Sun Java™ System

Portal Server Mobile Access 7 Administration Guide

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

Part No: 819-5305

Contents

About This Guide

This guide explains how to manage the administration functions of Sun Java™ System Portal Server Mobile Access (formerly known as Sun™ ONE Portal Server, Mobile Access) software. This chapter includes the following sections:

Who Should Use This Book

You should review this book if you are Portal Server administrator or system administrator responsible for managing Mobile Access software at your site.

What You Need to Know

Before you administer Mobile Access, you must be familiar with the following concepts:

- Basic Solaris™ administration procedures
- Basic UNIX® administration procedures
- LDAP (lightweight directory access protocol)
- Markup languages used to create portal content appropriate for mobile and voice environments, such as HTML, cHTML, and VoiceXML
- Solaris™ 8 Operating System (SPARC® Platform Edition) or Solaris™ 9 Operating System (SPARC® Platform Edition) or Solaris™ Operating System (x86 Platform Edition)
- Sun Java™ System Directory Server (formerly Sun ONE Directory Server)
- Sun Java™ System Access Manager (formerly Sun ONE Identity Server, and Sun Java System Identity Server)
- Sun Java™ System Portal Server 7
- Sun Java™ System Portal Server Secure Remote Access 7

Depending on the Web container that you are using, you should be familiar with one or more of the following:

- Sun Java™ System Web Server (formerly Sun ONE Web Server)
- Sun Java™ System Application Server
- BEA WebLogic Server™ 8.1 SP2 / SP4
- IBM WebSphere® 5.1

How This Book Is Organized

This book contains the following chapters and appendixes:

- Chapter 1, “Overview,” describes the key features of the Mobile Access software, as well as Mobile Access functions and Portal Server desktop, the primary end user interface.
- Chapter 2, “Managing Mobile Devices,” provides information about identifying and managing mobile devices, managing the client database, and using Sun Java System Access Manager client detection interface to manage client type data.
- Chapter 3, “Configuring Mobile Authentication,” describes authentication modules that can be useful to portal sites offering mobile access.

- Chapter 4, “Managing the Mobile Portal Desktop,” provides an overview of the wireless desktop dispatcher and the mobile Portal desktop properties.
- Chapter 5, “Configuring Mobile Applications,” provides an overview on how you can manage the application preferences
- Chapter 6, “Configuring Voice Access,” explains Mobile Access software’s support for voice access to portal sites.

A glossary and an index are also provided.

Conventions Used in This Book

The guide uses typographical conventions to represent types of information presented.

Monospaced Font

`Monospaced font` is used for any text that appears on the computer screen or text that you should type. It is also used for file names, distinguished names, functions, and examples.

Bold Monospaced Font

`Bold monospaced font` is used to represent text within a code example that you should type. For example, you might see something like this:

```
./pssetup
************************************************************************
Sun Java™ System Portal Server
************************************************************************
Installation log at
/var/sadm/install/logs/ipsinstall.13343/install.log
This product will run without a license. However, you must either purchase
a Binary Code License from, or accept the terms of a Binary Software
Evaluation license with, Sun Microsystems, to legally use this product.
Do you accept? yes/[no]
```

In this example, **./pssetup** is what you would type from the command line. The rest is what would appear as a result.

Italicized Font

An *italicized font* is used to represent text that you enter using information that is unique to your installation (for example, variables). It is used for server paths and names and account IDs.

Square or Straight Brackets

Square (or straight) brackets [] are used to enclose optional parameters. For example, in Portal Server software documentation, you will see the usage for the dpadmin command described as follows:

```
dpadmin list|modify|add|remove [command-specific options]
```

The presence of [command-specific] indicates that optional parameters can be added to the dpadmin command.

Command-Line Prompts

Command-line prompts (for example, % for a C-Shell, or $ for a Korn or Bourne shell) are not displayed in examples. Depending on which operating system environment you are using, you will see a variety of different command-line prompts. However, you should enter the command as it appears in the document unless specifically noted otherwise.

Related Documentation

The http://docs.sun.com℠ web site enables you to access Sun technical documentation online. You can browse the archive or search for a specific book title or subject.

Books in This Documentation Set

The following table summarizes the books included in the Portal Server Mobile Access core documentation set.

Book Title	Description
Sun Java System Portal Server Mobile Access Deployment Planning Guide	Describes how to plan for and deploy Sun Java System Portal Server Mobile Access software.
Sun Java System Portal Server Mobile Access Administration Guide	Describes how to administer Sun Java System Portal Server Mobile Access 7 using the administration console and the command line.
Sun Java System Portal Server Release Notes	Available after the product is released. Contains last-minute information, including a description of what is new in this current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.
Sun Java System Portal Server Mobile Access Developer's Guide	Describes a developer's perspective of the mobile access software. It also provides information about three mobile applications that is shipped with the software: Calendar, Address Book, and Mail, all of which can be fully customized by developers.
Sun Java System Portal Server Mobile Access Developer's Reference Guide	Provides reference to developer's on how they can customize the three mobile application that is shipped along with the software: Calendar, Address Book, and Mail.
Sun Java System Portal Server Mobile Access Tag Reference Library	Provides detailed information on the Sun Java System Portal Server Mobile Access tag libraries (in the software).

Other Portal Server Documentation

Other Portal Server books include:

- *Sun Java System Portal Server 7 Deployment Planning Guide*
- *Sun Java System Portal Server 7 Desktop Customization Guide*
- *Sun Java System Portal Server 7 Developer's Guide*
- *Sun Java System Portal Server 7 Administration Guide*

- *Sun Java System Portal Server 7 Secure Remote Access Administration Guide*
- *Sun Java System Portal Server 7 Technical Reference Guide*
- *Sun Java System Portal Server 7 Installation Guide*
- *Sun Java System Portal Server 7 Command Line Reference*

Use the following URL to view all the Sun Java System Portal Server 7.0 documentation:

http://docs.sun.com/coll/1303.1

Accessing Sun Resources Online

For product downloads, professional services, patches and support, and additional developer information, go to the following:

- Download Center
  http://wwws.sun.com/software/download/
- Professional Services
  http://www.sun.com/service/sunps/sunone/index.html
- Sun Enterprise Services, Solaris Patches, and Support
  http://sunsolve.sun.com/
- Developer Information
  http://developers.sun.com/prodtech/index.html

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in the product documentation, go to http://www.sun.com/service/contacting.

Related Third-Party Web Site References

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.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to `http://docs.sun.com` and click Send Comments. In the online form, provide the document title and part number. The part number of this guide is 819-5305 and can be found on the title page of this book or at the top of the document. For example, the title of this book is *Sun Java System Portal Server Mobile Access 7 Administration Guide*, and the part number is 819-5305.

Chapter 1

Overview

Sun Java™ System Portal Server Mobile Access (formerly known as Sun™ ONE Portal Server, Mobile Access) software extends the services and capabilities of Sun Java System Portal Server platform to mobile devices, such as mobile phones and personal digital assistants. It also provides a framework for voice access. Mobile Access software enables portal site users to obtain the same content that they access using browsers that require HyperText Markup Language (HTML). It supports Sun Java™ System Portal Server Secure Remote Access software and uses Sun Java™ System Access Manager software's administration console.

The features of the Mobile Access product are integrated seamlessly into Portal Server software. If you know how to administer Portal Server software, understanding how to administer Mobile Access software will not be difficult.

This chapter provides an overview of the key features for Mobile Access software, as well as a description of Mobile Access functions added to Sun Java System Access Manager software's administration console, portal server console, and Portal Server Desktop, the primary end user interface. The topics discussed in this chapter are:

Mobile Access Software

Knowledge of the following Mobile Access software features and how they extend the functions of Portal Server software are useful:

- Authentication
- Voice Access
- Channels, Containers, and Providers
- Rendering
- Mobile Applications

The Portal Desktop

Your portal site provides a mobile Portal Desktop and a voice Portal Desktop as well as a standard Portal Desktop. A wireless desktop dispatcher, which is a component of the Mobile Access software, controls them. The Portal Server desktop servlet forwards requests to the wireless desktop dispatcher.

The wireless desktop dispatcher uses display profile configuration data to determine which Portal Desktop—standard, mobile, or voice—is the appropriate one to route user requests to.

Regardless of how the user accesses a portal site, the Portal Desktop is the user's interface for the portal site. When a portal site user accesses a portal site with a mobile device, the mobile Portal Desktop appears. When a portal site user accesses a portal site with a telephone, the voice Portal Desktop responds.

These channels are available and visible by default on the mobile Portal Desktop:

- User Information
- Bookmark
- Personal Notes
- Sample XML

For more details on the mobile Portal Desktop, see Chapter 4, "Managing the Mobile Portal Desktop."

Client Types

Mobile Access software supports virtually every mobile device available. It uses a client profile to identify each mobile device, or client. It assigns each client a unique identifier called *client type*, based on the device markup language the device's browser uses.

These markup languages include:

- HDML (Handheld Device Markup Language)
- cHTML (compact Hypertext Markup Language)
- iHTML (i-mode Hypertext Markup Language)
- JHTML (J-Sky Hypertext Markup Language)
- XHTML (Extensible Hypertext Markup Language)
- VoiceXML (Voice Extensible Markup Language)
- WML (Wireless Markup Language)

Mobile Access software certifies WML support for the Nokia 6310i client and cHTML support for the Handspring Treo 180 client, although users can access portal content with any mobile device that uses one of these markup languages.

The Client Manager, which is part of the administration console of Access Manager, is used for managing client profiles. For details about mobile client type and device detection, see Chapter 2, “Managing Mobile Devices.”

Authentication

Mobile Access software supports the authentication modules that Portal Server software provides, but it also allows you to:

- Enable users to bypass the password prompt when logging into the mobile Portal Desktop.
- Enable users to log on as anonymous users.

For details on using these authentication modules, see Chapter 3, “Configuring Mobile Authentication.”

Voice Access

Mobile Access software provides the framework for VoiceXML applications. To access voice functionality, you must configure a voice server to provide speech recognition, text-to-speech, and a VoiceXML browser.

For details about voice access, see Chapter 6, “Configuring Voice Access.”

Channels, Containers, and Providers

Mobile Access software uses providers, channels, and containers to present content to the mobile Portal Desktop.

This topic provides information on:

- Channels
- Container Channels
- Providers

Channels

Channels display content in the mobile Portal Desktop. A *channel* consists of the provider object, configuration settings, and data files (such as templates) required to support the channel.

Container Channels

A container, or *container channel*, is a channel that displays content in the mobile Portal Desktop by aggregating the content of other channels. Mobile Access software adds the following default container channels to those included with Portal Server software:

- `JSPNativeContainer`
- `JSPRenderingContainer`
- `TemplateNativeContainer`
- `VoiceJSPDesktopContainer`
- `WirelessDesktopDispatcher`

Providers

Providers are the underlying implementation that present channel content to users on the mobile Portal Desktop. They adapt the interfaces of generic resources.

Provider content sources can include:

- Content in a file
- Output from an application
- Output from a service

Providers, which are Java™ class files, deliver content in the proper format for each type of mobile device. As a mobile Portal Desktop is created, each provider is queried for the content of its associated channel.

The default providers include:

- `JSPRenderingProvider`
- `RenderingWrappingProvider`

The following new providers are added to the default containers:

- `JSPRenderingContainerProvider`
- `JSPSingleRenderingContainerProvider`
- `WirelessDesktopDispatcherProvider`
- `WirelessJSPDesktopProvider`
- `WirelessTemplateClientConfigProvider`
- `WirelessTemplateContentProvider`
- `WirelessTemplateDesktopProvider`
- `WirelessTemplateLayoutProvider`

For details on using channels, containers, and providers to configure the mobile Portal Desktop, see Chapter 4, “Managing the Mobile Portal Desktop.”

Rendering

Using a mobile device, portal site users can access the same content that they would access using any HTML browser. The process that enables this is a translation process called *rendering*. Rendering allows you to create content once and display it appropriately on a variety of unique mobile devices.

The mobile rendering component detects devices and formats output for display on mobile devices. It consists of four subcomponents:

- *Client detection* determines the capabilities and characteristics of each mobile device that is used to access the portal. To do this, it uses the composite capability and preference profiles (CC/PP) specification, UAProf, or preconfigured data.

- A *rendering filter* passes content to the rendering engine and passes translated device-specific content back to the client, using the content type value set in the JavaServer Pages™ (JSP™) software template. It is a servlet filter that is applied to all authentication and application JSP software templates.
- The *rendering engine* converts AML, a device-independent markup language, to whatever device-specific markup language is appropriate for the client.

  When rendered content exceeds the page size of the target device, the rendering engine paginates it and stores the pages in the response buffer.
- The *response buffer* stores large output streams as separate, smaller responses so that they fit limited device buffers. The authentication, desktop, and mobile application components use the response buffer.

  When a client device makes a request for another page, it responds with the next page.

Mobile Access software supports both native and rendering channels and containers. *Native channels* are based onJSP technology and templates that are specific to Nokia WML clients. Clients that support HTML, VoiceXML, and WML use templates for a native Portal Desktop.

Rendering channels also use JSP technology. They enable a user to view a Portal Desktop that displays rendered content that is unique to a specific mobile device. This feature is made possible with the use of Abstract Markup Language (AML) templates that are passed through Mobile Access software's rendering process. Clients that support cHTML, iHTML, JHTML, XHTML, and HDML require AML templates for a rendered Portal Desktop.

Mobile Applications

The Mobile Access software provides four default applications that users can access in the mobile Portal Desktop. These are:

- Address Book
- Calendar
- Mail
- Fax

These applications run on a back-end server with the mobile Portal Desktop acting as the user interface. Once the link to an application is established, the application runs outside the control of Portal Server software. When the user is finished using the application, the user can return to the mobile Portal Desktop to work with other providers.

The Administration Console

Mobile Access software administrators use the Access Manager's administration console to complete most ongoing, day-to-day administrative tasks.

This section provides the following details about using the administration console to manage mobile access to a portal site:

- Mobile Access Software Features
- Typical Administrator Functions
- Logging In

For details about the Access Manager administration console and how to manage Access Manager software, see the *Sun Java™ System Access Manager 7 2005Q4 Administration Guide*.

Mobile Access Software Features

The features needed to manage mobile access to a portal site are part of the administration console and portal console.

The features that are a part of the administration console include:

- Conditional properties pages for each mobile device client type.
- Markup language categories and properties definitions for mobile device client types

The features that are a part of the portal console include:

- Default container channels for the mobile Portal Desktop
- Providers for mobile Portal Desktop
- Service Configuration pages for the mobile address book, mail and calendar applications

Typical Administrator Functions

Some of the functions that you can perform to manage a mobile and voice Portal Desktop with the administration console include:

- Editing device profiles through Access Manager software's client detection interface
- Controlling how channels are loaded with the mobile Portal Desktop by modifying properties of the desktop container
- Enabling users to bypass password identification by editing the NoPassword authentication module

Typical functions that are a part of the portal console include:

- Creating channels that are dynamically rendered for a particular mobile device
- Editing the default Mobile Access container to support non-default containers and their own unique channel lists

Logging In

The default URL for the Access Manager administration console is:

`http://`*`server`*`:`*`port`*`/amconsole`

The default URL for the Portal administration console is:

`http://`*`server`*`:`*`port`*`/psconsole`

Administrators log in to both the administration consoles as amadmin.

For more information on Portal Server software's use of the administration console, see the *Sun Java™ System Portal Server 7 Administration Guide*.

Chapter 2

Managing Mobile Devices

Sun Java System Portal Server Mobile Access software uses Sun Java System Access Manager client detection module to identify and manage the various clients, or mobile devices, that portal site users employ to access a portal site.

This chapter provides the following topics:

Understanding Client Detection

Client detection determines the capabilities and characteristics of each mobile device that is used to access the portal site. To do this, it uses the composite capability and preference profiles (CC / PP) specification, UAProf, or preconfigured data.

Mobile Access software requires that three properties be defined for every client. They are:

- `clientType`—A name that provides a unique index for the client data. `Nokia6310i_1.0` is the `clientType` value for the Nokia 6310i mobile phone.
- `parentId`—ID of the immediate parent for a device. (For an object with no parent, the value is the same as `clientType`.) `Nokia` is the `parentId` value for the Nokia 6310i mobile phone.

- userAgent—The HTTP user-agent string. This value can be empty for base and style information. Nokia6310/1.0 is the userAgent value for the Nokia 6310i mobile phone.

Mobile Access software also uses conditional properties to store and retrieve specific property values for client types. One example is the desktopContainer conditional property. The wireless desktop dispatcher reads this property to determine what the desktop container is for the requested client type.

Mobile Access software imports client type data from the file /etc/opt/SUNWam/config/ldif/sunAMClient_data.ldif into the LDAP directory and uses Access Manager software APIs to identify clientType. Matches are determined in the following order:

1. An exact match
2. A partial match
3. A keyword match

You can also dynamically apply UAProf profile against your base profile. Users need to retain FEDIClientDetector and do one of the following:

- configure your firewall to allow access from Mobile Access system to the public internet or selective handset vendor sites
- configure the Mobile Access system JVM to use a proxy server to access the public internet or selective handset vendor sites (Please refer to the technical note below)
- publish the UAProf profiles (RDF files) on an internal web server accessible to the Mobile Access system, e.g. within the DMZ, and configure DNS on the Mobile Access system to use the internal web server instead of the public internet for all UAProf requests.

Technical Note to configure proxy server to selectively access public internet:

JVM provides an option to specify proxy server details for external connection from the web container using an external proxy. It also allows you to specify the hosts that should not use the specified proxy. You can configure the Mobile Access system JVM to use a proxy server to access the public Internet.

Use the following JVM options in the web container:

```
Dhttp.proxyHost=<your-proxy-server-host>
Dhttp.proxyPort=<your-proxy-server-port>
```

Use the following option for bypassing proxy server for certain domains and hosts:

```
Dhttp.nonProxyHosts="*.<domain-name>|<hostname>|localhost"
```

Managing the Client Database

Client data are stored in two locations:

- An internal library or database containing all default client data definitions. The internal library is defined and supported by Access Manager software.
- An external library containing customized client data definitions. It overrides client data within the internal library.

When Mobile Access software is installed, all client data are stored in the internal library. Once you customize a device, an override version of the device is created and stored in the external library. The data in these libraries are merged at server startup time.

The Mobile Access product provides patches to update the internal library on a quarterly basis by adding new client data definitions and updating existing clients when their properties change. This process enables you to update the internal library at periodic intervals. It does not alter the customized data stored in the external library.

The name for this patch is PortalMAClientDeviceUpdateRev*nn*. The number for this patch is 116412-*nn*. The value for *nn* is the patch sequence number. For example, *nn* is 01 for the first patch and 02 for the second patch.

To Update the Client Database

1. Go to http://www.sun.com.
2. Click the Support & Training link.
3. Click the Patches link to go to the SunSolve patch database.
4. Follow the instructions provided.

Using the Client Manager

The Access Manager administration console provides a Client Manager that enables you to manage properties for mobile devices.

This section explains the following types of information that the Client Manager provides about client types:

- Markup Languages
- Styles
- Device Information
- Filter Option
- Client Editor

This section also explains how:

- To Launch the Client Manager
- To View Style Properties

Markup Languages

Mobile Access software supports these markup languages used by mobile client browsers:

- HDML (Handheld Device Markup Language)—Openwave's proprietary language, for mobile devices that use Openwave browsers. It uses Openwave's Handheld Device Transport Protocol (HDTP).

  Examples of devices in this category include RIM 950 and those using the UP.Browser 3.0 or earlier.

- JHTML (J-Sky Hypertext Markup Language)—Vodafone's proprietary language for Japanese J-Sky devices.

  Examples of devices in this category include J-Phone 2.0, J-Phone 3.0, and Mitsubishi V101D.

- VoiceXML (Voice Extensible Markup Language)—a standard for creating audio dialogs in interactive voice response applications.

  Devices in this category include any telephone or any Session Initiation Protocol (SIP) software-based phone.

- WML (Wireless Markup Language)—based on XML (Extensible Markup Language) and part of the Wireless Application Protocol (WAP).

  Examples of devices in this category include Motorola i95, Nokia 6310i, and Siemens S40.

- XHTML (Extensible Hypertext Markup Language)—a reformulation of HTML 4.0 that anyone can extend by adding new elements and defining new attributes.

  Examples of devices in this category include: Motorola T720, Nokia 3560, and Sony Ericsson T68.

- cHTML (compact Hypertext Markup Language)—a simpler version of HTML (Hypertext Markup Language) to accommodate mobile devices.

  Examples of devices in this category include Handspring Treo 180, Palm i705 Handheld, and Toshiba e400 Series.

- iHTML (inline Hypertext Markup Language)—the markup language used with NTT DoCoMo's Japanese i-mode service. It is similar to cHTML but provides proprietary extensions.

  Examples of devices in this category include NTT DoCoMo phones.

Styles

A Style is a set of properties for an associated group of devices for a markup language. For example, a `Nokia` Style is applied to all WML devices manufactured by Nokia.

At least one Style exists for each markup language. Some markup languages have multiple styles.

You cannot override Style properties. If you use an existing client as a template for a new devices when you create it, the new client inherits the existing client's Style properties.

Device Information

Device information is device-specific client type data that you can update.

When you change the device information for a default client type, you create a new and separate version of the default client type. This custom information is stored in the external library, while the default device information remains in the internal library. Two asterisks are added to the client type name of each custom device to differentiate it from devices in the internal library.

Filter Option

The Filter option is a search field that enables you to find and list groups of specific client types assigned to a specific Style.

Client Editor

The Client Editor enables you to create and customize a client type, and to manage client properties.

The Client Editor organizes properties in the following groups:

- General
- Hardware Platform
- Software Platform
- Network Characteristics
- BrowserUA
- WapCharacteristics
- PushCharacteristicsNames
- Additional Properties

To Launch the Client Manager

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

For information about managing the client data base, see “Managing the Client Database” on page 25.

To View Style Properties

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

5. From the tabs at the top, click the markup language for the device whose properties you want to examine (for example, `WML`).

   If client types using the markup language you selected are in the database, they appear in alphabetical order.

6. From the Style pull-down menu, pick the style that you want (for example, `Nokia`).

   The list of client types already in the database appears for the selected style.

7. Click the Current style properties link.

   The Edit *style* page appears. The Styles for General properties are displayed by default.

8. From the Properties pull-down menu, click the properties type that you want to view (for example: `Software Platform`).

NOTE Properties type choices include General, Hardware Platform, Software Platform, Network Characteristics, BrowserUA, WapCharacteristics, PushCharacteristicsNames, and Additional Properties.

9. To return to the Client Manager page, click Cancel.

Managing Client Type Data

You use the Client Manager in the administration console to manage client type data.

You can change client type properties, create new client types to accommodate new devices, set up client types with names and other properties that are customized for your site, and remove custom client types.

If you choose to create a new device based on an existing device, a process called *inheriting*, you must base the new device on either the styles or the properties of the existing device. Examine your new device and the existing device to decide which option -- styles or properties -- is prefereable. Both choices require you to customize device definitions.

NOTE The client type database consists of internal and external libraries. When you change or add to default client type information in the internal library, your updates are stored in the external library. Two asterisks added to the client type name indicate that it is a customized client type.

This section provides instructions for completing the following tasks:

- To Edit Client Types
- To Create a New Device by Inheriting Styles
- To Create a New Device by Inheriting Properties
- To Remove a Custom Device
- To Identify Selected Client Types for a Portal User

To Edit Client Types

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

5. From the tabs at the top, click the markup language for the device you want to edit (for example, `WML`).

   If client types using the markup language you selected are in the database, they appear in alphabetical order.

6. From the Style pull-down menu, pick the Style that you want (for example, `Nokia`).

   The list of client types already in the database appears for the selected style.

7. From the Client Type list, scroll down to find the client that you want to edit (for example, `Nokia6310i_1.0`).

TIP Clients are listed in alphabetical order.

To go directly to a specific client type, or to a group of client types, use the Filter option. In the Filter text box, type in the first character or first few characters of the client type you want to view and then click the Filter button. (For example: To find client types that start with the letter S, type in S*.)

To go to specific pages, scroll to the bottom and use the arrows or the Go option.

8. Click the Edit link in the Actions column for the client that you want to edit.

   The Edit *client-type* page is displayed. The General properties are displayed by default.

9. From the Properties pull-down menu, select the type of properties you want to change (for example, `Software Platform`).

10. Change or add values for each property you want to alter.

TIP To clear your changes and start over, click Reset. To return to the display of client types without making any changes, click Cancel.

11. Click Save to make these changes.

NOTE	If you do not click Save, your changes are not made. You must change one property type at a time and save those changes before you change another property type.

The properties for this device are now changed, and the list of client types for this style appears.

12. To verify that its properties are changed, find your client type in the Client Type list. Two asterisks added to the client type name indicate that you have customized this client type.

NOTE	Whenever you change a default client type, a Default link is added to the Actions column. The Default link points to the internal library. To remove your changes and reset the client type's properties to their default values, click this link. A prompt asking whether you want to complete this action is not provided.

To Create a New Device by Inheriting Styles

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

5. From the tabs at the top, click the markup language for the device you want to set up (for example, `WML`).

   If client types using the markup language you selected are in the database, they appear in alphabetical order.

6. From the Style pull-down menu, pick the Style that you want (for example, `Nokia`).

   The list of client types already in the database appears for the selected style.

7. Click the New Device button to display the Create New Device page.

8. If Style choices are required, click the button for the Style you want to assign (for example, `Nokia`).

9. Type in the Device User Agent value.

10. Click Next.

    The Device User Agent value you provided appears in the Client Type Name and The HTTP user-agent string fields.

11. If appropriate, change these values.

12. Click OK to save these properties.

    Your new device is now defined, and the Edit *Style* page appears. Displayed here are default properties inherited from the parent Style you assigned.

13. From the Properties pull-down menu, select the properties type that you want to modify (for example: `Software Platform`).

NOTE	Properties type choices include General, Hardware Platform, Software Platform, Network Characteristics, BrowserUA, WapCharacteristics, PushCharacteristicsNames, and Additional Properties.

14. Click Save to save your changes to these values.

TIP	To clear your changes and start over, click Reset. To return to the display of client types without making any changes, click Cancel.

15. Search the Client Type list to verify that your client type is available. Two asterisks added to the client type name indicate that you have customized this client type.

NOTE Whenever you add a new client type, a Delete link is added to the Actions column. The Delete link points to the external library.

To remove your new client type, click this link. A prompt asking whether you want to complete this action is not provided.

To Create a New Device by Inheriting Properties

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame and Organizations is selected in the Navigation frame.

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

5. From the tabs at the top, click the markup language for the device you want to copy (for example, `WML`).

   If client types using the markup language you selected are in the database, they appear in alphabetical order.

6. From the Style pull-down menu, pick the default Style that you want (for example, `Nokia`).

   The list of client types already in the database appears for the selected style.

7. From the Client Type list, scroll down to find the specific client that you want to use as a template for a new client type (for example, `Nokia6310i_1.0`).

TIP Clients are listed in alphabetical order.

To go directly to a specific client type, or to a group of client types, use the Filter option. In the Filter text box, type in the first character or first few characters of the client type you want to view and then click the Filter button. (For example: To find a client type that starts with the letter S, type in S*.)

To go directly to specific pages, scroll to the bottom and use the arrows or the Go option.

8. Click the Duplicate link in the Actions column for the client type that you want to use as a template for a new client type.

   The Duplicate Device page is displayed. The Client Type and Device User Agent properties for the device you are copying are displayed, with the prefix `Copy_of_` added to its name. (For example, `Copy_of_Nokia6310i_1.0`)

9. If appropriate, type in new names for these properties.

10. Click Duplicate to make these changes.

    The Edit *client-type* page is displayed. The General properties are displayed by default. The values for all properties views available here are inherited from the client type that you used as the master for this new client type.

TIP	To return to the display of client types without making any changes, click Cancel.

11. From the Properties pull-down menu, select which type of properties you want to change (for example, `Software Platform`).

12. Change or add values for each property you want to alter.

TIP	To clear your values and start over, click Reset. To return to the display of client types without making any changes, click Cancel.

13. Click Save to make these changes.

NOTE	If you do not click Save, your changes are not made. You must change one property type at a time and save those changes before you change another property type.

    The properties for this device are now changed, and the list of client types for this style appears.

14. Search the Client Type list to verify that your client type duplicate is available. Two asterisks added to the client type name indicate that you have customized this client type. (For example, `Copy_of_Nokia6310i_1.0 **`)

NOTE Whenever you add a new client type, a Delete link is added to the Actions column. The Delete link points to the external library.

To remove your new client type, click this link. A prompt asking whether you want to complete this action is not provided.

To Remove a Custom Device

TIP If you set up a custom device incorrectly and do not want to modify it, you can use these steps to remove it entirely.

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Click the Service Configuration tab.

3. From the Service Configuration frame on the left, under the Access Manager Configuration heading, click the arrow for Client Detection.

   The Client Detection global preferences appear in the Data frame on the right.

4. Click the Edit link following the Client Types label.

   The Client Manager interface appears. Details about HTML devices are displayed by default.

5. From the tabs at the top, click the markup language for the device you want to delete (for example, `WML`).

   If client types using the markup language you selected are in the database, they appear in alphabetical order.

6. From the Style pull-down menu, pick the Style that you want (for example, `Nokia`).

   The list of client types already in the database appears for the selected style.

7. From the Client Type list, scroll down to find the customized client that you want to remove (for example, `Copy_of_Nokia6310i_1.0`).

TIP	Clients are listed in alphabetical order. To go directly to a specific client type, or to a group of client types, use the Filter option. In the Filter text box, type in the first character or first few characters of the client type you want to view and then click the Filter button. (For example: To find a client type that starts with the letter S, type in S*.) To go directly to specific pages, scroll to the bottom and use the arrows or the Go option.

8. In the Actions column for the customized client that you want to remove, click the Delete link.

   The revised list of client types for this style is displayed.

9. Search the Client Type list to verify that your client type is no longer available.

To Identify Selected Client Types for a Portal User

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. From the View menu in the Navigation frame on the left, choose Users.

   A list of Access Manager users appears in the Navigation frame on the left.

3. Find the name of the user who is using the mobile device you want to identify the client type for and click the arrow for the user's name under Full Name.

   Information for this user appears in the Data frame on the right.

4. From the View menu in the Data frame on the right, choose Portal Desktop.

5. Click the Edit link.

   The Portal Desktop page pops up.

6. Click the Manage Channels and Containers link for the User Display Profile.

   The Channels page pops up.

7. Under the Container Channels section, click the Edit Properties link for `WirelessDesktopDispatcher`.

   The container's property settings page is displayed.

8. Scroll down to `selectedClients` and click the link.

   The `selectedClients` property's edit properties page is displayed. It lists client types for the devices that the user has used to access your portal site.

9. Review the list to locate the client type string for the device whose client type you want to identify.

Chapter 3

Configuring Mobile Authentication

Portal Server Mobile Access software supports the authentication modules provided by Sun Java System Portal Server software. This chapter describes three authentication modules that can be useful to portal sites offering mobile access:

- NoPassword Authentication
- Anonymous Authentication
- MSISDN Authentication

NoPassword Authentication

If your site specifications require it, you can allow users to log in to the mobile Portal Desktop without being prompted for a userID.

To Enable the NoPassword Module

1. Log in to the Sun Java System Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame and Organizations is selected in the Navigation frame.

2. Click the Service Configuration tab.

3. In the Navigation frame on the left, click the arrow for Core under Access Manager Configuration.

   The Core attributes page displays in the Data frame on the right.

4. In the Pluggable Authentication Module Classes text box, type in the following:

   `com.sun.identity.authentication.modules.nopassword.NoPassword`

5. Click Add and verify that the new class appears in the list.
6. Click Save.
7. Scroll down to the Organization Authentication Modules list and verify that NoPassword is now present.
8. Click the Identity Management tab in the Header frame.
9. From the View field in the Navigation frame, click Services.
10. In the Navigation frame on the left, click the arrow for Core under Access Manager Configuration.
11. From the Organization Authentication Modules field, click NoPassword and the other authentication modules you use.

TIP To select more than one module, press and hold the shift key when you click.

12. Click the Save button.
13. Verify that the modules you selected are highlighted.

Once NoPassword is enabled, users can bypass the password prompt by using the following URL to log in:

`http://`*server*`:`*port*`/portal/UI/Login?module=NoPassword`

Although NoPassword is enabled in the administration console, the module is not displayed in the list of available modules during user login when `authlevel=0` is the authentication level setting. To verify this, go to:

`http://`*server*`:`*port*`/amserver/UI/Login?authlevel=0`

For details about accessing authentication modules with specific authentication levels, See the *Sun Java™ System Access Manager 7 2005Q4 Administration Guide*.

Anonymous Authentication

If you want a user to access your portal site to explore what the experience of an authenticated user is, you can allow users to log in to the mobile Portal Desktop as anonymous users.

This feature presents a snapshot of the mobile and voice Portal Desktop of a user with an authenticated session.

NOTE Anonymous users cannot change, store, or alter the content or configuration of channels with stateful data. If you support anonymous authentication, make sure that these channels are not available to these users.

To implement anonymous authentication, see the *Sun Java™ System Portal Server 7 Administration Guide.*

The Portal Desktop for anonymous authentication uses the `WirelessDesktopDispatcher` as well as device-specific containers for both JavaServer Pages™ (JSP™) software and templates. All channels to be displayed to the anonymous user must be included in these containers, just as they are for authenticated users.

TIP To support a new device that may need a client-specific mobile or voice Portal Desktop for an anonymous user, do the following:

1. Create the appropriate device-specific container.
2. Alter the `WirelessDesktopDispatcher` in the anonymous user's display profile to use the new container for that particular device type.

MSISDN Authentication

The users of an organization can be confiured to authenticate using MSISDN - Mobile Station ISDN, a standard international telephone number used to identify a given subscriber. This allows the users to log into the mobile portal desktop without the user passing authentication credentials.

This feature limits the format of the login URL. We recomend the following format for the URL:

```
http://<access-manager-host>:<port>/<service-deploy-URI>/UI/Login?module=MSISDN&org=<org__name>
```

To implement MSISDN authentication and how to configure it, see the *Sun Java™ System Access Manager 7 2005Q4 Administration Guide.*

Chapter 4

Managing the Mobile Portal Desktop

Portal Server Mobile Acess software uses the Access Manager software's administration console and portal administration console to manage the mobile Portal Desktop.

This chapter provides the following topics:

Understanding the Wireless Desktop Dispatcher

Once you install Mobile Access software, your Portal Server site provides a mobile Portal Desktop and a voice Portal Desktop as well as a standard Portal Desktop. At the time a user logs in to Portal Server, the wireless desktop dispatcher, which is a component of Mobile Access software, determines which Portal Desktop is the appropriate one to route user requests to.

The wireless desktop dispatcher uses an XML Display Profile configuration to determine which Portal Desktop—standard, mobile, or voice—is the appropriate one to route user requests to. The wireless desktop dispatcher:

- Determines the client type of the desktop request
- Uses a display profile configuration to match that client to the appropriate container
- Routes the request to the appropriate container

The default channel for the mobile Portal Desktop is the `WirelessDesktopDispatcher`. You can edit the `WirelessDesktopDispatcher` container to support other containers for particular devices.

Wireless Desktop Dispatcher Properties

The wireless desktop dispatcher properties include:

- `desktopContainer`

  The `desktopContainer` property maps mobile devices to appropriate containers. This mapping identifies how requests are routed.

  By default, HTTP requests from devices that display native content (for example, Nokia devices that use WML) are routed to the `JSPNativeContainer`. HTTP requests from devices that display rendered content are routed to the `JSPRenderingContainer`.

- `selectedClients`

  The `selectedClients` property tracks the mobile devices used to access your portal site. Whenever anyone uses a new device to access your portal site, the client type of that device is added the selectedClients property s collection.

  This property is also used to display a list of devices on the Mobile Devices edit page in the standard Portal Desktop. Individual users can view what devices they have used, and they can add to the list simply by logging into the mobile Portal Desktop with other devices.

Conditional Properties

Conditional properties for client types enable administrators to specify properties for a channel or container channel that are specific to a client type. Conditional properties for client types can also be hierarchical, just as client data is hierarchical.

The syntax for a conditional property is `client=`*clientType*. For example, `client=WML` is the name of the conditional property for WML client types.

The `desktopContainer` property for the wireless desktop dispatcher is an example of a client conditional property for the client type `client=WML`. By default the definition for this property is `desktopContainer=JSPRenderingContainer`.

Here is a hierarchical representation of the default desktopContainer property for Nokia devices:

```
client=WML -> desktopContainer=JSPRenderingContainer
       client=Nokia -> desktopContainer=JSPNativeContainer
```

Devices within the WML client style use the JSPRenderingContainer. The subset of WML clients defined by the Nokia client style use a different desktopContainer definition, however. They use the JSPNativeContainer.

Channel State Properties

These properties indicate the state of a channel to both the JSPNativeContainer and the JSPRenderingContainer. They allow an end user to display only a channel s title bar on a mobile Portal Desktop instead of loading a channel s content inline.

NOTE On the standard Portal Desktop, you can provide buttons on a channel so that the user can minimize or maximize its content. This is not currently supported with the mobile Portal Desktop.

These properties include:

- defaultChannelIsMinimizable and defaultChannelIsMaximizable

These properties determine whether the Load Channels with desktop check box is to be displayed on the user's Mobile Devices edit page in the standard Portal Desktop. The default value of both properties is true. The check box thus is displayed. If either property is false, the check box is not displayed.

NOTE To display the Load Channels with desktop check box, both values must be true. If either is false, the check box is not displayed.

- defaultChannelIsMinimized

  This property determines whether the Load Channels with desktop check box is to be checked on the user's Mobile Devices edit page in the standard Portal Desktop. The default value for this property is true. The check box thus is not checked, and all channels in the container have a window state of minimize.

When this property is set to false, the check box is checked, and all channels in the container have a window state of normal.

JSPRenderingContainer Properties

Two advanced properties for the JSPRenderingContainer specify how error pages and edit pages are displayed. These properties are:

- `errorChannel`

  This property indicates what channel is used to render an error page in case of a desktop error when using the `JSPRenderingContainer` and the rendering engine.

- `editContainerName`

  This property indicates what channel is used to render an edit page for a channel that is marked editable for a particular client.

Chapter 5

Configuring Mobile Applications

Portal Server Mobile Access software provides the following default applications that end users can access from the mobile Portal Desktop. These are:

- Address Book
- Calendar
- Mail
- Fax

These applications run on a server on which Portal Server software is installed. The mobile Portal Desktop acts as the user interface. Once the link to the application is established, they run outside the control of Portal Server software. When the users are finished using the application, they can return to the mobile Portal Desktop to work with other applications or channels.

To obtain detailed information on the Service Configuration and Access Manager features, refer to the *Sun Java™ System Access Manager 7 2005Q4 Administration Guide*. To obtain detailed information on configuring communications channels, refer to the *Sun Java™ System Portal Server 7 Administration Guide*.

Using Service Configuration Attributes

Within the Service Configuration level, you can view and change default application preferences as well as control which preferences users can edit themselves.

Two types of preferences for mobile applications are stored in Service Configuration:

- **Global** These preferences establish global defaults and control the user's ability to edit them.

- **Dynamic** These preferences establish the default pattern of information that is eventually stored at the user level.

Using Access Manager Attributes

Within the Access Manager level, you can view and change default application preferences for any organization or user.

Preferences stored here are copies of the dynamic preferences stored at the Service Configuration level. For the mobile mail application, an organizational preference is stored here also.

To Edit Identity Management Users Attributes

1. Log in to the Access Manager administration console as the administrator.

   By default, Identity Management is selected in the Header frame (the top horizontal frame) and Organizations is selected in the Navigation frame (the left vertical frame).

2. Choose Users from the View menu in the Navigation frame on the left.

   The list of Access Manager users appears in the Navigation frame on the left.

3. From the UserId column, select the box next to the name of the user and click the arrow following the user's full name.

   The user's information appears in the Data frame on the right.

4. Choose the mobile application from the View menu in the Data frame on the right.

   The mobile application frame with its preferences is displayed.

5. Enter the values, if you wish to change a user's preferences.

6. Click Save.

   You can now view the new settings.

TIP	To clear all of your values and start over, click Reset. To undo your entries, click Delete.

About Mobile Application Templates

Mobile application templates exist to establish the rules governing the storage of application preferences. The templates are represented as uniform resource locators (URLs) described in RFC 1738 published by the World Wide Web Consortium (W3C).

Administrators can edit template strings to assign values to properties within the strings and to apply certain rules of use to those properties.

A template string must start with the word `default` followed by the pipe symbol `|`. The string provides the name of the template configuration and preferences that can be changed to alter the application's behavior. These preferences are set to default values when Mobile Access is installed.

Code Example 5-1 is an example of an address book template. Template strings appear in the field as a single, long string. This example divides the template string into separate lines for readability purposes. Line breaks have been added preceding each ampersand (&).

Code Example 5-1

```
default|undef:///?configName=MA-AB-APP
&default=sortBy
&default=sortOrder
&merge=sortBy
&merge=sortOrder
&sortBy=cn
&sortOrder=asc
```

The name of the template in this example is `MA-AB-APP`. This template includes two preferences—`sortOrder` and `sortBy`. It provides default values for them and rules permitting user definitions of these preferences.

Configuring Fax

The fax feature enables the hand held device users to send a document to a fax machine. The fax machine can be any where, including a local fax machine where the user wants the document in a printed form. In this release, we address one feature where the user can fax one or more documents that are available as a received email attachment(s).

NOTE The capability of faxing more than one e-mail attachment is dependent on the Fax Service provider. The default implementation (Fax1.com) does not support this feature as the Fax service provider does not allow the same.

Users can send a complete email message to the service provider which can be printed as a Fax. Users can also be able to specify a different service provider than the one that is configured, in which case, the following information shall be provided by the user:

- Fax Number: The number to which the user intends to send the e-mail as fax.

  The fax number may be suffixed with the fax service provider's DNS domain name. If this information is not provided, the default, configured domain name shall be used

  For example: <fax#@domain>

- From Address: The e-mail address should match with the fax service account that the user has created with the service provider.
- Subject: Contains the required credential information for service provider to verify the request.

The following limitations are applicable for this feature:

- The type of the document are set by the service providers
- User can only fax the complete email message. They don't have an option to choose selective attachments in the message and fax them.
- If the message format which the user is trying to fax is non-plain text such as HTML, then the fax contents would not appear as expected. In case of HTML, the fax content would have the HTML code with HTML elements in it. Hence, users must ensure that the fax message is of type plain text. This is a requirement from Fax1
- Since there is no standard which dictates how the email should be formatted for the email to fax services, all service providers define their own format. Formats of the mail differ from provider to provider. Supported attachment types to be faxed may be found by visiting the respective service provider site since not all file types are supported.

Chapter 6

Configuring Voice Access

This chapter explains Portal Server Mobile Access software's support for voice accessibility. It contains the following sections:

Understanding Voice Functionality

Mobile Access software's support for voice accessibility allows users to access voice-enabled content by phone, or with software that enables Session Initiation Protocol (SIP) and supports Voice over IP (VoIP).

This software provides the following voice-enabled functionality:

- **Voice Authentication**—Allows users to speak (or key in) their account number and PIN to identify themselves to your portal site. This authentication process enables users to access the same content that is available to them on the standard Portal Desktop or the mobile Portal Desktop.
- **Channel management**—Allows users to select from the list of voice-enabled applications. Users can add and remove voice-enabled applications from their voice Portal Desktop.
- **Notes**—Allows users to hear a list of messages published to all Portal Server users.

- **Personal Notes**—Allows users to hear personal messages associated with their Portal Server account.
- **Mail Voicelet**—Allows users to hear a summary of e-mail messages such as the number of e-mail messages, number of read / unread messages. It also allows users to fetch emails from a pre-configured email server. Users can listen to the content of a specific e-mail message such as the headers and body of the message and also delete a specific message.

Configuring Voice Access

Mobile Access software has built-in support for providing voice accessibility. Its voice components are certified against the Nuance Voice Platform, which includes a VoiceXML 2.0-compliant voice browser.

To access the voice functionality, a *voice server* must be configured to provide speech recognition, text to speech, and a VoiceXML browser.

This section discusses the following available configuration options:

- Using a Voice Service Provider
- Using a Telephony System
- Using Session Initiation Protocol
- Using Native Audio

Using a Voice Service Provider

If your Portal Server software is accessible from the Internet, the simplest option is to use a voice application service provider, or a voice hosting service. The service provider runs the voice server and provides a phone number for calling your Portal Server software.

To enable voice service provider access, perform the following steps:

1. Install Sun Java System Portal Server Mobile Access software, and make sure the system is accessible from the Internet. You may need to open your firewall to HTTP traffic for the port assigned to Portal Server software web interface.

2. Identify a voice service provider that uses the Nuance Voice Platform. The platform must use the VoiceXML browser in the Nuance Voice Web Server, not just the core Nuance recognition platform. Contact Nuance for a complete list of voice service providers that support the Nuance Voice Platform.
3. Create an account with your voice service provider, and specify the HTTP URL of your Portal Server software. The service provider will assign a phone number and possibly a PIN for each service that you create. Most voice service providers allow the creation of evaluation accounts.

Using a Telephony System

If you plan to access Portal Server software using a phone, you must obtain a Digital Signal Processing (DSP) telephony card that is compatible with your telephony network (or switch) and your voice server hardware platform. NMS Communications has a range of DSP solutions that are compatible with the Solaris™ Operating System (SPARC® Platform Edition), as well as Windows Intel platforms.

To use telephony system access, perform the following steps:

1. Install the DSP hardware, device drivers, and software using the manufacturer's recommendations. If you use an NMS Communications DSP card, you will also need the NMS Natural Access 2002-1 software.
2. Test the DSP card configuration using a test utility, usually supplied with the DSP software.
3. Provision a port or range of ports in your telephony switch, and configure the ports to use a protocol compatible with the DSP card.
4. Assign phone numbers to the ports established in step 3 above.
5. Connect the DSP card to the switch, and test its connectivity by dialing the numbers established in step 4 above. Use the test programs provided with the DSP hardware to verify correct operation.
6. Refer to the section Installing a Nuance Voice Web Server to install the Nuance components.

Using Session Initiation Protocol

Session Initiation Protocol (SIP) allows users to access your portal site using Voice over IP (VoIP) from any computer equipped with a microphone and speakers.

To enable this, you must install a SIP software-based phone to communicate with the voice server. Pingtel has a SIP-enabled software-based phone application, InstantExpressa, which can be used for this purpose. A number of public domain SIP-based phones are available on the Internet.

To set up a system for SIP access, perform the following steps:

1. Locate the computer that will be used to communicate with Portal Server software. Ensure that it can communicate with your voice server using the SIP protocol. You may need to configure your firewall to open the SIP ports.
2. Install a SIP-based phone on this computer, and test it by connecting to a SIP service, if available. Many SIP-based phones include simple test servers that you can install for testing purposes.
3. Refer to the section Installing a Nuance Voice Web Server on page 58 to install the Nuance components.

Using Native Audio

The term *native audio* refers to accessing Portal Server software directly from the system that is running the voice server software. This means that the voice server must have an audio card and a microphone. Users can only interact with the native audio locally. Remote access is not possible.

For systems running Microsoft Windows operating systems, the sound card must be SoundBlaster compatible. To use native audio access, perform the following steps:

1. Refer to the section "Installing a Nuance Voice Web Server," to install the Nuance components.
2. Configure the Nuance Voice Web Server for native audio. Refer to the Nuance Voice Web Server documentation for details.

Installing a Nuance Voice Web Server

NOTE If you are using a voice service provider, you can skip this section.

Installing a Nuance Voice Web Server involves selecting an appropriate hardware platform, installing and configuring the voice recognition software, and providing connectivity through a phone or IP network interface.

To install the server, complete the following steps:

1. Select a dedicated system to use as the voice server. Refer to the Nuance Voice Web Server 2.0 Release Notes for hardware requirements.
2. Obtain the Nuance Voice Platform software. Refer to the Nuance documentation for hardware requirements.
3. Install and configure the Nuance software and additional software on the voice server. Follow the Nuance installation instructions. The order in which the software components and service packs are installed is important.
4. Configure the Nuance Voice Web Server for native audio, SIP, or telephony access.
5. Start the Nuance services, followed by the Nuance Voice Web Serve.
6. Access the voice server using a microphone, a telephone, or your SIP software, and verify that you can access the Nuance voice demo applications. At this point, it may be necessary to tune the Nuance software to improve voice recognition performance. Refer to the Nuance documentation for instructions.

When you have completed your installation, configure the Nuance software to access Portal Server software.

To do so, complete the following steps:

1. Shut down the Nuance Voice Web Server if it is running.
2. Locate the `browser.conf` file in the Nuance Voice Web Server installation. On Microsoft Windows operating systems, you will find this file in the following directory:

   ```
   C:\Nuance\VWS\conf\browser.conf
   ```

3. Open the file using a text editor, and locate the following entry:

   ```
   browser.initialPage=%CONTENT%/%LOCALE/initial/dialogs/main.vxml
   ```

4. Modify this entry to contain the URL of your Portal Server software installation.

   For example:

   ```
   browser.initialPage=http://portal.example.com:58080/amserver/UI/Login
   ```

   where `portal.example.com` is the host name of the system where Portal Server software is installed, and `58080` is the port assigned to the portal web interface.

5. Save the `browser.conf` file.

6. Restart the Nuance Voice Web Server. Voice Web Server initializes and waits for incoming calls.

Creating Voice-Accessible User Accounts

To use the voice functionality, you must create user accounts on Portal Server software. For voice access, you must assign numeric user IDs and passwords to accounts that will be voice-accessible.

Create a 10-digit account number. The account password is used as the PIN, so assign a numeric password to the account. The PIN must be a four-digit number.

Refer to the *Sun Java™ System Portal Server 7 Administration Guide* for information about creating user accounts.

Accessing Portal Server Software

This section describes the following ways of accessing Portal Server software:

- Using a Voice Service Provider
- Using a Phone
- Using Session Initiation Protocol
- Using Native Audio

Using a Voice Service Provider

Call the number assigned by the service provider to your account.

If your software is correctly configured, you will hear the following dialog. Speak (or key in) the account number and PIN assigned to your Portal Server account.

System:

This is Voice Portal, by Sun.

Please say or key in your account number.

User:

415 555 5940

System:

Got it.

And what s your PIN?

User:

1234

System:

Hello, John. You're signed in.

Here are the portal channels you can choose from: Personal Notes, Notes.

You can also say add a channel.

Which would you like?

Using a Phone

To access Portal Server software by phone, simply dial the number assigned to the voice server by your telecommunications service provider.

Using Session Initiation Protocol

If you are using a SIP software phone, you must specify the SIP address of your Voice Server software (not Portal Server software). Use the following URL format:

```
sip://vws@voiceserver.example.com:5060
```

Where `vws` means "access the Nuance Voice Web Server service" on the server named `voiceserver.example.com`, and `5060` is the port that Nuance Voice Web Server is listening on for SIP connection requests.

Using Native Audio

If you have configured the Nuance Voice Web Server for native audio, start it using the Nuance `vws` command. The server starts and immediately accesses Portal Server software.

Glossary

Refer to the *Java Enterprise System Glossary* (http://docs.sun.com/doc/816-6873) for a complete list of terms that are used in this documentation set.

Index

D

E

F

G

H

I

J

M

N

P

R

S

T

U

V

W

X