

Installing Novell Integration Manager

To be submitted.

Manually deploying the Integration Manager Enterprise Server to supported application servers

The Integration Manager installation program provides automated installation of the Integration Manager Enterprise Server (also known as the Integration Manager runtime) to JBoss application servers. You must manually install the Integration Manager Enterprise server to other supported application servers. The following sections provide instructions for manually installing the Integration Manager Enterprise Server to supported application servers.

Runtime Files

To install the Integration Manager Enterprise Server to a supported application server, you need the Integration Manager runtime files, which are contained in zip files (JBossComposerLinux.zip or JBossComposerWindows.zip) which are installed with your Integration Manager installation. Extract the contents of the zip file to a temporary directory. This will create the following directory structure:

\conf

\deploy

\lib

The files in these directories will be used in the procedures that follow.

JBoss

The Integration Manager installation program provides automated installation of the Integration Manager Enterprise Server (also known as the Integration Manager runtime) to JBoss application servers. You can install the Integration Manager Enterprise server to a JBoss application server at any time by running the Integration Manager installation program. If for any reason you need to install the Integration Manager Enterprise server to JBoss Version 4 manually, follow these steps:

NOTE: Note: The steps to manually deploy the Integration Manager Enterprise Server from versions of Integration Manager before version 5.2.1 are different. Contact Novell Customer Support for instructions.

- 1 Copy the contents of the Runtime Files **conf** directory (see “Runtime Files” on page 1) to the {JBoss install directory}\server\{server configuration directory}\conf directory.
NOTE: The JBoss server **default** directory is set as the server configuration directory in the supplied files. If you want to use a different JBoss server configuration, you will have to modify the xconfig.xml located in the \server\{server configuration directory}\conf directory. Locate the line <XC_DIR Desc="Location of eXtendComposer Installation">../server/default/deploy</XC_DIR> in the xconfig.xml file, and change "default" to the name of the server configuration directory that you want to use.
- 2 Copy the contents of the Runtime Files **deploy** directory (see “Runtime Files” on page 1) to the {JBoss install directory}\server\{server configuration directory}\deploy directory.
- 3 Copy the contents of the Runtime Files **lib** directory (see “Runtime Files” on page 1) to the {JBoss install directory}\server\{server configuration directory}\lib directory.
- 4 Create a text file named **users.properties** with the contents "novell=novell" in the {JBoss install directory}\server\{server configuration directory}\conf directory.

- 5 Create a text file named **roles.properties** with the contents "novell=ComposerAdministrators" in the {jboss install directory}\server\{server configuration directory}\conf directory.
- 6 Start JBoss.
You can now start the Integration Manager Enterprise Server using the URL "http://localhost:8080/exteNdComposer". Use "novell" for both the User ID and Password.

WebLogic 8.1.1

Follow these steps to manually deploy the Integration Manager Enterprise Server to WebLogic 8.1.1.

- 1 Create a directory named Composer in the weblogic81 directory.
- 2 Create the following subdirectories in the Composer directory:
 - bin
 - console
 - lib
 - log
 - tmp
- 3 Copy the contents of the Runtime Files **lib** directory (see **"Runtime Files" on page 1**) to the weblogic81\Composer\lib directory.
- 4 Move the **exteNdComposer.ear** file from the weblogic81\Composer\lib directory to the weblogic81\Composer\console directory.
- 5 Copy the **xconfig.xml** file from the Runtime Files **conf** directory (see **"Runtime Files" on page 1**) to the weblogic81\Composer\lib directory.
- 6 Edit the **xconfig.xml** file in the weblogic81\Composer\lib directory and make the following changes:
 - 6a Locate the **XC_DIR** element (near the beginning of the xconfig.xml file) and change the path contained in this element from "../server/default/deploy" to the path to the Composer directory that you created in step 1 (for example, D:\bea\weblogic81\Composer).
 - 6b Locate the **JBOSS_40** element and replace everything between <JBOSS_40> and </JBOSS_40> with the following:


```
<WEBLOGIC_600>
  <ADMIN_HOST>Your_Host_Name_or_IP</ADMIN_HOST>
  <ADMIN_PORT>7001</ADMIN_PORT>

  <HELPER_CLASS>com.sssw.b2b.xs.deploy2.wl70.GXSWL70DeployHelper</HELPER_CLASS>
  <WEBLOGIC_HOME>D:\bea\weblogic81</WEBLOGIC_HOME>
  <STAGING_LOCATION>D:\bea\weblogic81\Composer\tmp</STAGING_LOCATION>
  <DEPLOY_INPUT>deploy/wl60/DeployInput.jsp</DEPLOY_INPUT>
  <DEPLOY_HANDLER>Composer.wl60.DeployHandler</DEPLOY_HANDLER>
</WEBLOGIC_600>
```

NOTE: Change the host name contained in the **ADMIN_HOST** element to match your environment. Change the paths contained in the **WEBLOGIC_HOME** and **STAGING_LOCATION** elements to match the paths in your environment.
- 7 Edit the **startWebLogic.cmd** file in the directory for your domain (for example, \weblogic81\user_projects\domains\{your domain}), and make the following additions:
 - 7a Before the **set CLASSPATH** statement, add a variable called **XCSCCLASSPATH**, and assign to the variable the path to the **xcs-all.jar** file in the Composer **lib** directory. For example:


```
set XCSCCLASSPATH=D:\bea\weblogic81\Composer\lib\xcs-
all.jar;D:\bea\weblogic81\Composer\lib\;
```
 - 7b Add the **XCSCCLASSPATH** variable to the **CLASSPATH** variable. For example (additions that you must add to the supplied files are shown in bold):

```

set
CLASSPATH=%XCS_CLASSPATH%;%WEBLOGIC_CLASSPATH%;%POINTBASE_CLASSPATH%;%JAVA_HOM
E%\jre\lib\rt.jar;%WL_HOME%\server\lib\webservices.jar;%CLASSPATH%

```

- 8** Edit the `fileRealm.properties` file in the `{install path}\bea81\user_projects\domains\{your domain}` directory and add the line `group.ComposerAdministrators=system` to the file.
- 9** Start the WebLogic application server.
- 10** Using the WebLogic Administration console, select Security | Realms | {your realm} | Groups.
 - 10a** Click on the **Configure New Group** link.
 - 10b** Create a Group named **ComposerAdministrators**.
 - 10c** Select the Membership tab.
 - 10d** Assign the ComposerAdministrators group to the Administrators group.
- 11** Using the WebLogic Administration console, select Security > Realms > {your realm} > Users.
 - 11a** Select the User name of the Composer Administrator.
 - 11b** Select the Groups tab.
 - 11c** Assign the user to the group ComposerAdministrators.
- 12** Using the WebLogic Administration console, select {your domain} | Servers | Deployments | Applications.
 - 12a** Select the **Deploy a New Application** link in the right frame of the WebLogic Administration console window.
 - 12b** Using the Location links, navigate to the `\weblogic81\Composer\console` directory, select the `exteNdComposer.ear` file, then select Continue.
 - 12c** Select Deploy.

You can now start the Integration Manager Enterprise Server using the URL "http://localhost:7001/exteNdComposer". Use the user name and password of the user that you assigned to the ComposerAdministrator role.

WebSphere 5

Follow these steps to manually deploy the Integration Manager Enterprise Server to WebSphere 5.

- 1** In the `\WebSphere\AppServer` directory, create the following directory structure:


```

Composer
  bin
  console
  lib
  log
  tmp
      
```
- 2** Copy the `updateLicense.bat` and `updateLicense.sh` files from the Runtime Files `conf` directory (see "Runtime Files" on page 1) to the `\WebSphere\AppServer\Composer\bin` directory.
- 3** Copy the contents of the `lib` directory from the Runtime Files `lib` directory (see "Runtime Files" on page 1) to the `\WebSphere\AppServer\Composer\lib` directory.
- 4** Move the `exteNdComposer.ear` file from the `\WebSphere\AppServer\Composer\lib` directory to the `\WebSphere\AppServer\Composer\console` directory.
- 5** Copy the `xconfig.xml` file from the Runtime Files `conf` directory (see "Runtime Files" on page 1) to the `\WebSphere\AppServer\Composer\lib` directory.
- 6** Edit the `xconfig.xml` file in the `\WebSphere\AppServer\Composer\lib` directory and make the following changes:

- 6a** Locate the **XC_DIR** element (near the beginning of the xconfig.xml file) and change the path contained in this element from "../server/default/deploy" to the path to the Composer directory that you created in step 1 (for example, D:\WebSphere\AppServer\Composer).
- 6b** Locate the **JBOSS_40** element and replace everything between `<JBOSS_40>` and `</JBOSS_40>` with the following:

```
<WEBSPHERE_40>
  <ADMIN_HOST>Your_Server_Name_or_IP</ADMIN_HOST>
  <ADMIN_PORT>2809</ADMIN_PORT>
  <STAGING_LOCATION>D:\WebSphere\AppServer\Composer\tmp</STAGING_LOCATION>
  <DEPLOY_INPUT>deploy/ws50/DeployInput.jsp</DEPLOY_INPUT>
  <DEPLOY_HANDLER>Composer.ws50.DeployHandler</DEPLOY_HANDLER>
</WEBSPHERE_40>
```

NOTE: Change the host name or IP address contained in the **ADMIN_HOST** element to match your environment. Change the path contained in the **STAGING_LOCATION** element to match the paths in your environment.

- 7** Start the WebSphere Application Server.
- 8** Start the WebSphere Administrative Console.
- 9** Select Environment > Manage WebSphere Variables.
- 10** Create a new variable with the following values:
Name: **EXTEND_COMPOSER_HOME**
Value: `$(WAS_INSTALL_ROOT)/Composer`
Description: The root directory for the Integration Manager installation.
- 11** Create another new variable, with the following values:
Name: **EXTEND_COMPOSER_LIB**
Value: `${EXTEND_COMPOSER_HOME}/lib`
Description: The Integration Manager directory where the system Jars can be located.
- 13** Select Servers > Application Servers, then select the name of the application server to which you want to install the Integration Manager Enterprise Server. The Administrative Console displays the configuration page for the application server.
- 14** Select **Process Definition**.
- 15** Select **Java Virtual Machine** from the **Additional Properties** list.
- 16** Select **Custom Properties** from the **Additional Properties** list.
- 17** Create a new custom property, with the following values:
Name: `ws.ext.dirs`
Value: `${EXTEND_COMPOSER_LIB}`
Description: The Integration Manager extension directory
- 18** Select Applications > Enterprise Applications.
- 19** Select Install.
- 20** Select the `exteNdComposer.ear` file in the `{WebSphere install directory}\AppServer\Composer\console` directory.
- 21** Accept the default settings for bindings.
- 22** Accept the default settings for AppDeployment Options.
- 23** Select the appropriate options for mapping virtual hosts to web modules, and mapping modules to application servers.
- 24** Map the ComposerAdministrators security role to the appropriate users and groups.
- 25** Select Finish to complete the installation.
- 26** Save the changes to the application server master configuration.
- 27** Stop and restart the application server before attempting to start the Integration Manager Enterprise Server.

You can now start the Integration Manager Enterprise Server using the URL "http://localhost:9080/exteNdComposer".

exteNd Application Server 5.2

Follow these steps to manually deploy the Integration Manager Enterprise Server to the exteNd Application Server Version 5.2.

- 1 In the \exteNd5\AppServer directory, create the following directory structure:
Composer
 bin
 console
 lib
 log
 tmp
- 2 Copy the contents of the **lib** directory from the Runtime Files **lib** directory (see “Runtime Files” on page 1) to the \exteNd5\AppServer\Composer\lib directory.
- 3 Move the **exteNdComposer.ear** file from the \exteNd5\AppServer\lib directory to the \exteNd5\AppServer\Composer\console directory.
- 4 Copy the **xconfig.xml** file from the Runtime Files **conf** directory (see “Runtime Files” on page 1) to the \exteNd5\AppServer\Composer\lib directory.
- 5 Edit the **xconfig.xml** file in the \exteNd5\AppServer\Composer\lib directory and make the following changes:
 - 5a Locate the **XC_DIR** element (near the beginning of the xconfig.xml file) and change the path contained in this element from "../server/default/deploy" to the path to the Composer directory that you created in step 1 (for example, D:\exteNd5\AppServer\Composer).
 - 5b Locate the **JBOSS_40** element and replace everything between <JBOSS_40> and </JBOSS_40> with the following:

```
<NOVL>  
  <BIN_DIR>../BIN_DIR</BIN_DIR>  
  <DEPLOY_INPUT>deploy/nov1/DeployInput.jsp</DEPLOY_INPUT>  
  <DEPLOY_HANDLER>Composer.nov1.DeployHandler</DEPLOY_HANDLER>  
</NOVL>
```

- 6 Copy the following code into a text file, and save the file in the \exteNd5\AppServer\Composer\console directory, using the name **exteNdComposerEARDeplPlan.xml**:

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE earJarOptions PUBLIC "-//SilverStream Software, Inc.//DTD J2EE EAR  
Deployment Plan 1.3//EN" "deploy-ear_1_3.dtd">  
  
<earJarOptions isObject="true">  
  <earJar isObject="true">  
    <version type="String">1.0</version>  
    <earJarName type="String">extendComposer.ear</earJarName>  
    <moduleList isObject="true">  
      <module isObject="true">  
        <warJar isObject="true">  
          <warJarName type="String">exteNdComposer.war</warJarName>  
          <isEnabled type="Boolean">True</isEnabled>  
          <deployToFilesystem>False</deployToFilesystem>  
          <urls type="StringArray">  
            <el>exteNdComposer</el>  
          </urls>  
          <deployedObject  
type="String">exteNdComposer</deployedObject>  
          <excludedJSPs type="StringArray">  
            <el>deploy/wl60/DeployInput.jsp</el>
```

```

        <el>deploy/wl60/DeployInput-page2.jsp</el>
        <el>deploy/wl60/DeployInput-page3.jsp</el>
        <el>deploy/wl60/DeployInput-validate-page2.jsp</el>
        <el>deploy/wl60/DeployInput-validate.jsp</el>
        <el>deploy/ws40/DeployInput.jsp</el>
        <el>deploy/ws40/DeployInput-page2.jsp</el>
        <el>deploy/ws40/DeployInput-page2-validate.jsp</el>
        <el>deploy/ws40/DeployInput-page3.jsp</el>
        <el>deploy/ws40/DeployInput-validate.jsp</el>
    </excludedJSPs>
    <roleMap isObject="true">
        <roleMapping isObject="true">
            <name type="String">ComposerAdministrators</name>
            <userOrGroupName
type="String">ComposerAdministrators</userOrGroupName>
            </roleMapping>
        </roleMap>
    </warJar>
</module>
</moduleList>
<roleMap isObject="true">
    <roleMapping isObject="true">
        <name type="String">ComposerAdministrators</name>
        <userOrGroupName
type="String">ComposerAdministrators</userOrGroupName>
    </roleMapping>
</roleMap>
</earJar>
</earJarOptions>

```

7 Locate the AgJars.conf file in the \exteNd5\AppServer\bin directory and open the file in a text editor.

8 Add the following lines to the "MODULE COMMON" section, near the end of the file:

```

$$$_LIB ../Composer/lib/xcs-all.jar
$$$_LIB ../Composer/lib/xcs-ejb-res-stub.jar
$$$_LIB ../Composer/lib/

```

NOTE: You may need to add other JARs, depending on the Integration Manager Connects that you are using (for example, \$\$\$_LIB ../Composer/lib/xci-way.jar, \$\$\$_LIB ../Composer/lib/iwpsci.jar). See your Integration Manager Connect documentation for the JARs needed for a specific Integration Manager Connect.

9 Start the application server.

10 From a command prompt, navigate to the \exteNd5\AppServer\bin directory.

11 Type the following command at the command prompt (replacing the information in brackets with the information that is appropriate for your environment):

```

silvercmd deployEAR {server name}:{http port number 80} SilverMaster50 {server
install path}\AppServer\Composer\console\exteNdComposer.ear -f {server install
path}\AppServer\Composer\console\exteNdComposerEARDeplPlan.xml -U {admin} -P
{admin password}.

```

For example, if your server is named "myserver", is set up to use port 80 for http (this is the default setting), you have installed the application server in the default location, and the administrator is named "admin" with the password "admin", you would type:

```

silvercmd deployEAR myservers:80 SilverMaster50 C:\Program
Files\Novell\exteNd5\AppServer\Composer\console\exteNdComposer.ear -f
C:\Program
Files\Novell\exteNd5\AppServer\Composer\console\exteNdComposerEARDeplPlan.xml
-U admin -P admin.

```

This command deploys the Integration Manager Enterprise Server to the application server. The next step is to add a security group and role for the Integration Manager Enterprise Server to the application server.

12 Start the exteNd application server Server Management Console.

- 13** Select **View > Security**.
- 14** Select **Users & Groups**.
- 15** Double-click **Silver Security**.
- 16** Double-click **Groups**.
- 17** Add a new group named **ComposerAdministrators**.
- 18** Click on **ComposerAdministrators**, then select the **Add User to Group** button from the buttons on the bottom of the page (hold the mouse pointer over a button to display a tool tip that gives the button name).
- 19** In the **Add Users to Group** dialog box, double-click **SilverSecurity**, then **Users**.
- 20** Click on the name of the user that you want to add to the **ComposerAdministrators** group, then select **Add**.
- 21** Select **Close**.
- 22** Select **File > Restart Server**.

After the application server has restarted, you can start the Integration Manager Enterprise Server using the URL "http://localhost:{port number}/exteNdComposer". The default port number is 80.

Apache Tomcat 4

Follow these steps to manually deploy the Integration Manager Enterprise Server to the Apache Tomcat application server.

- 1** Make sure that the Tomcat application server is *not* running.
- 2** In the Tomcat directory, create the following directory structure:


```
Composer
  bin
  console
  lib
  log
  tmp
```
- 3** Copy the contents of the **lib** directory from the Runtime Files **lib** directory (see **“Runtime Files” on page 1**) to the {Tomcat install directory}\Composer\lib and the {Tomcat install directory}\shared\lib directories.
- 4** Use an archive tool (for example, WinZip, File Roller) to extract the **exteNdComposer.war** file from the **exteNdComposer.ear** file, and place one copy of the **exteNdComposer.war** file in the {Tomcat install directory}\Composer\console directory, and another copy in the {Tomcat install directory}\webapps directory.
- 5** Copy the **xconfig.xml** file from the Runtime Files **conf** directory (see **“Runtime Files” on page 1**) to the {Tomcat install directory}\Composer\lib directory.
- 6** Edit the **xconfig.xml** file in the {Tomcat install directory}\Composer\lib directory and make the following changes:
 - 6a** Locate the **XC_DIR** element (near the beginning of the **xconfig.xml** file) and change the path contained in this element from **"../server/default/deploy"** to the path to the Composer directory that you created in step 1 (for example, **D:\Tomcat 4.1.24\Composer**).
 - 6b** Locate the **JBOSS_40** element and replace everything between **<JBOSS_40>** and **</JBOSS_40>** with the following:

```
<TOMCAT_41>
  <ADMIN_HOST>localhost</ADMIN_HOST>
  <ADMIN_PORT>8080</ADMIN_PORT>
  <TOMCAT_HOME>D:\Tomcat 4.1.24</TOMCAT_HOME>
  <STAGING_LOCATION>D:\Tomcat 4.1.24\Composer\tmp</STAGING_LOCATION>
  <DEPLOY_INPUT>deploy/tc41/DeployInput.jsp</DEPLOY_INPUT>
```

```
<DEPLOY_HANDLER>Composer.tc41.DeployHandler</DEPLOY_HANDLER>
</TOMCAT_41>
```

NOTE: Change the host name or IP address contained in the **ADMIN_HOST** element to match your environment. Change the paths contained in the **TOMCAT_HOME** and **STAGING_LOCATION** elements to match the paths in your environment.

- 7 Using a text editor, open the **tomcat-users.xml** file in the {Tomcat install directory}\conf directory, and add the following lines:

```
<user password="novell" roles="admin,manager,ComposerAdministrators"
username="novell"/>
<role rolename="ComposerAdministrators"/>
```

NOTE: Substitute any desired username and user password for "novell".

- 8 Using a text editor, open the **setclasspath.bat** (Windows) or **setclasspath.sh** (Linux) file in the {Tomcat install directory}\bin directory, and add the path to the \Composer\lib directory to the CLASSPATH variable. For example, if you installed Tomcat in a directory named Tomcat 4.1.24, the CLASSPATH variable assignment would look like this (additions that you must add to the supplied files are shown in bold):

Windows

```
set CLASSPATH=%JAVA_HOME%\lib\tools.jar;D:\Tomcat 4.1.24\Composer\lib
```

Linux

```
CLASSPATH="$JAVA_HOME"/lib/tools.jar:home/jdoe/jakarta-tomcat-4.1.24/Composer/lib
```

- 9 From a command prompt, navigate to the {Tomcat install directory}\bin directory, and start Tomcat by running the **startup.bat** (Windows) or **startup.sh** (Linux\Unix) file.
- 10 You can now start the Integration Manager Enterprise Server using the URL "http://localhost:8080/exteNdComposer". Use the user name and password of the user that you assigned to the ComposerAdministrator role in **Step 7**.