



Sun Java™ System

# Application Server Platform Edition 8.1 Quick Start Guide

---

2005Q1

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

819-1004

Copyright © 2004-2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

THIS PRODUCT CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF SUN MICROSYSTEMS, INC. USE, DISCLOSURE OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF SUN MICROSYSTEMS, INC.

U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, the Duke logo, the Java Coffee Cup logo, the Solaris logo, the SunTone Certified logo and the Sun ONE logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon architecture developed by Sun Microsystems, Inc.

Legato and the Legato logo are registered trademarks, and Legato NetWorker, are trademarks or registered trademarks of Legato Systems, Inc. The Netscape Communications Corp logo is a trademark or registered trademark of Netscape Communications Corporation.

The OPEN LOOK and Sun(TM) Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this service manual are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

---

Copyright © 2004-2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuels relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs des brevets américains listés à l'adresse <http://www.sun.com/patents> et un ou des brevets supplémentaires ou des applications de brevet en attente aux Etats - Unis et dans les autres pays.

CE PRODUIT CONTIENT DES INFORMATIONS CONFIDENTIELLES ET DES SECRETS COMMERCIAUX DE SUN MICROSYSTEMS, INC. SON UTILISATION, SA DIVULGATION ET SA REPRODUCTION SONT INTERDITES SANS L'AUTORISATION EXPRESSE, ECRITE ET PREALABLE DE SUN MICROSYSTEMS, INC.

Cette distribution peut comprendre des composants développés par des tierces parties.

Des parties de ce produit peuvent être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Java, Solaris, JDK, Java Naming and Directory Interface, JavaMail, JavaHelp, J2SE, iPlanet, le logo Duke, le logo Java Coffee Cup, le logo Solaris, le logo SunTone Certified et le logo Sun[tm] ONE sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. Legato, le logo Legato, et Legato NetWorker sont des marques de fabrique ou des marques déposées de Legato Systems, Inc. Le logo Netscape Communications Corp est une marque de fabrique ou une marque déposée de Netscape Communications Corporation.

L'interface d'utilisation graphique OPEN LOOK et Sun(TM) a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une license non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de ce manuel d'entretien et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes biologiques et chimiques ou du nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou reexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font l'objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

# Quick Start Guide

Welcome to the *Quick Start Guide* for the Sun Java™ System Application Server Platform Edition 8.1 software. This guide describes basic steps for starting the Application Server software, and packaging and deploying applications. It also provides information about the Admin Console and command-line tools.

The intended audience for this guide is the system administrator, network administrator, Application Server administrator, and developer who is interested in learning the basics about Application Server software.

This document contains information about the following topics:

- [Starting and Verifying the Server](#)
- [Packaging and Deploying Applications](#)
- [Using Command-line Tools](#)
- [Where to Go Next](#)

Information about [Documentation Conventions](#) used in this guide is also included.

## Starting and Verifying the Server

This section provides information for administrators who want to start and verify the Sun Java System Application Server software:

- [Starting the Admin Server](#)
- [Confirming That the Server is Running](#)
- [Deploying the hello Application](#)

For administration tasks, the Application Server software provides these tools, which enable administrators to manage server instances:

- The asadmin utility, a command-line tool
- The Admin Console, a browser-based graphical user interface (GUI)

A *domain* is a collection of configuration settings and deployed applications that make up a server instance. You specify a domain when starting a server instance. That instance then uses the configuration settings and runs the applications that make up the domain. A default domain called *domain1* is created when you install Application Server.

## Starting the Admin Server

To start the server, follow the steps below. Solaris and Linux steps are in the first column, and Windows steps are in the second column:

Solaris/Linux	Windows
1. Add the <i>install_dir/bin/</i> directory to the PATH environment variable:	From the Start menu, choose Programs -> Sun Microsystems -> Application Server PE-> Start Default Server.
C Shell: <code>% setenv PATH <i>install_dir/bin</i>:\$PATH</code>	
Bourne Shell: <code>PATH=<i>install_dir/bin</i>:\$PATH export PATH</code>	
2. Start the server by entering this command from the <i>install_dir</i> : <code>% asadmin start-domain domain1</code>	

## Confirming That the Server is Running

When the server has completed startup, this message appears: Domain domain1 is ready to receive client requests. Additional services are being started in the background.

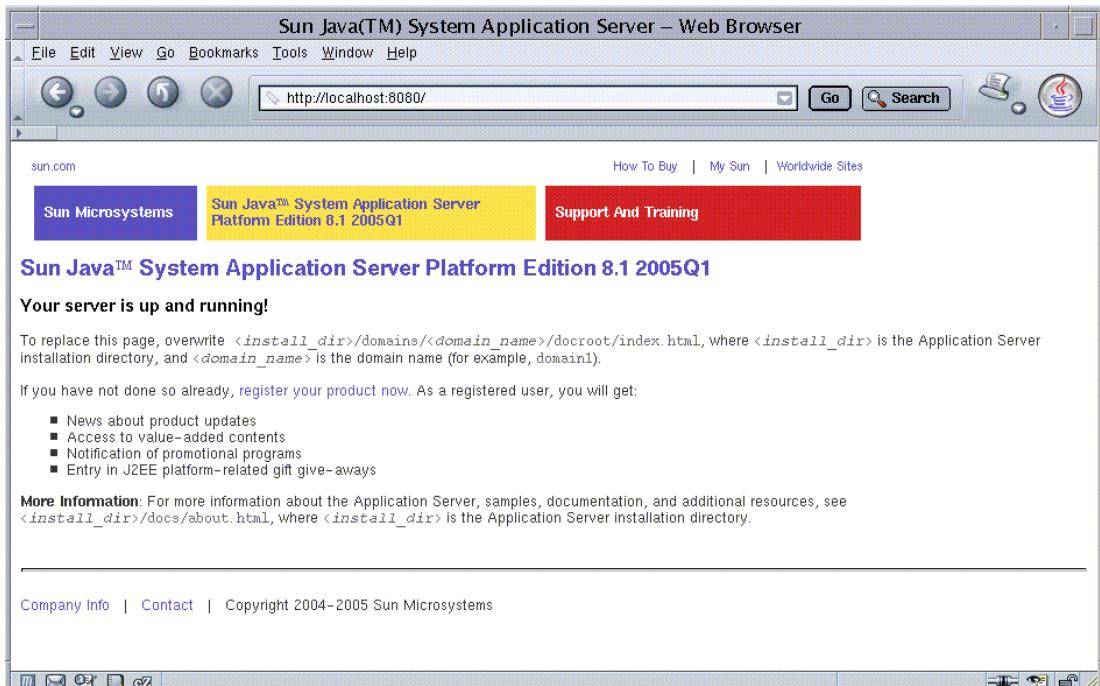
To verify that the server is running on your system, click this URL:

<http://localhost:8080>

The URL has two parts:

- The name *localhost* refers to the system the browser is currently running on. If the Application Server is running on another system, substitute the name of that system in the URL.
- The default port number is 8080. If you changed the port number during installation, use that number in your browser's address field instead.

You should see the server Welcome page, shown below.



This might take a few seconds as the server completes initialization.

When you see this page, you know that the server is running. If the server is not running, but your system is operating normally, you get a server not found error. (In some browsers, that error automatically launches a search engine.) For other problems, you may need to check your server and browser settings, as described in the *Sun Java System Application Server Platform Edition 8.1 2005Q1 Troubleshooting Guide*.

Later on, to stop the server, follow the steps below. Solaris and Linux steps are in the first column, and Windows steps are in the second column:

---

Solaris/Linux	Windows
From the <i>install_dir/bin</i> directory, type this command: % <b>asadmin stop-domain</b>	From the Start menu, choose Programs -> Sun Microsystems -> Application Server PE -> Stop Default Server.

---

## Deploying the hello Application

The server installation includes a pre-packaged application that says “Hello,” taken from *The J2EE™ 1.4 Tutorial*. To deploy it, you must give the Application Server software the information it needs to load the application, map a URL to it, and connect it to the resources it uses. That information is contained in deployment descriptors, which form part of the packaged application. The hello application is contained in a Web ARchive (WAR) file in the `samples` directory. Packaged applications can also be in the form of a Java ARchive (JAR) or Enterprise ARchive (EAR) file.

Dragging or copying to a domain’s `autodeploy` directory lets you put a pre-packaged application into use immediately, with minimal effort.

To deploy using `autodeploy`:

1. Find *install\_dir/samples/quickstart/hello.war*
2. Copy it to *install\_dir/domains/domain1/autodeploy/*

---

**TIP** On Windows, you can create a shortcut to the `autodeploy` directory on your desktop and then drag and drop the `hello.war` file onto the shortcut.

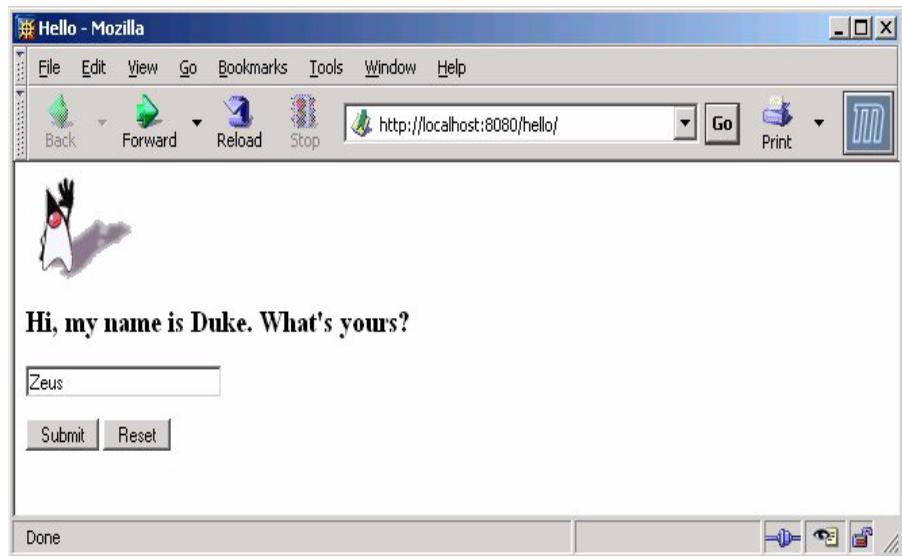
---

When the server has finished deploying the application, it creates a file named `hello.war_deployed` in the `autodeploy` directory. Depending on the speed of your system, the process can take a few seconds. Until that file appears, a 404-File Not Found error occurs when you try to visit the application page.

Click this URL to visit the application page:

<http://localhost:8080/hello>

You should now see the application’s first page, which asks you to fill in your name:



Type your name, and click Submit.

You should now see the customized response, giving you a personal Hello.

---

**NOTE** You can also use the `asadmin` command-line utility to deploy from the command line or in a shell script. The command is `asadmin deploy`.

---

## Using the Admin Console

The Admin Console is a web-based interface that simplifies a variety of administration and configuration tasks. It is commonly used to:

- Deploy and undeploy applications
- Enable and disable applications
- Identify which applications are currently running
- Configure JDBC resources and other server settings
- Select and view recent entries in the log file

For further information about using the Admin Console, consult the online help or the *Sun Java System Application Server Platform Edition 8.1 2005Q1 Administration Guide*.

To log in to the Admin Console:

1. Click this URL:

<http://localhost:4848/asadmin>

- The *localhost* variable is the name of the system that the browser is currently running on. If Application Server software is running on another system, substitute the name of that system for *localhost*.
- 4848 is the Admin Console's default port number. If you changed the port number during the installation, use that number instead.

---

**TIP**

Windows users can also use the Start menu option: Programs -> Sun Microsystems -> Application Server ->Admin Console.

---

2. When the log in window appears, enter the admin user name and password.
3. Click Log In.

When the Admin Console appears, it looks like this:



You can use the options at the top of the information panel to configure a variety of operational settings for Application Server software.

To see the server's log file, click "Search Log Files" under the Common Tasks heading in the right-hand pane. When the Log Viewer window appears, you can see status messages from the server and output from your J2EE application.

This ends the first section of the *Quick Start Guide*. You have successfully started the Application Server, deployed the Hello application, and viewed the Admin Console.

# Packaging and Deploying Applications

This section provides information for packaging and deploying applications in the Application Server software.

This section describes two techniques for packaging and deploying applications:

- [Using the deploytool Utility](#)
- [Deploying from a Development Directory](#)

For more information, see “Getting Started with Web Applications” in *The J2EE 1.4 Tutorial*.

## Using the deploytool Utility

The deploytool utility is recommended for configuring complex Java™ 2 Platform, Enterprise Edition (J2EE™ platform) applications that require security controls, context parameters, enterprise beans, property-group settings (for HTML templates), or database connections.

The deploytool utility lets you:

- Create XML deployment descriptors without manually editing XML files.
- Package the classes and web content for a J2EE application into a Java archive (JAR), web application archive (WAR) or enterprise archive (EAR) file.
- Edit an existing archive file to modify its contents.
- Deploy a packaged application onto the Sun Java System Application Server software, or onto other application servers.

### Get Ready

The `install_dir/samples/quickstart/` directory contains the files used in the Hello World application.

The `web` subdirectory contains the `index.jsp` and `response.jsp` files and the `images` directory, which contains the `duke.waving.gif` file.

The `web/WEB-INF` subdirectory contains the `sun-web.xml` and `web.xml` files.

The `src/conf` subdirectory contains the `LocalStrings.properties` file.

The Hello World application contains two JSP™ pages and the two files that they reference.

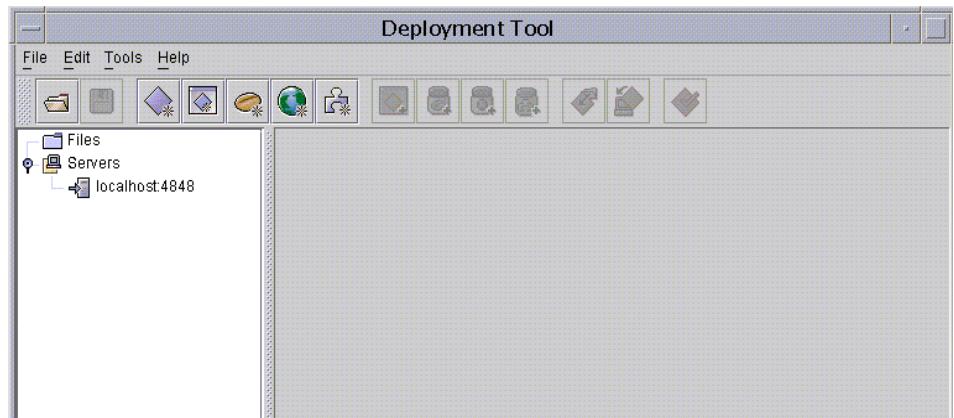
## Package the Application

In this section, you practice packaging the Hello World application using `deploytool`.

Begin by starting `deploytool` using the instructions below. Solaris and Linux steps are in the first column, and Windows steps are in the second column:

Solaris/Linux	Windows
From the <code>install_dir/bin</code> directory, start the tool: <code>% deploytool</code>	From the Start menu, choose Programs -> Sun Microsystems -> Application Server PE -> Deploytool.

When `deploytool` starts, it looks like this:



The tree structure in the pane at the left shows currently active files. It also indicates that the admin server is available to carry out deployment operations at its assigned port number.

To do the packaging, complete the following steps:

1. Select File -> New -> Web Component.

The wizard's Introduction appears.

**2. Click Next to display the Wizard's main screen.**

You will create a new stand-alone WAR module, so accept the default selection for the WAR Location panel.

With the `hello.war` file open, the default WAR File location is shown as `install_dir/samples/quickstart/`. With no WAR files open, the field is blank. Either way, click the Browse button to specify the name of the WAR file to create. (“WAR File” must specify the full path to a WAR file.)

- 3. If you are not already there, navigate to `install_dir/samples/quickstart`.**
- 4. Enter the filename as `hello2` and click “Create Module File” to create the WAR file and return to the Wizard.**

The Wizard now shows WAR Name as `hello2`. WAR Display Name is the title that appears in administration utilities like `deploytool` and the Admin Console. It defaults to the same name as the WAR file in the Wizard, but it can be different.

- 5. In Context Root enter `/hello2` to define the URL for the web application as `http://server:port/hello2`. In most production systems, of course, the Application Server software uses a default port, so the port number is not part of the URL that users type.**
- 6. Click Edit Contents to add the files that make up the application.**  
The Edit Content dialog appears.  
You can press the Control key and click multiple files.
- 7. Navigate to `install_dir/samples/quickstart`.**

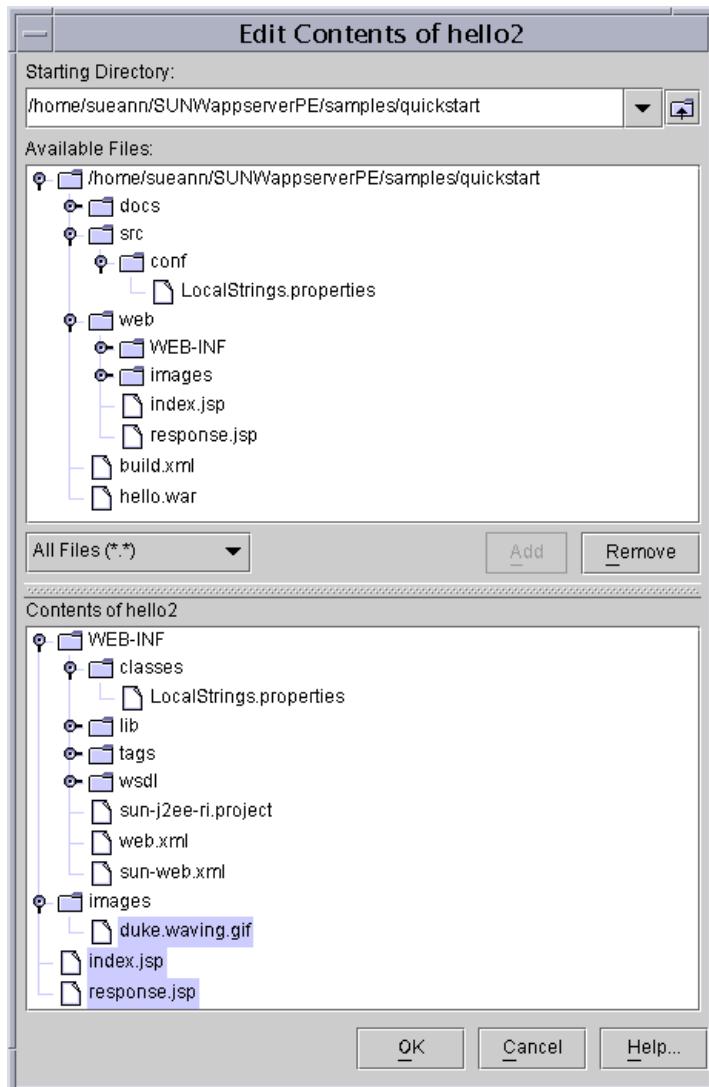
---

**NOTE** You can use the drop-down list of shortcuts.

---

- 8. In the lower panel, “Contents of `hello2`,” click the WEB-INF node to expand it.**
- 9. In the upper panel, “Available Files,” under `src/conf`, click the file `LocalStrings.properties` and drag and drop it on the WEB-INF/classes directory in the lower panel.**  
This adds the `LocalStrings.properties` file under the WEB-INF/classes directory.
- 10. In the web/ directory, select the `images/` directory and the `index.jsp` and `response.jsp` files.**

11. Click Add. The dialog shows that the selected files have been added.



12. Click OK to return to the Wizard's WAR File definition window.  
13. Click Next to complete the process of defining the WAR file.

The Choose Component Type dialog appears.

14. Select JSP Page and click Next.

The Component General Properties screen appears.

15. From the drop-down list for “JSP Filename,” select `index.jsp`.

This selection populates the other fields with default values based on the name.

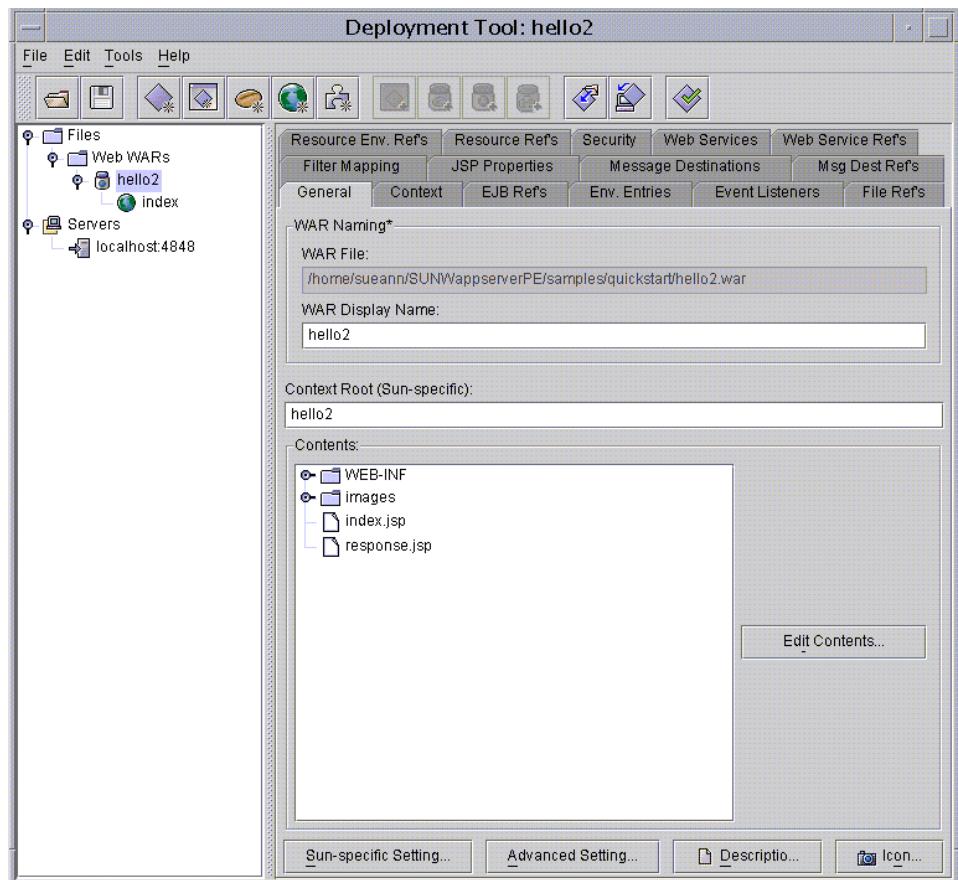
16. Click Next and read the notes about the next steps you can take.

17. Click Finish to end the Wizard and return to the deploytool screen.

You have just packaged a simple application. In *The J2EE 1.4 Tutorial*, you can learn how to package more complex applications, but the general outline of the process is similar. Now that it is packaged, deploying the application is a simple operation. You will do that next.

### Deploy the Packaged Application

The deploytool utility now shows the `hello2` application, and has it selected.



To deploy the application, complete the following steps:

1. From the toolbar, choose the Tools option and then Deploy.  
The security dialog appears.
2. Supply the admin user name and password. (If you chose “Don’t prompt for username” during the installation, these fields are already filled in.)
3. Click OK.

The server now starts loading the application. A dialog appears to inform you of the server’s progress, so that you know when the deployment operation has completed. Click Close to close the dialog.

4. Verify that the application is running by going to this URL:  
`http://localhost:8080/hello2`.

You should now see the hello application's first page, which provides a greeting from Duke, asking you to fill in your name.

5. Type in your name, and click Submit.

You should now see the customized response, giving you a personal `Hello`.

## Deploying from a Development Directory

You can deploy an application directly from a development directory, if the appropriate directory hierarchy and deployment descriptors have been created. Because the process is complex for non-trivial applications, it is a procedure that is recommended only for advanced users, or simple applications. But when it is feasible to do so, deploying directly from a directory, instead of using the `deploytool` utility, can speed up the development cycle.

The process can be performed interactively by using the Admin Console, or by using `asadmin deploydir` from the command line or in a script. The steps for the command line are presented here, since most developers want to automate the procedure in command scripts.

To deploy from a development directory, follow these steps:

1. In the `install_dir/samples/quickstart/` directory, create a new directory called `build/`.
2. Unzip the contents of the `hello.war` file into the `build/` directory.

Preserve the directory structure of the `hello.war` file when you unzip it. It is a template for the kind of file structure you need to deploy directly from a directory.

3. If you already have an application `hello` deployed, undeploy it now using the command `asadmin undeploy hello`.
4. Issue the following command to deploy the application:

```
asadmin deploydir install_dir/samples/quickstart/build
```

---

**NOTE** Specify the full path to the `build/` directory.

If you did not install the Application Server with the “Don’t prompt” option, also specify `--user username --password password`, where *username* is the admin user name and *password* is the admin password.

---

5. Verify that this message appears to confirm that the redeployment succeeded:  
Command `deploydir` executed successfully.
6. Verify that the application is running by going to this URL:  
`http://localhost:8080/hello`.

## Using Command-line Tools

The Application Server software offers a variety of command-line tools for performing administrative functions, in addition to the Admin Console. This section explains what command-line tools are available.

To launch a tool, type the name of the tool in a command window. This table lists tools by name in the first column and describes them in the second column.

Name of Tool	Description
<code>appclient</code>	Launches the Application Client Container and invokes the client application packaged in the application Java archive (JAR) file.
<code>asadmin</code>	Application Server admin tool for configuring the Application Server software.
<code>asant</code>	Launches the Jakarta Ant tool, so that you can automate repetitive development and deployment tasks.
<code>asupgrade</code>	Application Server admin tool for upgrading the Application Server software.
<code>capture-schema</code>	Extracts schema information from a database and produces a schema file that the server can use for Container Managed Persistence (CMP).
<code>deploytool</code>	Creates deployment descriptors for J2EE applications, packages them into JAR, Web archive (WAR), and enterprise archive (EAR) files, and deploys them on the server.  This tool also provides a graphical user interface.

---

Name of Tool	Description
jspc	Compiles JSP pages.
package-appclient	Packages the application client container libraries and JAR files.
verifier	Validates the J2EE deployment descriptors with the DTDs. This tool also provides a graphical user interface. To see the GUI, specify the <code>-u</code> option. Some Windows systems launch a driver verifier utility with the same name. To launch the Application Server verifier, you must be in the <code>install_dir/bin</code> directory.
wscompile	Takes the service definition interface and generates the client stubs or server-side skeletons for JAX-RPC; or generates a Web Services Description Language (WSDL) description for the provided interface.
wsdeploy	Generates an implementation-specific, ready-to-deploy WAR file for web services applications that use JAX-RPC.

To configure the Windows environment to use the command-line tools:

1. From the Explorer window or desktop, right click My Computer.
2. Choose Properties to display the System Properties dialog.
3. Click the Advanced tab.
4. Click Environment Variables.

In the User variables section,

If a PATH variable exists:

Click Edit.

In Variable Value, enter the path to the server's bin directory, separated from other entries by a semicolon:

`install_dir/bin;other_entries`

If a PATH variable is not present:

Click New.

In Variable Name, type PATH.

In Variable Value, type the path to the server's bin directory:

`install_dir/bin`

Add an environment variable AS\_ADMIN\_USER and set it to the Administrative User Name that you assigned during installation.

5. Click OK to commit the change and close the remaining open windows.

# Where to Go Next

Other resources for learning about and using Application Server are available. They include:

- The server's welcome page at <http://localhost:8080> or `install_dir/docs/about.html`.

See this document for the latest information on what is new, and pointers to tutorials and other educational services.
- *Sun Java System Application Server Platform Edition 8.1 2005Q1 Release Notes* on at <http://docs.sun.com/>.

See this document for information regarding this release.
- *Sun Java System Application Server Platform Edition 8.1 2005Q1 Administration Guide* at <http://docs.sun.com/>.

This document contains the reference for performing administrative functions using the Admin Console.
- *Sun Java System Application Server Platform Edition 8 2005Q1 Utility Reference* (man pages).

See this document for reference information on Application Server's command-line utilities, such as `asadmin`. Go to <http://docs.sun.com/>.
- *The J2EE 1.4 Tutorial*

This tutorial covers the process for building and deploying Java™ 2 Platform, Enterprise Edition (J2EE™ platform) applications. Go to <http://docs.sun.com/>.
- Java™ BluePrints guidelines for the Enterprise

See the Java BluePrints guidelines for a comprehensive set of examples that demonstrate operations of the Application Server software and that can be used as application templates. Go to <http://java.sun.com/blueprints>.

# Documentation Conventions

This section describes the types of conventions used throughout this guide.

## General Conventions

The following general conventions are used in this guide:

- **File and directory paths** are given in UNIX® format (with forward slashes separating directory names). For Windows versions, the directory paths are the same, except that backslashes are used to separate directories.
- **URLs** are given in the format:

`http://server.domain/path/file.html`

In these URLs, *server* is the server name where applications are run; *domain* is your Internet domain name; *path* is the server's directory structure; and *file* is an individual filename. Italic items in URLs are placeholders.

- **Font conventions** include:
  - The `monospace` font is used for sample code and code listings, API and language elements (such as function names and class names), file names, path names, directory names, and HTML tags.
  - *Italic* type is used for code variables.
  - *Italic* type is also used for book titles, emphasis, variables and placeholders, and words used in the literal sense.
  - **Bold** type is used as either a paragraph lead-in or to indicate words used in the literal sense.
- **Installation root directories** for most platforms are indicated by *install\_dir* in this document.

By default, the location of *install\_dir* on **most** platforms is:

- Solaris and Linux file-based installations, non-root user:  
*user's home directory*/SUNWappserver
- Solaris and Linux file-based installations, root user:  
`/opt/SUNWappserver`
- **Windows**, all installations:

*system drive*: \Sun\AppServer

- **Domain root directories** are indicated by *domain\_dir* in this document, which by default is an abbreviation for the following:

*install\_dir*/domains/

However, for package-based installations, the directory containing all the domains can be changed from *install\_dir*/domains/ to another directory during installation. In configuration files, you may see *domain\_dir* represented as follows:

`${com.sun.aas.instanceRoot}`

