



Java CAPS Domain and Runtime Installation



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

Part No: 821-0457-13
September 2009

Copyright 2009 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, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries 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 SunTM 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.

Contents

Java CAPS Domain and Runtime Installation	5
Overview of New Domain Installation	6
Creating a New Java CAPS Domain	6
▼ To Create a New Java CAPS Domain	6
Installing a Java CAPS Runtime on a New Domain	7
▼ To Install Runtime Components on a New Domain	7
Adding the New Domain to NetBeans	8
▼ To Add the New Domain to NetBeans	8
Installing Java CAPS JBI Components on a New Domain	9
▼ To Install JBI Components on a New Domain From the Admin Console	9
▼ To Install JBI Components on a New Domain From NetBeans	11
▼ To Install JBI Components on a New Domain From the Command Line	13
Finalizing the New Domain Installation	14
▼ To Finalize the Installation	14
Enabling CAPS Repository-Based Applications on a Stand-Alone GlassFish 2.1 Instance	15
▼ Adding the CAPS Runtime Environment to GlassFish 2.1	15

Java CAPS Domain and Runtime Installation

This topic provides instructions on how to create a new Java CAPS domain and how to install the runtime components on new Java™ Composite Application Platform Suite (Java CAPS) domains. If you have any questions or problems, see the Java CAPS Support web site at <http://goldstar.stc.com/support>.

To access all the Java CAPS documentation in HTML or PDF format, see the [Java CAPS documentation on DSC](#).

What You Need to Know

The following topic provides a brief overview of the installation process:

- [“Overview of New Domain Installation” on page 6](#)

What You Need to Do

Perform these steps in the order given to create a new GlassFish domain for Java CAPS:

- [“Creating a New Java CAPS Domain” on page 6](#)
- [“Installing a Java CAPS Runtime on a New Domain” on page 7](#)
- [“Adding the New Domain to NetBeans” on page 8](#)
- [“Installing Java CAPS JBI Components on a New Domain” on page 9](#)
- [“Finalizing the New Domain Installation” on page 14](#)
- [“Enabling CAPS Repository-Based Applications on a Stand-Alone GlassFish 2.1 Instance” on page 15](#)

Overview of New Domain Installation

The Java CAPS runtime pieces are automatically installed with a complete Java CAPS installation. During this installation, a domain named `domain1` is automatically created with its own runtime. You can create additional domains in GlassFish, but these domains will not include the Java CAPS runtime components. You need to install the runtime pieces for new domains manually.

The steps required to create a complete Java CAPS domain include creating the physical domain using `asadmin` commands and then installing the Java CAPS runtime framework. You also need to add the new domain to NetBeans so you can work with and configure the components from the IDE, and install the Java CAPS JBI components on the new domain. To finalize the installation, you need to configure the JVM options for the new domain, and you can optionally install the ETL Monitor to monitor Data Integrator projects. The JBI components can be installed from the command line, GlassFish Admin Console, or NetBeans IDE; all three methods are documented here.

Creating a New Java CAPS Domain

You can create a new Java CAPS domain by running an `asadmin` command. The instructions below provide an example of a very simple command with few parameters, but you can specify additional information such as the location for the domain files, a password file, and so on. For a complete list of options, type `asadmin create-domain` at the command line.

Note – Do not abbreviate the following instructions by running `asadmin` and then entering all command at the `asadmin` prompt. Run each command separately as noted below to avoid any loss of information.

▼ To Create a New Java CAPS Domain

1 Do one of the following:

- **Add the following path to your PATH environment variable:** *appserver/bin*, where *appserver* is the path to your GlassFish home directory for Java CAPS.
- **Navigate to** *appserver/bin*.

2 Run the following command:

```
asadmin create-domain --portbase PortBaseNumber --savemasterpassword=true  
--savelogin=true DomainName
```

where *PortBaseNumber* is the port number with which port assignment starts and *DomainName* is the name you want to give the new domain.

Note – Use the `--savemasterpassword` option to encrypt and save the master password so you do not need to enter it for `asadmin` commands. Use the `--savelogin` option to encrypt and save the admin password in `UserHomeDirectory/.asadminpass`. This enables the Access Manager to configure itself after the domain is first started.

For more information about `create-domain` options, see [create-domain\(1\)](#).

3 Enter the administrator login credentials when prompted.

4 (Optional) If you need STCMS (Java CAPS JMS) support in the new domain, run the following command:

```
asadmin create-stcms --domainname DomainName --stcmsport PortNumber
STCMSName
```

where:

- *DomainName* is the name you assigned to the new domain above
- *PortNumber* is the port number for STCMS (typically the base port number plus seven)
- *STCMSName* is a unique name for the message service

5 Run the following command to start the new domain:

```
asadmin start-domain DomainName
```

6 Continue to “[Installing a Java CAPS Runtime on a New Domain](#)” on page 7.

Installing a Java CAPS Runtime on a New Domain

Whenever you create a domain after the initial Java CAPS installation you need to add the runtime to the new domain in order to deploy any Java CAPS applications on it. This requires you to modify the `install.properties` files, which contains the domain configuration, and defines where to install and deploy components, as well as how to connect to the application server. You can also specify the new values for the domain name, port number, and username on the command line.

▼ To Install Runtime Components on a New Domain

Before You Begin Complete the steps under “[Creating a New Java CAPS Domain](#)” on page 6.

- 1 **Open the file `appserver/addons/caps/install.properties` in a text editor (where `appserver` is the home directory for the Java CAPS GlassFish server installation).**
- 2 **Modify the properties in the file to match the properties of the new domain.**

Tip – You will likely need to change the following properties:

- `caps.domain.name`.
- `caps.admin.port`. This is typically your base port plus 48. You can look the value up in `domains/DomainName/config/domain.xml`. It is the port number for the admin-listener.
- `caps.admin.user`, if you changed the default administrator user name.
- `AS_ADMIN_PASSWORD` and `AS_ADMIN_MASTERPASSWORD`, if you changed the default passwords.

In addition, you need to enable the UDDI installation if you require UDDI support, and enable the STCMS installation and specify an operating system if you required STCMS support.

- 3 **From a command line, navigate to the `appserver/bin` directory.**
- 4 **Run the following command:**

```
asant -f ../addons/caps/install.xml
```
- 5 **Restart the domain.**
- 6 **Continue to “[Installing Java CAPS JBI Components on a New Domain](#)” on page 9.**

Adding the New Domain to NetBeans

After you create a new domain, you can add it to the Servers list on the NetBeans Services window so you can work with the new domain from NetBeans.

There are multiple options for registering the domain, including registering a remote domain, registering a local domain that is not located in the default `appserver/domain` directory, or registering a local domain that is located in the default directory. The following instructions describe how to add a local default domain. Selecting one of the other options will add another step to the wizard. Click the Help button to determine which option to use.

▼ To Add the New Domain to NetBeans

Before You Begin Complete the steps listed under “[Installing a Java CAPS Runtime on a New Domain](#)” on page 7.

- 1 **In NetBeans, open the Services window.**

- 2 Right-click on Servers, and select Add Server.**
The Add Server Instance Wizard appears.
- 3 Select GlassFish V2 and enter a unique name for the domain.**
- 4 Click Next.**
The Platform Folder Location window appears.
- 5 In the Platform Location field, enter the path to the application server home directory.**
- 6 Select Register Local Default Domain, and select the domain you just created.**
- 7 Click Next.**
The Domain Admin Login Info window appears.
- 8 Enter the admin username and password, and then click Finish.**
The new domain appears in the Servers list.

Installing Java CAPS JBI Components on a New Domain

After you install the Java CAPS runtime on your new domain, you can install the JBI components, which includes shared libraries, Service Engines, Binding Components, and so on. These components are all installed individually. The components to install are located in *appserver/addons/jbi-components*. Install the shared libraries first.

You can install the JBI components using any one or a combination of the following methods:

- [“To Install JBI Components on a New Domain From the Admin Console” on page 9](#)
- [“To Install JBI Components on a New Domain From NetBeans” on page 11](#)
- [“To Install JBI Components on a New Domain From the Command Line” on page 13](#)

▼ To Install JBI Components on a New Domain From the Admin Console

Before You Begin Complete the steps under [“Adding the New Domain to NetBeans” on page 8](#).

- 1 Launch the GlassFish Admin Console.**
- 2 In the navigation panel on the left, expand JBI.**
- 3 To install the shared libraries, do the following:**

Note – Shared libraries must be installed first.

- a. **Under JBI, select Shared Libraries.**
 - b. **In the JBI Shared Libraries table, click Install.**
 - c. **Browse to the location of the shared libraries**
(*JavaCAPSHome/appserver/addons/jbi-components*), **and then select** `encoderlib.jar`.
 - d. **Click Next and then click Finish.**
 - e. **Repeat the above steps to install the following files:**
 - `saxonlib.jar`
 - `wsdlxtlib.jar`
- 4 To install the Service Engines and Binding Components, do the following:**
- a. **Under JBI on the Admin Console, select Components.**
 - b. **In the JBI Binding Components and Service Engines table, click Install.**
 - c. **Browse to the location of the components**
(*JavaCAPSHome/appserver/addons/jbi-components*), **and then select** `bpelserviceengine.jar`.
 - d. **Click Next.**
 - e. **If needed, modify the configuration properties of the component.**
 - f. **Click Finish.**
 - g. **Repeat the above steps to install the following files:**
 - `databasebc.jar`
 - `edmse.jar`
 - `etlserviceengine.jar`
 - `filebc.jar`
 - `ftpbc.jar`
 - `iepserviceengine.jar`
 - `jmsbc.jar`
 - `ldapbc.jar`
 - `schedulerbc.jar`
 - `xsltseviceengine.jar`

Note – Installing Sun Data Integrator (`etlserviceengine.jar`) on the Admin Console might throw an exception. If you run into this exception, install Data Integrator through NetBeans (described below).

- 5 Continue to [“Finalizing the New Domain Installation” on page 14.](#)

▼ To Install JBI Components on a New Domain From NetBeans

Before You Begin Complete the steps under [“Adding the New Domain to NetBeans” on page 8.](#)

- 1 Launch NetBeans and open the Services window.
- 2 Expand Servers, expand the new domain, and then expand JBI.
- 3 To install the shared libraries, do the following:

Note – Shared libraries must be installed first.

- a. Right-click Shared Libraries and then select install.
 - b. Browse to the location of the shared libraries
(*JavaCAPSHome/appserver/addons/jbi-components*), and then select `encoderlib.jar`.
 - c. Click Next and then click Finish.
 - d. Repeat the above steps to install the following files:
 - `saxonlib.jar`
 - `wsdlexlib.jar`
- 4 To install the Service Engines, do the following:
- a. Right-click Service Engines, and then select Install.
 - b. Browse to the location of the components
(*JavaCAPSHome/appserver/addons/jbi-components*), and then select `bpelserviceengine.jar`.
The Installation Configuration window appears.
 - c. If needed, modify the configuration properties of the component.

d. Click Install.

e. Repeat the above steps to install the following files:

- edmse.jar
- etlserviceengine.jar
- iepserviceengine.jar
- xsltseviceengine.jar

5 To install the Binding Components, do the following:

a. Right-click Binding Components, and then select Install.

b. Browse to the location of the components

(*JavaCAPSHome/appserver/addons/jbi-components*), and then select *databasebc.jar*.

The Installation Configuration window appears.

c. If needed, modify the configuration properties of the component.

d. Click Install.

e. Repeat the above steps to install the following files:

- filebc.jar
- ftpbc.jar
- jmsbc.jar
- ldapbc.jar
- schedulerbc.jar

6 Continue to [“Finalizing the New Domain Installation” on page 14.](#)

▼ To Install JBI Components on a New Domain From the Command Line

Tip – To expedite the process of installing components through the command line, create a password file containing the administrator and master password. Otherwise, you will be prompted for login information for each component to install. Enter the following text in the file:

```
AS_ADMIN_PASSWORD=adminadmin
```

```
AS_ADMIN_MASTERPASSWORD=changeit
```

Remember to replace *adminadmin* and *changeit* with the actual passwords if you have changed them.

Before You Begin Complete the steps under “[Adding the New Domain to NetBeans](#)” on page 8.

- 1 **Navigate to** *JavaCAPSHome/appserver/bin*.
- 2 **To install the shared libraries, do the following:**

Note – Shared libraries must be installed first.

a. Run the following command:

```
asadmin install-jbi-shared-library --port PortNumber --user AdminUser
--passwordfile FileName ../addons/jbi-components/encoderlib.jar
```

where:

- *PortNumber* is the Admin Console port number.
- *AdminUser* is the administrator username (admin, by default).
- *FileName* is the name of the password file you created.

b. Repeat the previous step to install the following files:

- *saxonlib.jar*
- *wsdlxtlib.jar*

- 3 **To install the Service Engines and Binding Components, do the following:**

a. Run the following command:

```
asadmin install-jbi-component --port PortNumber --user AdminUser
--passwordfile FileName ../addons/jbi-components/bpel-serviceengine.jar
```

b. Repeat the above steps to install the following files:

- databasebc.jar
- edmse.jar
- etlserviceengine.jar
- filebc.jar
- ftpbc.jar
- iepserviceengine.jar
- jmsbc.jar
- ldapbc.jar
- schedulerbc.jar
- xsltseviceengine.jar

4 Continue to [“Finalizing the New Domain Installation” on page 14.](#)

Finalizing the New Domain Installation

Once you have everything installed on your new domain, you need to configure the JVM properties for the domain. You also have the option of installing the ETL Monitor, which is recommended if you plan to use the Data Integrator Service Engine.

▼ To Finalize the Installation

Before You Begin Complete the steps under [“Installing Java CAPS JBI Components on a New Domain” on page 9.](#)

1 To modify the JVM properties, do the following:

- a. **Navigate to the new domain's home directory**
(JavaCAPSHome/appserver/domains/DomainName/config.
- b. **Open domain.xml in a text editor.**
- c. **Remove the line `<jvm-options>-client</jvm-options>` under the `<java-config>` element.**
- d. **Replace the above line with `<jvm-options>-Xms512m</jvm-options>`.**
- e. **Save and close the file.**

2 To install the ETL Monitor, copy the file `etlse-monitor.war` from *JavaCAPSHome/appserver/bin* to *JavaCAPSHome/appserver/domains/DomainName/autodeploy.*

Enabling CAPS Repository-Based Applications on a Stand-Alone GlassFish 2.1 Instance

This section describes how to deploy classic Java CAPS projects to a stand-alone instance of GlassFish 2.1 (Sun Java Application Server 9.1.1). To make the GlassFish 2.1 instance compatible with Java CAPS Repository-based projects requires the following manual steps.

▼ Adding the CAPS Runtime Environment to GlassFish 2.1

- 1 Install the stand-alone GlassFish 2.1 (Sun Application Server 9.1.1).
- 2 Add the Java CAPS Runtime Environment to GlassFish 2.1 as follows:
 - a. Extract the `caps.zip` file into the domain addons directory, for example `domains\domain1\addons`. This creates the `caps` folder in the addons directory.
The `caps.zip` file is available from Sun Support.
 - b. Configure the `install.properties` file in the `caps` folder, so that the values match those of the domain created by GlassFish 2.1.
- 3 Start the application server.
- 4 From the command line, change directories to the application server's bin directory. For example, `cd AppServer\bin`.
- 5 Execute the following command to install the CAPS specific libraries into `domains\domain1\lib`. Replace `\` with `/` for UNIX systems.

```
asant -f ..\domains\domain1\addons\caps\install.xml
```


The `domains\domain1\lib` directory should now include the new JARs used by the CAPS projects.
- 6 Turn off the Security Manager in the application server.
- 7 Restart the domain.
- 8 Deploy the CAPS specific WLM project into the Application Server.

