Sun Java System Application Server Platform Edition 8.2 Quick Start Guide

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

Part No: 819–4710

Copyright 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 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 U.S. patents or pending patent applications in the U.S. and in other countries.

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 other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, J2EE, JDBC, NetBeans, Java, and Solaris 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 an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ 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 publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or 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 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle 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 brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

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

Certaines composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, J2EE, JDBC, NetBeans, Java et Solaris 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.

L'interface d'utilisation graphique OPEN LOOK et Sun 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 licence 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 cette publication et les informations qu'il contient sont régis par la legislation 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 chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations 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 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 legislation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement designés, 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.

070802@18135

Quick Start Guide

Welcome to the *Sun Java™ System Application Server Quick Start Guide*. This guide describes basic steps for starting the Sun Java System Application Server (henceforth referred to as 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 interested in learning the basics about Application Server software.

This document contains information about the following topics:

- "Starting and Verifying the Server" on page 3
- "Packaging and Deploying Applications" on page 8
- "Using Command-Line Tools" on page 12
- "Where to Go Next" on page 14

For information about conventions used in this document, see "Documentation Conventions" on page 14.

Starting and Verifying the Server

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

- "Starting the Application Server" on page 4
- "Confirming That the Server Is Running" on page 4
- "Deploying the Hello Application" on page 5
- "Using the Admin Console" on page 7

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)

Starting the Application Server

You start the server by starting the domain. A *domain* provides authentication and administration for the server instance that belongs to it. Starting the domain starts the server instance in the domain. A default domain called *domain1* is created when you install Application Server.

Note – A default domain does not get created in certain distributions (such as the Application Server bundled with Solaris 10 distributions). If a default domain has not been created, use the create-domain command to create an administrative domain. See `create-domain`(1).

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 and Linux	Windows
1. Add the *install-dir*`/bin/` directory to the `PATH` environment variable. 2. Start the server by entering this command from the *install-dir*: `asadmin start-domain domain1`.	From the Start menu, choose Programs ⇒ Sun Microsystems ⇒ Application Server PE⇒ Start Default Server.

Confirming That the Server Is Running

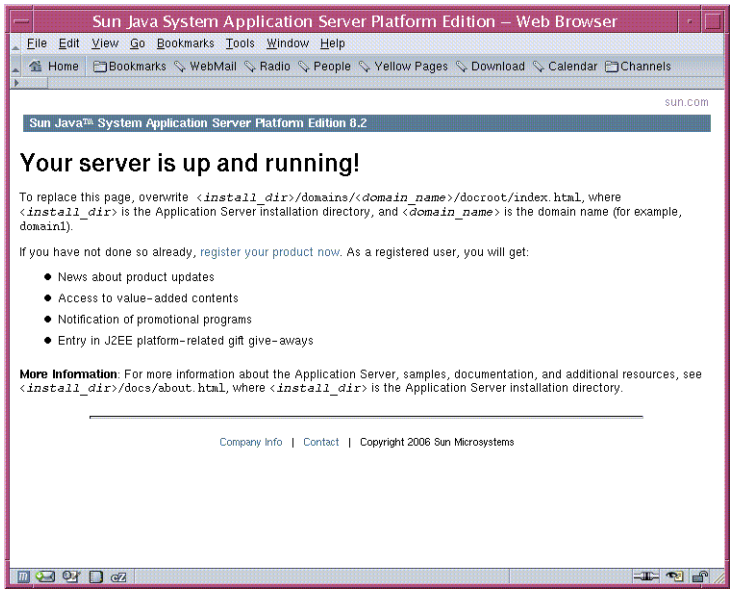
When the server has completed startup, this message appears: `Domain domain1 is ready to receive client requests. Additional services are being started in the background.`

To verify that the server is running on your system, access this URL: `http://localhost:8080`.

The URL has two parts:

- The name *localhost* refers to the system on which the browser is currently running. 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 to display 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 might need to check your server and browser settings, as described in the *Sun Java System Application Server Platform Edition 8.2 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 and Linux	Windows
From the *install-dir*`/bin` directory, type this command: `asadmin stop-domain`	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*. The Hello web 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`**.**

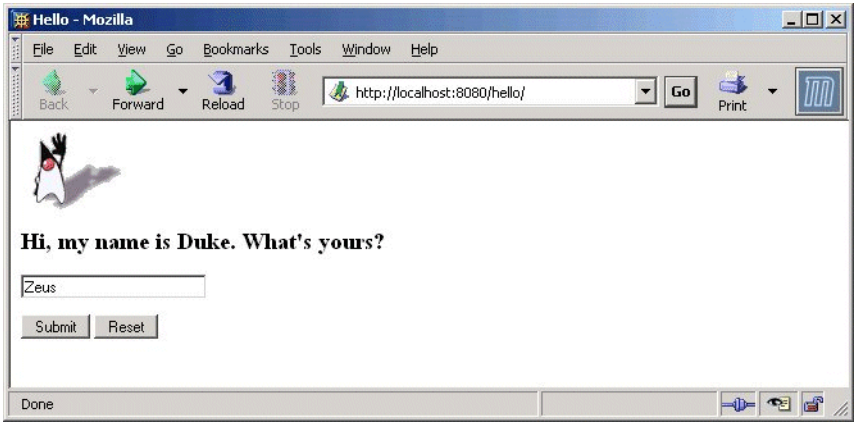
2 **Copy it to** *install-dir*`/domains/domain1/autodeploy/`**.**

Tip – On Windows, you can create a shortcut to the `autodeploy` directory on your desktop, 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.

3 **Access this URL to visit the application page:** `http://localhost:8080/hello`**.**

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



4 **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`. For more information, see the `deploy` command documentation in the *Sun Java System Application Server Platform Edition 8.2 Reference Manual*.

Other deployment methods are discussed later in the Guide in "To Deploy From the Admin Console" on page 10, and "To Deploy From a Development Directory" on page 11. It is also possible to dynamically reload changes to your application. For more information, see "To configure dynamic reloading" in *Sun Java System Application Server Platform Edition 8.2 Administration Guide*.

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 accomplish the following:

- Deploy and undeploy applications
- Enable and disable applications
- Identify which applications are currently running
- Configure Java DataBase Connectivity (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 8.2 Administration Guide*.

▼ To Log In to the Admin Console

1 **Access `http://localhost:4848`.**

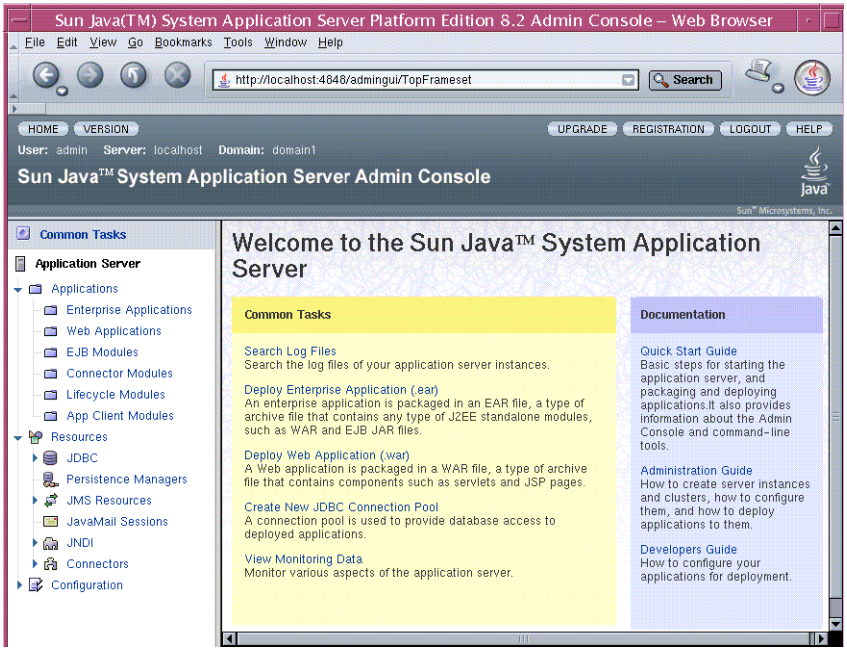
- The *localhost* variable is the name of the system that the browser is currently running on. If Application Server software is running on another system, substitute the name of that system for *localhost*.
- `4848` is the Admin Console's default port number. If you changed the port number during the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

Tip – Windows users can also use the Start menu option: Programs ⇒ Sun Microsystems ⇒ Application Server PE ⇒Admin Console.

2 **When the Login window appears, enter the admin user name and password.**

3 **Click Log In.**

When the Admin Console appears, it looks like this:



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 pane. When the Log Viewer window appears, you can see status messages from the server and output from your J2EE application.

Next Steps You have successfully completed the first section of the *Quick Start Guide*. You started the Application Server, deployed the Hello application, and viewed the Admin Console. To continue with the Quick Start tasks, go on to "Packaging and Deploying Applications" on page 8.

Packaging and Deploying Applications

Before applications can be deployed, they must be packaged into Java ARchive (JAR), Web ARchive (WAR), or Enterprise ARchive (EAR) files. The packaged application includes deployment descriptors, which 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. This section describes the following techniques for packaging and deploying applications:

- "Using NetBeans IDE to Develop and Deploy Applications" on page 9
- "To Package the Hello Web Application Using asant" on page 9

- "To Deploy From the Admin Console" on page 10
- "To Deploy From a Development Directory" on page 11

For more information, see "Getting Started With Web Applications" in *The J2EE 1.4 Tutorial*.

Using NetBeans IDE to Develop and Deploy Applications

NetBeans™ IDE is an Integrated Development Environment you can use to create, package, and deploy complex Java 2 Platform, Enterprise Edition (J2EE platform) applications. The NetBeans IDE is available for download from `http://www.netbeans.org`. NetBeans IDE integrates the development environment with the deployment environment, and automates many steps for producing J2EE applications. Once you register the Application Server with the NetBeans IDE, you can easily deploy applications, make changes, and redeploy them.

For more information on NetBeans IDE, see the documentation at: `http://www.netbeans.org`.

▼ To Package the Hello Web Application Using asant

If applications are not packaged with a tool such as NetBeans IDE, you can use the `asant` utility shipped with the product. The `asant` utility is used to automate repetitive deployment tasks. It is a shell script that initializes the Application Server environment and launches the Jakarta Ant tool. For more information on the `asant` utility, see the *Sun Java System Application Server Platform Edition 8.2 Reference Manual*.

The Hello application deployed earlier was already packaged into a WAR file. This section covers packaging the application's components into another WAR file. The `build.xml` file included with the Hello application contains the build targets.

1 **If you have not already added** *install-dir*`/bin` **to your PATH environment variable, do so now.**

2 **Go to the** *install-dir*`/samples/quickstart/` **directory.**

3 **Type** `asant war`**.**

This command assembles the WAR file for the application in *install-dir*`/samples/quickstart/build/assemble/war`.

▼ To Deploy From the Admin Console

Now that you have packaged the Hello application, you can deploy it. The following section discusses deploying the Hello application using the Admin Console. You can also deploy using `asant`. For more information, see *Sun Java System Application Server Platform Edition 8.2 Developer's Guide*.

1. **Access** `http://localhost:4848/`**.**
   - *localhost* is used if the both the browser and the Application Server are running on the same system. If the Application Server is running on another system, substitute the name of that system in the URL.
   - 4848 is the Admin Console's default port number. If you changed the port number during the installation, use that number instead.
2. **Enter the admin user name and password.**
3. **In the left pane, click the Applications node to expand it.**
4. **Click Web Applications.**
5. **If you already have a Hello application deployed, undeploy it now by selecting the checkbox next to it and clicking Undeploy.**
6. **To deploy the newly assembled Hello application, on the Web Applications page, click Deploy.**
7. **Select the Specify a package file to upload to the Application Server radio option and click Browse.**
8. **Navigate to the** *install-dir*`/samples/quickstart/build/assemble/war` **directory and select** `hello.war`**.**
9. **Click Next.**
10. **On the Deploy Web Module page, click OK.**
    The application appears in the Web Applications list.
11. **To verify that it was deployed properly, click Launch.**
    The application's first page appears.

▼ To Deploy 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 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, because most developers want to automate the procedure in command scripts.

Note – Before using `asadmin` on Windows, see "To Configure the Windows Environment to Use the Command-Line Tools" on page 13.

---

1. **In the *install-dir*`/samples/quickstart/` directory, create a new directory called `hello2/`.**

2. **Unzip the contents of the `hello.war` file into the `hello2/` 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 a Hello application deployed, undeploy it now using the command `asadmin undeploy hello` at the command line.**

4. **Issue the following command to deploy the application:**

   ```
   asadmin deploydir install-dir/samples/quickstart/hello2
   ```

   Note – Specify the full path to the `hello2/` directory.

   If you installed the Application Server with the "Prompt for Administration User Name" option, also specify the option `--user` *username*, where *username* is the admin user name. You are prompted for the password.

   The following message appears: `Command deploydir executed successfully.`

5. **Verify that the application is running by going to this URL: `http://`*localhost*`:8080/hello`.**

Next Steps You have completed the second section of the *Quick Start Guide*. See "Using Command-Line Tools" on page 12 for more information on the command-line tools available in Application Server. See "Where to Go Next" on page 14 for information on other resources for learning about Application Server.

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 in the first column and describes them in the second column.

Name of Tool	Description
`appclient`	Launches the Application Client Container and invokes the client application packaged in the application Java archive (JAR) file.
`asadmin`	Application Server administration tool for configuring the Application Server software.
`asant`	Launches the Jakarta Ant tool, so that you can automate repetitive development and deployment tasks.
`asupgrade`	Application Server administration tool for upgrading the Application Server software.
`capture-schema`	Extracts schema information from a database and produces a schema file that the server can use for Container Managed Persistence (CMP).
`deploytool`	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.
`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 `-u` option. Some Windows systems launch a driver verifier utility with the same name. To launch the Application Server verifier, you must be in the *install-dir*`/bin` 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.**
5. **In the User variables section, add or update the `PATH` variable.**
   - **If a `PATH` variable exists:**
     a. **Click Edit.**
     b. **In Variable Value, enter the path to the server's `bin` directory, separated from other entries by a semicolon. For example:**
        install-dir`/bin;`*other_entries*
   - **If a `PATH` variable is not present:**
     a. **Click New.**
     b. **In Variable Name, type `PATH`.**
     c. **In Variable Value, type the path to the server's `bin` directory:**
        install-dir`/bin`
6. **In the User variables section, add the environment variable `AS_ADMIN_USER` and set it to the Administrative User Name that you assigned during installation.**
7. **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, including the following:

- The server's Installation Complete page at *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 8.2 Release Notes*

  See this document for late-breaking changes and other information regarding this release.

- *Sun Java System Application Server Platform Edition 8.2 Administration Guide*

  See this guide for a reference for performing administrative functions using the Admin Console.

- *Sun Java System Application Server Platform Edition 8.2 Reference Manual* (man pages)

  See this document for reference information on Application Server's command-line utilities, such as `asadmin`.

- *The J2EE 1.4 Tutorial* (`http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html`)

  See this document for a tutorial that covers the process for building and deploying Java 2 Platform, Enterprise Edition (J2EE™ platform) applications.

- Java BluePrints (`http://java.sun.com/blueprints`) guidelines for the Enterprise

  See this site for a comprehensive set of examples that demonstrate operations of the Application Server software and that can be used as application templates.

- NetBeans.org (`http://www.netbeans.org`)

  See this site to download the NetBeans IDE and view the documentation, including installation instructions and Quick Start Guides.

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 following 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. Emphasis items in URLs are placeholders.

- **Font conventions** include the following:
  - 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.
  - *Emphasis* type is used for code variables.
  - *Emphasis* 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}`