authentication**: Oracle COREid Authenticator ■ **SPNEGO authentication**: SPNEGO ■ **SiteMinder authentication**: CA SiteMinder Authenticator
ACCESSRESAUTHENTICATIONMETHOD	The authentication method used by Oracle WebCenter Ensemble to access the proxied resource. Possible values are: ■ Autologin Disabled ■ Basic ■ Form ■ None Selected

9.2.2 Auditing Creation, Modification, and Deletion of Oracle WebCenter Ensemble Resources

Audit information regarding the creation, modification, and deletion of Oracle WebCenter Ensemble resources is stored in two tables in the Oracle WebCenter Ensemble database: **RESOURCECONFIGAUDITRECORDS** and **RESOURCECONFIGDATA**.

RESOURCECONFIGAUDITRECORDS stores information about who modifies which resources, and when.

RESOURCECONFIGDATA stores snapshot of the properties of the resource. This allows you to see how the resource changes with each modification.

9.2.2.1 Example SQL Queries

The following query displays all creation, modification, or deletion of resources by a specific *username*. Replace *owner* with the database owner of the RESOURCECONFIGAUDITRECORDS table.

```
select ID, CREATE_DATE, USERNAME, USERTYPE, OWNERNAME, POLICYOWNERNAME, ENABLED_
FLAG, SERVICENAME, RESOURCE_ID, RESOURCENAME, ACTIONTYPE
from owner.RESOURCECONFIGAUDITRECORDS
where USERNAME='username';
```

The following query displays the details of how a specific *resource* was modified.

```
select owner.RESOURCECONFIGDATA.record_id, owner.RESOURCECONFIGDATA.pageNumber,
owner.RESOURCECONFIGAUDITRECORDS.USERNAME,
owner.RESOURCECONFIGAUDITRECORDS.RESOURCENAME, owner.RESOURCECONFIGDATA.properties
from owner.RESOURCECONFIGAUDITRECORDS, owner.RESOURCECONFIGDATA
where owner.RESOURCECONFIGAUDITRECORDS.ID=owner.RESOURCECONFIGDATA.record_id
and owner.RESOURCECONFIGAUDITRECORDS.RESOURCENAME='resource';
```

You should create custom queries to meet your reporting needs.

9.2.2.2 Schema Description

The following table describes the RESOURCECONFIGAUDITRECORDS schema.

Table 9–2 RESOURCECONFIGAUDITRECORDS

Column	Description
ID	A unique identifier for the audit record in this table.
CREATE_DATE	The date and time the audit event was created.
USERNAME	The name of the user creating, modifying, or deleting the resource.
USERTYPE	The highest administrative role of the user that allows him to make the resource configuration change. Possible values, from highest to lowest, are: 1. Administrators 2. Managers 3. Resource Owners
OWNERNAME	GUID of the resource owner.
POLICYOWNERNAME	GUID of the policy owner of the policy associated with this resource.
SERVICENAME	The fully-qualified domain name of the Ensemble Proxy server.
RESOURCE_ID	The ID of the resource in the RESOURCES table.
RESOURCENAME	The name of the resource.
ACTIONTYPE	An integer from 0-2 that indicates what has been done to the resource: ■ 0: Resource created. ■ 1: Resource modified. ■ 2: Resource deleted.

Note: OWNERNAME and POLICYOWNERNAME GUIDs come from the Oracle WebCenter Interaction portal database. These values are stored in the PTMIGRATION table, which can be joined with the PTUSERS table to match user names with GUIDs.

The following table describes the RESOURCECONFIGDATA schema:

Table 9–3 RESOURCECONFIGDATA

Column	Description
record_id	Associates this entry with a record in RESOURCECONFIGAUDITRECORDS.
properties	A CR-delimited string of name/value pairs that provides a snapshot of the resource's configured values.
pageNumber	The properties column may require multiple rows. In cases where multiple rows are required, pageNumber is incremented for each additional row for a given record_id.

9.2.3 Auditing Creation, Modification, and Deletion of Oracle WebCenter Ensemble Policies

Audit information regarding the creation, modification, and deletion of Oracle WebCenter Ensemble policies is stored in two tables in the Oracle WebCenter Ensemble database: **AUTHORIZATIONCONFIGAUDITRECS** and **AUTHORIZATIONCONFIGDATA**.

AUTHORIZATIONCONFIGAUDITRECS stores information about who modifies which policies, and when.

AUTHORIZATIONCONFIGDATA stores snapshot of the properties of the policy. This allows you to see how the policy changes with each modification.

9.2.3.1 Example SQL Queries

The following query displays all creation, modification, or deletion of policies by a specific *username*. Replace *owner* with the database owner of the AUTHORIZATIONCONFIGAUDITRECS table.

```
select ID, CREATE_DATE, USERNAME, USERTYPE, OWNERNAME, POLICYOWNERNAME, ENABLED_
FLAG, SERVICENAME, RESOURCE_ID, RESOURCENAME, ACTIONTYPE
from owner.AUTHORIZATIONCONFIGAUDITRECS
where USERNAME='username';
```

The following query displays the details of how a specific policy *policy* was modified.

```
select owner.AUTHORIZATIONCONFIGDATA.record_id,
owner.AUTHORIZATIONCONFIGDATA.pageNumber,
owner.AUTHORIZATIONCONFIGAUDITRECS.USERNAME,
owner.AUTHORIZATIONCONFIGAUDITRECS.RESOURCENAME,
owner.AUTHORIZATIONCONFIGDATA.properties
from owner.AUTHORIZATIONCONFIGAUDITRECS, owner.AUTHORIZATIONCONFIGDATA
where owner.AUTHORIZATIONCONFIGAUDITRECS.ID=owner.AUTHORIZATIONCONFIGDATA.record_
id
and owner.AUTHORIZATIONCONFIGAUDITRECS.RESOURCENAME='policy';
```

Custom queries should be created to meet your reporting needs.

9.2.3.2 Schema Description

The following table describes the AUTHORIZATIONCONFIGAUDITRECS schema.

Table 9–4 AUTHORIZATIONCONFIGAUDITRECS

Column	Description
ID	A unique identifier for the audit record in this table.
CREATE_DATE	The date and time the audit event was created.
USERNAME	The name of the user creating, modifying, or deleting the resource.
USERTYPE	The highest administrative role of the user that allows him to make the resource configuration change. Possible values, from highest to lowest, are: 1. Administrators 2. Managers 3. Resource Owners
OWNERNAME	GUID of the resource owner this policy is associated with.
POLICYOWNERNAME	GUID of the policy owner of the policy set.

Table 9–4 (Cont.) AUTHORIZATIONCONFIGAUDITRECS

Column	Description
SERVICENAME	The fully-qualified domain name of the Ensemble proxy server.
RESOURCE_ID	The ID of the policy set in the POLICIES table.
RESOURCENAME	The name of the policy set.
ACTIONTYPE	An integer from 0-2 that indicates what has been done to the resource: ■ 0: Policy set created. ■ 1: Policy set modified. ■ 2: Policy set deleted.

Note: OWNERNAME and POLICYOWNERNAME GUIDs come from the Oracle WebCenter Interaction portal database. These values are stored in the PTMIGRATION table, which can be joined with the PTUSERS table to match user names with GUIDs.

The following table describes the AUTHORIZATIONCONFIGDATA schema:

Table 9–5 AUTHORIZATIONCONFIGDATA

Column	Description
record_id	Associates this entry with a record in AUTHORIZATIONCONFIGAUDITRECS.
properties	A CR-delimited string of name/value pairs that provides a snapshot of the policy set's configured values.
pageNumber	The properties column may require multiple rows. In cases where multiple rows are required, pageNumber is incremented for each additional row for a given record_id.

10

Oracle WebCenter Analytics

This chapter describes how to configure Oracle WebCenter Analytics to receive usage data from Oracle WebCenter Ensemble. This involves the following steps:

1. Configure Oracle WebCenter Analytics to accept Oracle WebCenter Ensemble events. For details, see *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Ensemble*.
2. Configure Oracle WebCenter Ensemble to send events to Oracle WebCenter Analytics. For details, see Section 10.1, "Configuring Oracle WebCenter Ensemble to Send Event Data to Oracle WebCenter Analytics."

You can also expose Oracle WebCenter Analytics reports through Oracle WebCenter Ensemble. To do so, you must import and edit the Oracle WebCenter Analytics resource migration file. For details, see Section 8.9, "Exposing Oracle WebCenter Analytics Pagelets through Oracle WebCenter Ensemble."

10.1 Configuring Oracle WebCenter Ensemble to Send Event Data to Oracle WebCenter Analytics

You configure Oracle WebCenter Ensemble to send events to Oracle WebCenter Analytics by using Oracle WebCenter Configuration Manager.

1. On the Oracle WebCenter Ensemble host, launch the Oracle WebCenter Configuration Manager.
2. Under AquaLogic Ensemble, click **ALI Analytics**.
3. Select **Enabled**.
4. Type the **Server** and **Port** of your Oracle WebCenter Analytics installation.

11

Extending Oracle WebCenter Ensemble

This chapter describes ways to extend Oracle WebCenter Ensemble, including customizing the user experience and developing web applications using Oracle WebCenter Ensemble extensions, and is divided into the following sections:

- Section 11.1, "Custom Login Resources"
- Section 11.2, "Oracle WebCenter Ensemble Adaptive Tags"

11.1 Custom Login Resources

What the user sees when she logs in and out of Oracle WebCenter Ensemble resources is controlled by customizing login resources. Based on the experience definition associated with the user, different pages can be delivered to the user at different times in the login or logout process. The following sections describe login resources and how to use them to communicate with Oracle WebCenter Ensemble.

- Section 11.1.1, "About Login Resources"
- Section 11.1.2, "Communicating With Oracle WebCenter Ensemble"

For details on using experience definitions to determine which login resource is used with a given user, see Chapter 7, "Experience Definitions."

11.1.1 About Login Resources

A login resource hosts pages associated with a user's experience authenticating with Oracle WebCenter Ensemble. The following table describes the types of pages that can be delivered by a login resource:

Table 11–1 Login Resource Pages

Page	Definition
Pre-login page	This page is displayed to the user prior to prompting the user for authentication the user.
Login page	This page is only applicable when form authentication is being used, and provides the form for login.
Post-login page	This page is displayed to the user after successful authentication and before the resource is accessed.
Error page	This page is displayed if there is an error in the login process.
Post-logout page	This page is displayed after the user logs out of the resource.

A web page on the login resource can be specified for any or all of these settings using experience definitions. For details on configuring experience definitions, see Section 7.4, "Login Resources and Interstitial Pages."

11.1.2 Communicating With Oracle WebCenter Ensemble

Login resource pages communicate with Oracle WebCenter Ensemble by using HTTP headers. The following table lists the available headers and describes how and when they are used in the login or logout process.

Table 11–2 Login Resource Headers

Page	Header	Values
Pre-login	runner_pre_interstitial_complete	■ **true** - indicates that the Pre-login page has completed successfully. The page is not displayed and Oracle WebCenter Ensemble proceeds to the login page. ■ **false** - Oracle WebCenter Ensemble does nothing when the header is set to false or if the header is not present. The pre-login page is displayed.
Login	runner_username	■ The username Oracle WebCenter Ensemble uses to authenticate the user.
Login	runner_password	■ The password Oracle WebCenter Ensemble uses to authenticate the user.
Login	runner_authentication_provider	■ The provider for authentication. For Oracle WebCenter Ensemble 10.3 the only valid value is **portal**. If the header is not present, the provider defaults to **portal**.
Login	runner_portal_authentication_source	■ The authentication source to authenticate the user against. This is the same as the authentication source the user would use to log in to the portal.
Post-login	runner_post_interstitial_complete	■ **true** - indicates that the Post-login page has completed successfully. The page is not displayed and Oracle WebCenter Ensemble proceeds to the login page. ■ **false** - Oracle WebCenter Ensemble does nothing when the header is set to false or if the header is not present. The post-login page is displayed.

Note: Error and Logout pages are considered terminal pages and do not communicate with Oracle WebCenter Ensemble using headers. Oracle WebCenter Ensemble does provide information to these pages using Oracle WebCenter Ensemble Adaptive tags. For details, see Section 11.2, "Oracle WebCenter Ensemble Adaptive Tags."

11.2 Oracle WebCenter Ensemble Adaptive Tags

Oracle WebCenter Ensemble Adaptive Tags allow you to insert data and logic into your proxied web application. Oracle WebCenter Ensemble transforms tags included in proxied web applications before the page is delivered to the user.

For additional details, see the topic on Oracle WebCenter Ensemble Adaptive Tags in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Ensemble*, which is located on the Oracle Technology Network at `http://download.oracle.com/docs/cd/E13158_01/ensemble/docs103/index.html`.

Index

M

N

O

P

R

S

T

U

W