Sun Java System Portal Server 7.1 Administration Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819–5022–10
February 2007

070223@16599

Contents

Figures

Tables

Examples

Preface

The *Sun Java™ System Portal Server 7.1 Administration Guide* provides information and instructions for administering the Sun Java System Portal Server 7.1.

Who Should Use This Book

This book is intended for IT administrators who are responsible for administering a portal server using Sun Java System servers and software.

Before You Read This Book

Readers should be familiar with the following products and concepts:

- Sun Java System Directory Server
- Sun Java System Access Manager
- Your web container
  - Sun Java System Application Server 8.2
  - Sun Java System Web Server 7.0
  - BEA WebLogic Server 8.1SP4
  - IBM WebSphere Application Server 5.1.1.6
- Your operating system
- Basic UNIX® administrative procedures
- LDAP (lightweight directory access protocol)
- Web Services for Remote Portlets (WSRP)

How This Book Is Organized

Chapters in the book are organized into three parts:

- Part I
  - Chapter 1 presents an overview of how Portal Server is managed.
  - Chapter 2 describes setting up and administering Portal Server. Instructions for creating and deleting instances of Portal Server are included.
  - Chapter 3, provides instructions for managing organizations and users and for using LDAP nodes.
  - Chapter 4, describes steps for setting up end-user content delivered using the Portal Server.
  - Chapter 5, provides information and instructions for using Web Services for Remote Portlets (WSRP).
  - Chapter 6, explains how to diagnose, troubleshoot, and analyze issues related to end-user activities and end-user interaction with various portal system components.
  - Chapter 7, explains how to obtain runtime information about the Desktop and Sun Java System Secure Remote Access server.
  - Chapter 8, describes how to control Portal Server logging.
  - Chapter 9, describes how to configure and administer subscriptions.
  - Chapter 10, presents information about using the SSO Adapter, which provides this configuration data for an authenticated connection to a portal, and the SSO Adapter service stores that data.
- Part II
  - Chapter 11, provides details about working with search categories and databases.
  - Chapter 12, describes the search server robot and its corresponding configuration files.
- Part III
  - Chapter 13, explains how to decentralize administrative functions.
  - Chapter 14, describes what reference information is available for the delegated administration tag library.

Related Books

- *Sun Java System Portal Server 7.1 Deployment Planning Guide*
- *Sun Java System Portal Server 7 Technical Overview*
- *Sun Java System Portal Server Secure Remote Access 7.1 Administration Guide*
- *Sun Java System Portal Server 7.1 Command Line Reference*

- *Tag Library for Delegated Administration*
- *Sun Java System Portal Server 7.1 Release Notes*
- *Sun Java System Portal Server 7.1 Community Sample Guide*
- *Sun Java System Portal Server 7.1 Developer Sample Guide*
- *Sun Java System Portal Server 7.1 Technical Reference*
- *Sun Java System Portal Server 7.1 Developer's Guide*

An introduction to Portal Server concepts and components is available in the *Sun Java System Portal Server 7 Technical Overview*.

Other Server Documentation

For other server documentation, go to the following:

- Directory Server documentation at (`http://docs.sun.com/coll/1224.1`)
- Access Manager documentation at (`http://docs.sun.com/coll/1292.2`)
- Web Server documentation at (`http://docs.sun.com/coll/1308.3`)
- Application Server documentation at (`http://docs.sun.com/coll/1310.3`)
- Web Proxy Server documentation at (`http://docs.sun.com/coll/1311.4`)

Searching Sun Product Documentation

Besides searching Sun product documentation from the docs.sun.com web site, you can use a search engine by typing the following syntax in the search field:

search-term **`site:docs.sun.com`**

For example, to search for "broker," type the following:

`broker site:docs.sun.com`

To include other Sun web sites in your search (for example, `java.sun.com`, `www.sun.com`, `developers.sun.com`), use "**`sun.com`**" in place of "**`docs.sun.com`**" in the search field.

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell	`machine_name%`
C shell for superuser	`machine_name#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell for superuser	`#`

PART I

Managing Sun Java System Portal Server

◆ ◆ ◆ CHAPTER 1

Understanding Portal Server Management

Portal Server administrators manage a variety of functions, including tasks for the following:

- Multiple portals and Portal Server instances
- The Desktop
- Search server
- Secure Remote Access server
- Single Sign-On (SSO) adapters

This chapter provides information about Portal Server components and the ways for managing a portal:

Understanding Portal Server Components

A Portal Server deployment has a number of components that affect portal administration. These components include the following:

- **Common agent container** – a standalone Java program that implements a container for Java management applications. For more information, see *Solaris 10 What's New*.
- **Portal Administration Server** – a management application that performs authentication and access control check for users accessing Portal Server MBeans. This server uses a JMX™ interface and is implemented as a common agent container module. A portal administration server instance runs on each host that the Portal Server product is installed.
- **Portal domain repository** – a hierarchical data store that contains information about how Portal Server MBeans are organized. Some Portal Server MBeans also store configuration data in this repository. The default Portal domain repository is a subtree in the same LDAP server that Access Manager uses.

On stand-alone Gateway installations, communicating with the LDAP server from the Gateway is prohibited. An additional Portal domain repository on the Gateway file system is used to contain only local Gateway MBeans information.

- **Portal data store** – Back-end storage, such as a relational database management system (RDBMS) or LDAP server, or in the File System, for configuration data and other Portal Server resources that facilitate content delivery by a portal.
- **Portal Administrative MBeans** – Loaded by Portal administration server in the common agent container server to perform portal administrative tasks.
- **Portal administration command-line interface** (`psadmin`) – Provides administrative tools for various Portal Server components. For more information, see "Using the `psadmin` Command-line Interface" on page 30.
- **Portal management console** (`psconsole`) – Provides a browser interface for administering various portal server resources. For more information, see "Using the Portal Server Management Console" on page 28.
- **Monitoring MBeans** – Help capture Portal Server runtime resource information. For more information, see Chapter 7, Monitoring Portal Server Activity
- **Local File System Data** – Portal data stored in the local file system. The data includes configuration files, provider-based templates and JSP™ syntax files, resource bundle files, and customized provider-based Java classes.

For more information about Portal Server components, see the *Sun Java™ System Portal Server 7.1 Deployment Planning Guide*.

Using the Portal Server Management Console

The Portal Server management console, which simplifies a variety of portal administration tasks, is a Java 2 Platform, Enterprise Edition (J2EE™) application that:

- Is accessible through a web browser
- Logs messages to a debug log according to configured debug level
- Logs setting changes that include name and value pairs
- Uses Java Management Extensions (JMX) technology to communicate with portal administrative MBeans in the Portal Administration Server to connect to the portal data store

The management console enables portal administrators to perform the following activities:

- Manage the Desktop and content delivery
- Track user behavior to help portal administrators diagnose, troubleshoot, and analyze issues related to end-user activities and how end users interact with various Portal Server components

- Obtain runtime statistics about Portal Server's Desktop and Secure Remote Access components
- Log information about Portal Server applications

About the Browser Interface

The management console's user interface arranges administration functions into pages. Across the top of each page is a tab strip. The tabs present pages that group management functions in an organized manner. To navigate from page to page, administrators click a tab. The tabs provided are the following:

- **Common Tasks** – Displays links that provide direct access to tasks that portal administrators frequently perform
- **Portals** – Lists deployed portals by their portal IDs so that portal administrators can select a specific portal
- **Search Server** – Lists names of specific search servers so that portal administrators can access pages for managing a specific search server
- **Secure Remote Access** – Allows portal administrators to manage how remote users securely access a portal and its services over the Internet
- **SSO Adapter** – Allows portal administrators to manage how end users gain authenticated access to applications after signing in once

Portal Server administrators can provide and limit access to content on a portal through the definitions of the identities of specific end users. You can set up portal pages, attributes and access policies so that portal content is available to specific entities. These entities include the following:

- A specific organization
- A specific suborganization
- A role
- An individual end-user

▼ To Log In to the Management Console

Only administrators with SuperAdmin permission can access the Portal Server management console. Users access the Portal Server management console using a browser client from a distinct uniform resource identifier (URI).

1 Type this URL in your browser: `http://`*hostname*`:`*port*`/psconsole`

hostname	The name of the system that the management console is running on.
port	The management console's port number assigned during installation.

2 **In the text boxes, type the Admin User Name and Password.**

The admin user should be a top-level administrator. A typical Admin User Name is `amadmin`.

3 **Click the Log In button.**

The management console's Common Tasks page is displayed.

Using the Portal Server Administration Tag Library and Portlets

Portal Server provides an administration tag library for developing administration portlets that enable a portal to be managed from the Desktop instead of from the management console. Administrators can use this tag library to do the following:

- Modify out-of-the-box administration portlets
- Develop portlets with new administration functionality
- Support user management, provider management, and portlet and WSRP management tasks
- Create and administer channels that are based on `JSPProvider`
- Write custom administration portlets with a custom user interface
- Write administrative portlets to manage any custom channel

Administrators can use administration portlets to grant delegated administration status to other users, called delegated administrators. Portal Server provides a sample set of administration portlets that can be used to design a basic Desktop for delegated administrators.

For more information, see *Sun Java System Portal Server 7.1 Developer Sample Guide* and *Tag Library for Delegated Administration*.

Using the `psadmin` Command-line Interface

Portal Server software provides a command-line interface (CLI). The CLI allows portal administrators to do the following:

- Perform administrative tasks by typing commands using the keyboard
- Automate regularly recurring management tasks by incorporating them into scripts

The CLI offers a number of `psadmin` subcommands for managing portal tasks. These include subcommands for:

- Managing multiple portals and portal instances

- Deploying portal and portlet WAR files
- Managing the search server
- Managing Secure Remote Access server
- Managing monitoring
- Managing portal logging

Most subcommands commands are written specifically to mimic functions in the browser interface. For management functions that have no special commands, administrators use standard UNIX commands.

Caution – If you installed Portal Server on Sun Java System Web Server, you must start the Web Server administration server before you invoke `psadmin` commands.

For information about all `psadmin` subcommands, see the *Sun Java System Portal Server 7.1 Command Line Reference*.

◆ ◆ ◆ CHAPTER 2

Managing Portals and Portal Server Instances

This chapter explains multiple portals and how to manage a portal and Portal Server instances. The topics provided include the following:

Understanding Multiple Portals

Multiple portals share the same user set. The features of multiple portals include the following:

- A portal is identified by a URL. For example: `http://hr.xyz.com/portal` or `http://eng.xyz.com/portal`
- Multiple portals share the same user repository — the same Access Manager and the Directory server. You use Access Manager to manage end users, and you do not need to synchronize end-user data in LDAP with any other repository. All related data related to end users resides in only one directory server.
- You can deploy multiple portals and Portal Server instances on one or more hosts. For example, one host may have two portal server instances serving content for one portal and three Portal Server instances serving another portal. Each Portal Server instance must run inside a different web container instance.

All portals share these components:

- Rewriter - Although this component is shared, you can define a different rule set for each portal.
- SSO Adapter - Although this component is shared, you can define a different adapter for each portal.
- All Secure Remote Access services

The following components have a one-to-one relationship with portals:

- Desktop - Each portal has an independent Desktop.
- Subscriptions - This is configured differently per portal.
- WSRP - Producer and Consumer - Independent set of Producers and Configured Producers for each portal.

Search can have a many-to-many relationship with portals:

- One portal can use one search server.
- Many portals can use a single search server.
- Each portal can use more than one search server.

End users see different content for different portals and can customize the each portal's Desktop. Single sign-on between portals is possible. An end user who has access to two portals at a corporation would typically experience the following sequence:

- Types in a URL for Portal One and authenticates using the corporate identify.
- Views personalized content on Portal One.
- Types in a URL for Portal Two without needing to provide authentication.
- Views personalized content on Portal Two.

Portals that use different Access Managers are *not* multiple portals. They are independent and unrelated portals, each with its own set of users.

Access Manger can be a collection of its own instances, all using the same set of Directory Server instances. Different Access Managers are two unrelated Access Managers, not different instances of the same Access Manager.

Setting Up Portals

A *portal* consists of one or more portal server instances that deliver the same content and are mapped to a single Uniform Resource Locator (URL). The content and services delivered by a portal are common to all its instances.

Multiple portals share the same user set. These portals can be deployed on one or more hosts, but they all share the same user repository — the same Access Manager and the Directory server.

Note – Portals that use different Access Managers are *not* multiple portals. They are independent and unrelated portals, each with its own set of users.

Access Manger can be a collection of its own instances, all using the same set of Directory Server instances. Different Access Managers are two unrelated Access Managers, not different instances of the same Access Manager.

This section explains how to complete the following tasks:

To List Portals

You can view a list of Portal Servers that are already set up.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**

More Information

Equivalent `psadmin` Command

```
psadmin list-portals
```

To Create a Portal

During Portal Server installation, a default portal named *portal1* is created. You can also create a new portal server using the Create Portal wizard.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Click the New Portal button to launch the wizard.**
4. **Provide a unique name for the Portal Server, for example, `portal5`.**
5. **Type a URI that enables end users to access the Portal Server, for example, `/portal`.**
6. **Select a web container type.**

   The available types are the following:

   - Sun Java™ System Web Server 6.0
   - Sun Java System Web Server 7.x
   - Sun Java System Application Server 8.x
   - BEA WebLogic 8.1SP4/SP5
   - IBM WebSphere 5.1.1.6

7. **(Optional) Change the default web container instance properties.**

   For information, see Creating a New Portal in *Sun Java System Portal Server 7.1 Configuration Guide*.

8. **Verify the information you supplied.**

9. **Click Finish to create the new portal.**

10. **(Optional) View the log file to monitor the process.**

    a. **Log in to the machine where portal is to be created.**

    b. **Run the `psadmin set-logger` command.**

       `./psadmin set-logger -u` *uid* `-f` *password* `-m` *component-type* `-O` *logger-name*

More Information Equivalent `psadmin` Command

```
psadmin create-portal
```

Templates for `webcontainer.properties` for supported web containers are in the `portal-install-dir/template` directory.

▼ To Delete a Portal

You can delete all existing instances of a portal on all hosts and clean up the portal's data in the Access Manager LDAP directory.

1. **Log in to the Portal Server management console.**

2. **Select the Portals tab.**

3. **From the list of portals, select the portal you want to remove, and click the Delete Portal button.**

More Information Equivalent `psadmin` Command

```
psadmin delete-portal
```

▼ To Export Portal Data

You can archive the following portal data in a `par` file:

- Data stored in the Access Manager directory

- Desktop file system files, located by default in the `/var/opt/SUNWportal/portals/`*portal-id*`/desktop` directory
- Desktop customized classes, located by default in the `/var/opt/SUNWportal/portals/`*portal-id*`/desktop/classes` directory
- Portal Server web applications, located by default in the `/var/opt/SUNWportal/portals/`*portal-id*`/war` directory
- Portal Server web source data, located by default in the `/var/opt/SUNWportal/portals/`*portal-id*`/web-src` directory

After you archive data, you can import the data to the same portal or to a different portal. To export a portal from psconsole:

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal from the table.**
4. **Click the Export button.**
5. **Specify the `par` file location on the Portal Server machine and what you want to export:**
   - **All Desktop data — the exported `par` includes file system data and display profile data**
   - **File system data only — the exported `par` file includes only the desktop file system data, which is data deployed into the portal desktop and portal `web-src`**
   - **Display profile data only — the exported `par` includes only display profile data**

More Information

Equivalent `psadmin` Command

```
psadmin export
```

Note – This command does not support user data in the Directory Server.

▼ To Import Portal Data to a Portal

You can import into any portal any portal data that you previously exported.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**

3. **Select a portal from the table.**

   The Import Desktop Data page appears.

4. **Click the Import button and specify the following:**

   - **The `par` file path for the imported data. The `par` file must be located on the Portal Server system.**
   - **Whether to continue if the storage structure of the portal does not match the archived file you want to import.**

5. **Redeploy the portal web applications.**

   a. **Schedule a time to run the `psadmin redeploy` command.**

      Plan to do this step off hours or in system maintenance mode, when your system is not in production. This action redeploys the portal `war` file, and it logs out users who are running a Desktop, causing them to lose their work.

   b. **Run the `psadmin redeploy` command.**

      ```
      psadmin redeploy -u amadmin -f passwordfile -p portalID --allwebapps
      ```

More Information

Equivalent `psadmin` Command

```
psadmin import
```

Note – This command does not support user data in the Directory Server.

Setting Up Portal Server Instances

A *Portal Server instance* is a web application deployed to a web container. An instance uses a particular Portal Server context URI to serve requests on a specific network port. Each Portal Server instance is associated with a single Portal.

A server instance listens on a particular port, bound to either one IP address or any IP address of the host. For the Portal Server, a server instance corresponds to a deployment container process listening on a port and running a single Java™ Virtual Machine (JVM™ software).

Note – Sun Java™ System Web Server and Sun Java™ System Application Server support multiple instances.

This section explains how to complete the following tasks:

To List Portal Server Instances

You can view a list of Portal Server instances that are already set up.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Click the name of Portal Server from the table.**
4. **Select the Server Instances tab.**
   The table displays all the instances of the Portal Server you selected.

More Information

Equivalent `psadmin` Command

```
psadmin list-portals
```

To Create a Portal Server Instance

Before You Begin

- Create a new instance for an existing Portal Server on your web container instance.
- Start the web container instance.
- Start the administration server of the web container.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select the name of a Portal Server.**
4. **Select the Server Instances tab.**
5. **Click on the New Instance button to launch the wizard.**
6. **Provide the name of the portal identifier.**
7. **Select a web container type.**
   The available types are the following:

- Sun Java System Web Server 7
- Sun Java System Application Server 8.2
- BEA WebLogic 8.1SP4
- IBM WebSphere 5.1.1.6

8 **(Optional) Change the default web container instance properties.**

For information, see Creating a Portal on the Same Node in *Sun Java System Portal Server 7.1 Configuration Guide*.

9 **Verify the information you supplied, and click Finish to create the new portal instance.**

A progress bar displays the status of this procedure. When the procedure is complete, a results page is provided.

10 **Click Finish to create your new portal instance.**

More Information

Equivalent `psadmin` Command

```
psadmin create-instance
```

▼ To Delete a Portal Server Instance

You can delete an instance of a Portal Server.

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select the name of a Portal Server.**

4 **Select the Server Instances tab.**

5 **From the table, select the instance you want to remove.**

6 **Click Delete Instance button.**

More Information

Equivalent `psadmin` Command

```
psadmin delete-instance
```

◆ ◆ ◆ CHAPTER 3

Managing Organizations, Roles, and Users

Portal Server administrators can provide and limit access to content on a portal through the definitions of the identities of specific end users. You can set up portal pages, attributes and access policies so that portal content is available to specific entities. These entities include the following:

- A specific organization
- A specific suborganization
- A role
- An individual end-user

To manage organizations, roles, and end-users, Portal Server administrators must use both the Portal Server management console and the Sun Java™ System Access Manager console. This chapter explains how Portal Server administrators can do this using Access Manager. This chapter provides the following topics:

Note – This chapter explains how to use Access Manager that is installed and configured to support Legacy Mode. For information about Legacy Mode and Realm Mode, see the *Sun Java System Access Manager Administration Guide*

Understanding How to Use Access Manager With Portal Server

Portal Server uses Sun Java System Access Manager services to manage attributes that are specific to Portal Server end users and applications. You must use the Access Manager console to manage tasks related to identity.

To control who has access to a portal site, Portal Server administrators must use the following tools:

- The Portal Server management console is a browser interface that allows administrators to manage the following:
  - Portals and portal instances
  - Search
  - Remote access
  - Single sign-on
  - Display profile documents
  - Containers and channels
- The Sun Java System Access Manager console is a browser interface that allows administrators with different levels of access to do the following:
  - Create and remove realms and organizations
  - Create and delete users to and from those organizations
  - Manage services
  - Set up enforcement policies that protect and limit access to organization resources

Portal Server administrators must use Access Manager to perform the following tasks:

- Manage identity-based objects, including users, roles, and organizations, to administer and assign appropriate access to users according to roles they have within organizations or suborganizations
- Delegate administrative functions to specific end users by authorizing the end users to administer organizations, suborganizations, users, policy, roles, and channels

Access Manager uses the lightweight directory access protocol (LDAP).

For information about Access Manager administration, see the *Sun Java System Access Manager 7.1 Administration Guide*.

Creating New Organizations for Portal Server

New organizations inherit services that are registered at the top-level Access Manager organization. Typical services that new organizations inherit include the following:

- **Access Manager Configuration**
  - Authentication Configuration
- **Authentication Modules**
  - Core
  - LDAP
  - Policy configuration

New organizations use LDAP authentication, and LDAP service settings are inherited from the corresponding global service.

For information about Access Manager administration, see the *Sun Java System Access Manager 7.1 Administration Guide*.

▼ To Create a New Organization to Use with Portal Server

1. **Log in to the Access Manager console.**

   For information about Access Manager administration, see the *Sun Java System Access Manager 7.1 Administration Guide*.

2. **Under Identity Management, select Organizations from the View menu.**

3. **Click New to create a new organization.**

4. **Specify the organization attributes.**

   For example:

Name	`TestOrganization`
Organization Aliases	`TestOrganization`

5. **Click OK.**

▼ To Access a New Organization

◗ **Type this URL in your browser:**

`http://`*host*`:`*port*`/amserver/UI/Login?org=`***organizationalias***

`host`	The name of the system that the console is running on.
`port`	The console's port number assigned during installation.
`organizationalias`	The value assigned to the Organization Alias attribute field.

Adding Portal Services to Organizations

Before the Portal is accessible, you must add several services to an organization. The services that you must add to the organization include the following:

- Portal Server configuration
  - `portalID` Desktop

- `portalID` Subscriptions
- SSO Adapter
- `portalID` WSRP Consumer

- Mobile Application configuration
  - Mobile Address Book
  - Mobile Calendar
  - Mobile Mail

Optional services that you can add include the following:

- Secure Remote Access configuration
  - Access List
  - NetFile
  - Netlet
  - Proxylet

▼ To Add Portal Services to an Organization

Portal requires several services to be added to an organization before the Portal Server is accessible to the organization. After you add Portal services to the organization, use the Portal Server management console to administer Portal Server settings.

1. **Log in to the Access Manager console.**

   For information about Access Manager administration, see the *Sun Java System Access Manager 7.1 Administration Guide*.

2. **Under Identity Management, select Organizations from the View menu.**

3. **Click your organization.**

   For example: `TestOrganization`

4. **In the View menu for the organization, select Services.**

5. **Click Add.**

6. **Select the following services, if they are available in your deployment:**

   - Mobile Application Configuration
     - Mobile Address Book
     - Mobile Calendar
     - Mobile Fax
     - Mobile Mail

- Portal Server Configuration
  - `portalID` Desktop
  - `portalID` Subscriptions
  - SSO Adapter
- Remote Portlets (WSRP)
  - `portalID` WSRP Consumer
- Secure Remote Access Configuration
  - Access List
  - NetFile
  - Netlet
  - Proxylet

7 Click OK.

▼ To Specify Required Portal Services for New Users

After you add all of the Portal services to an organization, you must use the Access Manager console to add the services to newly created end-users so that they can access the Portal Desktop and whatever Portal services they need.

The Access Manager Administration service allows you to specify which services are dynamically added to end-user entries when they are created. If your Portal deployment allows users to be created, such as a "Sign-Me Up" feature, specify the Required Services setting in the Access Manager console for your organization.

Before You Begin Add Portal services to the organization. See "Adding Portal Services to Organizations" on page 43.

1 Log in to the Access Manager console.

For information about Access Manager administration, see the *Sun Java System Access Manager 7.1 Administration Guide*.

2 Add the Administration Service.

a. Under Identity Management, select Organizations from the View menu.

b. Click your organization.

For example: `TestOrganization`

c. In the View menu for the organization, select Services.

d. Click Add.

e. Select the Administration service and Click OK.

3 Specify the setting for Administration Service Required Services.

This setting specifies whether to assign all services in the required services list to a new end user.

a. Select the Administration service setting.

b. For the Required Services setting, specify the following services:

- `SunPortalportalIDDesktopService`
- `SunPortalportalIDSubscriptionsService`
- `SunMobileAppABService`
- `SunMobileAppCalendarService`
- `SunMobileAppMailService`
- `SunSSOAdapterService`

c. Click Save.

4 Log out of the Access Manager console.

Navigating to Specific Nodes

Portal Server uses Access Manager services to store application and user-specific attributes. To enable you to administer portal-related functions for an LDAP directory node (DN), the Portal Server management console provides details about the DN in a *location bar*, a horizontal strip below the row of tabs.

The location bar enables you to do the following:

- Identify the currently selected node
- View up to 10 organization DNs
- Change to another directory name

A directory name can be a organization, role, or user name.

Understanding the Location Bar

The location bar provides the following functions:

- **Select DN** – Use this drop-down menu to display the following directory node types:
  - Default organizations defined when Portal Server was installed.
  - Nodes that administrators set up using the Add DNs button.
- **Selected DN** – Identifies which DN is currently chosen.
- **Enter DN** – Enables you to go to any DN that is already defined by typing in its full name.

▼ To Set a New Directory Node

You can select a new DN without adding it to the location bar.

1. **Log in to the Portal Server management console.**
2. **Select the Add button next to the location bar.**
3. **Select the name of the DN using one of the following methods:**
   - **Select a DN listed in the window.**
   - **Use the Search utility:**
     a. **Type the search string.**
        You can use wildcard characters.
        Search results are displayed by short name and corresponding directory node.
     b. **Click the Search button.**
4. **Click the Set Current DN button.**
   The window closes, and the Selected DN field displays the new directory node. The directory node is not added to the location bar selections.

▼ To Add a Directory Node to Location Bar Selections

When you add a directory node to the location bar menu, it is stored as a cookie so that the directory node is available in the same browser across sessions.

1. **Log in to the Portal Server management console.**

2. **Select the name of the DN using one of the following methods:**

   - **Using the Add button:**

     a. **Click the Add button next to the Select DN menu.**
     The Add to DNs List pop-up window opens and displays a list of available directory nodes.

     b. **Select the desired DN.**

   - **Using the Search utility:**

     a. **Use the Search menu to select the object type.**

     b. **Type the Search string.**
     You can use wildcard characters.

     Search results are displayed by short name and corresponding DN.

     c. **Select the desired DN.**

3. **Select the name of the directory node.**

4. **(Optional) Edit the short name field to change the name that the directory node in the drop-down menu displays.**

5. **Click the Add button.**
   The directory node is added to the Select DN menu.

▼ To Remove a Directory Node From Location Bar Selections

You can delete a directory node from the drop-down list displayed in the location bar. The directory node itself is not removed. To remove a directory name from the LDAP database, you must use Access Manager.

You cannot remove default organizations that were defined during installation.

1. **Log in to the Portal Server management console.**

2. **From the Select DN drop-down menu, select the DN that you want to delete.**

3. **Click the Delete button next to the Select DN drop-down menu button.**
   The selected directory node is removed.

▼ To Display Information for a Directory Node

1 Log in to the Portal Server management console.

2 Display information about a directory node using one of the following methods:

- Type the name of the directory node in the Enter DN text box, and click the Go button.
- Select the name of the directory node from the Select DN menu.

◆ ◆ ◆ CHAPTER 4

Managing the Portal Server Desktop

This chapter describes the Sun Java™ System Portal Server Desktop and how to manage it.

Understanding Portal Server Desktop Management

This section describes the key components of Portal Server desktop. The following topics are discussed:

Understanding the Display Profile

While installing Portal Server, you create an initial organization. The installer then imports the display profile global level document, and the default organization display profile, based on the input parameters you specify.

After that, each time you create a new organization, suborganization, or role, the display profile is not automatically loaded. However, the new organization, suborganization, or role inherits the display profile defined from its parent. If there are specific entries to the newly created organization, suborganization, or role, you must manually load the display profile.

The display profile creates the display configuration for the standard Desktop by defining the following three items:

Provider definition	Specifies the name and the Java class for the provider. A provider is a template used to generate content, which is displayed in the channel.
Channel definition	Specifies the run time configuration of an instance of the provider class. A channel is a unit of content, often arranged in rows and columns. You can also have channels of channels, called *container channels*.
Provider and channel property definitions	Specify the values for provider and channel properties. Properties defined in a provider usually specify default values for the channels that are derived from the provider. The display configurations for the channels include properties such as the title, description, channel width, and so on. The properties defined in the channel usually specify the specific value for that channel that is different from the default value. Container properties define the display definition about how to display the contained channels in the container, including: the layout of the container (thin-wide, wide-thin, or thin-wide-thin); a list of the contained channels; the position of the channel (the row and column number); and the window state of the contained channels (minimized or detached).

The display profile exists only to provide property values for channels. It does not actually define the overall layout or organization of what users see on their Desktops. However, the display profile does indirectly control some aspects of channel presentation, such as column layout for a table container or how the table container draws channels in a table.

The system reports errors when you try to save a display profile document containing invalid XML. The error messages appear as a title, a message, and a sub-message. The title of the message box is "Invalid XML document." The message appears as one of the following:

- Failed to parse XML...
- Missing doctype in the XML
- Failed to sore DP...
- Invalid XML input...

If you receive an "Invalid XML document" error, you must correct the error to be able to save the XML document.

The display document syntax is as follows:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<DOCTYPE DisplayProfile SYSTEM " jar://resources/psdp.dtd">

<DisplayProfile version="1.0" priority="xxx">
    <Properties>
    ...
    </Properties>
    </Channels>
    ...
    </Channels>
    <Providers>
    ...
    </Providers>
</DisplayProfile>
```

Understanding Desktop Attributes

The Desktop merges all documents in a user's display profile merger set and uses the result to configure the user's desktop. A display profile merger set consists of all the display profile documents associated with a user. Display profiles are defined at different levels in the Portal Server organization tree. Display profile documents from the various levels of the tree are merged or combined to create the user's display profile.

For example, the user's display profile document is merged with the role display profile documents (if any), the organization's display profile document, and the global display profile document to form the user's display profile.

The Desktop display profile and other configuration data are defined as service attributes such as parent container, desktop type and edit container of the portal Desktop service under the Sun Java System Access Manager service management framework. When an organization adds for the Portal Desktop service from the Sun Java System Access Manager management console, all users within the organization inherit the Portal Desktop service attributes in their user profiles. These attributes are queried by the Portal Desktop to determine how information will be aggregated and presented in the Portal Desktop.

See “Managing Desktop Attributes” on page 66

Managing Portal Server Desktop Content

This section discusses how to manage the desktop content. For more information on the desktop, see Understanding the Standard Desktop in *Sun Java System Portal Server 7 Technical Overview*.

Administering Portlets

This section describes how to deploy and undeploy portlets, and how to modify portlet preferences.

Portlets are web applications that process requests and generate content within the context of a portal. Portlets are managed by the Portlet Container (an implementation of the Portlet Specification as defined by the JSR 168 Expert Group).

A portlet can only be deployed on a selected DN node once. If a portlet has already been deployed on the same DN node, you should undeploy the portlet and deploy it. If your require a portlet to be on multiple sub organizations or roles, then deploy the portlet on the portal global DN or the parent organization.

▼ To Deploy a Portlet

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **From the Select DN drop-down menu, select any DN.**
5. **Click Deploy Portlet to start the wizard.**
   a. **Ensure the selected portal and selected DN are the ones where you want to deploy the portlet, and click Next.**
   b. **Specify a portlet war file, the roles file, and the users file.**

Note – The roles file and the users files are optional. The war file, the roles file, and the users file can be located either on the local machine, or on the remote portal server system.

c. **Select the button for either the local system or the remote portal server system.**

- **If the upload file is from the local machine, use the browse dialog box to select the file from the local machine.**
- **If the upload file is from a remote portal server system, use the file chooser dialog to choose a file from the remote machine**

d. **Verify the information provided, and click Next.**

e. **An information page appears when the portlet is deployed.**

6 **Follow the instructions to deploy a portlet.**

More Information

Equivalent `psadmin` Command

```
psadmin deploy-portlet
```

▼ To Undeploy a Portlet

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server from Portals.**

4 **From the Select DN drop-down menu, select any DN.**

5 **Click Undeploy Portlet to launch the wizard.**

6 **Modify the configuration attributes as necessary.**

7 **Click Undeploy to record the changes.**

More Information

Equivalent `psadmin` Command

```
psadmin undeploy-portlet
```

▼ To Modify Portlet Preferences

1. **Log in to the Portal Server management console.**
2. **Click the Common Tasks tab, then Manage Channel and Containers from the submenu.**
3. **Select a portal and the DN where the portlet is deployed.**
   The navigation tree with available channels and portlets is displayed.
4. **From the navigation tree on the left frame, select the portlet channel.**
   The preferences table and properties table is displayed on the right frame.
5. **In the preferences table, click Edit Values link of a preference you want to modify.**
6. **In the preferences wizard, type the new value in the text field, and click OK.**
   - **To remove a value, select the value from the list and click Remove.**
7. **When you done with modifying preferences, click Save.**
8. **Click Close.**

Managing Channels and Containers

This section describes how to manage portal server channels and containers from the management console.

The following topics are discussed:

Viewing Channels and Containers

The desktop for a user is rendered by starting a desktop parent container. You can customize the parent container attribute at every organization, role and user DNs. The content for a desktop at a particular DN is provided by iterating the child containers and channels that selected to be to be displayed inside the desktop parent container.

Usually, the desktop parent container contains a few tab or table containers. Each tab container under the list of selected nodes of the parent container will display a tab on the user desktop. The channels that appear under the tab are the channels inside the tab container.

The bottom left frame of the Channels and Container Management in the portal management console has two components:

- View Type menu
- Channels and Container tree

Items in the View Type menu and the nodes displayed in the tree are dependent on content of the merged Display Profile XML.

The tree contains container and channel nodes. There are three types of channels that deliver content to the desktop:

- Provider (native) channels
- Portlet channels
- Remote portlet channels

You can click on any of the node links in the tree to display properties and actions on the right frame.

There are two types of items in the View Type menu:

- Display Profile XML Tree
- Desktop Views

See "To View Display Profile XML Tree and Desktop Views" on page 58

Display Profile XML Tree

The tree displays a complete set of channels and containers in the merged Display Profile (DP) XML. The root element in the DP XML Tree is DP_ROOT, which is the parent of all the channels and containers of the display profile. You can create a channel directly under DP_ROOT, or in a container under DP_ROOT.

The nodes listed under the DP XML Tree is not always displayed on the desktop. Some nodes in the display profile are never referenced or included in the hierarchy of the desktop container.

For example, the desktop default container JSPTabContainer has two containers, *tab1* and *tab2*. If *tab1* contains *ch1* and *ch2*, and *tab2*contains *ch3* and *ch4*, then there are five channels defined in the DP XML Tree. The DP XML Tree references *ch1* to *ch4* in the container hierarchy, but *ch5* is not. So, only *ch1* to *ch4* will display on the desktop.

Desktop Views

Desktop views are top level containers available in the merged display profile. You can set each desktop views as the parent container for the desktop at the DN. When you select a desktop view, the tree provides a visual hierarchy of the channels and containers that has a role in rendering content to the desktop.

Channels and containers displayed under the desktop views have two states:

- Selected and visible on the desktop
- Available for selection

Note – In this state, channels and containers icons are displayed in grey color.

You can change the state of channels and containers in a desktop view by clicking the task link on the right frame. To display a tool tip about the state, place the mouse over a container or channel icon. The tool tip also displays the fully qualified name of the node.

▼ To View Display Profile XML Tree and Desktop Views

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals, then any DN from the Select DN drop-down menu.**
   - **You can also select the organization from Select DN menu in the Manage Containers and Channels page.**
4. **Under Tasks, click Manage Containers and Channels.**
5. **From the View Type drop-down menu select DP XML Tree or a Desktop View.**

Modifying Channels and Container Properties

This section discusses the properties of channels and containers, and how to modify them.

You can perform the following tasks:

- "To Create a Property" on page 60
- "To Edit a List" on page 61
- "To Modify Portlet Preferences" on page 56

- "To Modify Channel and Container Properties" on page 62
- "To Upload a Display Profile" on page 69

Understanding Properties

The properties displayed when you click on the node in the tree are top level properties or channel level properties. These properties are defined at the provider level and you can customize these properties for a channel. However, new properties added to a channel cannot be added to the provider. This is the reason you cannot add new properties at the channel level.

The properties table displays client type and locale. There is no column to show the type of the property, however, the following convention is followed:

String	Value column has a wide text field for a maximum of 30 characters.
Integer	Value column has a narrow text field for a maximum of 5 characters.
Boolean	Value is a radio button.
Map	Name is a link.
List	Value column has an Edit Values link. Clicking this link opens a wizard to add and remove values.
Empty Collection	The name is a link showing Edit Values link. Name and value pairs may be added to an empty collection to behave like a map, and the Edit Values disappears. If values are added to an empty collection using Edit Values wizard, the collection behaves as a List and the name link disappears.

In addition to the Name and Value columns, the properties table has two more columns:

Category — Displays if the property is advanced or basic. The advanced properties generally are for experienced administrators.

State — Any property may be in three possible states:

- Default – Value assigned at the provider.
- Inherited – Values modified at some level above. For example, if the current node is a role, then the property may have been customized at the organization of the role. This organization may be the parent organization, or parent of the parent organization. When the property is inherited it is a link. Clicking this link shows all the possible parent nodes in the hierarchy from where this property was inherited from.
- Customized – Value defined at this node.

There are buttons in the properties table:

Remove Customization — Removes values defined at this node from the display profile. This may result in properties to be inherited from some parent in the

	hierarchy if the properties are customized there. If the value has not been customized anywhere in the hierarchy, the value defined at the provider is displayed and the state will show as Default.
Save	Saves additions, deletions, and changes of value.
Reset	Ignores changes and resets values to last saved state from the data store.
Clear All Sorts	Clears all sorts.

Tip – Table may be sorted by clicking on any column title. When you click the Name button first to sort by name, a + appears next to the Category and State buttons. Click the + to apply the next sort criteria.

Table Preferences	Sets the table preferences. Unless modified, the client type and locale are set to default.

▼ To Create a Property

From the New Property wizard you can edit the values and save. You can also add new name and value pairs.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal from Portals.**
4. **From Select DN drop-down menu, select any DN.**
5. **Under Task, click Manage Channels and Containers.**
6. **Select a container in the tree on left frame to display Edit Properties page on the right frame.**
7. **Click the New Property button to launch the wizard.**
8. **Select the property type, and click Next.**
9. **Type a Name, select a Value, and specify if the property is advanced or not.**

Note – Collection property behaves like a map when it contains name and value pairs. Property of type Collection can be nested. The property path above the table will change to display the current nesting and you can navigate back.

Any trailing values are optional. For example, the value may be `en` or `en_US`, but cannot be `US` only. The standard Java format for specifying a locale is followed.

10 **Click Finish to create the property.**

11 **Click Close to display the new property in the table.**

▼ To Edit a List

Collection property behaves like a List when it contains only values.

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal from Portals.**

4 **From Select DN drop-down menu, select any DN.**

5 **Under Task, click Manage Channels and Containers.**

6 **Select a container in the tree on left frame to display Edit Properties page on the right frame.**

7 **Click the Edit Values link of a property to launch the wizard.**

8 **Make your changes.**

- **To add a value, type the name of the value in the New Value text box, and click Add.**
- **To delete a value, select a value from the Values list, and click Remove.**

9 **Click Close.**

The edit properties page will update number of values in the list.

▼ To Modify Channel and Container Properties

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal from Portals.**
4. **From Select DN drop-down menu, select any DN.**
5. **Under Task, click Manage Channels and Containers.**
6. **Select a channel or container in the tree on left frame to display Edit Properties page on the right frame.**
7. **Change the properties, and click Save.**

More Information

Equivalent `psadmin` Command

```
psadmin modify-dp
```

Creating and Deleting Channels and Containers

This section discusses how to create and delete channels and containers from the portal management console.

- "To Create a Channel or Container" on page 62
- "To Delete a Channel or Container" on page 64

▼ To Create a Channel or Container

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal from Portals.**
4. **From Select DN drop-down menu, select any DN.**
5. **Under Task, click Manage Channels and Containers.**
6. **Select a container in the tree on left frame to display Edit Properties page on the right frame.**

7. **Under Tasks, click New Channel or Container to launch the wizard.**

   In the wizard, ensure that the selected portal and selected DN is where you want to create the channel or container and click Next.

8. **Create a container or channel from the wizard.**

   - **To create a container, perform the following steps:**

     a. **Select a provider from the Container Provider drop-down menu, and click Next.**

     b. **Type a name in the Channel or Container Name text field, and click Next.**

     c. **Review your selections, and click Finish.**

        A message confirms the creation of the container.

     d. **Click Close**

   - **To create a channel, perform the following steps:**

     a. **Select a channel type.**

        Select a channel from the following three types:

        - If you select Provider Channel, a list of provider channels are displayed.
        - If you select JSR 168 Portlet Channel, a list of portlet channels are displayed.
        - If you select WSRP Remote Portlet Channel, select the registered producer and the remote portlet from the drop-down menu.

     b. **Type a name in the Channel or Container Name text field, and click Next.**

     c. **Review your selections, and click Finish.**

        A message confirms the creation of the channel.

     d. **Click Close.**

More Information

Equivalent `psadmin` Command

```
psadmin add-dp
```

▼ To Delete a Channel or Container

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal from Portals.
4. From Select DN drop-down menu, select any DN.
5. Under Tasks, click Manage Channels and Containers.
6. Select a container in the tree on left frame to display Edit Properties page on the right frame.
7. Under Tasks, click Select Channels or Containers to Delete.
8. Under Type, select Channel or Container.
   Available channels and containers are displayed.
9. Select a channel or container, and click Delete.

More Information

Equivalent `psadmin` Command

```
psadmin remove-dp
```

Creating a Tab

This section describes how to create a tab form the portal server management console.

▼ To Create a Tab

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal from Portals.
4. From Select DN drop-down menu, select ay DN.
5. Under Tasks, click Manage Channels and Containers.

6. **From the tree on the left frame, select a tab container.**
7. **Under Tasks in the right frame, click New Tab to launch the wizard.**

Displaying Channels and Containers

This section discusses how to display channels and containers on the end-user Desktop. Channels and containers can also be made available on the content page so that the end user can select them to display on the Desktop.

▼ To Display Channels and Containers on Desktop

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal from Portals.**
4. **Under Tasks, click Manage Containers and Channels.**
5. **Select a container in the tree on left frame to display Edit Properties page on the right frame.**
6. **Under Tasks, click Show or Hide Channels and Containers on Portal Desktop.**
7. **Under Ready For Use, select a channel or container.**
8. **Using the Add button, move the channels to appear on the Content Page or Portal Desktop.**
   - **Using the Remove button, you can move the channels or containers back to Ready For Use.**
9. **Click Save.**

More Information

Equivalent `psadmin` Command

`psadmin modify-dp`.

Managing Desktop Attributes

This section discusses how to manage Desktop attributes. For more information, see "Understanding Desktop Attributes" on page 53.

Desktop attributes for the top level organization is different from different levels of the organization tree. You can change the location bar to TopLevel to see global Desktop attributes, and then select other distinguished names for organization or role Desktop attributes.

▼ To Set Up Desktop Attributes

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals, then Desktop.**
4. **From the Select DN drop-down menu, select any DN.**
5. **Modify the configuration attributes as necessary under Desktop Attributes.**

   The following options are available:

COS Priority	Sets the conflict resolution level for the Desktop service template used to resolve conflicts when multiple Desktop templates are merged. This attribute applies only to Organizations and Roles and doesn't apply to Users and Global DN.
Parent Container	Identifies which default container is rendered when the Desktop is called with an unspecified provider. The value for the Parent Container can be one of the containers which is defined as a TopLevelContainer that can draw a header and footer on the portal page. A container is a Top Level container if the display profile property TopLevel is set to true.
Edit Container	Specifies which default edit container to use to wrap the content when one is not specified in the URL. This container will be used by the parent container to draw the edit pages when the edit link is clicked on the channel title bar.
Desktop Type	The comma separated list used by the Desktop lookup operation when searching for templates and JSPs. The lookup starts at the first element in the list and each element represents a sub directory under the Desktop template base directory. e.g., "sampleportal,foo"

	in which case the lookup would be sampleportal directory, foo directory, default directory in that order.
Desktop Attributes	Specifies whether the Desktop attributes are displayed to the users associated with the role. This dynamic attribute is mainly used for role-based delegated administration in administration tag library. This attribute enabled to show, allows the delegated administrators to administer channels/containers inherited from the parent organizations. This attribute applies only to Organizations and Roles.
Display Profile Priority	Sets the priority of the display profile document. Display profile documents are merged from low priority to high priority. A lower number represents a lower priority. For example, a 1 is a lower priority than a 2. High priority documents override values set in lower priority documents using merge semantics (unless a lower priority document has locked the object for merging).

Note – The display profile priority is not stored as Desktop service attribute.

The following attributes apply only to Global (top level) DN.

XML Parsing Validation	Enables the validation for XML parsing.
Federation	Enables Identity Federation so that a user can associate, connect or bind multiple internet service providers, local identities, enabling them to have one network identity.
Hosted Provider ID	Specifies the unique identifier of the host that provides the network identity of a user.
Session Reap Interval	Specifies the session reap interval in seconds.
Session Idle Time	Specifies the idle time in seconds after which the session is terminated.
Maximum Number of Client Sessions	Specifies the maximum number of client sessions allowed at any given time.
Anonymous Desktop	When enabled, allows anonymous Desktop for the selected portal.
Anonymous Access for Federated Users	Prevents users with a network identity on a hosted provided to access the portal Desktop by providing a user name and password.

Valid UIDs for Anonymous Desktop	List of User IDs authorized to access the Desktop without authenticating.

6 **Click Save to record the changes.**

Otherwise, click Reset to undo any edits.

Note – To modify global attributes, Change the DN in the location bar drop-down to TopLevel.

More Information Equivalent `psadmin` Command

```
psadmin undeploy-portlet
```

Administering the Display Profile

This section describes how to manage the Sun Java System Portal Server display profile. For more information, see "Understanding the Display Profile" on page 51.

You can perform the following tasks from the portal management console:

▼ To Download a Display Profile

You can download the display profile to a file.

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server under Portals.**

4 **From Select DN drop-down menu, select any DN.**

5 **Click Download Display Profile under Tasks.**

The browser's download window pops up.

6 **Select a location and Click Save.**

Note – This step may vary from browser to browser.

More Information For equivalent `psadmin` Command

```
psadmin get-attribute
```

▼ To Upload a Display Profile

You can upload the display profile to a file.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals.**
4. **From Select DN drop-down menu, select any DN.**
5. **Click Upload Display Profile under Tasks.**
6. **Choose a display profile file to upload using the Browse button.**

Note – The file should be located on local machine based on the user's browser settings.

7. **Click Upload.**

More Information Equivalent `psadmin` Command

```
psadmin modify-dp.
```

▼ To Remove a Display Profile

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals.**
4. **From Select DN drop-down menu, select any DN.**

5. **Click Remove Display Profile under Tasks.**
6. **Click OK in the warning dialog box to confirm deletion.**

More Information

Equivalent `psadmin` Command

```
psadmin remove-dp
```

◆ ◆ ◆ CHAPTER 5

Web Services for Remote Portlets

Sun Java™ System Portal Server supports Web Services for Remote Portlets (WSRP). This chapter presents guidelines and best practices for using WSRP. This chapter contains the following sections:

Understanding the WSRP Standard

WSRP 1.0 is an OASIS standard that simplifies integration of remote applications and content into portals. The WSRP standard defines presentation-oriented, interactive web services with a common, well-defined interface and protocol for processing user interactions and for providing presentation fragments suited for mediation and aggregation by portals as well as conventions for publishing, finding and binding such services.

Because the WSRP interfaces are common and well-defined, all web services that implement the WSRP standard plug into all WSRP compliant portals – a single, service-independent adapter on the portal side is sufficient to integrate any WSRP service. As a result, WSRP is the means for content and application providers to provide their services to organizations running portals with no programming effort required.

See the WSRP 1.0 standard for more information:

`http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrp`

The implementation of the WSRP 1.0 standard in Portal Server includes both the WSRP consumer and the WSRP producer. The WSRP producer implementation supports publishing JSR 168 portlets for use by a remote WSRP consumer. The JSR 168 portlets are deployed locally on a portal server. These portlets can be published by an instance of the WSRP producer.

Another portal server, through its WSRP consumer, can subscribe to these remote portlets. While local portlets can be expected to provide a large part of the base functionality for portals, remote portlets allow the potential to bind to a variety of remote portlets without installation effort or code running locally on the consuming portal server.

Administering the Producer

This section discusses the following topics:

Create a producer if you want to offer locally deployed portlets remotely to other portals that act as WSRP consumers. A portal can host multiple producers. The consumer can import remote portlets offered by a producer. Based on the portlets that you want to provide to WSRP consumers, you may create one or more producers. A producer can support registration or it does not require registration. If a producer supports registration, then consumers must register to work with the producer.

Creating a Producer That Supports Registration

Registration is used to build a technical or business relationship between the consumer and the producer. While creating a producer, you can define any one of the following registration mechanisms: in-band registration or out-of-band registration:

If the producer requires registration and enabled in-band registration: the consumer can provide the details through WSRP interface and register with the producer. Consumer is also provided an option to register through out-of-band communication. That is, consumer can provide the registration handle obtained through out-of-band communication.

If the producer requires registration and enabled out-of-band registration: the consumer should obtain the registration handle through out-of-band communication and provide the registration handle during registration. Out-of-band registration happens with manual intervention such as phone calls, email, and so on. For a producer that supports out-of-band registration, the producer gets the details about the consumer through out-of-band communication, and it creates a registration handle for the consumer. The registration handle is communicated to the consumer through out-of-band communication.

▼ To Create a Producer That Supports Registration

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server from Portals.**

4 **Click the WSRP tab.**

5 **From the Select DN drop-down menu select any DN, and click the Producer tab.**

The WSRP Producers table displays all producers that are created.

Note – Organizations are created in Sun Java System Identity Server. Select the DN of an organization or suborganization based on the availability of portlets.

---

6 **Click New to create a new producer.**

7 **Type the name to identify the producer.**

8 **Select Required for Registration.**

9 **Select Supported for Inband Registration if you wish the consumer to enter the details, while adding the configured producer, using Sun Java System Portal Server application interface.**

10 **To add a registration property, click Add Row. Enter the values. Enter the name of the registration property and description.**

Note – Registration properties are the details that you want to get from the consumer while the consumer registers to a specific producer. The registration properties entered by the consumer can be validated through the Registration Validation class.

---

11 **Select Supported for out-of-band Registration if you wish the consumer to provide the details through out-of-band communication, such as phone calls, email, and so on.**

12 **Click Next.**

The Review screen displays the details that you entered. Review details. You can click Previous and change the details you entered.

13 **Click Finish.**

More Information Equivalent `psadmin` Command

```
psadmin create-producer
```

Creating a Producer That Does Not Support Registration

For a producer that does not require registration, consumer is not required to enter any information or get any information through out-of-band communication. In this case, the consumer can not customize (or edit) the portlets offered by the producer. The producer that does not support registration provides Read-Only portals to the consumers.

▼ To Create a Producer That Does Not Support Registration

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click the WSRP tab.**
5. **Select DN.**
   The Configured Producers table displays all producers that are already configured.
6. **Click New.**
7. **Type the name of the producer.**
8. **Select Registration not required.**
9. **Click Finish.**

More Information Equivalent `psadmin` Command

```
psadmin create-producer
```

Enabling and Editing WSRP Producer Properties

A newly created Producer should be enabled for a consumer to register. A producer can be enabled by adding one or more portlets.

A producer can be disabled. But, all the consumers registered with the disabled producer will not be able to access the portlets offered by the producer.

▼ To Enable and Edit the Producer's Properties

1. **In the Producer tab, click the producer name link.**

   The Edit Properties screen appears. The screen displays WSDL (Web Services Definition Language) URL. WSDL URL is a unique URL for a specific producer through which the consumer accesses the producer.

2. **Add one or more published portlets to the producer.**

   Note – The producer must have at least one published portlet to enable it. The screen displays all published portlets associated with the portal in which the producer is created.

3. **Select a portlet, and click Add.**

4. **Edit the Registration Validation Class field if required.**

   `Registration Validator` is used to validate the registration properties that are entered by the consumer. You can also customize this class based on the needs.

5. **Click Save. Now, the Enable check box displayed in the screen can be edited. Select Enable and click Save.**

   Note – You can also edit other properties of the producer.

More Information

Equivalent `psadmin` Command

```
psadmin set-attribute
```

Customizing Registration Validation Class

You can customize the `RegistrationValidator` class. Using this class, you can process the registration properties. For example, verifying the zip code of the customer. RegistrationValidator is the SPI for registration validation in the WSRP producer. For more information on customizing the validation class, see http://portalID/portal/javadocs/desktop. You can also refer to WSRP: Validating Registration Data in *Sun Java System Portal Server 7.1 Developer's Guide*.

Generating a Registration Handle

For a producer that supports registration, a registration handle needs to be generated for a specific consumer. After generating the registration handle, it needs to be communicated to the consumer to register with the producer through out-of-band communication. Consumer needs to enter the registration handle, while registering with the producer.

▼ To Generate a Registration Handle

1. **Click the Consumer Registration tab.**

   The screen displays all consumers that are already registered to the specific producer.

2. **Click New.**

3. **Type details, such as name, status, consumer agent, and method.**

Consumer name	A unique name to identify the consumer.
Status	Can be Enabled or Disabled.
Consumer Agent	Specifies the name and version of the consumer's vendor. Consumer Agent Name should be `ProductName.MajorVersion.MinorVersion`, where ProductName identifies the product the consumer installed for its deployment, and majorVersion and minorVersion are vendor-defined indications of the version of its product. This string can then contain any additional characters/words the product or consumer wishes to supply.
Method	Specifies whether the Consumer has implemented portlet URLs in a manner that supports HTML markup containing forms with method, get.

4. **Click Next.**

   The screen displays the registration property values that are specified while creating the producer.

5. **Enter the values, and click Next. Click Finish.**

Publishing Producer Details to ebXML Registry

Publishing a producer stores producer details in any one of the repositories, such as Sun Java System Service Registry Server or an ebXML Registry server. After a producer is published, you can search for the details of the producer using the application interface or using the command-line interface. For details on setting up Sun Java System Service Registry Server, see the *Service Registry 3.1 Administration Guide*.

You need to configure Sun Java System Portal Server for Registry to publish the producer details to the registry.

To Configure Sun Java System Portal Server for Registry

1. **Create the directory, `/soar/3.0/jaxr-ebxml/security`, in the machine where Portal Server is installed.**
2. **Copy `keystore.jks` from Registry Server's `/var/opt/SUNWsrvc-registry/3.0/data/security` directory to `/soar/3.0/jaxr-ebxml/security`.**
3. **Log in to the Portal Server management console.**
4. **Select the Portals tab.**
5. **Select a portal server from Portals.**
6. **Click SSO Adapter from the submenu.**
7. **Click JES-REGISTRY-SERVER.**
   The Edit Meta-adapter - JES-REGISTRY-SERVER screen appears.
8. **Type the details.**
   If you are accessing the registry server through a proxy:

http.proxy.host	Hostname of the proxy server.
http.proxy.password	Proxy password if proxy server required authentication.
http.proxy.port	Port on which proxy server is available.
http.proxy.user	Proxy username if proxy server required authentication.

   If you are not using a proxy server:

registry.keypassword	Password that is required to get the key from the keystore.
registry.keystorealias	The key alias that is present in the keystore that is to be used for authenticating with the registry server.
registry.keystorelocation	Location of the keystore relative to `/soar/3.0/jaxr-ebxml/`.
registry.keystorepassword	Password used to open the keystore.
registry.publishurl	URL of the registry server where publish request should be sent. This URL should accept SOAP requests.

registry.queryurl	URL of the registry server where search request should be sent. This URL should accept SOAP requests.

To Publish Producer Details to Registry

The following steps explain how to publish a producer to the Registry Server:

1. **Create organization data and producer data files.**

   Organization data file can contain the following entries:

   `org.name=`*Sun Microsystems*

   `org.description=`*Description*

   `org.primarycontact.name=`*Henry*

   `org.primarycontact.phoneno=`*1234567*

   `org.primarycontact.email=`*someone@host.com*

   Note – The `org.name` and `org.description` should be similar as that of the details in Identity Server unless the Registry is deployed internally.

   The producer data file should have the following entries:

   `producer.name=`*Producer_name*

   `producer.description=`*Producer_Description*

   `producer.id=`*Producer_ID*

   Note – It is not a must that you should create all the data files. But, for searching the details of producer, organization, or portlet, you should have created at least one file associated with that.

2. **Stop and restart the common agent container:**

   `/usr/lib/cacao/bin/cacaoadm stop`

   `/usr/lib/cacao/bin/cacaoadm start`

3. **To publish the produce details, use the following command:**

   `./psadmin publish-registry -u` *amadmin* `-f` *password_file* `-p` *portal1* `-m` *producer* `-U` *producer_data_file* `-O` *organization_data_file* `-T` *portlet* `-L --debug`

Note – The portlet file specifies the portlets that are offered by WSRP producer. The portlets list is specified as a string within double quotes and elements separated by space. For example, "NotepadPortlet BookmarkPortlet WeatherPortlet."

Note – You can check the log file by using the following command: `more var/opt/SUNWportal/logs/admin/portal.admin.cli.0.0.log`

More Information

Equivalent `psadmin` Command

```
psadmin publish-registry
```

Finding a Producer

The following section explains how to search for a producer:

▼ To Search a Producer

1. **Create a Search Producer data file.**

   Search Producer data file can contain the following:

   `producer.name=`*producer_name*

   `producer.description=producer_description`

   Note – The Search Producer data file contains a description of the producer to search for in the registry. Use the character `%` as a wildcard. For example, `%acme%` in producer.name any WSRP Producer that contains the string "acme" in its name.

2. **To search the registry, use the following command:**

   `./psadmin search-registry -m` *consumer* `-u` *amadmin* `-f` *ps_password* `-C` *search_producer_datafile* `-p` *portal1*

3. **Create a search Portlet data file.**

   Search Portlet data file can contain the following:

   `portlet.name=`*portlet_name*

   `portlet.description=`*portlet_description*

Note – The Search Portlet data file contains a description of the portlet to search for in the registry. Use the character `%` as a wildcard. For example, `%stock%` in portlet.name locates any Portlet that contain the string "stock" in its name.

4 **To search based on portlet details, use the following command:**

`./psadmin search-registry -m` *consumer* `-u` *amadmin* `-f` *ps_password* `-D` *search_portlet_datafile*

5 **Create a Search Organization data file.**

Search Organization data file should contain the following:

`organization.name=`*organization_name*

`organization.description=`*organization_description*

Note – The Search Organization data file contains a description of the organization to search for in the registry. Use the character `%` as a wildcard. For example, `%acme%` in organization.name locates any organization that contains the string "acme" in its name.

6 **To search based on the organization data file, use the following command:**

`./psadmin search-registry -m` *consumer* `-u` *amadmin* `-f` *ps_password* `-L` *search_organization_datafile* `-p` *portal1*

More Information

Equivalent `psadmin` Command

```
psadmin search-registry
```

Administering the Consumer

This section explains the activities need to be performed at the consumer side.

The following topics are discussed:

- "Configuring Proxies" on page 87

Adding a Configured Producer

To communicate with the portlets offered by the producer, a consumer needs to add a configured producer. If a producer requires registration, add a configured producer using the following methods:

- By entering the registration property values (in-band registration)
- By entering the registration handle (out-of-band registration)

If the producer does not require registration, the consumer is not required to enter any details while adding a configured producer.

▼ To Add a Configured Producer

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server from Portals.**

4 **Click the WSRP tab.**

5 **Select any DN and click New.**

6 **Type the configured producer name. Select the identity propagation mechanism. By default, None is selected.**

Note – Identity propagation mechanism allows the users of the consumer portal to present their credentials to the producer portal. It is a mechanism by which users can federate their identity from consumer portal to the producer portal.

---

7 **Type the WSDL URL, and click Next.**

Note – You can also search for a WSDL URL based on the producer or portlet. The search result displays WSDL URL of a producer only if the producer is published.

---

8 **If the producer requires registration, you can register the producer in two methods: by entering the registration property values (in-band registration) or entering the registration handle (out-of-band registration). Click Next.**

9. **If you selected the first method in step 7, enter the registration properties and click Next. If you selected the second method, enter the registration handle obtained through out-of-band communication, and click Next.**

10. **Review the details and click Finish.**

More Information

Equivalent `psadmin` Command

```
psadmin create-configured-producer
```

Identity Propagation Mechanism

Identity propagation is a mechanism by which the WSRP consumer supplies the identity of the user to the WSRP producer web service. It is a federation mechanism where the user federates its identity between the consumer and producer. After a successful federation, the consumer portal propagates the user identity to the producer portal. The WSRP producer, after receiving the user credentials from the consumer, validates the credentials and allows or denies access to the resource in the specified user context.

The user has two identities for each portal. That is, one for producer portal and the other for consumer portal. The user federates these identities using the identity propagation mechanism provided. This provides a single-sign on mechanism for the consumer and the producer portal. When the user logs into the portal through the consumer portal, the user gets the content that the user gets when logs directly into the producer portal. The changes that the user makes using the federated identity would be available when the user logs into the producer portal.

Sun Java System WSRP producer supports the following identity propagations:

- SSO Token: Select if both the producer portal and the consumer portal are connected to the same Access Manager instance. Typically recommended in configurations where both the producer portal and consumer portal are deployed within the same organization.
- WSS User Name Token Profile (username only): Uses the WSS specification where the user name is propagated as WS Security headers from the consumer portal to the producer portal.
- WSS User Name Token Profile (with password digest): WS Security headers send the user ID that is targeted at the producer with the password in the Digest form.
- WSS User Name Token Profile (with password text): WS Security headers send the user's user ID that is targeted at the producer with the password in the Text form.

In the above list, the last three options implement the OASIS WSS Username token profile specification. This specification describes how to use the Username Token with the Web Services. WSS specification describes how a web service consumer can supply a Username

Token by identifying the requestor by username, and optionally using a password to authenticate that identity to the web service producer.

Note – Many portal vendors support and implement the OASIS WSS Username token profile specification. Use one of the three options when interoperability is required.

There are two levels of identity propagation mechanism in Portal Server. First, the administrator of the consumer portal discovers that the producer portal supports one of the above specified identity propagation mechanisms. The administrator may allow the users to send their identity. Portal Server consumer supports all the above mentioned Identity Propagation Mechanisms.

After the consumer is created, the administrator has to create remote channels based on the identity propagation mechanism supported by the consumer. After the channels are available on the user Desktop, they are ready to accept identity propagation.

The identity propagation mechanism is set at the producer automatically. Portal Server checks for authentication from Sun SSO, then OASIS user name token profile, and then the No Identity Propagation mode.

Configuring Digest Passwords

Only new users can use the Digest Password facility after running the `configuration` command to store the LDAP passwords in plain text

Creation of a consumer should involve selecting the WSSO Username Token Profile (with Digest Password) option for User Identity Propagation Mechanism.

The Web Services SSO Portlet must be edited to select the appropriate Web service URL (producer) and provide the new username and password.

▼ To Configure the Accept Digest Passwords

Do the following to configure Sun Java System WSRP Producer to accept Digest Passwords.

1 **Run the command** `/opt/SUNWdsee/ds6/bin/dscfg set-server-prop pwd-storage-scheme:CLEAR` **to change the password storage scheme of the Directory Server so that plain text passwords are stored.**

Note – It is assumed that the default installed location of the Directory Server is `/opt/SUNWdsee`.

---

2 **Create a new user in the AM console, to ensure that the Username Token Profile with Password Digest can be used.**

More Information Recommendations

- When using the WSS User Name Token Profile (with PasswordDigest), communication between the producer portal and consumer portal should be secure because the password is sent in plain text between the consumer and the producer.
- Two different consumers that point to the same producer URL should use the same identity propagation mechanism types.

Creating User Token Profiles Using WebServices SSO Portlet

You can create user token profiles to authenticate user credentials if the user uses identity propagation mechanism. You can define the user name and password for specific Web service that the producer offers.

▼ To Provide User Credentials Using WebServices SSO Portlet

1 **Log in to Portal Server Desktop.**

2 **In the WebServices SSO Portlet, click the Edit button.**

3 **In the Create NewToken Profile section, select the WebService URL for which you want to create a user token profile.**

4 **Type the user name and password. Click Add.**

You can also edit or remove an existing user token profile.

Updating Service Description

After the consumer configures the producer, use the Update Service Description option to update any changes made to the producer later. For example, addition of new portlets or changes to the registration properties after the registration.

To Update Service Description

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal server from Portals.
4. Click the WSRP tab.
5. Select DN (Distinguished Name).
6. Click the configured producer link.
7. In the Edit Configured Producer screen, click Update Service Description.

More Information

Equivalent `psadmin` Command

```
psadmin update-configured-producer-service-description
```

Mapping User Categories to Roles

WSRP supports the concept of user categories, which are included in the service description of the producer. Mapping user categories to the roles allows the user to map the roles that are defined in the consumer portal to the roles that are defined in the portlet. Sun Java System Portal Server maps Java System Access Manager's roles to the portlet's roles. These roles can be mapped to the corresponding WSRP user categories.

You can perform the following tasks:

- "To Create Roles in Portlets" on page 85
- "To Map User Categories to Role" on page 86

Roles can be defined in the portlet while deploying the portlet.

Note – The roles defined in the portlet must exist in the Access Manger of the producer.

To Create Roles in Portlets

The following task creates a role in amconsole in Sun Java System Access Manager and Portlets.

1. Log in to the Access Manager console.

2. **Create a role and add a user to it.**

3. **In webxml of the portlet application, add the following code:**

   ```
   <security-role>
   <role-name>PS_TEST_DEVELOPER_ROLE<role-name>
   </security-role>
   ```

4. **Add the following lines in** `portlet.xml` **of the portal.**

   ```
   <security-role-ref>
   <role-name>PS_TEST_DEVELOPER_ROLE<role-name>
   <role-link>PS_TEST_DEVELOPER_ROLE<role-link>
   </security-role-ref>
   ```

5. **Create the portlet application war file.**

6. **Create a roles file with the following entry.**

   ```
   cn\=AM_TEST_DEVELOPER_ROLE,o\=DeveloperSample,dc\=india,dc\=sun,dc\=com=PS_TEST_DEVELOPER_
   ```

7. **Deploy the portlet using the following command.**

   `/opt/SUNWportal/bin/psadmin deploy-portlet -u` *amadmin* `-f` *ps_password* `-d` *"o=DeveloperSample,dc=india,dc=sun,dc=com"* `-p` *portal1* `-i` *stockprice-8080* `--rolesfile` *rolesfile* `TestPortlet.war`

More Information Equivalent `psadmin` Command

```
psadmin deploy-portlet
```

▼ To Map User Categories to Role

Do the following to map user categories to role:

1. **In the Consumer tab, click the producer name link.**

   The Edit Configured Producer screen displays the following: User Category: The roles in the producer portlet. Local Roles: The roles that are defined at the consumer's Sun Java System Access Manager.

2. **In the User Categories to Role Mapping section, map user categories to the roles defined at the consumer, and click OK.**

Mapping Consumer Attributes

The Sun Java System Portal Server implementation of WSRP Consumer maps common user attributes stored in the user entry on the Sun Java System Directory Server to the standard set of user attributes that the WSRP specification mandates.

If a consumer portlet uses any of the attributes that are not specified in the LDAP schema, create a custom object class to store these attributes and add this object class to the user entry. After attributes are created, map the LDAP attribute to the corresponding WSRP attribute using Sun Java System Access Manager management console.

Configuring Proxies

Proxies need to be configured for consumer and for web container XML files.

You can perform the following tasks:

- "To Configure Proxy for Consumers in Common Agent Container" on page 87
- "To Configure Web Container XML file" on page 87

▼ To Configure Proxy for Consumers in Common Agent Container

1. **Run** `./cacaoadm get-param java-flags`**.**
2. **Copy the values and paste it to** `./cacaoadm set-param java-flags`**.**
3. **Now add the following to the command:** `-Dhttp.proxyHost=`*webcache.canada.sun.com* `-Dhttp.proxyPort=`*8080* `-Dhttp.proxyUser=`*Proxyuser* `-Dhttp.proxyPassword=`*Password*
4. **Press Enter.**
5. **Restart the common agent container server.**

▼ To Configure Web Container XML file

1. **Edit the following file:**

   ```
   vi /var/opt/SUNWappserver/domains/domain1/config/domain.xml
   ```

2. **Set the following JVM options:**
   - Dhttp.proxyHost
   - Dhttp.proxyPort

- Dhttp.proxyUser
- Dhttp.proxyPassword

Administering the WSRP Producer

This section describes how to administer the Sun Java System Portal Server Web Services for Remote Portlets (WSRP) service. The tasks to administer a WSRP producer are:

▼ To Create a WSRP Producer

A WSRP producer is created with the following:

- Name of the producer instance (must be unique for the entire portal server)
- Whether registration is required. When registration is required, all WSRP consumers must register with this producer instance before making requests. Requests from unregistered WSRP consumers will be denied.
- Whether in-band registration is supported. In-band registration allows WSRP consumers to register programmatically. Otherwise, out-of-band registration is required with manual contact (such as email or telephone) between the WSRP consumer administrator and the WSRP producer administrator to set up and exchange access to a registration handle.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Producers from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **From WSRP Producers click New to launch the wizard**
7. **Follow the instructions to create the specified producer.**

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*

More Information Equivalent `psadmin` Command

```
psadmin create-producer
```

▼ To Edit a WSRP Producer

You can edit the WSRP Producer as follows:

- Add or remove portlets from the published list
- Change the requirement on registration

Caution – This option should be modified for an existing producer.

- Enable or disable in-band registration
- Specify the Registration Validator Class. The registration validator class is used by the WSRP Producer to validate that the values sent by the WSRP consumer are acceptable.
- Add new registration properties. Any change in properties will apply to subsequent consumers registering with the producer.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Producers from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Select a WSRP producer and modify the configuration attributes as necessary**

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*
7. **Click Save to record the changes.**

More Information Equivalent `psadmin` Command

```
psadmin set-attribute
```

▼ To Create a Consumer Registration

Each consumer registration represents a remote WSRP consumer that has established a relationship with the WSRP producer. A WSRP producer that supports allows multiple WSRP consumers to register with it. The registration mechanism allows a WSRP consumer to describe its capabilities to a WSRP producer.

A WSRP consumer is added out of band (such as by email or telephone). The information entered when adding a consumer registration must match the capabilities of the WSRP consumer that is given the registration handle. Consumer registrations allow a WSRP producer to scope artifacts (such as portlet preferences) that a WSRP consumer creates on the WSRP producer.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Producers from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Select a WSRP producer, then Consumer Registrations.**
7. **Click New to launch the wizard.**
8. **Follow the instructions to create the specified consumer registration.**

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*

More Information

Equivalent `psadmin` Command

```
psadmin create-consumer-registration
```

▼ To Edit a Consumer Registration

You can edit existing consumer registrations manually. Note that this could also be done via in-band registration from the WSRP Consumer end. Ensure that both out of band and in band registration are not used simultaneously.

1. **Log in to the Portal Server management console.**

2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Producers from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Select producers, then select a WSRP producer, then Consumer Registrations.**
7. **Select a consumer registration and modify the configuration attributes as necessary.**

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*
8. **Click Save to record the changes.**

Administering the WSRP Consumer

This section describes the tasks to administer the WSRP Consumer:

- "To Add a Configured Producer" on page 91
- "To Edit a Configured Producer" on page 92
- "To Specify the Consumer Name" on page 92

▼ To Add a Configured Producer

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Producers from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Under Configured Producer click New to launch the wizard**
7. **Follow the instructions to create the specified configured producer.**

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*

More Information

Equivalent `psadmin` Command

```
psadmin create-configured-producer
```

▼ To Edit a Configured Producer

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**
4. **Click WSRP, then Consumer from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Select a configured producer and modify the configuration attributes as necessary.**

   Note – Use the Update Service Description option to update any changes made to the producer. See "Updating Service Description" on page 84.

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*
7. **Click Save to record the changes.**

More Information

Equivalent `psadmin` Command

```
psadmin set-attribute
```

▼ To Specify the Consumer Name

The WSRP consumer sends the consumer name to producers during registration. The value specified for the consumer name is used as the default unless a value is specified for consumer name at the organization or suborganization level.

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server from Portals.**

4. **Click WSRP, then Consumer from the submenu.**
5. **From Select DN drop-down menu choose any DN.**
6. **Under WSRP Consumer, click Edit.**
7. **Specify the consumer name.**
8. **Click OK.**

More Information

Equivalent `psadmin` Command

```
psadmin set-attribute
```

◆ ◆ ◆ CHAPTER 6

Managing Portal Server End-User Behavior Tracking

This chapter describes how to track Sun Java System Portal Server user behavior.

This chapter contains the following sections:

Understanding Portal Server User Behavior Tracking

Portal Server user behavior tracking (UBT) provides a way to track end-user activity on the Portal Server application. User activity on Portal Desktop is captured into a `ubt` log file. The `ubt` log file is recorded in a W3C standard Extended Log File Format. From this log file, you can create various end-user behavior tracking reports using the Portal Server console or the `psadmin generate-ubt-report` command. You can also use third-party tools such as`AWStats` to generate UBT reports.

You can also enable UBT from the `UBTConfig.properties` file. Go to `/var/opt/SUNWportal/portals/portalID/config/UBTConfig.properties` and set `com.sun.portal.ubt.enable=true`.

The table shows the list of UBT reports, their description, and the available format of the reports.

TABLE 6–1 User Behavior Tracking Reports

Report Name	Report Description	Report Formats
Portal User Identity Report	This report lists users along with time of their last portal access. Users are grouped as per the server they accessed, domain they belong to, and relative DN.	HTML or PDF
Portal User Login Rate	This report shows the rate of logins into portal.	
Portal Channel View Report	This report lists users viewing a channel along with number of times they viewed that channel. The channels are grouped as per the containers they belong to.	HTML or PDF
User Customization of Portal Containers	This report shows the portal container customization. Container customization usually refers to content, layout or theme changes on the Desktop.	HTML or PDF
Portal Request Rate	This report shows the rate of request of each top container every hour over a period of time. The top container request is considered a page request.	HTML or PDF
User Customization of Portal Channels	This report lists end users along with the actions they performed on the channels. Users are grouped by the containers they access, and by channels on which they performed actions.	HTML or PDF
Portlet Actions Report	This report shows the rate of portlet action requests in the portal.	HTML or PDF
Portlet Render Report	This report shows the number of times a portlet is displayed in a portlet mode in a particular window state. In MINIMIZED window state, a portlet is not rendered, and the count for this state is not displayed.	HTML or PDF
Portal User Login Rate Report	This report shows the rate of logins into the portal.	HTML or PDF

Setting Up Portal Server User Behavior Tracking

This section has information on how to enable user behavior tracking and generate reports.

You can perform the following tasks from the portal server management console:

- "To Enable the User Behavior Tracking Logging" on page 97
- "To Generate User Behavior Tracking Reports" on page 97

▼ To Enable the User Behavior Tracking Logging

By default, UBT logging on a Portal Server application is not enabled.

1. **Log in to the Portal Server management console.**
2. **Select the Common Tasks tab.**
3. **Under Reports and Logs, click Portal Usage Reports to launch the wizard.**
4. **From Select Portal drop-down menu select a portal instance, and click OK.**

   The User Behavior Tracking page is displayed.
5. **Click the Settings submenu and enable UBT logging under Common Properties.**

   For more information on Common Properties, Handler Properties and Event Settings, see *Sun Java System Portal Server 7.1 Technical Reference*

   Note – For all other properties, default values are already set and are sufficient for UBT to work. To apply the changes to all instances of Portal Server, click the Apply to All Instances button. Otherwise, click the Apply to Selected Instance button.

   ---
6. **Access the portal Desktop and make sure user behavior tracking log files are generated.**

   By default, user behavior tracking logs are written into `/PortalData-Dir/portals/PortalID/logs/instanceID/ubt.0.0.log` file.

▼ To Generate User Behavior Tracking Reports

1. **Log in to the Portal Server management console.**
2. **Select the Common Tasks tab.**
3. **Under Reports and Logs, click Portal Usage Reports to launch the wizard.**

4. **From Select Portal drop-down menu select a portal instance, and click OK.**

   The User Behavior Tracking page is displayed.

5. **Click the Reports submenu.**

   Eight reports are listed. All these reports can be generated either in PDF or HTML format. See Table 6–1 for more information.

More Information

Equivalent `psadmin` Command

```
psadmin generate-ubt-report
```

◆ ◆ ◆ CHAPTER 7

Monitoring Portal Server Activity

This chapter describes how to set up the Sun Java™ System Portal Server monitoring.

This Chapter contains the following sections:

Understanding Portal Server Monitoring

Monitoring helps record runtime resource information about portal server. Desktop monitoring keeps record of information on requests received by portal server for content, edit, and process types. It also records information on the minimum, maximum and average response time for each type of request for the different channels of portal server.

Information gathered from monitoring portal activity is useful to optimize portal response time either by moving channels that need a higher response time to separate secondary tab, or by setting the time-out property for Desktop channels based on cache hits.

The Java Virtual Machine (JVM) in a portal server collects monitoring data for the Desktop. Monitoring information can be viewed on portal server management console, or can be accessed using `psadmin` monitoring subcommands. See *Sun Java System Portal Server 7.1 Command Line Reference*.

Monitoring uses Java Management Extensions (JMX™ technology) and registers Management Beans (MBeans) in the portal server instance's MBeansServer that represents portal server Desktop and portal Desktop channels. Each MBean attribute represents monitoring data collected for each resource. The portal management console and `psadmin` monitoring subcommands communicate with MBeans to collect and present monitoring data for a portal server instance.

Setting Up Portal Server Monitoring

Monitoring can be configured by accessing monitoring properties stored in `/var/opt/SUNWportal/portals/`*portalID*`/config/instanceID/monitoring.properties` file. Monitoring is enabled by default. To disable monitoring, set `com.sun.portal.monitoring.MonitoringContext.monitoring.disable` property to true. When the JVM restarts, monitoring is disabled.

You can also enable or disable monitoring from the portal management console.

▼ To Enable or Disable Portal Monitoring

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals.**
4. **Click the Monitoring tab.**
5. **Click Settings submenu.**
6. **Select a portal server instance.**
7. **Click Enable Monitoring or Disable Monitoring button.**

▼ To View Desktop Statistics

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals.**
4. **Click the Monitoring tab.**
5. **Click Desktop Request/Response Statistics from the submenu.**

▼ To View Channel Statistics

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal server under Portals.
4. Click the Monitoring tab.
5. Click Channel Action Statistics from the submenu.
6. From Select DN drop-down menu choose an organization.
7. Select the server from the Server Instance drop-down menu.

Collecting Portal Server Monitoring Data

Monitoring collects seven types of data requests received by the Desktop. Each type of request is represented as MBean with type `DesktopRequestStatistic`, and name MBean property as the request type. For example, `type=DesktopRequestStatistics,name=Content` name properties help identify Desktop content request statistics.

Desktop Statistics

The seven types of requests are explained in the following list:

Content	The number of times Desktop successfully served content requests, and the time taken for it.
Edit	The number of times Desktop successfully served edit requests , and the time taken for it.
Exception	The number of times Desktop could not serve a request due to some exception during request processing. Exception information is logged in portal server log files.
LocalAuth	The number of times Desktop responded to local authentication requests.
Logout	The number of times user logged out from portal server, and how long it took to log out
PreLogin	The number of times Desktop responded to pre-login requests.
Process	The number of times Desktop processed edit requests, and the time taken for it

You can view the Desktop statistics from the portal management console.

Channel Statistics

Each type of channel action is represented as MBean with type `ChannelActionStatistic` along with additional name properties that identify the channel. To know the full MBean name, use the command `psadmin get-monitoring-mbean-names`.

Portal Desktop presents cached content view for a channel based on time-out channel property

The types of channel actions that are monitored for each Desktop channel are explained in the following list:

Content — The number of times channel provider successfully generated the content view, and the response time for it.

Edit — The number of times channel provider successfully presented the edit view, and the response time for it.

Process — The number of times channel provider processed the edit view.

You can view the Desktop statistics from the portal management console.

◆ ◆ ◆ CHAPTER 8

Managing Portal Server Logging

This chapter describes how to obtain Sun Java™ System Portal Server log information.

This chapter contains the following sections:

Understanding Portal Server Logging

Portal Server supports logging across all components. The logs and log configuration are uniform across portal components. Seven standard log levels range from severe to fine grain. The logs can be routed to different files or data sinks and can consist of a single file or multiple files; that is, one for each component.

Log levels can be set for each module and sub-module, and logs can be routed to separate files for each module and sub-module within each component.

Managing Portal Server Logging

You can set up and manage Portal Server logging using the following components:

- Log Viewer
- Common Logger settings
- Specific Logger settings

You can manage portal logging from the portal management console.

▼ To Manage the Log Viewer

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server under Portals.**

4 **Click Logging, then Log Viewer from the submenu.**

5 **From the Instance Name drop-down menu, select a portal instance.**

The Search Criteria and Search Results page for the log viewer is displayed.

6 **Enter the values for the Search Criteria, and click Search.**

The following search options are available:

Log File Name	File name that has the log content.
Log Level	Messages at the selected level or higher appear in the log. The available levels are SEVERE, WARNING, INFO, CONFIG, FINE, FINER, and FINEST. The default level is INFO, so the log will contain messages of INFO, WARNING, or SEVERE levels. To ensure that the messages you want to view appear in the log, first set the appropriate log levels on the Specific Logger Settings page.
Timestamp	Displays log messages of a certain time period. You can view 100 most recent log entries, or type a time period in the From and To text boxes. If you choose a Specific Range: ■ Both the From Date and To Date values are required ■ The From Date value cannot be later than the To Date value ■ The To Date value cannot be later than Today's Date ■ The From Time and To Time values are optional. If the From Time value is specified, then the To Time value has to be specified. For the Time value, the syntax must take the form hh:mm:ss.SSS. SSS stands for milliseconds. For example, 18:20:10.000

More Information

Equivalent `psadmin` Command

```
psadmin set-logger
```

▼ To Customize the Log Display

You can customize the Search Results page using the following steps:

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a Portal Server under Portals.**
4. **Click Logging, then select a portal server from the Instance Name drop-down menu.**
5. **In the Log Viewer Results table, click the Timestamp column header to sort the messages.**
6. **Click the details link to view a formatted log message in a new window.**

▼ To Manage Common Logger Settings

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a Portal Server under Portals.**
4. **Click Logging, then Common Logger settings from the submenu.**
5. **From the Instance Name drop-down menu, select a portal instance.**
6. **Modify the configuration attributes as necessary.**

   The following options are available:

   General

   Log Level — You can choose what information to view in a log file by selecting a log level setting.

   The choices for Log level include:

   - Severe - errors visible to users
   - Warning - user warnings
   - Info - informative for users
   - Config - static setup information for developers
   - Fine - basic tracing information
   - Finer - detailed tracing information
   - Finest - complete tracing information

- Off - can be used to turn off logging
- All - indicates that all messages should be logged

File Handler Properties

- Limit — Specify the size of the log file in bytes. If the log file size exceeds this value, the log file will be rotated based on file count. The default value is 5 megabytes.
- File Count — When the log reaches the specified size in bytes, create a new empty file with the generation number (%g in the File Pattern) incremented by 1. The default value is 2. To turn off log file rotation, set the value to 0.
- Append — Specify whether the new message is to be appended to the existing file. Default is true.
- Filter — To filter log records that are sent to destinations such as portal log or a destination specified by a custom log handler, you can plug in a custom log filter. The custom filter must implement the interface `java.util.logging.Filter`. Type the absolute class name of the filter in the field. Also put the filter class in the Application Server classpath so that the filter is installed during server startup.

Other

- Custom Handlers — To send logs to a destination other than portal log, you can plug in a custom log handler. The custom handler must extend the class `java.util.logging.Handler` (a JSR 047 compliant API). Type the absolute class name of the handler in the field. Also put the handler class in the Application Server classpath so that the handler is installed during server startup. You can specify more than one handler. Use comma to separate multiple names.
- Use Web Container Log File — To disable portal logging administration and route all logs to the web container log file, chose Yes, other chose No. Default is No.

7 Click Apply to the Selected Instance or Apply to All Instances to record the changes.

More Information

Equivalent `psadmin` Command

```
psadmin set-logger
```

▼ To Manage Specific Logger Settings

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a Portal Server from Portals.**
4. **Click Logging, then Specific Logger settings from the submenu.**
5. **From the Instance Name drop-down menu, select a portal instance.**
6. **Modify the configuration attributes as necessary.**

   The following options are available:

   Logger Settings

   - Logger Name – Click the logger name to get the configuration details of the logger.
   - Log Level – You choose what information to view in the log file for the logger by selecting a log level setting or you can inherit the log level from the parent logger. For example, if the log level of `debug.com.sun.portal` is INFO and the log level of `debug.com.sun.portal.desktop` is Inherit Parent Logger Level, then its value will also be INFO.
   - Log File Merge Strategy – For a logger, you can choose whether you want the log messages in the same log file as parent (Log to Parent Log File) or the log should go to a separate file (Log to Separate Log File).
   - Parent Handler – For a logger, if the Log File Merge Strategy is set to Log to Separate Log File, you can choose whether you want the messages to be logged to both the separate log file as well as the parent log file (Inherit Parent Handlers) or log to separate file only (Do not Inherit Parent Handlers).
   - Parent Handler – For a logger, if the Log File Merge Strategy is set to Log to Separate Log File, you can choose whether you want the messages to be logged to both the separate log file as well as the parent log file (Inherit Parent Handlers) or log to separate file only (Do not Inherit Parent Handlers).
   - Stacktrace – For a logger, you can choose whether you want the stacktrace to be logged for all levels (Print Stack Trace for All Levels) or for only till WARNING log level (Print Stack Trace till Warning Level).

Note – If Log File Merge Strategy value is Log to Parent Log File, Parent Handler and stacktrace values are ignored. If Log File Merge Strategy value is Log to Separate Log File, and if Parent Handler value is Inherit Parent Handlers, the Stacktrace value Print Stack Trace for All Levels is not valid.

7 Click Apply to the Selected Instance or Apply to All Instances to record the changes.

More Information

Equivalent `psadmin` Command

```
psadmin set-logger
```

◆ ◆ ◆ CHAPTER 9

Managing Portal Server Subscriptions

This chapter describes the Sun Java™ System Portal Server subscriptions component and how to manage it. The chapter contains following topics:

Understanding Portal Server Subscriptions

Subscriptions enable end users to create a profile covering many sources of information, including categories, discussions, and searchable documents. The profile is updated with the latest information each time the end user accesses the Subscriptions channel. The Subscriptions channel summarizes the number of items of relevant information that match each profile entry that the end user defines for categorized document or discussions.

You can match the following types of content using the search server:

- New documents in a target category from a specified range of days
- New relevant comments within a discussion from a specified range of days
- Document hits against saved searches

The result is displayed as a link that shows the number of matching information to the profile entry. This link redirects the end user to a more detailed view of the match itself.

In case of a category subscription, the link redirects the end user to the search channel, which summarizes the specific documents of interest in a standard category search result format. The Subscriptions channel acts as the doorway to a more detailed view for the end user.

The Profiler function provides email notifications when the content of specified interests has changed. The Profiler obtains subscription details for end users from the Access Manager, fetches the results from the Search server, and sends email notifications to end users. You can schedule the Profiler to run at a specific time at the organization level.

Setting Up Subscriptions

You can enable or disable subscriptions. Subscriptions can be set up at the:

- Root Level
- Organization Level
- End-User Level

▼ To Set Up Subscriptions

1. **Log in to the Portal Server management console.**
2. **Select the Portals tab.**
3. **Select a portal server under Portals.**
4. **Click the Subscriptions tab.**
5. **Set the subscriptions level by choosing one of the following, and set the default values:**

   - **From the Select DN drop-down menu, choose TopLevel [Global].**

     Note – Administering subscriptions at the TopLevel sets the system-wide default maximum number of subscriptions for each type, or for categories, discussions, and saved searches.

     Maximum number of Categories subscriptions
     Specifies the maximum number of categories that a user can subscribe to.

     Maximum number of Discussion subscriptions
     Specifies the maximum number of discussions that a user can subscribe to.

     Maximum number of Saved searches
     Specifies the maximum number of searches that can be saved.

   - **From the Select DN drop-down menu, choose any Organization.**

     Note – Administering Subscriptions at the Organization level overwrites the system-wide default maximum number of subscription per type (that is, for categories, discussions, and for saved searches).

Profiler SMTP	The host system that serves as the SMTP server to route Email notifications to the end users.

Profiler Email	Subscription profiler email address from which the user receives email notification. Email should be in the form ID@domain.
Profiler Provider	The URL of the Profiler channel that is used to render the content of the Email notification to the user. It should be in the form of `http://HOST:PORT/portal/dt?provider=profiler&desktop.suid=UID_OF_AUTHLESSANO`
Profiler Default Search	The URL of the default search server. Profiler Default Search is only used for backward compatibility with user profiles created with Portal Server 6.3.x. It should be in the format `http://HOST:PORT/search1/search`
Profiler Max Hits	The maximum number of result hits that any given end user subscriptions in the organization will see in email notification sent to a user. For example, if the value is 5, a saved search with a large scope like "*" is limited with five most relevant results.
Maximum Category subscriptions	The maximum number of categories that a user can subscribe to.
Maximum Discussion subscriptions	The maximum number of discussions that a user can subscribe to.
Maximum Saved Searches	The maximum number of searches the end user can save.

- **From the Select DN drop-down menu, choose any User.**

Note – Administering Subscriptions at the Organization User level edits user's Subscriptions settings. The administrator can maintain the user's service data.

- Update user subscriptions
- Delete user subscriptions

Profiler Enabled Allows users to receive email notifications by selecting Enabled.

For each type of subscription, add or remove subscriptions. The format of:

Category subscription

```
label | target category | scope | lapsed time | rating | server | database | status
```

where

`label`	Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.
`target category`	Must be of the string format *ABC:DEF:GHI*
`scope`	Refers to a search query and it must be of a string format that is a valid search string, including search operators.
`lapsed time`	Must be one of the following numbers: ■ 0 = forever ■ 1 = since yesterday ■ 7 = since last week ■ 30 = since last month ■ 180 = since last 6 months ■ 365 = since last year
`rating`	This is the minimum rating that a matching document should be to be selected as a match for the subscription. Values are number ■ –1 = irrelevant ■ 0 = routine ■ 1 = interesting ■ 2 = important ■ 3 = must read
`server`	This is the URL of the search server that will be queried to find content matching subscription's criteria.
`database`	Target search server database where subscription searches for potential matches. This is a single value database.
`status`	Boolean value that marks whether the subscriptions is active or inactive. ■ Active means the subscriptions is to be evaluated.

- Inactive means the subscriptions is dormant.

Discussions subscriptions

```
label | target discussion | scope | lapsed time | rating | server | database | status
```

where:

`label`	Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.
`target discussion`	Parent node of the discussion thread from which subscriptions will try to find matching content for other defined criteria.
`scope`	Refers to a search query. `scope` must be a string format that is a valid search string, including search operators.
`lapsed time`	Must be one of the following numbers: ■ `0` = forever ■ `7` = since last week ■ `30` = since last month ■ `180` = since last 6 months ■ `365` = since last year
`rating`	This is the minimum rating that a matching document should be to be selected as a match for the subscription. Values are number ■ –1 = irrelevant ■ 0 = routine ■ 1 = interesting ■ 2 = important ■ 3 = must read
`server`	This is the URL of the search server that will be queried to find content matching subscription's criteria.

`database` — Target search server database where subscription searches for potential matches. This is a single value database.

`status` — Boolean value that marks whether the subscriptions is active or inactive.

- Active means the subscriptions is to be evaluated.
- Inactive means the subscriptions is dormant.

Saved searches

```
label | scope | lapsed time | rating | server | database | status
```

where

`label` — Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.

`scope` — Refers to a search query and if must be of a string format that is a valid search string, including search operators.

`lapsed time` — Must be one of the following numbers:

- `0` = forever
- `1` = since yesterday
- `7` = since last week
- `30` = since last month
- `180` = since last 6 months
- `365` = since last year

`rating` — This is the minimum rating that a matching document should be to be selected as a match for the subscription.

Values are number

- –1 = irrelevant
- 0 = routine
- 1 = interesting
- 2 = important
- 3 = must read

`server` — This is the URL of the search server that will be queried to find content matching subscription's criteria.

`database`	Target search server database where subscription searches for potential matches. This is a single value database.
`status`	Boolean value that marks whether the subscriptions is active or inactive. ■ Active means the subscriptions is to be evaluated. ■ Inactive means the subscriptions is dormant.

6 **Click Save.**

More Information Equivalent `psadmin` Command

```
psadmin set-attribute
```

Administering Portal Server Discussions

This section describes the discussions channel and how to manage it.

This section contains the following:

Understanding DiscussionProvider

The Discussions channel is based on the DiscussionProvider, similar to the search channel's JavaServer Pages™ (JSP™) files. The discussion channel has a query portion and a display portion, and uses Desktop themes.

The DiscussionProvider:

- Uses the Desktop themes
- Is based on JSP technology
- Retrieves data from the back-end Search service using search tag libraries and API

Discussions and comments are stored as different Resource Descriptors (RDs) in the discussion database. The DiscussionProvider supports:

- A full view (using the Discussions channel) and an abbreviated view (using the DiscussionLite channel) that:
- Starts a new discussion from the discussion channel

- Posts replies to an existing discussion
- Starts a new discussion based on web documents from the search channel
- A Discussion List that:
  - Retrieves main posts sorted by last-modified date
  - Has pagination so users can access older discussion
- A discussion view that displays each discussion subtree. The main item is displayed in detail and the subtree is displayed below the main item. View discussion includes:
  - Several filters on the page. A document display can be based on filters such as document rating (irrelevant, routine, interesting, important, and must read).
  - Display preference can be set to threaded or flat display.
  - Expansion threshold to help control displayed items in the subtree. The users can choose to expand only highly rated documents, or expand all or collapse all. Default value is collapse all. Expand all displays all the filtered comments, shows a description of the discussion, provides a menu for rating the discussion, and allows the user to post a reply.
  - Support to search within a discussion. The user also has the option to set these preferences through the channel edit page.
- Commenting and rating a discussion. For example, users can:
  - Add a comment on an existing discussion.
  - Rate all discussions and comments. User rating is not immediately visible. The rating calculation is based on an algorithm, and the rating for any comment goes up gradually. For example, a comment must be rated important three times before it is marked as important.
  - Searching all discussions and within a discussion. These functions are routed to the search provider. Users can also search by rating in Advance Search.
  - Subscriptions. Authenticated users can choose to subscribe to a particular discussion by selecting the subscribe link. The request is handled by the SubscriptionProvider.

Administering the DiscussionProvider

You can create a DiscussionProvider channel and manage it from the portal server management console:

- "To Create a Channel from DiscussionProvider" on page 117
- "To Delete a DiscussionProvider Channel" on page 117
- "To Configure a DiscussionProvider Channel" on page 118

End users can configure the discussion channel using the channel edit page.

To Create a Channel from DiscussionProvider

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal server under Portals.
4. From the Select DN drop-down menu, select any DN.
5. Select the container where you want to create the channel.
   The container Task and Properties are displays on the right panel.
6. Under Tasks, click New Channel or Container to launch the wizard.
   a. From the Select Portal drop-down menu, select a portal server.
   b. From the Select DN drop-down menu, select any DN.
   c. Under Type, select channel, and click Next.
   d. Under Channel Type, select Provider Channel, and click Next.
   e. From the Provider drop-down menu, select DiscussionProvider, and click Next.
   f. Type a name for the channel in the text box, and click Next.
   g. Review the channel information, and click Finish.
   h. Click Close.

   The channel based on DiscussionProvider is created.

To Delete a DiscussionProvider Channel

1. Log in to the Portal Server management console.
2. Select the Portals tab.
3. Select a portal server under Portals.
4. From the Select DN drop-down menu, choose the DN where the DiscussionProvider channel resides.

Tip – Select DP XML Tree as the View Type from the drop-down menu for a listing of all the channels and containers under DP_ROOT.

5 **Select the container where the channel resides.**

The container Tasks and Properties page displays.

6 **Click Select Channel or Container to delete.**

7 **Select the DiscussionProvider channel.**

8 **Click Delete.**

▼ To Configure a DiscussionProvider Channel

1 **Log in to the Portal Server management console.**

2 **Select the Portals tab.**

3 **Select a portal server under Portals.**

4 **Choose DN organization where the DiscussionProvider channel resides from Select DN drop-down menu.**

Tip – Select DP XML Tree as the View Type from the drop-down menu for a listing of all the channels and containers under DP_ROOT.

5 **Select the DiscussionProvider channel you want to configure.**

For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*.

DiscussionLite Channel

The DiscussionLite channel displays the top 20 recent discussion titles and the date. Discussions are sorted by creation date (last modified), and the newest discussion is displayed first. Titles can be reconfigured.

The DiscussionLite channel view has links for:

- Viewing each discussion.
- Viewing all discussions that target the Discussions Channel.

- Starting a discussion.

By default, the channel is displayed in a single container, and all links are brought up in a `JSPDynamicSingleContainer`.

Properties can be configured from the management console. By default, the end user cannot edit properties of this channel.

◆ ◆ ◆ CHAPTER 10

Managing the Portal Server Single Sign-On Adapter

This chapter describes how to configure the single sign-on (SSO) adapter in order to adjust options available to end users. This chapter contains the following sections:

Overview of the Single Sign-On Adapter

The single sign-on adapter service allows end users to use applications, such as a portal server provider or any other web application, to gain authenticated access to various resource servers after signing in once. The resource servers that can be accessed depend on the implementations of the SSO Adapter interface that are available in the system.

Portal Server provides SSO Adapters for the following resource servers: Address Book, Calendar, and Mail. Single Sign-On for the Instant Messaging channel is not achieved through SSO Adapter but through the use of the Sun Java System Portal Server authentication method. For information on this method, see the `authMethod` property in Instant Messaging Channel . The Address Book, Calendar, and Mail services are available through the products:

- Sun Java System Calendar Server 5.1.1, 6.0, 6 2006Q2
- Sun Java System Sun Java System Messaging Server 5.2, 6.0, 6 2006Q2

Resource servers are typically accessed by an application using a standard application programming interface (API), such as the JavaMail™ API for accessing a mail server. To create an authenticated connection using the API, the API must be provided the configuration data for the connection. The purpose of the SSO Adapter is to provide this configuration data, and the SSO Adapter service is used to store that data.

The SSO Adapter service defines two levels of data, meta-adapters and adapters. A meta-adapter defines a class of connections that are going to be made available to users. A single

meta-adapter is used by many users. It defines data values that are the same for all users that use the meta-adapter including default values and identification of what values can be edited by a user. Therefore, meta-adapters are defined at a global service level.

An adapter builds upon a meta-adapter by providing data values that are specific to an organization, role, or user. An adapter references a meta-adapter, and takes data values from the meta-adapter for those properties that are not editable by the user. When an end user changes the user-editable properties of an adapter, that adapter would then apply only to that one user.

A Sun Java System Sun Java System Portal Server communication channel that uses the SSO Adapter service references either a meta-adapter or an adapter to get data values needed to obtain a connection to a resource server. If the channel references a meta-adapter, and the user saves configuration information, the reference is changed to refer to an adapter instead. The adapter then references the meta-adapter.

All administration for the SSO Adapter is done either through the Portal Server console web application or the `psadmin` command-line interface. The default deployment URI for Portal Server console is `/psconsole`. The default location for the psadmin CLI is `/opt/SUNWportal/bin` for Solaris.

Managing Meta-Adapters

A meta-adapter defines a class of connections that are going to be made available to users. A single meta-adapter is used by many users.

You can perform the following tasks using meta-adapters:

- "To View Meta-Adapters" on page 122
- "To Create a Meta-Adapter" on page 123
- "To View Adapters" on page 123

▼ To View Meta-Adapters

1 **Log in to the Portal Server management console.**

2 **Select the SSO Adapter tab.**

The list of meta-adapters is shown in the table.

More Information

Equivalent `psadmin` Command

```
psadmin list-ssoadapters
```

▼ To Create a Meta-Adapter

1. Log in to the Portal Server management console.
2. Select the SSO Adapter tab.
3. From List of Meta-Adapters click New Meta—Adapter to launch the wizard.
4. Follow the instructions and then click OK to create the specified Meta-Adapter.

More Information

Equivalent `psadmin` Command

```
psadmin create-ssoadapter-template
```

▼ To View Adapters

1. Log in to the Portal Server management console.
2. Select the SSO Adapter tab.
   - To view adapter for a DN, click View Adapter for Locations.
     a. From the Select DN drop-down menu, choose any DN.
        The adapters for selected DN are listed.
   - To view adapters for a meta—adapter, select a meta-adapter under List of Meta-Adapters.
     a. Click View Adapters for Selected Meta-adapter.

More Information

Equivalent `psadmin` Command

```
psadmin list-ssoadapters
```

Note – The only list of adapters allowed by the CLI is by DN.

Managing Adapters

An adapter builds upon a meta-adapter by providing data values that are specific to an organization, role, or user. An adapter references a meta-adapter, and takes data values from the meta-adapter for those properties that are not editable by the user. When an end user changes the user-editable properties of an adapter, that adapter would then apply only to that one user.

You can perform the following tasks using SSO Adapter configurations:

▼ To Create an Adapter

1. **Log in to the Portal Server management console.**
2. **Select the SSO Adapter tab.**
3. **Select a meta-adapter under List of Meta-adapters.**
4. **Click View Adapters for Selected Meta-adapter.**
5. **Click New Adapter.**
   The New adapter page appears.
6. **Provide the configuration attributes as necessary.**
7. **Click OK.**

More Information

Equivalent `psadmin` Command

```
create-ssoadapter-config
```

▼ To Edit an Adapter Configuration Property

1. **Log in to the Portal Server management console.**
2. **Select the SSO Adapter tab.**
3. **Click View Adapters for Locations.**

4. **From the Select DN drop-down menu, choose any DN.**
   The list of Adapters appears.
5. **Select an adapter and modify the configuration attributes as necessary.**
6. **Click OK.**

More Information Equivalent `psadmin` Command

```
psadmin set-ssoadapter-property
```

Creating Anonymous Users

Without logging in, end users have access to any read-only communication channels that administrators have configured. However, end users are usually prevented from editing these channels.

▼ To Create a List of Anonymous Users

1. **Log in to the Portal Server management console.**
2. **Select the SSO Adapter tab.**
3. **From SSO Adapter Tasks, click Edit list of users allowed to access SSO Adapters without authentication.**
4. **From User locations, click Add Users.**
5. **From Users Found table, choose users.**
6. **Click Add Selected Users.**

Note – The Anonymous Users function is available only through Portal Server management console.

PART II

Managing the Search Server

◆ ◆ ◆ CHAPTER 11

Managing the Search Server

This chapter describes how to configure and administer the Sun Java™ System Portal Server Search Server.

This chapter contains these sections:

Understanding the Search Server

The Portal Server Search Server is a taxonomy and database service designed to support search and browse interfaces similar to popular Internet search servers such as Google and Alta Vista. The Search Server includes a robot to discover, convert, and summarize document resources. The Portal Server Desktop includes a search user interface based on JavaServer Pages™ (JSP™). The Search Server includes administration tools for configuration editing and command-line tools for system management. Configuration settings can be defined and stored through the Portal Server management console.

Note – The management console permits an administrator to configure a majority of the search server options, but it does not perform all the administrative functions available through the command-line interface.

Search Database

User query the search server's databases to locate resources. Individual entries in each database are called resource descriptions (RDs). A resource description provides summary information about a single resource. The database schema determines the fields of each resource description.

The search server is based on open Internet standards such as Resource Description Messages (RDM) and the Summary Object Interchange Format (SOIF) to ensure that the search server can operate in a cross-platform enterprise environment.

Database Taxonomy Categories

Users interact with the search system in two ways. They can type direct queries to search the database, or they can browse through the database contents using a set of categories that you design. A hierarchy of categories is sometimes called a *taxonomy*. Categorizing resources is like creating a table of contents for the database.

Browsing is an optional feature in a search system. That is, you can have a perfectly useful Search system that does not include browsing by categories. You need to decide whether adding categories that users can browse is useful to the users of your index, and, if so, what kind of categories you want to create.

The resources in a Search database are assigned to categories to reduce complexity. If a large number of items are in the database, grouping related items together is helpful. Doing so allows users to quickly locate specific kinds of items, compare similar items, and choose which ones they want.

Such categorizing is common in product and service indexes. Clothing catalogs divide men's, women's, and children's clothing, with each of those further subdivided for coats, shirts, shoes, and other items. An office products catalog could separate furniture from stationery, computers, and software. And advertising directories are arranged by categories of products and services.

The principles of categorical groupings in a printed index also apply to online indexes. The idea is to make it easy for users to locate resources of a certain type, so that they can choose the ones they want. No matter what the scope of the index you design, the primary concern in setting up your categories should be usability. You need to know how users use the categories. For example, if you design an index for a company with three offices in different locations, you might make your top-level categories correspond to each of the three offices. If users are more interested in, say, functional divisions that cut across the geographical boundaries, it might make more sense to categorize resources by corporate divisions.

Once the categories are defined, you must set up rules to assign resources to categories. These rules are called *classification rules*. If you do not define your classification rules properly, users cannot locate resources by browsing in categories. You need to avoid categorizing resources incorrectly, but you also should avoid failing to categorize documents.

Managing Search Servers

Sun Java System Portal Server can support one or more search servers.

▼ To Create a Search Server

During Portal Server installation, a default search server (*search1*) is created. You can also create a new search server using the Create Search Server wizard.

Before You Begin You will need to know configuration information specific to the web container instance that you use:

- Sun Java System Web Server 7
- Sun Java System Web Server 6
- Sun Java System Application Server 8.1
- BEA Weblogic 8
- IBM WebSphere 5

1. **Log in to the Portal Server management console.**
2. **Select Search Servers and then New from the menu bar.**

   The New Search Server wizard appears.
3. **Follow the instructions and then click Finish to create the specified search server.**

More Information For equivalent `psadmin` Command

```
psadmin create-search-server.
```

▼ To Delete a Search Server

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar.**
3. **Select a search server and click Delete.**

More Information For equivalent `psadmin` Command

```
psadmin delete-search-server
```

Overview of the Database

The search server stores its descriptions of resources in a database. A search database is a document collection index. They are created by the indexer (command `rdmgr`, or search server itself). For example, by default the robot can be setup to crawl web sites and the robot indexes whatever it finds into the default" search database where users can search for the data. The data or index into other databases too.

The following are some configuration and maintenance tasks you may need to perform to administer the database:

Importing to a Database

Normally, items in your search database come from the robot. You can also import databases of existing items, either from other Portal Server Search servers, from iPlanet Web Servers or Netscape™ Enterprise Servers, or from databases generated from other sources. Importing existing databases of RDs instead of sending the robot to create them anew helps reduce the amount of network traffic. Doing so also enables large indexing efforts to be completed more quickly by breaking the effort down into smaller parts. If the central database is physically distant from the servers being indexed, it can be helpful to generate the RDs locally and periodically import the remote databases to the central database.

The search server uses import agents to import RDs from another server or from a database. An *import agent* is a process that retrieves a number of RDs from an external source and merges that information into a local database.

Before you can import a database, you must create an import agent. Once an agent is created, you can start the import process immediately or schedule a time to run the import process on a regular basis.

Editing the Database Schema

A *schema* determines what information your search server maintains on each resource, and in what form. The design of your schema determines two factors that affect the usability of your index:

- The way users can search for resources
- The ways users view resource information

The schema is a master data structure for Resource Descriptions in the database. Depending on how you define and index the fields in that data structure, users have varying degrees of access to the resources.

The schema is closely tied to the structure of the files used by the search server and its robot. You should change only the data structure by using the schema tools in management console. Never edit the schema file directly.

You can edit the database schema of the search server to add a new schema attribute, to modify a schema attribute, or to delete attributes.

The schema includes the following attributes:

- Editable – If checked, this attribute indicates that the attribute appears in the Resource Description Editor, and you can change its values.
- Indexable – This attribute indicates that users can search for values in this particular field. An indexable fields may also appear in the pop-up menu in the Advanced Search screen.
- Description – This attribute is a text string to use to describe the schema. You can use it for comments or annotations.
- Aliases – This attribute allows you to define aliases to convert imported database schema names into your own schema.
- Score Multiplier – A weighting field for scoring a particular element. Any positive value is valid.
- Data Type – Defines the data type.

Defining Schema Aliases

You might encounter discrepancies between the names used for fields in database schemas. When you import Resource Descriptions from one server to another, you cannot always guarantee that the two servers use identical names for items in their schemas. Similarly, when the robot converts HTML `<meta>` tags from a document into schema fields, the document controls the names.

The search server allows you to define schema aliases for your schema attributes, to map these external schema names into valid names for fields in your database.

Viewing Database Analysis

The search server provides a report with information about the number of sites indexed and the number of resources from each in the database.

Re-indexing the Database

You might need to re-index the Resource Description database for the search server if you have edited the schema to add or remove an indexed field or if a disk error corrupts the index file. It may also be necessary to re-index if a discrepancy occurs between the database content and its index for any other reason. For example, a system failure while indexing.

Re-indexing a large database can take several hours. The time required to re-index the database corresponds to the number of records in the database. If you have a large database, perform re-indexing at a time when the server is not in high demand.

Expiring the Database

Removing Resource Descriptions that are out of date is *expiring* the database. Resource Descriptions are removed *only* when you run the expiration. Expired Resource Descriptions are deleted, but the database size is not decreased.

One attribute of a Resource Description is its expiration date. Your robots can set the expiration date from HTML `<meta>` tags or from information provided by the resource's server. By default, Resource Descriptions expire in three months from creation unless the resource specifies a different expiration date. Periodically your search server should purge expired Resource Descriptions from its database.

Purging the Database

Purging allows you to remove the contents of the database. Disk space used for indexes is recovered, but disk space used by the main database is not recovered. Instead it is reused as new data are added to the database.

Partitioning the Database

The search server allows you to put the physical files that make up each search database on multiple disks, file systems, directories, or partitions. By spreading databases across different physical or logical devices, you can create a larger database than would fit on a single device.

By default, the search server sets up the database to use only one directory. The command-line interface allows you to perform two kinds of manipulations on the database partitions:

- Adding New Partitions
- Moving Partitions

The search server does not perform any checking to ensure that individual partitions have space remaining. It is your responsibility to maintain adequate free space for the database.

You can add new database partitions up to a maximum of 15 total partitions.

Note – Once you increase the number of partitions, you must delete the entire database if you want to reduce the number later.

However, partitions are not recommended as long as you have enough disk space.

To change the physical location of any database partition, specify the name of the new location. Similarly, you can rename an existing partition. Use the `rdmgr` command to manipulate the partitions. See the *Sun Java System Portal Server 7.1 Command Line Reference* for information on the `psadmin` command.

Managing Databases

Use the following instruction to manage a database:

▼ To Create a Database

1 **Log in to the Portal Server management console.**

2 **Select Search Servers tab, then select a search server.**

3 **Click Databases, then Management from the menu bar.**

4 **Click New.**

The New Database page displays.

5 **Type the name of the new database, and click OK.**

More Information

For equivalent `psadmin` Command

```
psadmin create-search-database
```

▼ To Create an Import Agent

1. **Log in to the Portal Server management console.**
2. **Select Search Servers tab, then select a search server.**
3. **Click Databases, then Import Agents from the menu bar.**
4. **Click New to launch the wizard.**
5. **Specify the Import Agent attributes.**
   For more information about the attributes, see Import Agents in *Sun Java System Portal Server 7.1 Technical Reference*
6. **Click Finish.**

More Information For equivalent `psadmin` Command

```
psadmin create-search-importagent
```

▼ To Create a Resource Description

1. **Log in to the Portal Server management console.**
2. **Select the Search Servers tab, then select a search server.**
3. **Click Databases, then Management from the menu bar.**
4. **Select a database and click Manage Resource Descriptions.**
5. **Click New and specify the attributes.**
   For more information about the attributes, see Schema in *Sun Java System Portal Server 7.1 Technical Reference*
6. **Click OK.**

▼ To Manage Resource Descriptions

1. **Log in to the Portal Server management console.**
2. **Select Search Servers tab, then select a search server.**
3. **Click Databases, then Management from the menu bar.**
4. **Select a database and click Manage Resource Descriptions.**
5. **Select a Resource Description to perform one of the following actions:**
   - Edit
   - Edit All
   - Delete

   For more information about the attributes, see Schema in *Sun Java System Portal Server 7.1 Technical Reference*
6. **Click Save.**

More Information For equivalent `psadmin` Command

```
psadmin modify-search-resourcedescription
```

Managing Reports

The search server provides a number of reports to allow you to monitor search activity.

▼ To View Reports

1. **Log in to the Portal Server management console.**
2. **Select the Search Servers tab , then select a search server.**
3. **Click Reports from the menu bar.**
4. **Click on a link in the menu bar to view a specific report.**

   The following options are available:
   - Logs
   - Advanced Robot Reports

- Popular Searches
- Excluded URLs

Managing Categories

The following tasks can be used to manage categories:

▼ To Create a Category

1 **Log in to the Portal Server management console.**

2 **Select Search Servers from the tab, then select a search server.**

3 **Select Categories, then Browse/Search from the menu bar.**

4 **Click New.**

The New Search Category dialog appears.

5 **Specify the attributes as necessary.**

For more information about the attributes, see Manage Categories in *Sun Java System Portal Server 7.1 Technical Reference*

6 **Click OK.**

▼ To Edit a Category

1 **Log in to the Portal Server management console.**

2 **Select the Search Servers tab, then select a search server.**

3 **Click Categories, then Browse/Search from the menu bar.**

4 **Select a category and click Edit to display the Edit** *Category* **page.**

For more information about the attributes, see Manage Categories in *Sun Java System Portal Server 7.1 Technical Reference*

To Run Autoclassify

1. Log in to the Portal Server management console.
2. Select the Search Servers tab, then select a search server.
3. Click Categories, then Autoclassify from the menu bar.
4. Click Run Autoclassify.

To Edit Autoclassify Attributes

1. Log in to the Portal Server management console.
2. Click the Search Servers tab, then select a search server.
3. Click Categories, then Autoclassify from the menu bar.
4. Modify the attributes as necessary.

   For more information about the attributes, see *Sun Java System Portal Server 7.1 Technical Reference*
5. Click Save.

◆ ◆ ◆ CHAPTER 12

Managing the Search Server Robot

This chapter describes the Sun Java™ System Portal Server Search Server robot and its corresponding configuration files. The chapter contains following topics:

Understanding the Search Server Robot

A Search Server robot is an agent that identifies and reports on resources in its domains. It does so by using two kinds of filters: an enumerator filter and a generator filter.

The *enumerator filter* locates resources by using network protocols. The filter tests each resource and if the resource meets the proper criteria, it is enumerated. For example, the enumerator filter can extract hypertext links from an HTML file and use the links to find additional resources.

The *generator filter* tests each resource to determine whether a resource description (RD) should be created. If the resource passes the test, the generator creates an RD that is stored in the Search Server database.

Configuration and maintenance tasks you might need to do to administer the robot are described in the following sections:

How the Robot Works

Figure 12–1 shows how the robot examines URLs and their associated network resources. Both the enumerator and the generator test each resource. If the resource passes the enumeration test, the robot checks it for additional URLs. If the resource passes the generator test, the robot generates a resource description that is stored in the Search Server database.

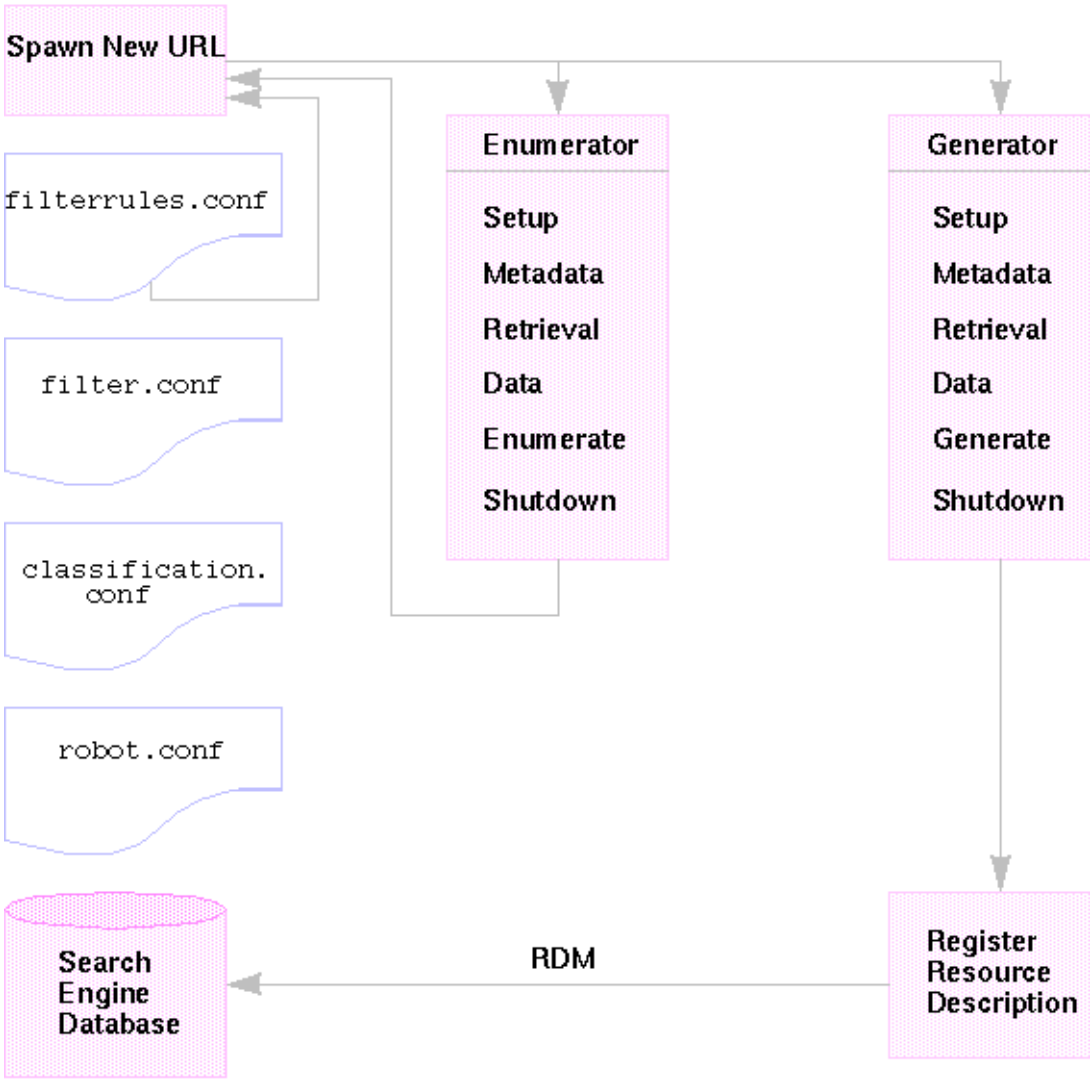

FIGURE 12–1 How the Robot Works

Robot Configuration Files

Robot configuration files define the behavior of the robots. These files reside in the directory `/var/opt/SUNWportal/searchservers/searchserverid/config`. The following list provides a description for each of the robot configuration files.

`classification.conf`	Contains rules used to classify RDs generated by the robot.
`filter.conf`	Defines the enumeration and generation filters used by the robot.
`filterrules.conf`	Contains the robot's site definitions, starting point URLs, rules for filtering based on mime type, and URL patterns.
`robot.conf`	Defines most operating properties for the robot.

Because you can set most properties by using the Search Server Administration interface, you typically do not need to edit the `robot.conf` file. However, advanced users might manually edit this file to set properties that cannot be set through the interface.

Defining Sites

The robot finds resources and determines whether to add descriptions of those resources to the database. The determination of which servers to visit and what parts of those servers to index is called a *site definition*.

Defining the sites for the robot is one of the most important jobs of the server administrator. You need to be sure you send the robot to all the servers it needs to index, but you also need to exclude extraneous sites that can fill the database and make finding the correct information more difficult.

Controlling Robot Crawling

The robot extracts and follows links to the various sites selected for indexing. As the system administrator, you can control these processes through a number of settings, including:

- Starting, stopping, and scheduling the robot
- Defining the sites the robot visits
- Crawling attributes that determine how aggressively it crawls
- The types of resources the robot indexes by defining filters
- What kind of entries the robot creates in the database by defining the indexing attributes

See the *Sun Java System Portal Server 7.1 Technical Reference* for descriptions of the robot crawling attributes.

Filtering Robot Data

Filters enable identify a resource so that it can be excluded or included by comparing an attribute of a resource against a filter definition. The robot provides a number of predefined filters, some of which are enabled by default. The following filters are predefined. Filters marked with an asterisk are enabled by default.

- Archive Files*
- Audio Files*
- Backup Files*
- Binary Files*
- CGI Files*
- Image Files*
- Java, JavaScript, Style Sheet Files*
- Log Files*
- Lotus Domino Documents
- Lotus Domino OpenViews
- Plug-in Files
- Power Point Files
- Revision Control Files*
- Source Code Files*
- Spreadsheet Files
- System Directories (UNIX)
- System Directories (NT)
- Temporary Files*
- Video Files*

You can create new filter definitions, modify a filter definition, or enable or disable filters. See "Resource Filtering Process" on page 149 for detailed information.

Using the Robot Utilities

The robot includes two debugging tools or utilities:

- Site Probe – Checks for DNS aliases, server redirects, virtual servers, and the like.
- Simulator – Performs a partial simulation of robot filtering on a URL. The simulator indicates whether sites you listed would be accepted by the robot.

Scheduling the Robot

To keep the search data timely, the robot should search and index sites regularly. Because robot crawling and indexing can consume processing resources and network bandwidth, you should

schedule the robot to run during non-peak days and times. The management console allows administrators to set up a schedule to run the robot.

Managing the Robot

This section describes the following tasks to manage the robot:

▼ To Start the Robot

1 **Log in to the Portal Server management console.**

2 **Choose Search Servers from the menu bar. Select a search server from the list of servers.**

3 **Click Robot from the menu bar, then Status and Control from the menu.**

4 **Click Start.**

More Information For equivalent `psadmin` command

```
psadmin start-robot
```

Note – For the command `psadmin start-robot`, the search robot does not start if no defined sites are available for the robot to crawl. The command `psadmin start-robot` indicates that no sites are available by displaying `Starting Points: 0 defined`.

▼ To Clear Robot Database

1 **Log in to the Portal Server management console.**

2 **Select Search Servers from the menu bar, then select a search server.**

3. **Select Robot from the menu bar then Status and Control.**
4. **Click Clear Robot Database.**

To Create a Site Definition

The robot finds resources and determines whether to add descriptions of those resources to the database. The determination of which servers to visit and what parts of those servers to index is called a *site definition*.

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Select Robot from the menu bar, then Sites.**
4. **Click New under Manage Sites and specify the configuration attributes for the site.**

   For more information about the attributes, see Sites in *Sun Java System Portal Server 7.1 Technical Reference*.
5. **Click OK.**

To Edit a Site Definition

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Click Robot from the menu bar, then Sites.**
4. **Click the name of the site you want to modify.**

   The Edit Site dialog appears.
5. **Modify the configuration attributes as necessary.**

   For more information about the attributes, see Sites in *Sun Java System Portal Server 7.1 Technical Reference*
6. **Click OK to record the changes.**

To Control Robot Crawling and Indexing

The robot crawls to the various sites selected for indexing. You control how the robot crawls sites by defining crawling and indexing operational properties.

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Click Robot from the menu bar, then Properties.**
4. **Specify the robot crawling and indexing attributes as necessary.**
   For more information about the attributes, see "Site Probe" in *Sun Java System Portal Server 7.1 Technical Reference* in *Sun Java System Portal Server 7.1 Technical Reference*.
5. **Click Save.**

To Run the Simulator

The simulator performs a partial simulation of robot filtering on one or more listed site sites.

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Click Robot from the menu bar, then Utilities.**
4. **Type the URL of a new site to simulate in the Add a new URL text box and click Add.**
   You can also run the simulator on existing sites listed under Existing Robot sites.
5. **Click Run Simulator.**

To Run the Site Probe Utility

The site probe utility checks for such information as DNS aliases, server redirects, and virtual servers.

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**

3. **Click Robot from the menu bar, then Utilities.**
4. **Type the URL of the site to probe.**
5. **(Optional) If you want the probe to return DNS information choose Show Advanced DNS information under Site Probe.**
6. **Click Run SiteProbe.**

Resource Filtering Process

The robot uses filters to determine which resources to process and how to process them. When the robot discovers references to resources as well as the resources themselves, it applies filters to each resource. The filters enumerate the resourceand determine whether to generate a resource description to store in the Search Server database.

The robot examines one or more starting point URLs, applies the filters, and then applies the filters to the URLs spawned by enumerating those URLs, and so on. The starting point URLs are defined in the `filterrules.conf` file.

Each enumeration and generation filter performs any required initialization operations and applies comparison tests to the current resource. The goal of each test is to allow or deny the resource. Each filter also has a shutdown phase during which it performs clean-up operations.

If a resource is allowed, then it continues its passage through the filter. The robot eventually enumerates it, attempting to discover further resources. The generator might also create a resource description for it.

If a resource is denied, the resource is rejected. No further action is taken by the filter for resources that are denied.

These operations are not necessarily linked. Some resources result in enumeration; others result in RD generation. Many resources result in both enumeration and RD generation. For example, if the resource is an FTP directory, the resource typically does not have an RD generated for it. However, the robot might enumerate the individual files in the FTP directory. An HTML document that contains links to other documents can result in an RD being generated, and can lead to enumeration of any linked documents as well.

The following sections describe the filter process:

- "Stages in the Filter Process" on page 150
- "Filter Syntax" on page 151
- "Filter Directives" on page 151
- "Writing or Modifying a Filter" on page 152

Stages in the Filter Process

Both enumeration and generation filters have five phases in the filtering process.

- **Setup** – Performs initialization operations. Occurs only once in the life of the robot.
- **Metadata** – Filters the resource based on metadata available about the resource. Metadata filtering occurs once per resource before the resource is retrieved over the network. Table 12–1 lists examples of common metadata types.

TABLE 12–1 Common Metadata Types

Metadata Type	Description	Example
Complete URL	The location of a resource	`http://home.siroe.com/`
Protocol	The access portion of the URL	`http`, `ftp`, `file`
Host	The address portion of the URL	`www.siroe.com`
IP address	Numeric version of the host	198.95.249.6
PATH	The path portion of the URL	`/index.html`
Depth	Number of links from the starting point URL	5

- **Data** – Filters the resource based on its data. Data is filtered once per resource after the data is retrieved over the network. Data that can be used for filtering include:
  - content-type
  - content-length
  - content-encoding
  - content-charset
  - last-modified
  - expires
- **Enumerate** – Enumerates the current resource in order to determine whether it points to other resources to be examined.
- **Generate** – Generates a resource description (RD) for the resource and saves it in the Search Server database.
- **Shutdown** – Performs any needed termination operations. This process occurs once in the life of the robot.

Filter Syntax

The `filter.conf` file contains definitions for enumeration and generation filters. This file can contain multiple filters for both enumeration and generation. The filters used by the robot are specified by the `enumeration-filter` and `generation-filter` properties in the file `robot.conf`.

Filter definitions have a well-defined structure: a header, a body, and an end. The header identifies the beginning of the filter and declares its name, for example:

```
<Filter name="myFilter">
```

The body consists of a series of filter directives that define the filter's behavior during setup, testing, enumeration or generation, and shutdown. Each directive specifies a function and, if applicable, properties for the function.

The end is marked by `</Filter>`.

Example 12–1 shows a filter named `enumeration1`.

EXAMPLE 12–1 Enumeration File Syntax

```
<Filter name="enumeration1>
   Setup fn=filterrules-setup config=./config/filterrules.conf
#  Process the rules
   MetaData fn=filterrules-process
#  Filter by type and process rules again
   Data fn=assign-source dst=type src=content-type
   Data fn=filterrules-process
#  Perform the enumeration on HTML only
   Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
#  Cleanup
   Shutdown fn=filterrules-shutdown
</Filter>
```

Filter Directives

Filter directives use robot application functions (RAFs) to perform operations. Their use and flow of execution is similar to that of NSAPI directives and server application functions (SAFs) in the Sun Java System Web Server's `obj.conf` file. Like NSAPI and SAF, data are stored and transferred using property blocks, also called *pblocks*.

Six robot directives, or RAF classes, correspond to the filtering phases and operations listed in "Resource Filtering Process" on page 149:

- Setup

- Metadata
- Data
- Enumerate
- Generate
- Shutdown

Each directive has its own robot application functions. For example, use filtering functions with the Metadata and Data directives, enumeration functions with the Enumerate directive, generation functions with the Generate directive, and so on.

The built-in robot application functions, as well as instructions for writing your own robot application functions, are explained in the *Sun Java System Portal Server 7.1 Developer's Guide*.

Writing or Modifying a Filter

In most cases, you can use the management console to create most of your site-definition based filters. You can then modify the `filter.conf` and `filterrules.conf` files to make any further desired changes. These files reside in the directory `/var/opt/SUNWportal/searchservers/searchserverid/config`.

To create a more complex set of properties, edit the configuration files used by the robot.

When you write or modify a filter, note the order of

- The execution of directives, especially the available information at each phase.
- The filter rules in `filterrules.conf`.

You can also do the following:

- Modify properties in `robot.conf` file.
- Modify robot application functions in `filter.conf` file.
- Create your own robot application functions.

For more information, see the *Sun Java System Portal Server 7.1 Developer's Guide*

Managing Filters

The following tasks to manage robot filters are described in this section:

- "To Create a Filter" on page 153
- "To Delete a Filter" on page 153
- "To Edit a Filter" on page 153
- "To Enable or Disable a Filter" on page 154

To Create a Filter

1. Log in to the Portal Server management console.
2. Select Search Servers from the menu bar, then select a search server.
3. Select Robot from the menu bar, then Filters.
4. Click New.
   The New Robot Filter wizard appears.
5. Follow the instructions to create the specified filter.
   a. Type a filter name and filter description in the text box, and click Next.
   b. Specify filter definition and behavior, and click Finish.
      For more information about filter attributes, see Filters in *Sun Java System Portal Server 7.1 Technical Reference*.
   c. Click Close to load the new filter.

To Delete a Filter

1. Log in to the Portal Server management console.
2. Select Search Servers from the menu bar, then select a search server.
3. Select Robot from the menu bar, then Filters.
4. Select a filter.
5. Click Delete.
6. Click OK in the confirmation dialog box that appears.

To Edit a Filter

1. Log in to the Portal Server management console.
2. Select Search Servers from the menu bar, then select a search server.

3. **Select Robot from the menu bar, then Filters.**
4. **Select a filter, and click Edit.**
   The Edit a Filter page appears.
5. **Modify the configuration attributes as necessary.**
   For more information about filter attributes, see Filters in *Sun Java System Portal Server 7.1 Technical Reference*.
6. **Click OK.**

▼ To Enable or Disable a Filter

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Select Robot from the menu bar, then Filters.**
4. **Select a filter.**
   - **To enable a filter, click Enable.**
   - **To disable a filter, click Disable.**

Managing Classification Rules

Documents can be assigned to multiple categories, up to a maximum number defined in the settings. Classification rules are simpler than robot filter rules because they do not involve any flow-control decisions. In classification rules you determine what criteria to use to assign specific categories to a resource as part of its Resource Description. A classification rule is a simple conditional statement, taking the form `if` *condition* `is true, assign the resource to` <*a category*>.

▼ To Create a Classification Rule

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**

3. **Select Robot from the menu bar, then Classification Rules.**
4. **Select Classification Rules and click New.**

   The New Classification Rule dialog box appears.
5. **Specify the configuration attributes as necessary.**

   For more information about the attributes, see Manage Classification Rules in *Sun Java System Portal Server 7.1 Technical Reference*.
6. **Click OK.**

▼ To Edit a Classification Rule

1. **Log in to the Portal Server management console.**
2. **Select Search Servers from the menu bar, then select a search server.**
3. **Select Robot, then Classification Rules from the menu bar.**
4. **Select a classification rule, and click Edit.**
5. **Modify the attributes as necessary.**

   For more information about the attributes, see Manage Classification Rules in *Sun Java System Portal Server 7.1 Technical Reference*.
6. **Click OK.**

Sources and Destinations

Most robot application functions (RAFs) require sources of information and generate data that go to destinations. The sources are defined within the robot and are not necessarily related to the fields in the resource description that the robot ultimately generates. Destinations, on the other hand, are generally the names of fields in the resource description, as defined by the resource description server's schema.

The following sections describe the different stages of the filtering process, and the sources available at those stages:

- "Enable Property" on page 157

Sources Available at the Setup Stage

At the Setup stage, the filter is set up but cannot yet obtain information about the resource's URL or content.

Sources Available at the MetaData Filtering Stage

At the MetaData stage, the robot encounters a URL for a resource but it has not downloaded the resource's content. Thus information is available about the URL as well as data that is derived from other sources such as the `filter.conf` file. At this stage, however, information about the content of the resource is not available.

TABLE 12–2 Sources Available to the RAFs at the MetaData Phase

Source	Description	Example
csid	Catalog server ID	`x-catalog//budgie.siroe.com:8086/alexandria`
depth	Number of links traversed from starting point	`10`
enumeration filter	Name of enumeration filter	`enumeration1`
generation filter	Name of generation filter	`generation1`
host	Host portion of URL	`home.siroe.com`
IP	Numeric version of host	`198.95.249.6`
protocol	Access portion of the URL	`http, https, ftp, file`
path	Path portion of the URL	`/, /index.html, /documents/listing.html`
URL	Complete URL	`http://developer.siroe.com/docs/manuals/`

Sources Available at the Data Stage

At the Data stage, the robot has downloaded the content of the resource at the URL and can access data about the content, such as the description and the author.

If the resource is an HTML file, the Robot parses the <META> tags in the HTML headers. Consequently, any data contained in <META> tags is available at the Data stage.

During the Data phase, the following sources are available to RAFs, in addition to those available during the MetaData phase.

TABLE 12–3 Sources Available to the RAFs at the Data Phase

Source	Description	Example
content-charset	Character set used by the resource	
content-encoding	Any form of encoding	
content-length	Size of the resource in bytes	
content-type	MIME type of the resource	text/html, image/jpeg
expires	Date the resource expires	
last-modified	Date the resource was last modified	
data in <META> tags	Any data that is provided in <META> tags in the header of HTML resources	Author, Description, Keywords

All of these sources except for the data in <META> tags are derived from the HTTP response header returned when retrieving the resource.

Sources Available at the Enumeration, Generation, and Shutdown Stages

At the Enumeration and Generation stages, the same data sources are available as in the Data stage. See Table 12–3 for information.

At the Shutdown stage, the filter completes its filtering and shuts down. Although functions written for this stage can use the same data sources as those available at the Data stage, the shutdown functions typically restrict their operations to robot shutdown and clean-up activities.

Enable Property

Each function can have an enable property. The values can be true, false, on, or off. The management console uses these parameters to turn certain directives on or off.

The following example enables enumeration for text/html and disables enumeration for text/plain:

```
#  Perform the enumeration on HTML only
Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
Enumerate enable=false fn=enumerate-urls-from-text max=1024 type=text/plain
```

Adding an `enable=false` property or an `enable=off` property has the same effect as commenting the line. These properties are used because the management console does not write comments.

Setup Functions

This section describes the functions that are used during the setup phase by both enumeration and generation filters. The functions are described in the following sections:

filterrules-setup

When you use the `filterrules-setup` function, use the `logtype` log file. The value can be `verbose`, `normal`, or `terse`.

Property

`config` Path name to the file containing the filter rules to be used by this filter.

Example

```
Setup fn=filterrules-setup

config="/var/opt/SUNWportal/searchservers/search1/config/filterrules.conf"
```

setup-regex-cache

The `setup-regex-cache` function initializes the cache size for the `filter-by-regex` and `generate-by-regex` functions. Use this function to specify a number other than the default of 32.

Property

`cache-size` Maximum number of compiled regular expressions to be kept in the regex cache.

Example

```
Setup fn=setup-regex-cache cache-size=28
```

setup-type-by-extension

The `setup-type-by-extension` function configures the filter to recognize file name extensions. It must be called before the `assign-type-by-extension` function can be used. The file specified as a property must contain mappings between standard MIME content types and file extension strings.

Property

`file` Name of the MIME types configuration file

Example

```
Setup fn=setup-type-by-extension

file="/var/opt/SUNWportal/searchservers/search1/config/mime.types"
```

Filtering Functions

Filtering functions operate at the Metadata and Data stages to allow or deny resources based on specific criteria specified by the function and its properties. These functions can be used in both Enumeration and Generation filters in the file `filter.conf`.

Each `filter-by` function performs a comparison and either allows or denies the resource. Allowing the resource means that processing continues to the next filtering step. Denying the resource means that processing should stop, because the resource does not meet the criteria for further enumeration or generation.

filter-by-exact

The `filter-by-exact` function allows or denies the resource if the `allow/deny` string matches the source of information exactly. The keyword `all` matches any string.

Properties

`src` Source of information

`allow/deny` Contains a string

Example

The following example filters out all resources whose content-type is `text/plain`. It allows all other resources to proceed:

```
Data fn=filter-by-exact src=type deny=text/plain
```

filter-by-max

The `filter-by-max` function allows the resource if the specified information source is less than or equal to the given value. It denies the resource if the information source is greater than the specified value.

This function can be called no more than once per filter.

Properties

The `filter-by-max`function lists the properties used with the `filter-by-max` function.

`src` Source of information: hosts, objects, or depth

`value` Specifies a value for comparison

Example

This example allows resources whose content-length is less than 1024 kilobytes:

```
MetaData fn-filter-by-max src=content-length value=1024
```

filter-by-md5

The `filter-by-md5` function allows only the first resource with a given MD5 checksum value. If the current resource's MD5 has been seen in an earlier resource by this robot, the current resource is denied. The function prevents duplication of identical resources or single resources with multiple URLs.

You can only call this function at the Data stage or later. It can be called no more than once per filter. The filter must invoke the `generate-md5` function to generate an MD5 checksum before invoking `filter-by-md5`.

Properties

None

Example

The following example shows the typical method of handling MD5 checksums by first generating the checksum and then filtering based on it:

```
Data fn=generate-md5

Data fn=filter-by-md5
```

filter-by-prefix

The `filter-by-prefix` function allows or denies the resource if the given information source begins with the specified prefix string. The resource doesn't have to match completely. The keyword `all` matches any string.

Properties

`src`	Source of information
`allow/deny`	Contains a string for prefix comparison

Example

The following example allows resources whose content-type is any kind of text, including `text/html` and `text/plain`:

```
MetaData fn=filter-by-prefix src=type allow=text
```

filter-by-regex

The `filter-by-regex` function supports regular-expression pattern matching. It allows resources that match the given regular expression. The supported regular expression syntax is defined by the `POSIX.1` specification. The regular expression `\\\\*` matches anything.

Properties

`src`	Source of information
`allow/deny`	Contains a regular expression string

Example

The following example denies all resources from sites in the `.gov` domain:

```
MetaData fn=filter-by-regex src=host deny=\\\\*.gov
```

filterrules-process

The `filterrules-process` function processes the site definition and filter rules in the `filterrules.conf` file.

Properties

None

Example

```
MetaData fn=filterrules-process
```

Filtering Support Functions

Support functions are used during filtering to manipulate or generate information on the resource. The robot can then process the resource by calling filtering functions. These functions can be used in enumeration and generation filters in the file `filter.conf`.

assign-source

The `assign-source` function assigns a new value to a given information source. This function permits editing during the filtering process. The function can assign an explicit new value, or it can copy a value from another information source.

Properties

`dst`	Name of the source whose value is to be change
`value`	Specifies an explicit value
`src`	Information source to copy to `dst`

You must specify either a `value` property or a `src`property, but not both.

Example

```
Data fn=assign-source dst=type src=content-type
```

assign-type-by-extension

The `assign-type-by-extension` function uses the resource's file name to determine its type and assigns this type to the resource for further processing.

The `setup-type-by-extension` function must be called during setup before `assign-type-by-extension` can be used.

Property

`src` Source of file name to compare. If you do not specify a source, the default is the resource's path

Example

```
MetaData fn=assign-type-by-extension
```

clear-source

The clear-source function deletes the specified data source. You typically do not need to perform this function. You can create or replace a source by using the `assign-source` function.

Property

`src` Name of the source to delete

Example

The following example deletes the path source:

```
MetaData fn=clear-source src=path
```

convert-to-html

The `convert-to-html` function converts the current resource into an HTML file for further processing if its type matches a specified MIME type. The conversion filter automatically detects the type of the file it is converting.

Property

`type` MIME type from which to convert

Example

The following sequence of function calls causes the filter to convert all Adobe Acrobat PDF files, Microsoft RTF files, and FrameMaker MIF files to HTML, as well as any files whose type was not specified by the server that delivered it.

```
Data fn=convert-to-html type=application/pdf

Data fn=convert-to-html type=application/rtf

Data fn=convert-to-html type=application/x-mif

Data fn=convert-to-html type=unknown
```

copy-attribute

The `copy-attribute` function copies the value from one field in the resource description into another.

Properties

`src`	Field in the resource description from which to copy
`dst`	Item in the resource description into which to copy the source
`truncate`	Maximum length of the source to copy
`clean`	Boolean property indicating whether to fix truncated text, to not leave partial words. This property is `false` by default

Example

```
Generate fn=copy-attribute \\

src=partial-text dst=description truncate=200 clean=true
```

generate-by-exact

The `generate-by-exact` function generates a source with a specified value, but only if an existing source exactly matches another value.

Properties

`dst`	Name of the source to generate
`value`	Value to assign `dst`

`src` Source against which to match

Example

The following example sets the classification to `siroe` if the host is `www.siroe.com`.

```
Generate fn="generate-by-exact" match="www.siroe.com:80" src="host"
value="Siroe" dst="classification"
```

generate-by-prefix

This `generate-by-prefix` function generates a source with a specified value if the prefix of an existing source matches another value.

Properties

`dst` Name of the source to generate

`value` Value to assign `dst`

`src` Source against which to match

`match` Value to compare to `src`

Example

The following example sets the classification to Compass if the protocol prefix is HTTP:

```
Generate fn="generate-by-prefix" match="http" src="protocol" value="World Wide
Web" dst="classification"
```

generate-by-regex

The `generate-by-regex` function generates a source with a specified value if an existing source matches a regular expression.

Properties

`dst` Name of the source to generate

`value` Value to assign `dst`

`src` Source against which to match

`match` Regular expression string to compare to `src`

Example

The following example sets the classification to `siroe` if the host name matches the regular expression `*.siroe.com`. For example, resources at both `developer.siroe.com` and `home.siroe.com` are classified as `Siroe`:

```
Generate fn="generate-by-regex" match="\\\\*.siroe.com" src="host"
value="Siroe" dst="classification"
```

generate-md5

The `generate-md5` function generates an MD5 checksum and adds it to the resource. You can then use the `filter-by-md5` function to deny resources with duplicate MD5 checksums.

Properties

None

Example

```
Data fn=generate-md5
```

generate-rd-expires

The `generate-rd-expires` function generates an expiration date and adds it to the specified source. The function uses metadata such as the HTTP header and HTML `<META>` tags to obtain any expiration data from the resource. If none exists, the function generates an expiration date three months from the current date.

Properties

`dst` Name of the source. If you omit it, the source defaults to `rd-expires`.

Example

```
Generate fn=generate-rd-expires
```

generate-rd-last-modified

The `generate-rd-last-modified` function adds the current time to the specified source.

Properties

`dst` Name of the source. If you omit it, the source defaults to `rd-last-modified`

Example

```
Generate fn=generate-last-modified
```

rename-attribute

The `rename-attribute` function changes the name of a field in the resource description. The function is most useful in cases where, for example, the `extract-html-meta` function copies information from a `<META>` tag into a field and you want to change the name of the field.

Property

`src` String containing a mapping from one name to another

Example

The following example renames an attribute from author to author-name:

```
Generate fn=rename-attribute src="author->author-name"
```

Enumeration Functions

The following functions operate at the Enumerate stage. These functions control whether and how a robot gathers links from a given resource to use as starting points for further resource discovery.

enumerate-urls

The `enumerate-urls` function scans the resource and enumerates all URLs found in hypertext links. The results are used to spawn further resource discovery. You can specify a content-type to restrict the kind of URLs enumerated.

Properties

`max` The maximum number of URLs to spawn from a given resource. The default is 1024.

`type` Content-type that restricts enumeration to those URLs that have the specified content-type. `type` is an optional property. If omitted, the function enumerates all

URLs.

Example

The following example enumerates HTML URLs only, up to a maximum of 1024:

```
Enumerate fn=enumerate-urls type=text/html
```

enumerate-urls-from-text

The `enumerate-urls-from-text` function scans text resource, looking for strings matching the regular expression: `URL:.*`. The function spawns robots to enumerate the URLs from these strings and generate further resource descriptions.

Property

`max` The maximum number of URLs to spawn from a given resource. The default, if `max` is omitted, is 1024

Example

```
Enumerate fn=enumerate-urls-from-text
```

Generation Functions

Generation functions are used in the Generate stage of filtering. Generation functions can create information that goes into a resource description. In general, they either extract information from the body of the resource itself or copy information from the resource's metadata.

extract-full-text

The `extract-full-text` function extracts the complete text of the resource and adds it to the resource description.

Note – Use the `extract-full-text` function with caution. It can significantly increase the size of the resource description, thus causing database bloat and overall negative impact on network bandwidth.

Example

```
Generate fn=extract-full-text
```

Properties

`truncate`	The maximum number of characters to extract from the resource
`dst`	Name of the schema item that receives the full text

extract-html-meta

The `extract-html-meta` function extracts any `<META>` or `<TITLE>` information from an HTML file and adds it to the resource description. A content-type may be specified to restrict the kind of URLs that are generated.

Properties

`truncate`	The maximum number of bytes to extract
`type`	Optional property. If omitted, all URLs are generated

Example

```
Generate fn=extract-html-meta truncate=255 type=text/html
```

extract-html-text

The `extract-html-text` function extracts the first few characters of text from an HTML file, excluding the HTML tags, and adds the text to the resource description. This function permits the first part of a document's text to be included in the RD. A content-type may be specified to restrict the kind of URLs that are generated.

Properties

`truncate`	The maximum number of bytes to extract
`skip-headings`	Set to `true` to ignore any HTML headers that occur in the document
`type`	Optional property. If omitted, all URLs are generated

Example

```
Generate fn=extract-html-text truncate=255 type=text/html skip-headings=true
```

extract-html-toc

The `extract-html-toc` function extracts table of contents from the HTML headers and adds it to the resource description.

Properties

`truncate` The maximum number of bytes to extract

`level` Maximum HTML header level to extract. This property controls the depth of the table of contents

Example

```
Generate fn=extract-html-toc truncate=255 level=3
```

extract-source

The `extract-source` function extracts the specified values from the given sources and adds them to the resource description.

Property

`src` Lists source names. You can use the `->` operator to define a new name for the RD attribute. For example, `type->content-type` would take the value of the source named `type` and save it in the RD under the attribute named `content-type`.

Example

```
Generate fn=extract-source src="md5,depth,rd-expires,rd-last-modified"
```

harvest-summarizer

The `harvest-summarizer` function runs a Harvest summarizer on the resource and adds the result to the resource description.

To run Harvest summarizers, you must have `$HARVEST_HOME/lib/gatherer` in your `path` before you run the robot.

Property

`summarizer` Name of the summarizer program

Example

```
Generate fn-harvest-summarizer summarizer=HTML.sum
```

Shutdown Function

The `filterrules-shutdown` function can be used during the shutdown phase by both enumeration and generation functions.

`filterrules-shutdown`

After the rules are run, the `filterrules-shutdown` function performs clean up and shutdown responsibilities.

Properties

None

Example

```
Shutdown fn=filterrules-shutdown
```

Modifiable Properties

The `robot.conf` file defines many options for the robot, including pointing the robot to the appropriate filters in `filter.conf` . For backward compatibility with older versions , `robot.conf` can also contain the starting point URLs.

Because you can set most properties by using the management console, you typically do not need to edit the `robot.conf` file. However, advanced users might manually edit this file to set properties that cannot be set through the management console. See "Sample `robot.conf` File" on page 177 for an example of this file.

Table 12–4 lists the properties you can change in the `robot.conf` file.

TABLE 12–4 User-Modifiable Properties

Property	Description	Example
`auto-proxy`	Specifies the proxy setting for the robot. It can be a proxy server or a JavaScript file for automatically configuring the proxy. .	`auto-proxy="http://proxy_server/proxy.pac"`
`bindir`	Specifies whether the robot adds a `bin` directory to the `PATH` environment. This is an extra `PATH` for users to run an external program in a robot, such as those specified by `cmd-hook` property.	`bindir=path`
`cmd-hook`	Specifies an external completion script to run after the robot completes one run. This must be a full path to the command name. The robot executes this script from the `/var/opt/SUNWportal/` directory. No default is set. At least one RD must be registered for the command to run.	`cmd-hook=”command-string”`
`command-port`	Specifies the port number that the robot listens to in order to accept commands from other programs, such as the Administration Interface or robot control panels. For security reasons, the robot can accept commands only from the local host unless `remote-access` is set to `yes`.	`command-port=port_number`
`connect-timeout`	Specifies the maximum time allowed for a network to respond to a connection request. The default is `120` seconds.	`command-timeout=seconds`
`convert-timeout`	Specifies the maximum time allowed for document conversion. The default is `600` seconds.	`convert-timeout=seconds`

TABLE 12–4 User-Modifiable Properties *(Continued)*

Property	Description	Example
`depth`	Specifies the number of links from the starting point URLs that the robot examines. This property sets the default value for any starting point URLs that do not specify a depth. The default is `10`. A value of negative one (`depth=-1`) indicates that the link depth is infinite.	`depth=integer`
`email`	Specifies the email address of the person who runs the robot. The email address is sent with the user-agent in the HTTP request header so that Web managers can contact the people who run robots at their sites. The default is `user@domain`.	`email=user@hostname`
`enable-ip`	Generates an IP address for the URL for each RD that is created. The default is `true`.	`enable-ip=[true | yes | false | no]`
`enable-rdm-probe`	Determines the server supports RDM. The robot decides whether to query each server it encounters by using this property. If the server supports RDM, the robot does not attempt to enumerate the server's resources that server is able to act as its own resource description server. The default is `false`.	`enable-rdm-probe=[true | false | yes | no]`
`enable-robots-txt`	Determines the robot should check the `robots.txt` file at each site it visits, if available. The default is `yes`.	`enable-robots-txt=[true | false | yes | no]`

TABLE 12–4 User-Modifiable Properties *(Continued)*

Property	Description	Example
`engine-concurrent`	Specifies the number of pre-created threads for the robot to use. The default is `10`. You cannot use the management console to set this property interactively.	`engine-concurrent=[1..100]`
`enumeration-filter`	Specifies the enumeration filter that is used by the robot to determine a resource should be enumerated. The value must be the name of a filter defined in the file `filter.conf`. The default is `enumeration-default`. You cannot use the management console to set this property interactively.	`enumeration-filter=enumfiltername`
`generation-filter`	Specifies the generation filter that is used by the robot to determine a resource description should be generated for a resource. The value must be the name of a filter defined in the file `filter.conf`. The default is `generation-default`. You cannot use the management console to set this property interactively.	`generation-filter=genfiltername`
`index-after-ngenerated`	Specifies the number of minutes that the robot should collect RDs before batching them for the Search Server. The default value is 30 minutes.	`index-after-ngenerated=30`

TABLE 12–4 User-Modifiable Properties *(Continued)*

Property	Description	Example
`loglevel`	Specifies the levels of logging. The `loglevel` values are as follows: ■ Level 0: log nothing but serious errors ■ Level 1: also log RD generation (default) ■ Level 2: also log retrieval activity ■ Level 3: also log filtering activity ■ Level 4: also log spawning activity ■ Level 5: also log retrieval progress The default value is `1`.	`loglevel=[0...100]`
`max-connections`	Specifies the maximum number of concurrent retrievals that a robot can make. The default is `8`.	`max-connections=[1..100]`
`max-filesize-kb`	Specifies the maximum file size in kilobytes for files retrieved by the robot.	`max-filesize-kb=1024`
`max-memory-per-url / max-memory`	Specifies the maximum memory in bytes used by each URL. If the URL needs more memory, the RD is saved to disk. The default is `64k`. You cannot use the management console to set this property interactively.	`max-memory-per-url=n_bytes`
`max-working`	Specifies the size of the robot working set, which is the maximum number of URLs the robot can work on at one time. You cannot use the management console to set this property interactively.	`max-working=1024`

TABLE 12–4 User-Modifiable Properties *(Continued)*

Property	Description	Example
`onCompletion`	Determines what the robot does after it has completed a run. The robot can either go into idle mode, loop back and start again, or quit. The default is `idle`. This property works with the `cmd-hook` property. When the robot is done, it performs the action of `onCompletion` and then runs the `cmd-hook` program.	`OnCompletion=[idle | loop | quit]`
`password`	Specifies the `password` used for `httpd` authentication and `ftp` connection.	`password=string`
`referer`	Specifies the property sent in the HTTP request if it is set to identify the robot as the referrer when accessing Web pages	`referer=string`
`register-user`	Specifies the user name used to register RDs to the Search Server database. This property cannot be set interactively through the Search Server Administration Interface.	`register-user=string`
`register-password`	Specifies the password used to register RDs to the Search Server database. This property cannot be set interactively through the management console.	`register-password=string`
`remote-access`	This property determines the robot can accept commands from remote hosts. The default is `false`.	`remote-access=[true | false | yes | no]`
`robot-state-dir`	Specifies the directory where the robot saves its state. In this working directory, the robot can record the number of collected RDs and so on.	`robot-state-dir="/var/opt/SUNWportal/searchservers/<searchserverid>/config/robot"`

TABLE 12–4 User-Modifiable Properties *(Continued)*

Property	Description	Example
server-delay	Specifies the time period between two visits to the same web site, thus preventing the robot from accessing the same site too frequently. The default is 0 seconds.	server-delay=delay_in_seconds
site-max-connections	Indicates the maximum number of concurrent connections that a robot can make to any one site. The default is 2.	site-max-connections=[1..100]
smart-host-heuristics	Enables the robot to change sites that are rotating their DNS canonical host names. For example, www123.siroe.com is changed to www.siroe.com. The default is false.	smart-host-heuristics=[true | false]
tmpdir	Specifies a place for the robot to create temporary files. Use this value to set the environment variable TMPDIR.	tmpdir=path
user-agent	Specifies the property sent with the email address in the http-request to the server.	user-agent=SunONERobot/6.2
username	Specifies the user name of the user who runs the robot and is used for httpd authentication and ftp connection. The default is anonymous.	username=string

Sample robot.conf File

This section describes a sample robot.conf file. Any commented properties in the sample use the default values shown. The first property, csid, indicates the Search Server instance that uses this file. Do not to change the value of this property. See "Modifiable Properties" on page 171 for definitions of the properties in this file.

Note – This sample file includes some properties used by the Search Server that you should not modify. The csid property is one example.

```
<Process csid="x-catalog://budgie.siroe.com:80/jack" \\
   auto-proxy="http://sesta.varrius.com:80/"
   auto_serv="http://sesta.varrius.com:80/"
   command-port=21445
   convert-timeout=600
   depth="-1"
   # email="user@domain"
   enable-ip=true
   enumeration-filter="enumeration-default"
   generation-filter="generation-default"
   index-after-ngenerated=30
   loglevel=2
   max-concurrent=8
   site-max-concurrent=2
   onCompletion=idle
   password=boots
   proxy-loc=server
   proxy-type=auto
   robot-state-dir="/var/opt/SUNWportal/searchservers/search1/robot" \\
   ps/robot"
   server-delay=1
   smart-host-heuristics=true
   tmpdir="/var/opt/SUNWportal/searchservers/search1/tmp"
   user-agent="iPlanetRobot/4.0"
   username=jack
</Process>
```

PART III

Managing Delegated Administration

◆ ◆ ◆ CHAPTER 13

Managing Delegated Administration Channels

Portal Server enables portal administrators to delegate the responsibility for managing various tasks in a particular organization to other individuals, called *delegated administrators*. Decentralizing administrative functions can improve portal management, especially in complex organizations. Portal administrators can set up channels for delegated administrators to use for managing the Desktop.

To perform administration tasks, delegated administrators use a set of administrative portlets on the Portal Server Desktop. This topic shows you how to set up these channels on the Developers Sample Desktop so that you can design a basic Desktop for delegated administrators.

Understanding Portal Delegated Administration

Portal Server provides a set of administrative portlets on the Portal Server Desktop. The portlets allow administrators to set up specialized channels for delegated administrators to use in managing the Desktop and end-user roles. The three delegated administration roles are the following:

- Organization admin role — Manages the Desktop content and end users in the defined organization.
- Content administrator role — Manages the Desktop content of end users in the defined organization.
- User administrator role — Manages end users in the defined organization and can assign or remove assignments of end-user roles.

This topic shows you how to set up these channels on the Developers Sample Desktop so that you can design a basic Desktop for delegated administrators.

Setting Up Delegated Administration Channels

This topic shows you how to set up delegated administration channels at the organization, role, and user level on the Developers Sample Desktop.

▼ To Set Up a Delegated Administration Channel

1 Set up access control instructions to allow or restrict access to the Desktop channel.

- **For administrator access at the organization level, access control instructions are set up by Access Manager by default.**
- **For administrator access at the role level or the user level, Portal Server administrators must set up access control instructions.**

a. Load the sample ACIs into the Directory Server.

Type `ldapmodify -D "cn=directory manager"-w -f acis.ldif`.

Here is the sample ACI content:

```
#

acis.ldif

dn:dc=sample,dc=siroe,dc=com
changetype:modify

# aci for JDCAdmin1 role

add:aci
aci: (target= "ldap:///ou=people,o=DeveloperSample,dc=red,dc=iplanet,dc=com")
(targetattr = "*")
(version 3.0; acl "Allow JDCAdmin1 Role to read and search users";
allow (read,search)
roledn = "ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com";)
-
add:aci
aci: (target="ldap:///dc=red,dc=iplanet,dc=com")
(targetfilter="(entrydn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com)")
(targetattr="*")
(version 3.0; acl "Allow JDCAdmin1 Role to read and search JDC Role";
allow (read,search)
roledn="ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com";)
-
add:aci
aci: (target="ldap:///ou=people,o=DeveloperSample,dc=red,dc=iplanet,dc=com")
```

```
(targetattr="nsroledn")
(targetfilter="(!(|(nsroledn=cn=Top-level Admin Role,dc=red,dc=iplanet,dc=com)
(nsroledn=cn=Top-level Help Desk Admin Role,dc=red,dc=iplanet,dc=com)
(nsroledn=
cn=Organization Admin Role,o=DeveloperSample,dc=red,dc=iplanet,dc=com)
(nsroledn=cn=Top-level Policy Admin Role,dc=red,dc=iplanet,dc=com)))")
(targattrfilters="add=nsroledn:
(nsroledn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com),
del=nsroledn:(nsroledn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com)")
(version 3.0; acl "Allow JDCAdmin1 Role to add/remove users to JDC Role";
allow (write)
roledn="ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com";)
-

# aci for JDCAdmin2 role

add:aci
aci:
(target="ldap:///cn=SunPortalportal1DesktopService,dc=red,dc=iplanet,dc=com")
(targetfilter=
(cn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com))(targetattr="*")
(version 3.0; acl "Allow JDCAdmin2 to edit display profile of JDC Role";
allow (all)
roledn="ldap:///cn=JDCAdmin2,o=DeveloperSample,dc=red,dc=iplanet,dc=com";)
-
add:aci
aci: (target="ldap:///dc=red,dc=iplanet,dc=com")(targetattr = "*")
(version 3.0; acl "Allow JDCAdmin2 to read and search all";
allow (read,search)
roledn = "ldap:///cn=JDCAdmin2,o=DeveloperSample,dc=red,dc=iplanet,dc=com";)

#
```

b. Find and replace every occurrence of `o=DeveloperSample,dc=red,dc=iplanet` **with** `dc=sample,dc=`*hostname*`,dc=com`**.**

2 Define the delegated administrator's role.

a. Log in to the Sun Java™ System Access Manager management console.

For information about the Access Manager console, see the *Sun Java System Access Manager 7.1 Administration Guide*.

b. Navigate to the DeveloperSample organization.

c. Create one of the following:

- **A new suborganization**

  When you create a new organization, Access Manager sets up an Organization Admin role for the organization.

  i. Register all the required services for this new organization.

  ii. Create a new user, and assign the Organization Admin role to this user.

- **New delegated administration roles:**

  i. Create the following new roles:

  - End-User Role — Create a role `JDC`, set Type to Service, and turn off access permissions.
  - Content Administration Role — Create a role `JDCAdmin2`, set Type to Administrative, and turn off access permissions.
  - User Administration Role — Create a role `JDCAadmin1`, set Type to User, and turn off access permissions.

  ii. Create the following new users:

  - `jdcuser` — Assign to the role `JDC`.
  - `jdcuadmin` — Assign to the role `JDCAadmin1`.
  - `jdctadmin` — Assign to the role `JDCAdmin2`.

d. (Optional) Log out of the Access Manager console.

3 Ensure that the Portal Desktop service attribute values for the admin role DNs match the Portal Desktop service attribute values for your Portal.

The Desktop service attribute values for the admin role DNs are:

- content.admin.role.dn
- user.admin.role.dn

If the Portal Desktop service attribute values do not match these values, when a user who belongs to the admin role authenticates to the Portal, the user can be presented with the incorrect Desktop.

For example, if you set the DeveloperSample Portal Desktop service attribute values to:

- Parent Container: JSPTabContainer
- EditContainer: JSPEditContainer
- Default Type: developer_sample

And you set both admin role DNs to:

```
cn=Organization Admin Role, o=DeveloperSample, dc=siroe, dc=com
```

You must set the Portal Desktop service attributes for the admin role DN to:

```
cn=Organization Admin Role, o=DeveloperSample, dc=siroe, dc=com
```

4 Edit the `taskadmin.properties` file.

a. Open the `taskadmin.properties` file in the *portal-base-directory*`/samples/taskadmin` directory.

b. Identify your values for the following variables:

- *am.admin.dn* — the top-level administrator DN (for example, amadmin)
- *default.org.dn* — the top-level or default organization (for example. dc=sun,dc=com)
- *ps.portal.id* — the portal identifier (for example, portal1)
- *ps.parent.tab.container* — the portal Desktop parent container name (for example, ASCTabContainer)
- *ps.default.type* — the portal Desktop type (for example, enterprise_sample)
- *content.admin.role.dn* — DN where the content admin channels and containers are loaded
- *user.admin.role.dn* — DN where the user admin channels and containers are loaded
- *managed.content.dn* — DN managed by the content admin role

c. Change the default values to match your deployment.

```
# ------------------------------------------------------
# General settings
# ------------------------------------------------------
#
# psadmin password file (file name and directory path)
# example: /tmp/password
#
psadmin.password.file=/tmp/password //password file contains the password
 #
# Portal configuration location
# example: /etc/opt/SUNWportal
#
ps.config.location=/etc/opt/SUNWportal
#
# Portal identifier
# example: portal1
#
ps.portal.id=portal1
```

```
#
# Access Manager admin dn
# example: uid=amAdmin,ou=People,dc=siroe,dc=com
#
am.admin.dn=uid=amAdmin,ou=People,dc=siroe,dc=com
#
# Access Manager default organization
# example: dc=siroe,dc=com
#
default.org.dn=dc=siroe,dc=com
#
# ----------------------------------------------------
# Task admin general settings
# ----------------------------------------------------
#
# Parent tab container
# example: JSPTabContainer
#
ps.parent.tab.container=JSPTabContainer
#
# Parent tab container provider
# example: JSPTabContainerProvider
#
ps.parent.tab.container.provider=JSPTabContainerProvider
#
# Portal default type
# example: developer_sample
#
ps.default.type=developer_sample
#
# ----------------------------------------------------
# Content admin settings
# ----------------------------------------------------
#
# Content admin role dn. The content admin channels and containers
# are loaded to this dn.
# example: see below
#
content.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# Managed content dn. The dn managed by the 'content.admin.role.dn'.
# example: see below
#
managed.content.dn=o=DeveloperSample,dc=siroe,dc=com
```

```
#
# ----------------------------------------------------
# User admin settings
# ----------------------------------------------------
#
# User admin role dn. The user admin channels and containers
# are loaded to this dn.
# example: see below
#
user.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# ====================================================
# Examples
# ====================================================
#
# Organization admin example:
#   content.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#   managed.content.dn=o=DeveloperSample,dc=siroe,dc=com
#   user.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# Role admin example:
#   content.admin.role.dn=cn=JDCAdmin2,o=DeveloperSample,dc=siroe,dc=com
#   managed.content.dn=cn=JDC,o=DeveloperSample,dc=siroe,dc=com
#   user.admin.role.dn=cn=JDCAdmin1,o=DeveloperSample,dc=siroe,dc=com
```

d. Run the `ant` command.

`/usr/sfw/bin/ant -f` *ps-base-directory*`/samples/taskadmin/build.xml -Dprops.location=/`*tmp*

tmp is the location of `taskadmin.properties` file

5 Verify the addition.

a. Log in to the new delegated administrator's user Desktop.

b. View the new delegated administration channel.

- **For an organization delegated administrator, verify that the administration channel appears for this organization in the Admin tab of the Developer Sample.**
- **For a role or user delegated administrator, verify that the administration channel appears for this user in the Admin tab of the Developer Sample.**

c. Log out of the user Desktop.

◆ ◆ ◆ CHAPTER 14

Using the Portal Server Delegated Administration Tag Library

The Portal Server delegated administration tag library allows you to do the following:

- Modify out-of-the-box delegated administration portlets
- Develop portlets that provide new delegated administration functions
- Write administration portlets with custom user interfaces
- Create and administer channels based on `JSPProvider`

Understanding the Delegated Administration Tag Library

The *Tag Library for Delegated Administration* describes the tags for writing delegated administration portlets and provides syntax for them. The tag library supports tasks for the following administrative functions:

- Provider management
- Portlet management
- User management
- WSRP management

▼ To Access the Reference for Delegated Administration Tags

The *Tag Library for Delegated Administration* provides tag names and syntax.

1 **Go to** *Tag Library for Delegated Administration*

2 **Select what contents you want to view.**

- Expand the title to view sections that you can select.
  - Tags for Desktop Channel and Container Management Tasks

- Tags for Portlet Management Tasks
- Tags for User Management Tasks
- Tags for Web Services for Remote Portlets (WSRP) Management Tasks

- Click the title link to view the beginning of the reference.

Index

E

F

R

S

T

U

V

W