How to use J2EE default server

By *Hamid Mosavi-Porasl*

Quick start for Sun Java System Application Server Platform J2EE

1. start default server
2. login in with Admin userId and password, i.e. myY+UserId & myPasswd
3. start database with command:
'asadmin start-database'

Server root

install_dir>/domains/*<domain_name>*/docroot/index.html, where
<install_dir>

<install_dir>/docs/about.html

Overview of Deployment

A typical development process for J2EE applications contains two steps:

1.   Create JavaServer Pages (JSP) components that handle the presentation logic of the application
and build servlet or Enterprise JavaBeans (EJB) components to manage the application's business
logic.

2.   Assemble those components into a single unit that can be deployed to a J2EE platform-compliant
application server, such as Application Server PE 8.

The deployable unit can be either of the following:

o   **J2EE module** -- A collection of J2EE components that reside in the same container. The
modules can be a Web archive (WAR), an EJB Java archive (JAR), a resource adapter archive
(RAR), or an application client JAR.
o   **J2EE application** -- A collection of J2EE modules along with application deployment
descriptors in the form of EAR files.

This article shows you the five ways in which to deploy J2EE applications or modules to Application Server
PE 8, some of which help reduce the length of the develop-deploy-debug cycle. However, this article does
not address the structure of the deployable units. For details on that structure, see *The J2EE 1.4 Tutorial*
and *Chapter 3, "Assembling and Deploying J2EE Applications"* in the *Sun Java System Application Server
Platform Edition 8 Developer's Guide.*

Method 1: Command Line

Application Server PE 8 includes a command-line interface (CLI), asadmin (short for Application Server
Administration), which, in addition to performing administration tasks on Application Server PE 8, (such as
creating and managing resources), can also deploy units. asadmin resides in the
Application_Server_install_dir/bin directory.

Before deploying with asadmin, ensure that your J2EE module or application is packaged according to the
specification. This method is useful for situations in which you desire to deploy from within scripts or when
your applications are packaged and deployment-ready.

For example, to deploy a WAR called hello.war and an EAR called test.ear, type:

```
% asadmin deploy --user admin_user --password admin_password hello.war
% asadmin deploy --user admin_user --password admin_password test.war
```

To undeploy, execute the `asadmin undeploy` command.

Note - If you elected during installation to have Application Server PE 8 remember the administrator password, you need not add the user and password options when you run `asadmin`.

`asadmin` offers other options, for example, compiling all JSP pages during deployment and keeping the application in a disabled state after deployment. Here is the syntax:

```
asadmin deploy --user admin_user [--password admin_password]
        [--host localhost] [--port 4848] [--passwordfile filename]
        [--secure|-s] [--virtualservers virtual_servers]
        [--type application|ejb|web|connector]
        [--contextroot context_root] [--force=true]
        [--precompilejsp=false]
        [--verify=false] [--name component_name] [--upload=true]
        [--retrieve local_dirpath] [--instance instance_name] filepath
```

For details on the options, see the `deploy`(1) man page or command-line help.

Method 2: Administration Console

You can also deploy applications or modules from the Administration Console of Application Server PE 8.

Reminder - For this deployment method to work, your application or module must be completely and properly packaged.

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`.

1. **Click http://localhost:4848/asadmin.**
   o *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.
   o 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 Package file to upload to the Application Server 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.**

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.

---

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.**

    If you did not install the Application Server with the "Don't prompt" option, also specify the option `--user` *username*, where *username* is the admin user name. You are prompted for the administration password.

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

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

 now check again via **http://localhost:4848/asadmin.** Hello2 **should be deployed, launch it.**

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 EE applications. The NetBeans IDE 5.0 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 Java EE 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 using NetBeans IDE with Application Server, see the documentation at: http://www.netbeans.org.

Structure of a WAR directory

```
WEB-INF/ -
            | -----Classes
            | -----web.xml
            | -----sun-web
Images/

META-INF/ –
            |----MANIFEST.MF
```