

NetBeans IDE Field Guide

Copyright © 2005 Sun Microsystems, Inc. All rights reserved.

Table of Contents

Introduction to J2EE Development in NetBeans IDE	1
Configuring the IDE for J2EE Development.....	2
Getting the Most from the Java BluePrints Solutions Catalog.....	4
Accessing the Java BluePrints Solutions Catalog.....	5
Navigating the Java BluePrints Solutions Catalog.....	5
Creating a NetBeans Project for a Java BluePrints Solution	7
Running a Java BluePrints Solution Project.....	11

Introduction to J2EE Development in NetBeans IDE

The Java 2 Platform, Enterprise Edition (J2EE) defines the standard for developing multi-tier enterprise applications. The J2EE platform simplifies enterprise applications by basing them on standardized, modular components, by providing a complete set of services to those components, and by handling many details of application behavior automatically, without complex programming. The J2EE platform is targeted for developers who want to write distributed transactional applications for the enterprise and leverage the speed, security, and reliability of server-side technology.

NetBeans IDE 4.1 introduces comprehensive support for the J2EE developer, from advanced wizards to create J2EE components like web applications, Servlets, Java Server Faces, or enterprise Java Modules and Enterprise JavaBeans components, or Web services, J2EE applications, to a complete runtime environment based on the Sun Java System Application Server 8.1.

The Sun Java System Application Server 8.1 which is bundled with NetBeans IDE 4.1 is a J2EE 1.4 compliant application server which is free for development, deployment and redistribution. It offers the ideal companion to the IDE for all the developers that need an integrated environment where complete J2EE applications can be developed, built, assembled, deployed and debugged.

Configuring the IDE for J2EE Development

In order to explore all the capabilities of NetBeans related to J2EE applications development or Web services development, you need to make sure your environment is correctly configured. If you have downloaded the cobundle product containing both NetBeans IDE and the J2EE 1.4 compliant Sun Application Server, you have a preconfigured development environment and you can skip the following steps. If you have downloaded a standalone NetBeans IDE, you will also need to install the Sun Java System Application Server 8.1, available from <http://java.sun.com/j2ee/1.4/download.html>. This application server is the core of the J2EE 1.4 SDK and is free for development, deployment and redistribution.

Before you can deploy an enterprise application, Web application, JSP, servlet, or EJB module, the server to which you are going to deploy needs to be registered with the IDE. By default, only the bundled Tomcat Web Server is registered with the IDE.

To register a Sun Java System Application Server:

1. In the IDE, choose “Tools | Server Manager” menu item.
2. In the Server Manager, click the “Add Server” button. The Add Server wizard (as shown in Figure 7-1) appears and displays the types of servers that are compatible with the IDE. The first time you register the Sun Application server, you will be asked for its installation directory.

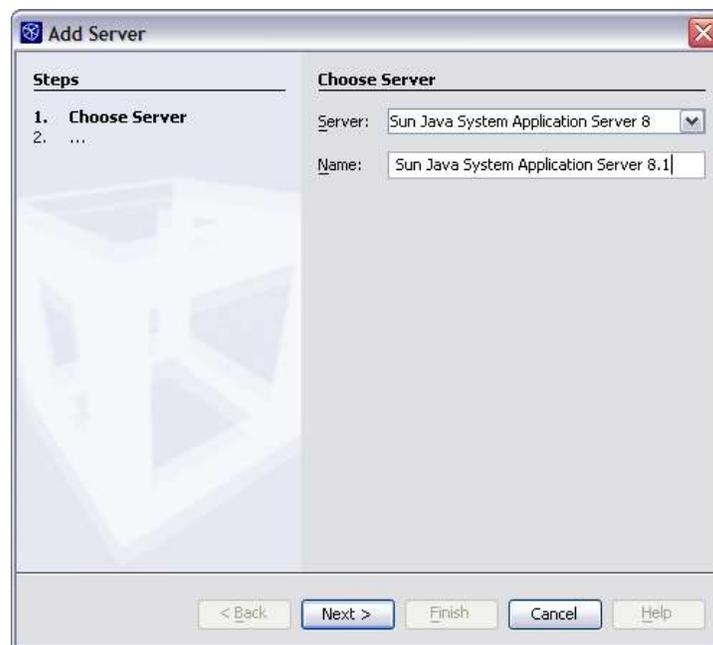


Figure 7-1
Add Server wizard, Choose Server page

3. In the dialog box, select the type of server you want to register (Sun Java System Application Server 8.1) and click Next.
4. On the Define Application Server Instance Properties page (as shown in Figure 7-2), specify server-specific information in the panels that follow and click Finish. (Remember the Admin user name is “admin” and the default password is “adminadmin”, unless you picked a different one at installation time.)

Figure 7-2



Add Server wizard, Define Application Server Instance Properties page

When you register a server with the IDE, you make its libraries available for production, or deployment, or both. If you deploy your applications to a remote instance of the Sun Java System Application Server, its libraries are available at runtime. However, during development, you might need a local instance of this server. You can have multiple instances of the SJS Application Server registered with the IDE. Once you have registered your local instance, you can register remote instances by selecting a remote machine name and its port number in the “location” entry field of the wizard, but you always have to specify the local folder that contains the local installation of the Sun Java System Application Server.

When a server is registered with the IDE, you can see its node in the Runtime window under the the Server Registry node. When you create a project in the New Project wizard, you select the server to which you want to deploy your application. After you create the application, you can change the server by right-clicking the project, choosing Properties, clicking Run, and selecting a different server.

You are now ready to create Web services, Enterprise JavaBeans, and J2EE applications.

Getting the Most from the Java BluePrints Solutions Catalog

NetBeans IDE provides a unique capability for learning and understanding best practices for Java application development with its integration of the Java BluePrints Solutions Catalog. The Java BluePrints Solutions Catalog has long been accepted as the source of Java application best practices and Java suggested guidelines. The Java BluePrints Solutions Catalog also illustrates these best practices and guidelines through various example applications. It provides a huge repository of example applications from which you can literally cut and paste source code or tailor for his or her own specific application to meet his or her application's needs.

In NetBeans IDE, you can directly access the catalog and install example Java BluePrints Solutions directly into the IDE as a new project. This feature provides you with a unique opportunity to quickly learn and understand various Java BluePrints best practices and recommended guidelines.

The following is a listing of the available solutions in the Java BluePrints Solutions Catalog listed by general category:

- Web Tier with the JavaServer Faces Technology
 - Making Web applications Accessible
 - Accessing UIComponent Instances Programmatically
 - Handling Command Submissions
 - Displaying and Navigating Database Query Results
 - Creating Tabbed Views
 - Creating a Common Look and Feel Using Templating
 - Storing Session State on the Client
 - Implementing Server-side Validation
 - Implementing Client-side Validation
- Service Oriented Architectures with Web services
 - Correlation Identifier
 - Designing Document Oriented Services
 - Idempotent Endpoints
 - Web service Context Information
 - Client Access to a Web service
 - Accessing Web services from a Stand-alone Java Client
 - Accessing J2EE Web services from a .NET Client
- Business Tier with the J2EE Platform
 - Refactoring Using the Unified JMS Client APIs
 - Portable Packaging of Utility Classes and Library jar Files Used By An Application
 - Using the J2EE 1.4 Timer Service for Scheduling Actions

In the following sections you will learn how to use the Java BluePrints Solutions Catalog in the NetBeans IDE.

Accessing the Java BluePrints Solutions Catalog

When you install NetBeans IDE, the Java BluePrints Solutions Catalog is available without any additional installation steps. You can access the catalog by choosing Help | BluePrints Solutions Catalog. The catalog is then displayed in the IDE's main document area (where the Source Editor also appears). When the Java BluePrints Solutions Catalog is loaded, the NetBeans IDE main window will look like the screen in Figure 7-3.



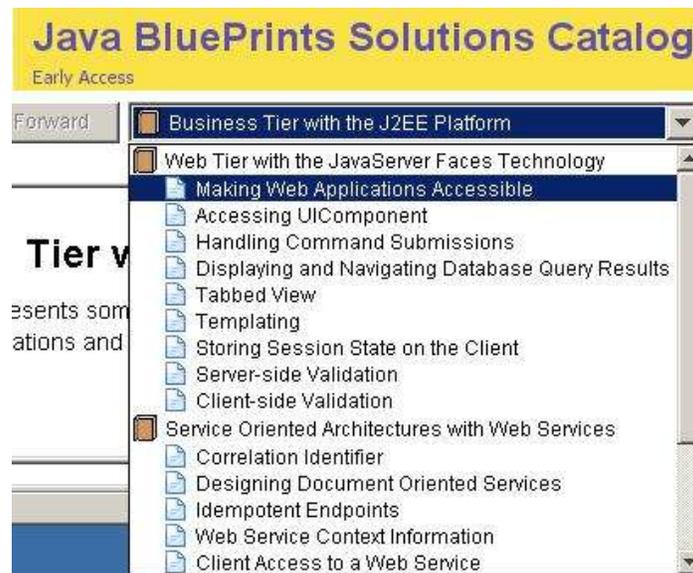
Figure 7-3
Java BluePrints Solutions Catalog in the NetBeans IDE main window

In the next section of this chapter, we will describe how to navigate the Java BluePrints Solutions Catalog.

Navigating the Java BluePrints Solutions Catalog

Since the Java BluePrints Solutions Catalog display in the NetBeans IDE main window follows the browser paradigm, it is easy to navigate. You will notice there is a dropdown list where you can select different Java BluePrints solutions (see Figure 7-4).

Figure 7-4



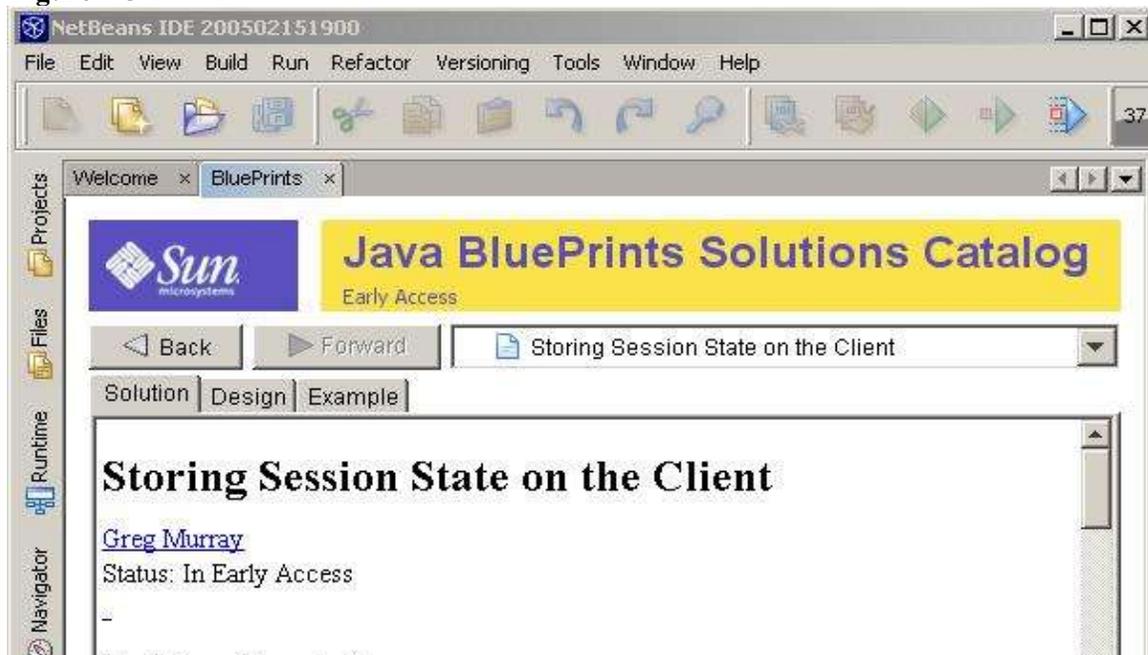
Java BluePrints Solutions Catalog with the combo box listing the different solutions available

You will notice there are currently three major categories of Java BluePrints Solutions available in the catalog. They are:

- Web Tier with the JavaServer Faces Technology
- Server Oriented Architectures with Web services
- Business Tier with the J2EE Platform

Under each major category there are many Java BluePrints Solutions. You can navigate within the three major categories by pressing the Back or Forward buttons or you may use the dropdown list to select a specific solution within a major category. In addition, for each of the solutions in a given major category, when a specific Java BluePrints Solution is selected there are two or three different tabs to choose from as shown in Figure 7-5.

Figure 7-5



A selected blueprint with the Solution tab selected

The first tab, the Solution tab, displays a description of the issue the Java BluePrints Solution is trying to solve. The second tab, the Design tab, when selected, describes the design of the solution so you can understand the implementation decisions made and the design best practices used in the solution. All of the solutions in the Java BluePrints Solutions Catalog have both the Solution and Design tabs.

To view the Solution or Design for a given Java BluePrints Solution, simply select the appropriate tab.

On some Java BluePrints Solutions there is a third tab called the Example tab which allows you to install an example implementation of the Java BluePrints Solution in the NetBeans IDE as a NetBeans project. This is a very useful capability since it allows you to see a running, working example implementation of the Java BluePrints Solution by being able to run or even possibly debug the solution. In addition, it is very easy to pull source code from a working implementation into your own specific application or project. In the next section, you will see how to install a Java BluePrints Solution into the NetBeans IDE as a project.

Creating a NetBeans Project for a Java BluePrints Solution

This section describes the steps to install a Java BluePrints Solutions example in the NetBeans IDE as a NetBeans project. In addition, the section describes how to run the example once it is installed as a NetBeans project.

NetBeans IDE Tip

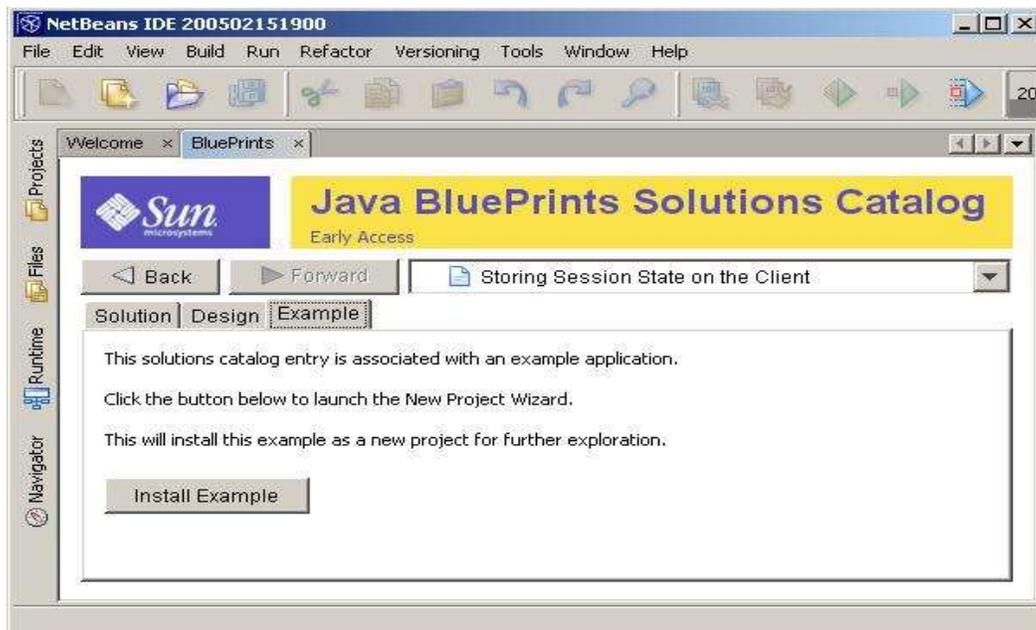
In NetBeans IDE 4.1, only Web tier solutions can be installed as projects. In future releases, the other categories should also be installable as examples.

To install a Java BluePrints Solution from the Java BluePrints Solutions Catalog:

1. Choose Help | BluePrints Solutions Catalog.

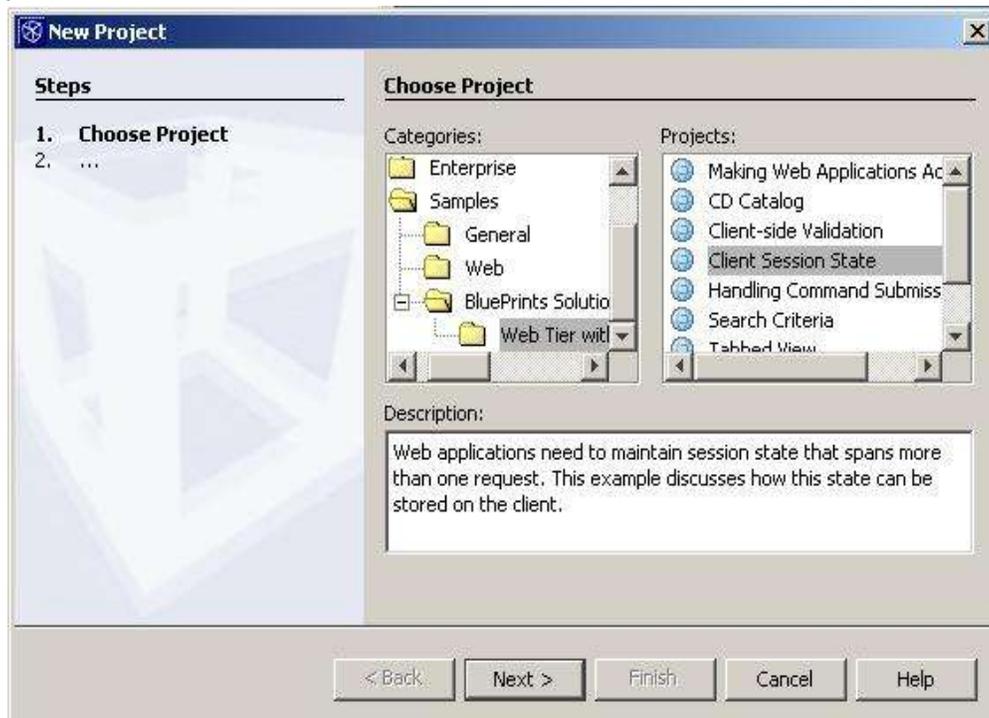
2. Once catalog is displayed, select the solution you wish to work with from the dropdown list in the display.
3. In the solution you have selected, click the Example tab (if there is an Example tab for that solution). If the Example tab exists and you press it, you will see a screen that looks similar to that shown in Figure 7-6.
If the Example button is not there, then this particular example can not be installed in the IDE.
4. On the Example screen for the solution, click the Install Example button.
This will begin the installation and setup of a new NetBeans project for the selected Java BluePrints Solution. The IDE's New Project wizard is launched with the Java BluePrints Solution you have chosen to install selected as the project. For example, the Client Session State project template would be chosen (as shown in Figure 7-7) if you pressed Install Example on the screen for the Storing session State on the Client template.

Figure 7-6



A solution with its Example tab selected

Figure 7-7



New Project wizard with the a Java BluePrints Solution selected

1. Press the Next button to continue the installation and setup of the Java BluePrints Solution. The next screen of the wizard asks you to supply a project name, or to accept the default name, a project location, or accept the default and whether to set this as the Main Project in the NetBeans IDE.
In almost all cases you should accept the defaults shown in the wizard. Only if you wish to change the name given to the project, its location or set it as the Main Project should you change the default settings displayed on this wizard.
2. Press the Finish button to complete the install and setup of the Java BluePrints Solution as a NetBeans project.
After completing the wizard, you might see the warning dialog box shown in Figure 7-8. This generally occurs if you do not have the Sun Java System Application Server registered with the IDE. It is possible to set the project to work with a different server (such as Tomcat), but some of the features in the server might not work if the server is not fully J2EE compliant.

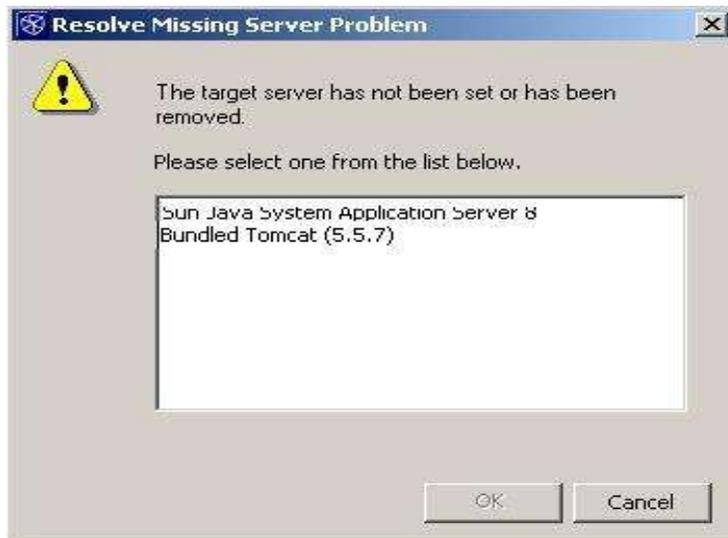
Figure 7-8

Warning dialog box that appears if the IDE does not detect an appropriate server for the project



3. If you see the warning dialog, follow the instructions given in the warning dialog. After clicking OK to close the dialog and open the project, right-click the newly created project in the Projects window and choose Resolve Missing Server Problem. A Resolve Missing Server Problem dialog box (as shown in Figure 7-9) appears).

Figure 7-9



Resolve Missing Server Problem dialog box

From the Resolve Missing Server Problem dialog, select a target server. If you have a Sun Java System Application Server installed with the NetBeans IDE, you will see an option for both the Sun Java System Application Server and for Tomcat. Choose the desired server for your project and press Ok.

NetBeans IDE Tip

If you would like to use Sun Java System Application Server with the Java BluePrints Solution you are installing, but the server does not appear in the list of servers in the Resolve Missing Server Problem dialog box, press Cancel to exit the dialog.

If necessary, download and install the application server. Then register the application server in the IDE's Server Manager (available through the Tools menu). After that, you can go back to the Resolve Missing Server

Problem dialog and select the Sun Java System Application Server as the target server.

Once you have the Java BluePrints Solution created as a NetBeans project you can perform operation such as building, deploying, and debugging. The next section describes how to run the project.

NetBeans IDE Tip

You can also install the available Java BluePrints Solutions as IDE projects straight from the New Project wizard. In the New Projects wizard, you can expand the Sample folder category to show BluePrints Solutions folder. The BluePrints Solutions folder contains the same Java BluePrints Solutions as the the ones in the catalog that have an Example tab.

Running a Java BluePrints Solution Project

To run a NetBeans project that has been created from the Java BluePrints Solutions Catalog, you perform the same operations as you would when running other J2EE applications in the IDE. That is, you open the Project window, right-click the newly created project for your Java BluePrints Solution, and select Run Project. The NetBeans IDE will build the newly created Java BluePrints Solution project, deploy it to the target server, and load the application's home page in your default web browser automatically.

Once you have the Java BluePrints Solution created as a project in the NetBeans IDE, you can perform a large number of operations. For instance, you can deploy the application to your target server as you have already seen, you can run the application in a debugger, you can use the HTTP Monitor to analyze the HTTP requests that are passed between your browser and the deployed application, you can make changes to the source files by editing the project source.

In fact, using the Java BluePrints Solutions Catalog and one of its applications is an excellent way to learn some of the J2EE technologies and best practices on how to use the technologies. In addition, these solutions are an excellent source from which you can cut and paste code for an application you are developing.