

---

---

# SilverStream eXtend Workbench

Tutorials

Version 4

July 2002

SilverStream<sup>®</sup>

---

Copyright ©2002 SilverStream Software, Inc. All rights reserved.

SilverStream software products are copyrighted and all rights are reserved by SilverStream Software, Inc.

SilverStream and jBroker are registered trademarks and SilverStream eXtend is a trademark of SilverStream Software, Inc.

Title to the Software and its documentation, and patents, copyrights and all other property rights applicable thereto, shall at all times remain solely and exclusively with SilverStream and its licensors, and you shall not take any action inconsistent with such title. The Software is protected by copyright laws and international treaty provisions. You shall not remove any copyright notices or other proprietary notices from the Software or its documentation, and you must reproduce such notices on all copies or extracts of the Software or its documentation. You do not acquire any rights of ownership in the Software.

Jakarta-Regexp Copyright ©1999 The Apache Software Foundation. All rights reserved. Ant Copyright ©1999 The Apache Software Foundation. All rights reserved. Xalan Copyright ©1999 The Apache Software Foundation. All rights reserved. Xerces Copyright ©1999-2000 The Apache Software Foundation. All rights reserved. Jakarta-Regexp, Ant, Xalan and Xerces software is licensed by The Apache Software Foundation and redistribution and use of Jakarta-Regexp, Ant, Xalan and Xerces in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "The Jakarta Project", "Jakarta-Regexp", "Xerces", "Xalan", "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [apache@apache.org](mailto:apache@apache.org) <<mailto:apache@apache.org>>. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of The Apache Software Foundation. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright ©1996-2000 Autonomy, Inc.

Copyright ©2000 Brett McLaughlin & Jason Hunter. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution. 3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [license@jdom.org](mailto:license@jdom.org) <<mailto:license@jdom.org>>. 4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management ([pm@jdom.org](mailto:pm@jdom.org) <<mailto:pm@jdom.org>>). THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Sun Microsystems, Inc. Sun, Sun Microsystems, the Sun Logo Sun, the Sun logo, Sun Microsystems, JavaBeans, Enterprise JavaBeans, JavaServer Pages, Java Naming and Directory Interface, JDK, JDBC, Java, HotJava, HotJava Views, Visual Java, Solaris, NEO, Joe, Netra, NFS, ONC, ONC+, OpenWindows, PC-NFS, SNM, SunNet Manager, Solaris sunburst design, Solstice, SunCore, SunNet, SunWeb, Sun Workstation, The Network Is The Computer, ToolTalk, Ultra, Ultracomputing, Ultraserver, Where The Network Is Going, SunWorkShop, XView, Java WorkShop, the Java Coffee Cup logo, Visual Java, and NetBeans are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

The software licensed hereby contains modified Sun NetBeans software which is available, together with the Sun Public License pursuant to which such NetBeans software may be used, at <http://www.silverstream.com/workbenchdownload>. Terms of this Agreement that differ from the terms of the Sun Public License are offered solely by SilverStream.

SilverStream eXtend Workbench software contains Sun NetBeans software that has been modified by SilverStream. The source code for such software may be found at <http://www.silverstream.com/workbenchdownload> together with the Sun Public License that governs the use of such modified software. The Original Code is NetBeans. The Initial Developer of the Original Code is Sun Microsystems, Inc. Portions Copyright 1997-2000 Sun Microsystems, Inc. All Rights Reserved. The Contributor to Covered Code is SilverStream Software, Inc.

IBM Jikes™ and Bean Scripting Framework (BSF) Copyright ©2001, International Business Machines Corporation and others. All Rights Reserved. This software contains code in executable form obtained pursuant to, and the use of which is subject to, the IBM Public License, a copy of which may be obtained at <http://oss.software.ibm.com/developerworks/opensource/license10.html>. Source code for Jikes™ is available at <http://oss.software.ibm.com/developerworks/opensource/jikes/>. Source code for BSF is available at <http://oss.software.ibm.com/developerworks/projects/bsf>.

Copyright ©2001 Extreme! Lab, Indiana University License. <http://www.extreme.indiana.edu>. Permission is hereby granted, free of charge, to any person obtaining a copy of the Indiana University software and associated Indiana University documentation files (the "IU Software"), to deal in the IU Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the IU Software, and to permit persons to whom the IU Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the IU Software. THE IU SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE IU SOFTWARE OR THE USE OR OTHER DEALINGS IN THE IU SOFTWARE.

Graph Layout Toolkit and Graph Editor Toolkit ©1992 - 2001 Tom Sawyer Software, Oakland, California, All Rights Reserved.

This Software is derived in part from the SSLava™ Toolkit, which is Copyright ©1996-1998 by Phaos Technology Corporation. All Rights Reserved.

SearchServer © 2000 Hummingbird Communications, Inc.

Copyright © 1994-2002 W3C® (Massachusetts Institute of Technology, Institut National de Recherche Informatique et en Automatique, Keio University), all Rights Reserved. <http://www.w3.org/consortium/legal>. This W3C work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions: Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make: 1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work. 2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body of any redistributed or derivative code: "Copyright © [date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>" 3. Notice of any changes or modifications to the W3C files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.) THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION. The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.



---

# Contents

---

## **Chapter 1 About the Tutorials 1**

Availability of the tutorials 1

Software requirements 1

## **Chapter 2 Workbench Deployment Instructions 3**

Purpose of these instructions 3

Setting up your deployment environment 3

Defining a server profile 4

Deploying to the SilverStream eXtend Application Server 8

Deploying to the Sun Reference Implementation server 13

Deploying to the Jakarta Tomcat server 18

Deploying to the BEA WebLogic server 20

Deploying to the IBM WebSphere server 24

Deploying to the Oracle9iAS server 28



---

# 1 About the Tutorials

---

SilverStream eXtend Workbench provides tutorials to help you get started with Workbench projects and different types of J2EE applications.

## Availability of the tutorials

SilverStream eXtend Workbench includes two tutorials:

- **Working with Web Services**

Illustrates how to create a Workbench project for a Web Service and use the Web Service Wizard to generate the Java and XML for the Web Service. It also describes how to create a J2EE WAR client application for the Web Service and demonstrates the Registry Browser and WSDL Editor.

- **Developing a Web Application**

Illustrates how to create a Workbench project for a WAR and how to use the Model-View-Controller (MVC) architecture to process user actions, data, and forms.

The tutorial has two parts:

- **Part I** shows you how to use Workbench to develop the sample Web application and how the MVC architecture is used in a Struts application.
- **Part II** teaches you how to write code in a Struts Web application that accesses data via the J2EE application server's connection pool and how to define forms that display that data.

## Software requirements

**SilverStream eXtend Workbench** To develop the tutorial applications, you need SilverStream eXtend Workbench. If you don't have it already (you are reading this from the SilverStream Web site), you can get the Workbench software at the Workbench Web page at <http://www.silverstream.com/workbench>. You will be asked to register, and then you can download the software.

**A J2EE application server and a DBMS** To run the tutorial applications, you need a J2EE application server for deploying the application. Some tutorials also need a DBMS for data storage.

- If you already have a J2EE application server, you can deploy the standards-based J2EE WARs to your application server using either the Workbench deployment commands or your server's deployment tools.
- If you don't have the required application server and/or DBMS, you can download the trial version of the SilverStream eXtend Application Server (which includes the Cloudscape DBMS) from [www.silverstream.com/downloads](http://www.silverstream.com/downloads) or the Workbench Web page mentioned above. For the tutorials, all you need is the Lite Edition (J2EE server and Cloudscape).



---

# 2

## Workbench Deployment Instructions

---

### Purpose of these instructions

SilverStream eXtend Workbench supports deployment to several different servers. It would make each tutorial too long to cover all those servers in detail in each situation. These instructions provide the detail, and the individual tutorials remind you of the commands to use and the information you need to supply.

You can refer to these instructions whenever you need server-specific information to supplement the deployment instructions in a tutorial.

**NOTE** Refer to these instructions *after* you have created your tutorial project and are ready to deploy it.

### Setting up your deployment environment

Using Workbench, you can create J2EE applications for any J2EE application server. For several popular application servers, Workbench supports deployment directly with a Deploy Archive menu item. For other J2EE application servers, you can build your archive in Workbench and use your server's deployment tools to deploy. As long as you stick to J2EE standards and avoid server-specific code, archives built by Workbench are completely J2EE-compatible and can be deployed to any J2EE server.

**NOTE** Be sure to check the *Release Notes* for additional deployment information.

Setting up and deploying consists of these tasks:

1. Defining a server profile
2. Preparing the runtime deployment information expected by your server
3. Specifying deployment settings
4. Deploying the archive

The following sections provide instructions for these tasks:

- Defining a server profile
- Deploying to the SilverStream eXtend Application Server
- Deploying to the Sun Reference Implementation server
- Deploying to the Jakarta Tomcat server
- Deploying to the BEA WebLogic server

- Deploying to the IBM WebSphere server
- Deploying to the Oracle9iAS server

**Deploying to other servers** For other servers, use your own server tools to deploy the archive you've built.

## Defining a server profile

A server profile stores information about your application server, so you don't have to provide the same information over and over again. You can use the same server profile for all the tutorial exercises.

If you've already created a profile for your server, skip to the deployment instructions for your server (listed in "Setting up your deployment environment" just above).

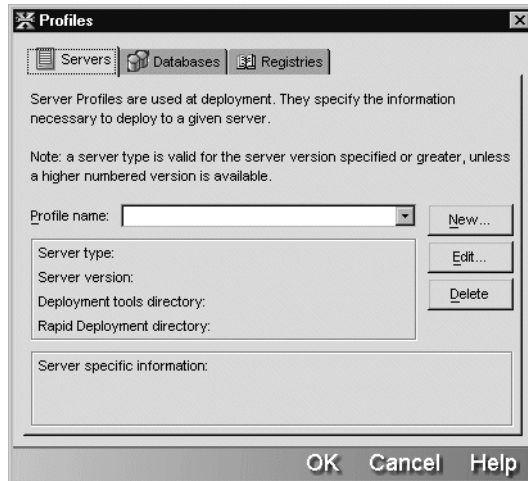
**Multiple profiles** If you want to deploy to more than one server, you should create a profile for each one. For SilverStream servers, you may also find it useful to define more than one profile for the same server—for example, if you want to deploy to different databases.



## EXERCISE 2-1: Create a server profile for your application server

In this exercise you will create a server profile.

1. Select **Edit>Profiles** from the menu.



2. On the **Servers** tab of the Profiles dialog, click **New**.

3. Fill in the Create a New Server Profile dialog with values like these (not all fields are present for all server types):

Option	Description
Profile name	Specify a name like <b>SilverStream 373 Tutorial</b> or <b>RI Tutorial</b>
Server type	<p>Choose the type that describes your server and version</p> <p>Server types are organized by brand and version number. The version number indicates the lowest version supported by a given server type. A server type is often valid for multiple subsequent versions as well.</p> <p>As a rule, you should select the server type for your brand that is closest to the target server's version, without being higher. For example, if your target is Version 3.7.5 of the SilverStream eXtend Application Server, the server type to select is SilverStream 3.7.4 or higher.</p>
Deployment tools directory	<p>Specify the directory that contains the server's deployment tools; typically, a path like <b>c:\SilverStream37\bin</b> or <b>c:\j2eesdk\bin</b></p> <p>If the server is located on another machine, you need either network access for running the tools or a copy of the tools in a local directory. For some servers, Workbench doesn't support remote deployment.</p>
Rapid deployment directory	Leave blank. Rapid deployment is not used in the tutorials.

- Specify server-specific information. For most servers, you specify only the server name and port number.

Option	Description
Server name	Specify the name of your server, which could be <b>localhost</b> (or the actual server name for your server)  If your server is not using the default port 80, specify <i>server-name:port-number</i>  For example, your server might be <b>localhost:8080</b>
Database name	(SilverStream only) Specify the name of a database on your server for the archive to be deployed into  <b>NOTE</b> The Proverbs Web application (the WAR tutorial application) has a database for the application's data. You can deploy any of the tutorial archives to that database. If you have set up the Cloudscape database provided with the tutorial files, specify <b>ProverbsCloud</b> . If you used the SQL script to create a database, specify its name. For database setup procedures, see Lesson 2 in the WAR tutorial.

A filled-in dialog for a SilverStream server looks something like this:



- Click **OK** to close both the Create a New Server Profile dialog and the Profiles dialog.

## Deploying to the SilverStream eXtend Application Server

To deploy to the SilverStream eXtend Application Server from Workbench, you create a server profile (described in “Defining a server profile” on page 4) and a deployment plan. Then you deploy the archive.

This is what you’ll do:

- EXERCISE 2-2: “Create a deployment plan for the SilverStream server”
- EXERCISE 2-3: “Specify deployment settings”
- EXERCISE 2-4: “Deploy the application to the SilverStream server”

**About the deployment plan** The deployment plan specifies actual values for names identified in the deployment descriptor. Before you can create the deployment plan, you must have built the archive and added appropriate items to the descriptor. If you change the descriptor later, Workbench modifies the plan to match when you open it in the editor.

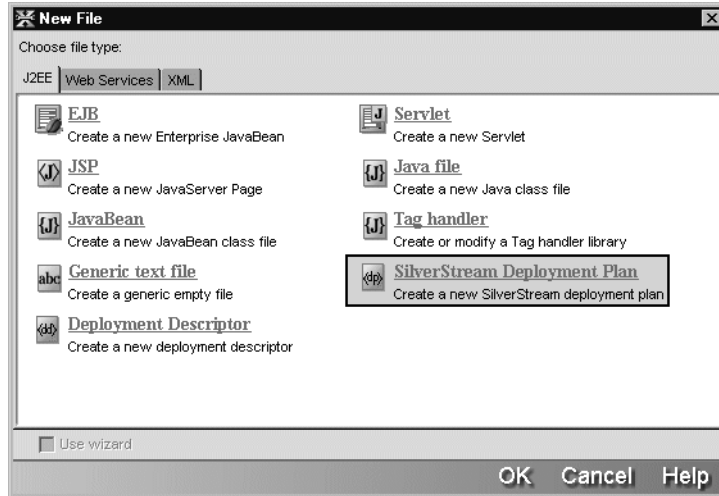


### **EXERCISE 2-2: Create a deployment plan for the SilverStream server**

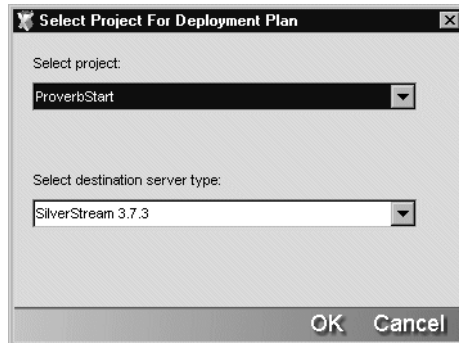
In this exercise you will create a SilverStream deployment plan.

1. If the SilverStream server isn’t running, start it now.
2. With your tutorial project open in Workbench, select **File>New** from the menu.

- In the New File dialog, select the **J2EE** tab; then select **SilverStream Deployment Plan** and click **OK**.

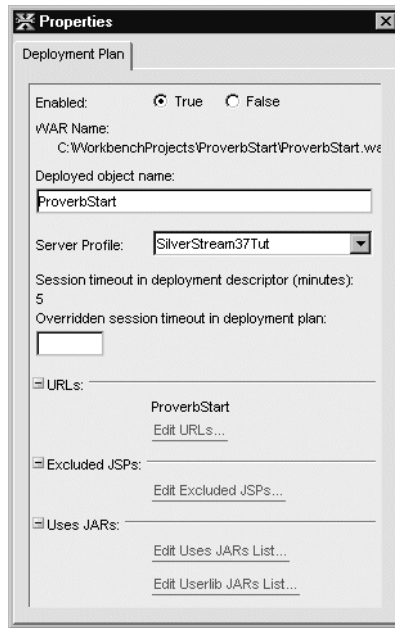


- In the **Select Project** text box of the Select Project For Deployment Plan dialog, accept the default project name—the only open project.
- In the **Select destination server type** list box, select the version of your SilverStream server. Then click **OK**.



- If Workbench displays the **Select Build Option** dialog, accept the defaults and click **OK**.  
The deployment plan displays in the Edit Pane. It includes a top-level element that corresponds with your project type. If there were elements in the deployment descriptor such as resource references, for each one you would see a corresponding element where you could provide its value.

7. Choose **File>Save** to save the deployment plan. In the Save As dialog, give the plan a name like *myprojectDeplPlan.xml*, where *myproject* is a name related to your tutorial project, and save it in the project's root directory.
8. Right-click the top-level archive element and select **Properties** from the popup menu. The property sheet displays properties of the deployment plan.



9. Set values on the property sheet using the values listed in your tutorial.
10. Save again and close the Deployment Plan Editor.

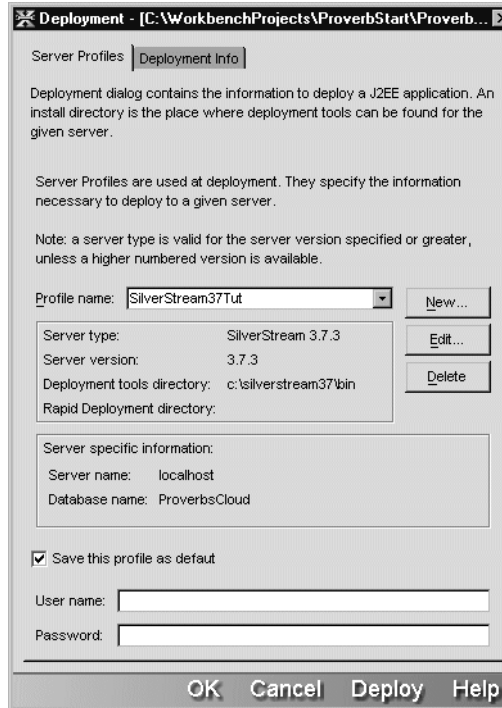


### EXERCISE 2-3: Specify deployment settings

In this exercise you will specify the deployment settings that will be used each time you deploy while you are developing and testing your tutorial application.

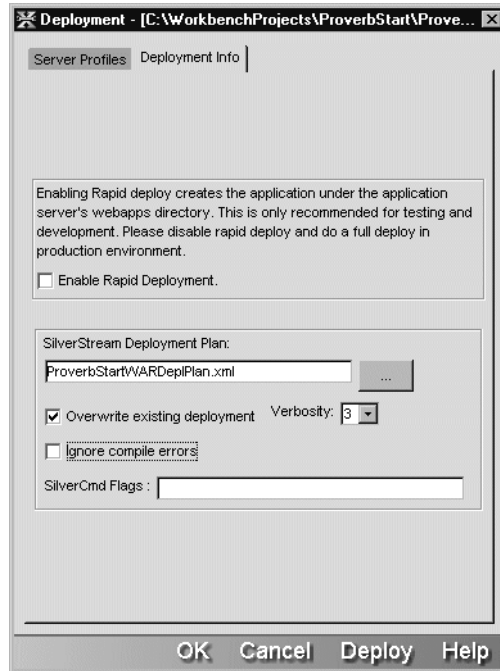


1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.



2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.  
Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change any of the profile definitions.
3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.
4. Select the **Deployment Info** tab.
5. Clear the **Ignore compile errors** check box.

6. Leave the other options as they are. In particular, do not check **Enable Rapid Deployment**. These instructions do not describe how to set up rapid deployment.



7. Click **OK** to save the deployment settings.  
**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item used in the next exercise.



### EXERCISE 2-4: Deploy the application to the SilverStream server

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

- Select **Project>Deploy Archive** from the menu.  
Workbench displays progress messages, errors, and warnings in the Output Pane.  
**TIP** The deployment will fail if your server is not running.

## Deploying to the Sun Reference Implementation server

To deploy to the Reference Implementation, you create a server profile (described in “Defining a server profile” on page 4) and a runtime deployment descriptor in a format defined by Sun Microsystems. The descriptor becomes part of your project, and its archive location is the META-INF directory.

This is what you’ll do:

- EXERCISE 2-5: “Create a runtime deployment descriptor”
- EXERCISE 2-6: “Include the WAR or EJB JAR in an EAR”
- EXERCISE 2-7: “Specify deployment settings”
- EXERCISE 2-8: “Deploy the application to the Reference Implementation server”

**Requirement** If you are deploying a WAR or an EJB, the Reference Implementation requires that the WAR be a module in an enterprise application archive (EAR). Then you specify deployment settings and deploy the EAR.

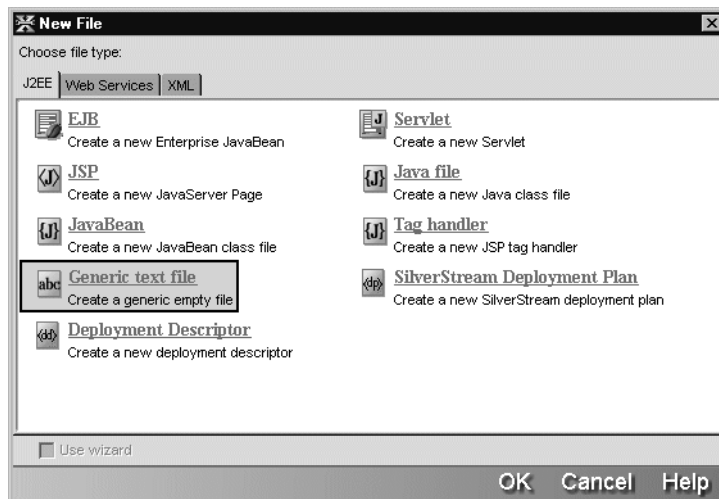
**About the runtime deployment descriptor** The runtime deployment descriptor specifies actual values for names identified in the deployment descriptor. You can create the file in the Edit Pane.



### EXERCISE 2-5: Create a runtime deployment descriptor

In this exercise you will create a new XML file for the Reference Implementation runtime deployment descriptor.

1. With your project open in Workbench, select **File>New** from the menu.
2. In the New File dialog, select **Generic text file** and click **OK**.



3. Select **File>Save**. In the Save As dialog, create a new directory in your project's root directory called **META-INF**. Save the file in that directory with the name **sun-j2ee-ri.xml**.
4. To add the new file to the project, select **Project>Add to Project>File**. If necessary, navigate to the new directory.
5. In the Add to Project dialog, select the file **sun-j2ee-ri.xml** and click **Open**.  
In the second Add to Project dialog, the correct values should already be set: **Add the file to the archive at this location** and **META-INF/sun-j2ee-ri.xml**.
6. Click **OK**.
7. In the Edit Pane, enter the XML for the descriptor, which looks something like this:

```
<?xml version="1.0" encoding="Cp1252"?>

<j2ee-ri-specific-information>
  <server-name></server-name>
  <rolemapping />
  <web>
    <display-name>ProverbsStart</display-name>
    <context-root>ProverbsStart</context-root>
    <resource-ref>
      <res-ref-name>jdbc/ProverbsDB</res-ref-name>
      <jndi-name>ProverbsCloud</jndi-name>
    </resource-ref>
  </web>
</j2ee-ri-specific-information>
```

Typically, you will specify a display name, context root, and values for resource references, if any. Each tutorial has specific instructions for these values.

8. Save again and close the file.



### EXERCISE 2-6: Include the WAR or EJB JAR in an EAR

If the tutorial application is a WAR or an EJB JAR, you need to put the archive in an EAR before deploying. After you create the EAR, you can work with the tutorial project from within the EAR or you can close the EAR and open the individual project.

1. Close the current project.
2. Select **File>New Project** from the menu.
3. In the New Project dialog, choose **EAR Enterprise Archive** and click **OK**.
4. Specify a project name and location. The rest of the dialog is filled in automatically. In Project J2EE Version, specify **j2ee 1.2 (ear 1.2)** so the application will run on any server that supports J2EE 1.2.

**New Project**

Enter the name and location (directory path) for the project, the archive file, and the deployment descriptor and select the desired J2EE version. (To use an existing archive as-is, create a deploy-only project instead.)

Project Name:  
ProverbEAR

Project Location:  
c:\WorkbenchProjects\ProverbEAR

Archive Name (e.g. office.war):  
ProverbEAR

Archive Location (directory):  
c:\WorkbenchProjects\ProverbEAR

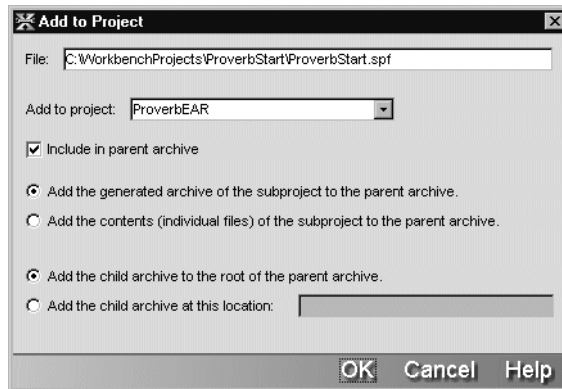
Deployment Descriptor Name:  
application.xml

Deployment Descriptor Location:  
c:\WorkbenchProjects\ProverbEAR\META-INF

Project J2EE Version:  
J2EE 1.2 (EAR 1.2)

< Back Next > Cancel Help

5. Click **Next** and then click **Finish**.
6. Select **Project>Add to Project>Subproject** from the menu.
7. In the Add to Project dialog, find the project root directory for the module you want to deploy, select the project file (it ends in SPF), and click **Open**.
8. In the second Add to Project dialog, select these options:
  - **Include in parent archive**
  - **Add the generated archive of the subproject to the parent archive**
  - **Add the child archive to the root of the parent archive**



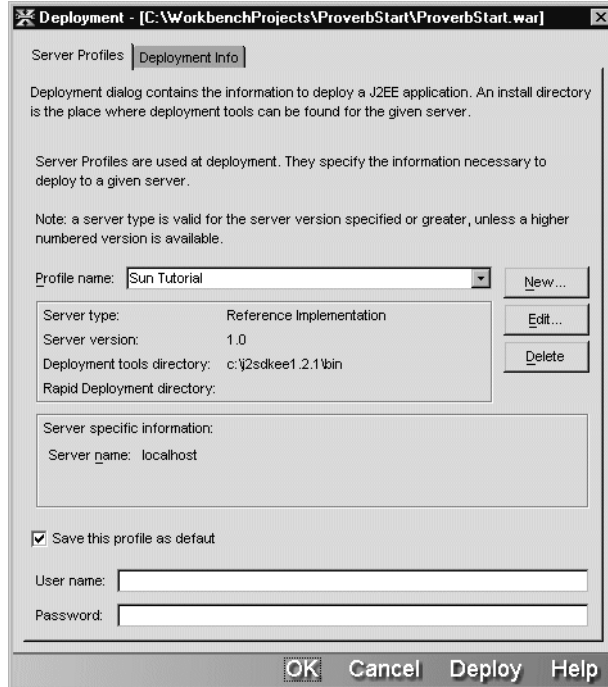
9. Open **application.xml** in the Deployment Descriptor Editor.  
The descriptor includes a WAR module that corresponds to the subproject you added.
10. Right-click **Enterprise Archive** and select **Properties**. In the property sheet, specify a display name for the application.
11. Select the WAR module. In the property sheet, specify a value for context root. Usually this is the same as the name of the archive.
12. Save and close the file.



### EXERCISE 2-7: Specify deployment settings

In this exercise you will specify the deployment settings that are used each time you deploy while you are developing and testing your tutorial application.

1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.



2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.

Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change the profile definitions.

3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.

You do not need to make any settings on the Deployment Info tab. These instructions do not describe how to set up rapid deployment.

4. Click **OK** to save the deployment settings.

**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item used in the next exercise.



### EXERCISE 2-8: Deploy the application to the Reference Implementation server

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

1. Select **Project>Deploy Archive** from the menu.

Workbench displays progress messages, errors, and warnings in the Output Pane.

**TIP** The deployment will fail if your server is not running.

2. Restart the server.

## Deploying to the Jakarta Tomcat server

To deploy to Tomcat, you create a server profile (described in “Defining a server profile” on page 4). Then you specify deployment settings and deploy the archive. The name of the archive becomes the context path for the application. There is no server-specific deployment information.

This is what you'll do:

- EXERCISE 2-9: “Specify deployment settings”
- EXERCISE 2-10: “Deploy the application to the Tomcat server”

**Limitations** Database connectivity, connection pools, and JNDI InitialContext are not supported by Tomcat Version 3.2.x, so these tutorials will not work:

- Proverbs Web application requires database support
- CalcWARClient Web application requires JNDI support (CalcWARClient does work with Tomcat 4)

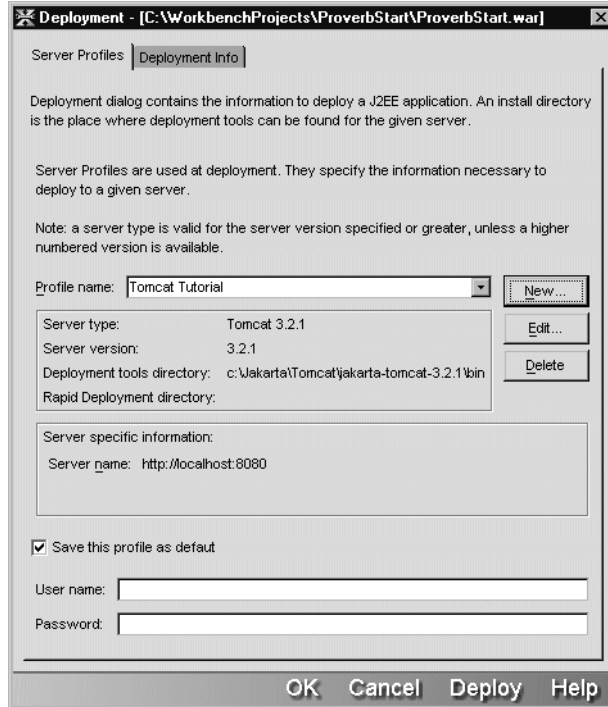


### EXERCISE 2-9: Specify deployment settings

In this exercise you will specify the deployment settings that will be used each time you deploy while you are developing and testing your tutorial application.

1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.





2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.  
Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change the profile definitions.
3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.  
You do not need to make any settings on the Deployment Info tab. These instructions do not describe how to set up rapid deployment.
4. Click **OK** to save the deployment settings.  
**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item you’ll use in the next exercise.



### EXERCISE 2-10: Deploy the application to the Tomcat server

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

1. Select **Project>Deploy Archive** from the menu.  
Workbench displays progress messages, errors, and warnings in the Output Pane.
2. Restart the Tomcat server.

## Deploying to the BEA WebLogic server

To deploy to the BEA WebLogic server, you create a server profile (described in “Defining a server profile” on page 4) and the weblogic.xml file that defines runtime information for the application. Then you specify deployment settings and deploy the archive.

This is what you'll do:

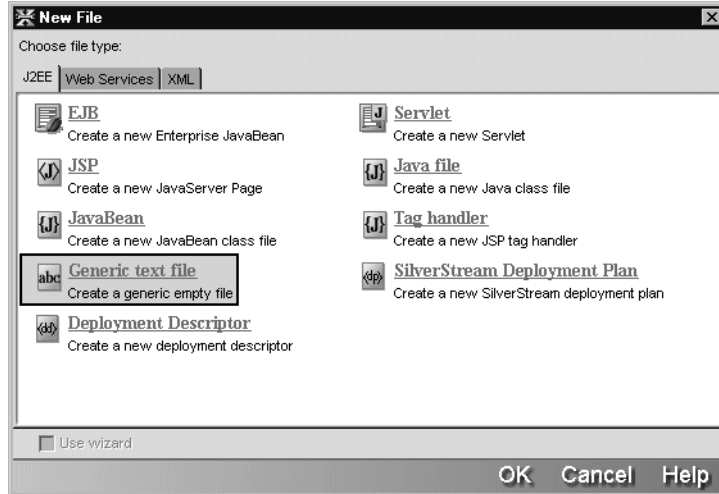
- EXERCISE 2-11: “Create a WebLogic deployment descriptor”
- EXERCISE 2-12: “Specify deployment settings”
- EXERCISE 2-13: “Deploy the application to the BEA WebLogic server”



### EXERCISE 2-11: Create a WebLogic deployment descriptor

In this exercise you will create a new XML file for the WebLogic deployment descriptor.

1. With your project open in Workbench, select **File>New** from the menu.
2. In the New File dialog, select **Generic text file** and click **OK**.



3. Select **File>Save**. In the Save As dialog, save the file in the project's WEB-INF directory with the name **weblogic.xml**.

**TIP** If the WEB-INF directory is already part of the project, the new file is included in the archive automatically.

4. In the Edit Pane, enter the XML for the descriptor.

Typically, you will specify a display name, context root, and values for resource references, if any. Each tutorial has its own values for these elements.

The XML looks something like this:

```
<!DOCTYPE weblogic-web-app PUBLIC "-//BEA
Systems, Inc.//DTD Web Application 6.0//EN"
"http://www.bea.com/servers/wls610/dtd/weblogic-web-jar.dtd">

<weblogic-web-app>
  <description>
    Text description of the Web App
  </description>
  <weblogic-version>
  </weblogic-version>

  <reference-descriptor>
    <resource-description>
      <res-ref-name>jdbc/ProverbsDB</res-ref-name>
      <jndi-name>ProverbsCloud</jndi-name>
    </resource-description>
  </reference-descriptor>
</weblogic-web-app>
```



For more information about the WebLogic descriptor, see Writing Web Application Deployment Descriptors in the WebLogic documentation set.

5. Save again and close the file.



### EXERCISE 2-12: Specify deployment settings

In this exercise you will specify the deployment settings that will be used each time you deploy while you are developing and testing your tutorial application.

1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.

Deployment dialog contains the information to deploy a J2EE application. An install directory is the place where deployment tools can be found for the given server.

Server Profiles are used at deployment. They specify the information necessary to deploy to a given server.

Note: a server type is valid for the server version specified or greater, unless a higher numbered version is available.

Profile name: WebLogic Tutorial [New...]

Server type: WebLogic 6.0 [Edit...]

Server version: 6.0 [Delete]

Deployment tools directory: c:\bea\wlserver6.0sp1\bin

Rapid Deployment directory:

Server specific information:

Server name: http://localhost:7001

Target Servers :

Save this profile as default

User name: [ ]

Password: [ ]

OK Cancel Deploy Help

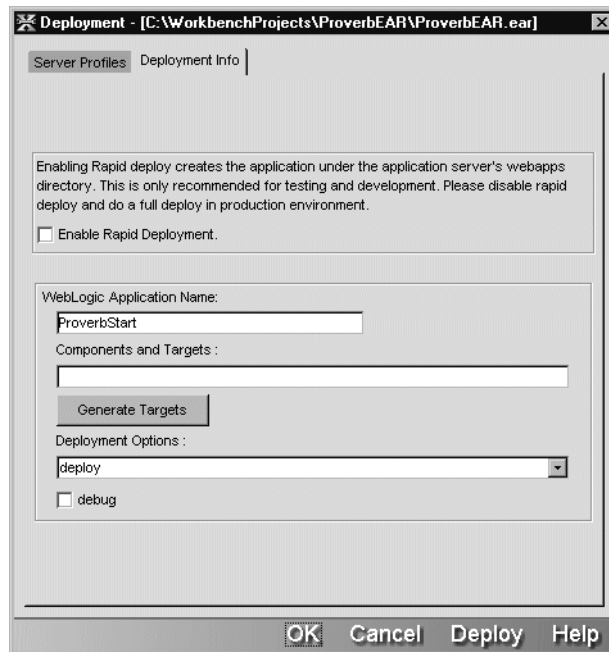
2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.

Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change the profile definitions.

3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.
4. Select the **Deployment Info** tab.
5. In the **WebLogic Application Name** text box, type the name your users will use in the URL for the application.

**NOTE** Do not check **Enable Rapid Deployment**. These instructions do not describe how to set up rapid deployment.

6. In the **Deployment Options** list box, select **deploy** if you're deploying the application for the first time. If a tutorial has you deploy the same application again, change the option to **update**.



7. Click **OK** to save the deployment settings.  
**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item you'll use in the next exercise.



### EXERCISE 2-13: Deploy the application to the BEA WebLogic server

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

- Select **Project>Deploy Archive** from the menu.  
Workbench displays progress messages, errors, and warnings in the Output Pane.

**TIP** The deployment will fail if your server is not running.

## Deploying to the IBM WebSphere server

To deploy to the IBM WebSphere server, you create a server profile (described in “Defining a server profile” on page 4) and specify deployment information. Then you specify deployment settings and deploy the archive.

This is what you'll do:

- EXERCISE 2-14: “Specify runtime deployment information”
- EXERCISE 2-15: “Specify deployment settings”
- EXERCISE 2-16: “Deploy the application to the IBM WebSphere server”

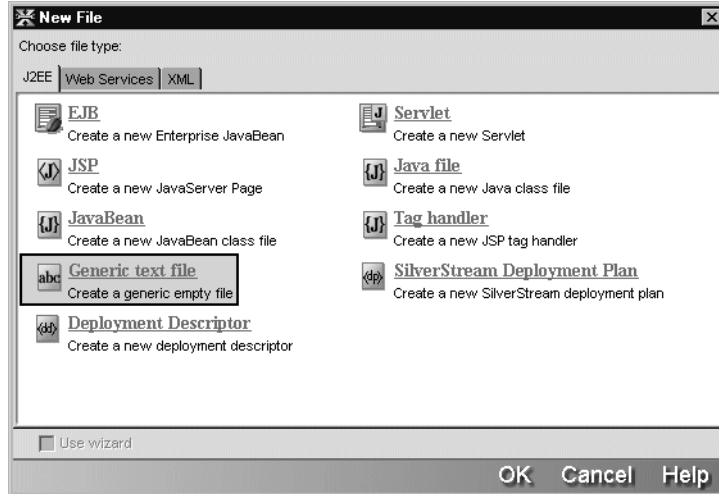
**Requirement** The WebSphere server requires that you deploy an EAR, not a WAR or an EJB archive. For a WAR, Workbench will automatically create an EAR that includes the WAR when you deploy.



### EXERCISE 2-14: Specify runtime deployment information

In this exercise you will create a new XML file for the WebSphere runtime deployment descriptor. Your J2EE application may not require this file.

1. With your project open in Workbench, select **File>New** from the menu.
2. In the New File dialog, select **Generic text file** and click **OK**.



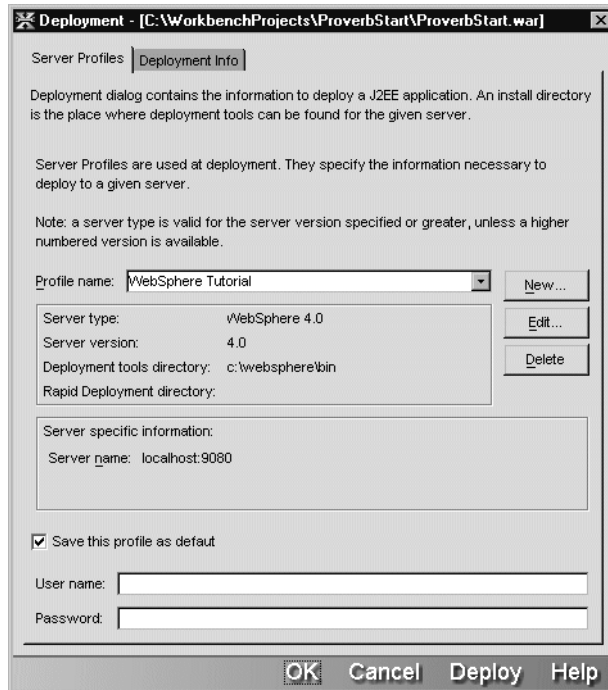
3. Select **File>Save**. In the Save As dialog, save the file in the project's WEB-INF directory with the name expected by the WebSphere server.  
**TIP** If the WEB-INF directory is already part of the project, the new file will be included in the archive automatically.
4. In the Edit Pane, enter the XML for the descriptor according to WebSphere requirements.
5. Save again and close the file.



### EXERCISE 2-15: Specify deployment settings

In this exercise you will specify the deployment settings that will be used each time you deploy while you are developing and testing your tutorial application.

1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.



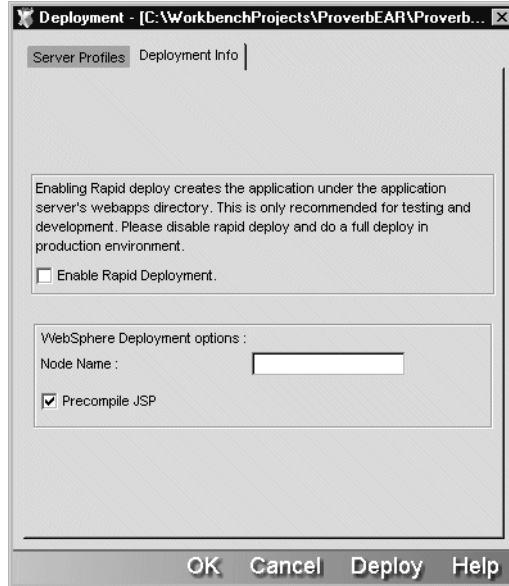
2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.

Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change the profile definitions.

3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.
4. Select the **Deployment Info** tab.
5. Leave the **Node Name** text box blank unless you have set up server nodes.
6. Check **Precompile JSP** so compilation errors are reported when you deploy.

**NOTE** Do not check **Enable Rapid Deployment**. These instructions do not describe how to set up rapid deployment.





7. Click **OK** to save the deployment settings.

**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item you'll use in the next exercise.



### EXERCISE 2-16: Deploy the application to the IBM WebSphere server

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

- Select **Project>Deploy Archive** from the menu.

Workbench displays progress messages, errors, and warnings in the Output Pane.

**TIP** The deployment will fail if your server is not running.

## Deploying to the Oracle9iAS server

To deploy to the Oracle9iAS server, you create a server profile (described in “Defining a server profile” on page 4) and specify deployment information. Then you specify deployment settings and deploy the archive.

This is what you’ll do:

- EXERCISE 2-17: “Specify runtime deployment information”
- EXERCISE 2-18: “Specify deployment settings”
- EXERCISE 2-19: “Deploy the application to the Oracle9iAS server”

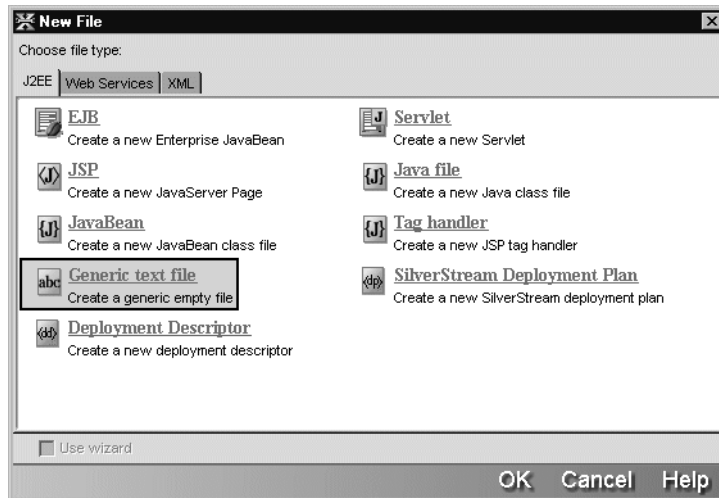
**Requirement** The Oracle server requires that you deploy an EAR, not a WAR or EJB archive. For a WAR, Workbench will automatically create an EAR that includes the WAR when you deploy.



### EXERCISE 2-17: Specify runtime deployment information

In this exercise you will create a new XML file for the Oracle runtime deployment descriptor.

1. With your project open in Workbench, select **File>New** from the menu.
2. In the New File dialog, select **Generic text file** and click **OK**.



3. Select **File>Save**. In the Save As dialog, save the file in the project's WEB-INF directory with the name expected by the Oracle server.  
**TIP** If the WEB-INF directory is already part of the project, the new file will be included in the archive automatically.
4. In the Edit Pane, enter the XML for the descriptor according to Oracle requirements.
5. Save again and close the file.



### EXERCISE 2-18: Specify deployment settings

In this exercise you will specify the deployment settings that will be used each time you deploy while you are developing and testing your tutorial application.

1. With your tutorial project open in Workbench, select **Project>Deployment Settings** from the menu.

Deployment - [C:\WorkbenchProjects\ProverbStart\ProverbStart.war]

Server Profiles | Deployment Info

Deployment dialog contains the information to deploy a J2EE application. An install directory is the place where deployment tools can be found for the given server.

Server Profiles are used at deployment. They specify the information necessary to deploy to a given server.

Note: a server type is valid for the server version specified or greater, unless a higher numbered version is available.

Profile name: Oracle Tutorial

Server type: Oracle9iAS 1.0.2.2

Server version: 1.0.2.2

Deployment tools directory: c:\oracle9ias\j2ee\home\lib

Rapid Deployment directory:

Server specific information:

Server name: http://localhost:8888

Save this profile as default

User name:

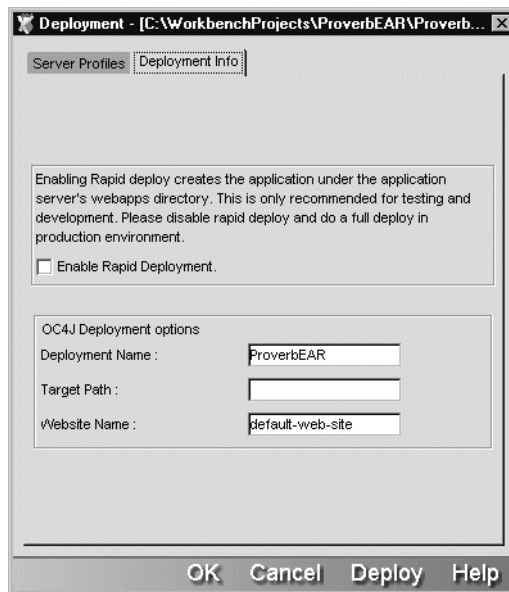
Password:

2. On the **Server Profiles** tab of the Deployment dialog, the profile you created in EXERCISE 2-1: “Create a server profile for your application server” should be selected. If it isn’t, select it now.

Values from the profile are displayed below the profile name. You can click New, Edit, or Delete to change the profile definitions.

3. If you have a secure server, fill in the **User name** and **Password** text boxes with the user name and password for the server.
4. Select the **Deployment Info** tab.
5. For **Deployment Name**, specify the name of the archive. Leave the other options as is.

**NOTE** Do not check **Enable Rapid Deployment**. These instructions do not describe how to set up rapid deployment.



6. Click **OK** to save the deployment settings.

**TIP** You could also click **Deploy**, which does the same thing as the **Deploy Archive** menu item you’ll use in the next exercise.



### **EXERCISE 2-19: Deploy the application to the Oracle9iAS server**

In this exercise you'll deploy your project using the values you specified in the Deployment dialog.

- Select **Project>Deploy Archive** from the menu.

Workbench displays progress messages, errors, and warnings in the Output Pane.

**TIP** The deployment will fail if your server is not running.



---

# Index

---

## A

Apache Tomcat  
  see Jakarta Tomcat 18

## B

BEA WebLogic server  
  deployment (tutorial) 20

## D

deployment  
  BEA WebLogic server (tutorial) 20  
  IBM WebSphere server (tutorial) 24  
  Jakarta Tomcat (tutorial) 18  
  Oracle9iAS server (tutorial) 28  
  setup (tutorial) 3  
  SilverStream eXtend Application Server (tutorial) 8  
  Sun J2EE Reference Implementation server  
    (tutorial) 13

## I

IBM WebSphere server  
  deployment (tutorial) 24

## J

J2EE application servers  
  supported (tutorial) 3  
Jakarta Tomcat  
  deployment (tutorial) 18

## O

Oracle9iAS server  
  deployment (tutorial) 28

## R

Reference Implementation server  
  see Sun J2EE Reference Implementation server

## S

servers  
  creating profiles (tutorial) 4  
  supported (tutorial) 3  
SilverStream eXtend Application Server  
  deployment (tutorial) 8  
Sun J2EE Reference Implementation server  
  deployment (tutorial) 13

## T

Tomcat server  
  see Jakarta Tomcat  
tutorials  
  about 1

## W

WebLogic server  
  see BEA WebLogic server  
WebSphere server  
  see IBM WebSphere server

