Using the IBM Solution Assembly Toolkit

Using the IBM Solution Assembly Toolkit

Contents

Figures

Tables

Chapter 1. Introduction to the IBM Solution Assembly Toolkit

The IBM® Solution Assembly Toolkit provides an easy method for integrating custom applications with middleware to deliver a total business solution as a single deliverable.

When you provide a total business solution as a single deliverable, you simplify the process of deploying a solution and guarantee that a solution is configured correctly. Using the IBM Solution Assembly Toolkit, you specify configuration parameters during the solution creation, which later prevent end users from installing a solution incorrectly. You can provide a solution directly to end users as needed, without requiring further assistance for configuration or integration. The IBM Solution Assembly Toolkit greatly reduces the complexity and time required to deploy a total business solution to mid-sized company customers.

IBM Solution Assembly Toolkit audience

This information is intended for solution providers who plan to create, configure, and deploy solutions. It assumes no previous experience with the IBM Solution Assembly Toolkit.

This information provides a description of the basic processes necessary to use the IBM Solution Assembly Toolkit effectively. This information provides an overview of the end-to-end process of using the IBM Solution Assembly Toolkit, and in-depth information on the following topics:

- The value of using the IBM Solution Assembly Toolkit.
- Tools and components included with the IBM Solution Assembly Toolkit.
- User roles typically required to use the IBM Solution Assembly Toolkit.
- The process for developing a solution.
- Options for deploying a solution.

Using the IBM Solution Assembly Toolkit to create, configure, and deploy total business solutions requires several core skills. A user role defines a set of the core skills required to perform a specific activity. The IBM Solution Assembly Toolkit users can be described by four user roles. One individual might complete the tasks of multiple user roles, but each user role must have at least one person capable of completing its necessary tasks. You need to identify individuals to take on the following user roles:

Application developer
The application developer creates the business partner application.

Solution developer
The solution developer builds solutions to integrate the installation, configuration, or integration of applications with required middleware. The solution developer should be familiar with Java™ and XML.

Solution distributor
The solution distributor distributes solutions to customers.

End user
The end user completes the deployment of a solution and might perform solution maintenance and administrative tasks.

Elements of Solution Assembly Toolkit

IBM Solution Assembly Toolkit provides the packaging and installation technology to create and deploy complete packages across multiple platforms. It can greatly reduce the time, effort and complexity associated with installing and configuring applications.

Solution Assembly Toolkit consists of the following elements:

- Express Runtime developer
- Deployment wizard
- IBM Remote Execution and Access (RXA)
- Solution launcher

Express Runtime developer

The Express Runtime developer is an Eclipse-based plug-in that provides a standard development environment.

To create solutions, you must develop or modify existing wrappers. The Express Runtime developer includes custom editors that allow you assemble solutions quickly and easily. The custom editors automatically generate wrappers as you provide simple information about applications and solutions.

What is a wrapper?

A wrapper is an Express Runtime developer project that refers to an XML file that defines the way an application or solution is installed and configured on the target computer.

An application wrapper helps to install an application on a target computer silently, and does so while exposing the least possible amount of variables to the user. Typically a wrapper corresponds to a compressed image that contains an installation program. A solution wrapper includes one or more application wrappers. You can use solution wrappers to define the way a solution is installed and configured, and include multiple items of middleware and custom applications to target computers.

Deployment packages:

A deployment package is a JAR file that contains the software image files needed to run the application installation.

You specify the software image files to include in a deployment package on the Files tab of the application project editor, and the deployment package is created when you select to **Generate Deployment Packages**.

The deployment package is stored outside of the application wrapper's project in the IRU_common_resources project. The IRU_common_resources project is a repository for common resources that multiple projects might need to access. Express Runtime developer creates the IRU_common_resources project as needed. The package is placed in the mediaJars folder of this project and the deployment package builder's log file is stored in the bin/*application ID*/log folder and is named softwareImageDeploymentPkgBuilder.log.

User programs:

User programs run during the application deployment to perform the main installation, and pre- or post-installation operations. A user program can be a Java program, a stand-alone executable, or a script, but it must be able to run silently.

All application projects require a main user program. Other user programs, such as a predeployment checker, an entry program, and an exit program, can also be included.

For more information about user programs, see "Providing user programs information" on page 88.

Deployment wizard

The deployment wizard is an easy-to-use tool for deploying solutions. It can be used by solution distributors and end users. End users can use solution launcher image CDs or DVDs to automatically start the deployment wizard. With the deployment wizard started, the end customer is guided through the deployment of your solution. A solution developer can choose to expose as much or as little of the configuration information of a solution as you believe necessary for end users. By limiting the configuration information, you install the solution with the configuration that optimizes the performance of the solution and provides the necessary security to run according to your recommendations.

The deployment wizard enables the deployment of solutions either locally or remotely and provides help with the following tasks:

- Selecting tasks to deploy
- Specifying target computers for each task
- Configuring deployment parameters for each task
- Reviewing task summary information
- Deploying one or more tasks
- Monitoring deployment status

IBM Remote Execution and Access (RXA)

The deployment wizard uses IBM Tivoli® Remote Execution and Access (RXA) to deploy solutions to target computers that you specify. The RXA component establishes a connection and logs on to the target computer, manipulates files and directories on the target computer, installs and uninstalls a deployment service or daemon and cleans up setup files that are not required for the deployed solution to function.

Solution launcher

The solution launcher is a utility that helps you start the deployment wizard and deploy a solution directly from a CD or DVD. With a solution launcher image, you can give your solution directly to end users for easy installation. Creating a solution launcher image is simple. The image can be created by an export wizard within the Express Runtime developer. The solution developer should complete the creation and testing of your solution. Then, export a completed solution as a solution launcher image. With the image exported, the solution distributor can use media burning software to create CDs or DVDs of a solution for distribution to end users.

Chapter 2. Overview of an integrated solution

To provide an environment for a business solution, middleware needs to be installed and configured to use with one or more custom applications.

To install a business solution manually, you need to perform several installations, and then complete a configuration process to enable all of the components of the solution to work together. Solution Assembly Toolkit enables you to create an integrated solution where all the components are installed and configured in one straight forward process.

Workflow for creating an integrated solution

The workflow for creating an integrated solution follows four basic steps. The goal of these four steps is to simplify the integration and deployment of an application with IBM middleware as one installation.

The following four steps provide a high-level, end to end view of the process for creating an integrated solution:

- “Step 1: Develop application wrappers for components of your solution ”
- “Step 2: Develop a solution wrapper for your solution”
- “Step 3: Package your solution as a solution launcher image” on page 6
- “Step 4: Deploy the solution to customers” on page 6

Step 1: Develop application wrappers for components of your solution

Use the Express Runtime developer to create application wrappers that describe how to automate the installation process for each component. An application wrapper includes the following information:

- A list of parts from the installation image
- The syntax of the command used to run the installation
- Identification of installation parameters you want to expose to the user
- Instructions for how to feed those parameters into the installation process

For J2EE applications, use the WebSphere Application Server deployment project wizard to automatically create the application wrapper.

This information is compiled into binary form in the application wrapper so that it can be included in an integrated solution.

Step 2: Develop a solution wrapper for your solution

Use the Express Runtime developer to create a solution wrapper that references the application projects that comprise your solution. This solution wrapper defines the contents, grouping, and sequence of the applications. The solution wrapper also defines which applications or groups of applications are selectable by the user.

This solution wrapper is compiled into binary form so that it can be included in a solution launcher image.

Step 3: Package your solution as a solution launcher image

Use the Express Runtime developer to export the solution wrapper in the format of an installable disk or set of disks.

Step 4: Deploy the solution to customers

There are two distinct ways to deploy a solution:

- Deploy a solution from one or more CDs or DVDs
- Deploy a solution from a staging server

When a solution is stored on one or more CDs or DVDs, the customer can load the CD or DVD and the deployment wizard starts and walks the customer through the deployment process for the solution.

When a solution is stored on a staging server, the deployment wizard runs on the staging server and walks the person who performs the deployment through the deployment process for the solution.

Chapter 3. IBM Solution Assembly Toolkit Tutorial

In this tutorial, you will work with the samples provided to perform tasks using components of IBM Solution Assembly Toolkit.

1. Use the Express Runtime developer application project editor to perform the following tasks:
   - Create a new application wrapper
   - Import additional resources into the application wrapper
   - Enable a string in the application wrapper for translation
   - Add software image files to the application wrapper
   - Specify the main program information for the application wrapper
   - Specify a predeployment checker program for the application wrapper
   - Add variables to the application wrapper
   - Generate the application and deployment packages
   - Import additional application wrappers to include in the solution wrapper
2. Use the Express Runtime developer solution project editor to perform the following tasks:
   - Create a new solution wrapper
   - Add task groups to the solution
   - Add install tasks to the solution
   - Configure solution tasks
   - Add a startup checkpoint to the solution
   - Share variables between applications in the solution
   - Export the solution to a solution launcher image
3. Use the deployment wizard to perform the following tasks:
   - Deploy the solution

This tutorial chapter is meant to get you started with the IBM Solution Assembly Toolkit and touches on only a portion of what you can do with it. For details about additional functionality, and in-depth explanation of how to accomplish tasks, see the Appendixes section at the back of this document.

Components of the tutorial

Before beginning the tutorial, acquire the necessary files.

To acquire the files needed to complete the tutorial, complete the following steps:

1. Install IBM Solution Assembly Toolkit from the CD provided. For more information about installing IBM Solution Assembly Toolkit, see Appendix F, “Installing the IBM Solution Assembly Toolkit from a disk or network location,” on page 135.
2. Save the samples.zip fle from the CD to your local hard drive and extract the contents.

Notes:

- Make a note of the locations to which you extract the sample files. This information is needed later in the tutorial.

- The versions of WebSphere Application Server Community Edition that are included with the sample files are included for sample purposes only and are not intended to be used outside of the tutorial. If you would like to use the WebSphere Application Server Community Edition application, download the latest version from the IBM developerWorks® Web site at http://www-128.ibm.com/developerworks/downloads/ws/wasce/?S_TACT=105AGX10 &S_CMP=WASCE. Ensure that you agree to the licensing terms before you install the application.

Creating an application wrapper for a sample application

Applications are the building blocks of solutions. To integrate an application into a solution, you must first create an *application wrapper*. An application wrapper is made up of several XML files, user programs, and, optionally, scripts, GIF files, and so on. You create an application wrapper using Express Runtime developer.

To create an application wrapper for the WebSphere® Application Server Community Edition application, complete the following steps:

1. Select **Start** → **Programs** → **IBM Solution Assembly Toolkit 2.2** → **Express Runtime Developer** to open Express Runtime developer.
2. Select **File** → **New** → **Application Project** to create a new project.

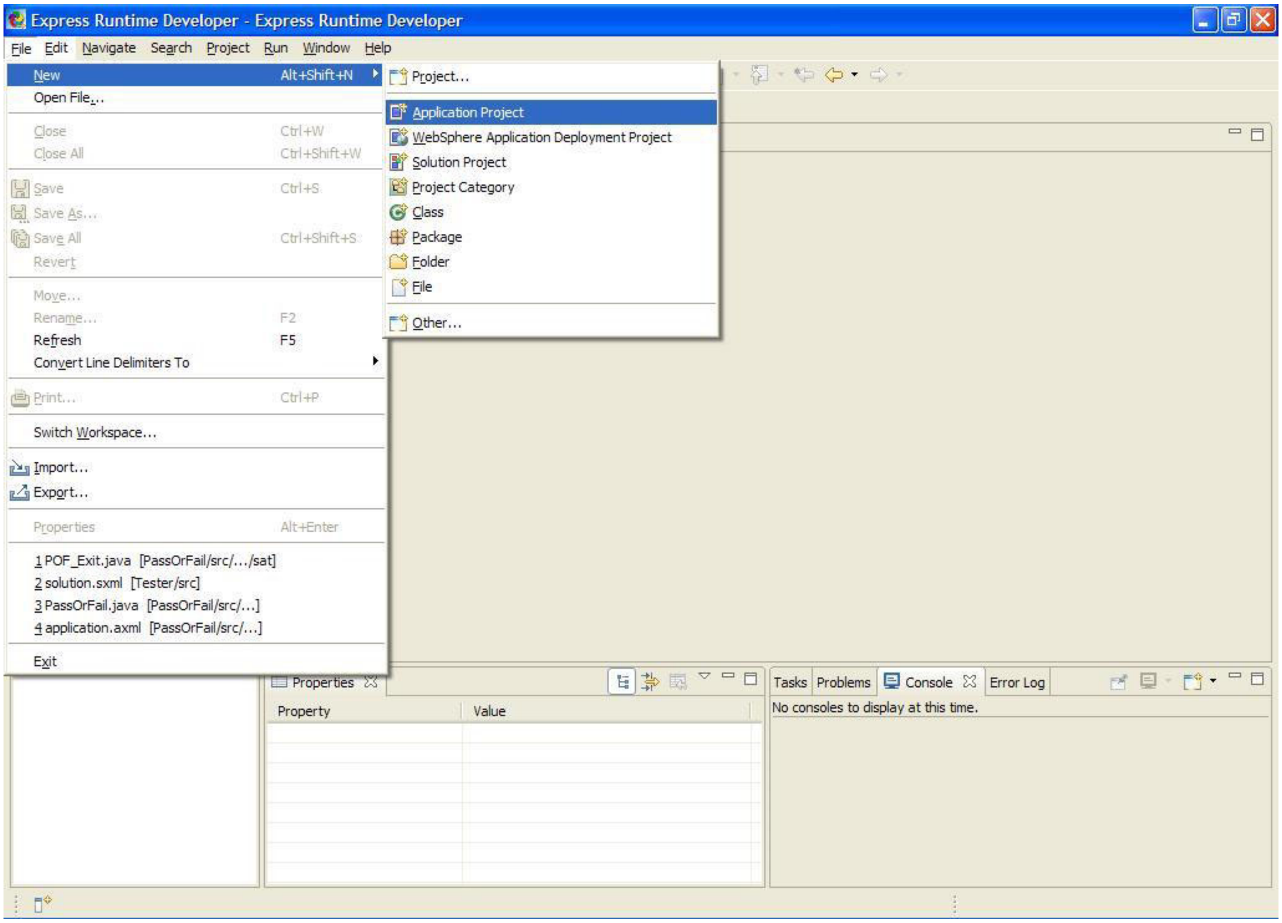
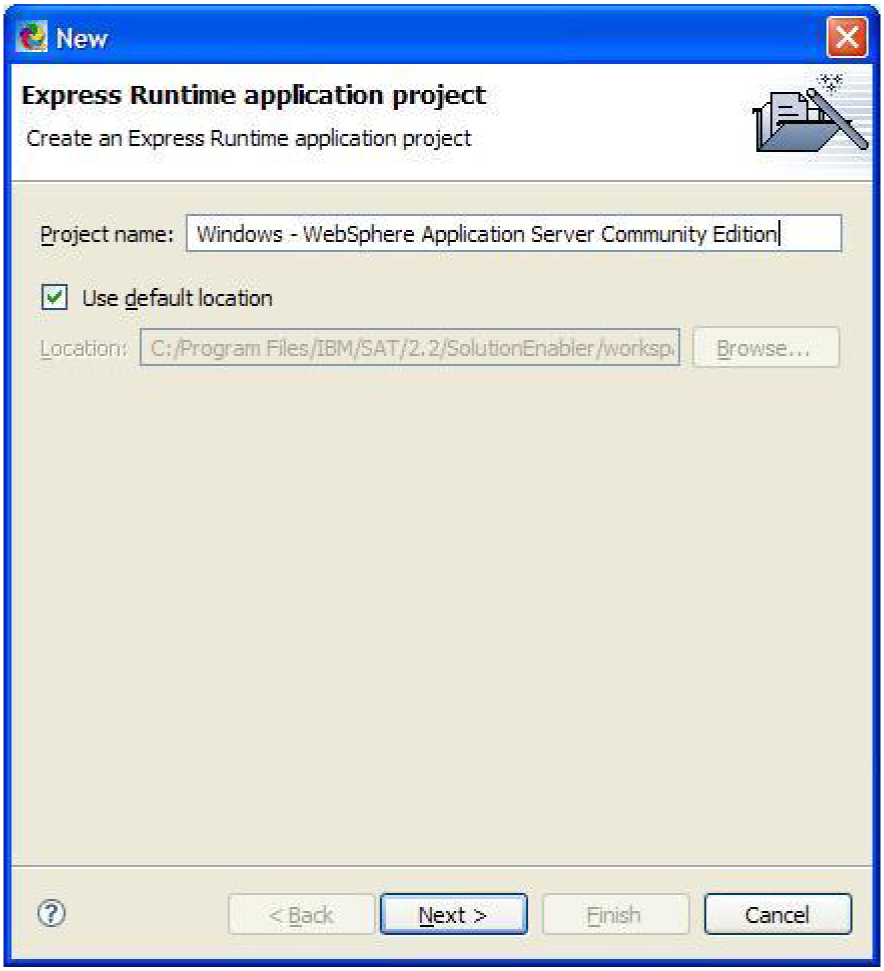


Figure 1. Creating a new application project

The New application project wizard opens.

3. In the **Project name** field, type `Windows - WebSphere Application Server Community Edition`.



Express Runtime application project

Create an Express Runtime application project

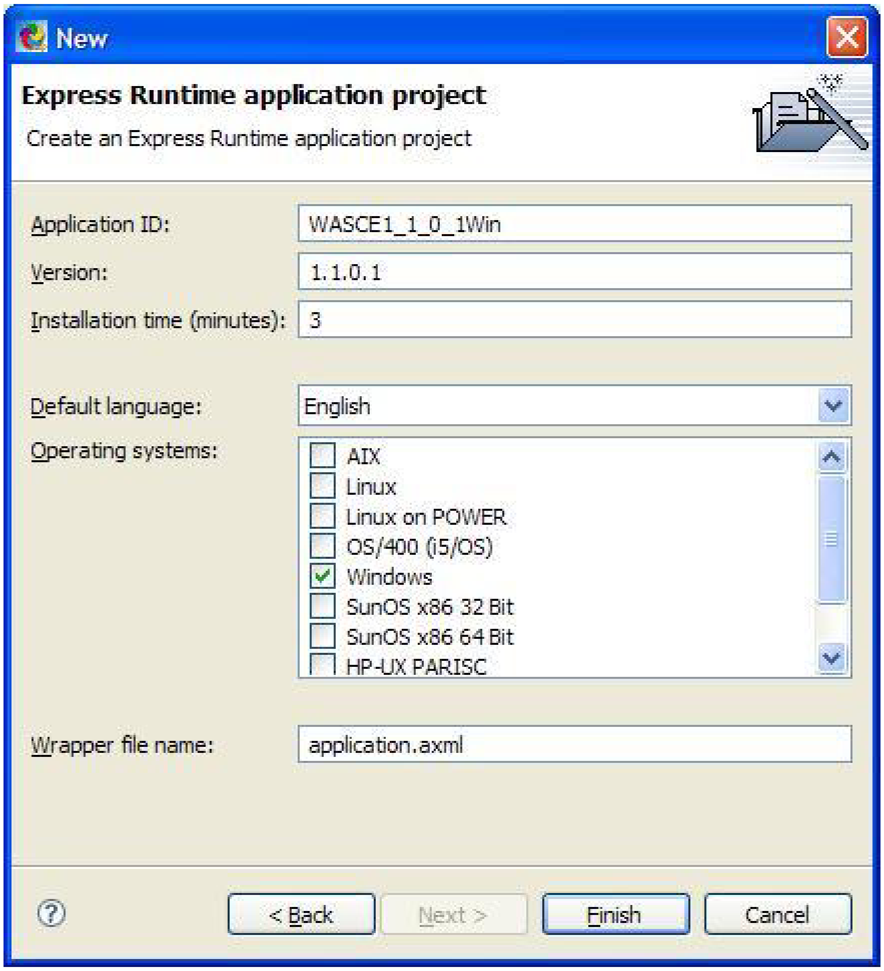
Project name: Windows - WebSphere Application Server Community Edition

☑ Use default location

Location: C:/Program Files/IBM/SAT/2.2/SolutionEnabler/worksp

Figure 2. New application project wizard: First page

4. Click **Next**.
5. On the next page of the wizard, type or select the following values, as shown in Figure 3 on page 10:
   - **Application ID**: `WASCE1_1_0_1Win`
   - **Version**: `1.1.0.1`
   - **Installation time (minutes)**: 3
   - **Default language**: English
   - **Operating systems**: Windows®
   - **Wrapper file name**: application.axml (default value)


New

Express Runtime application project

Create an Express Runtime application project

Application ID: WASCE1_1_0_1Win

Version: 1.1.0.1

Installation time (minutes): 3

Default language: English

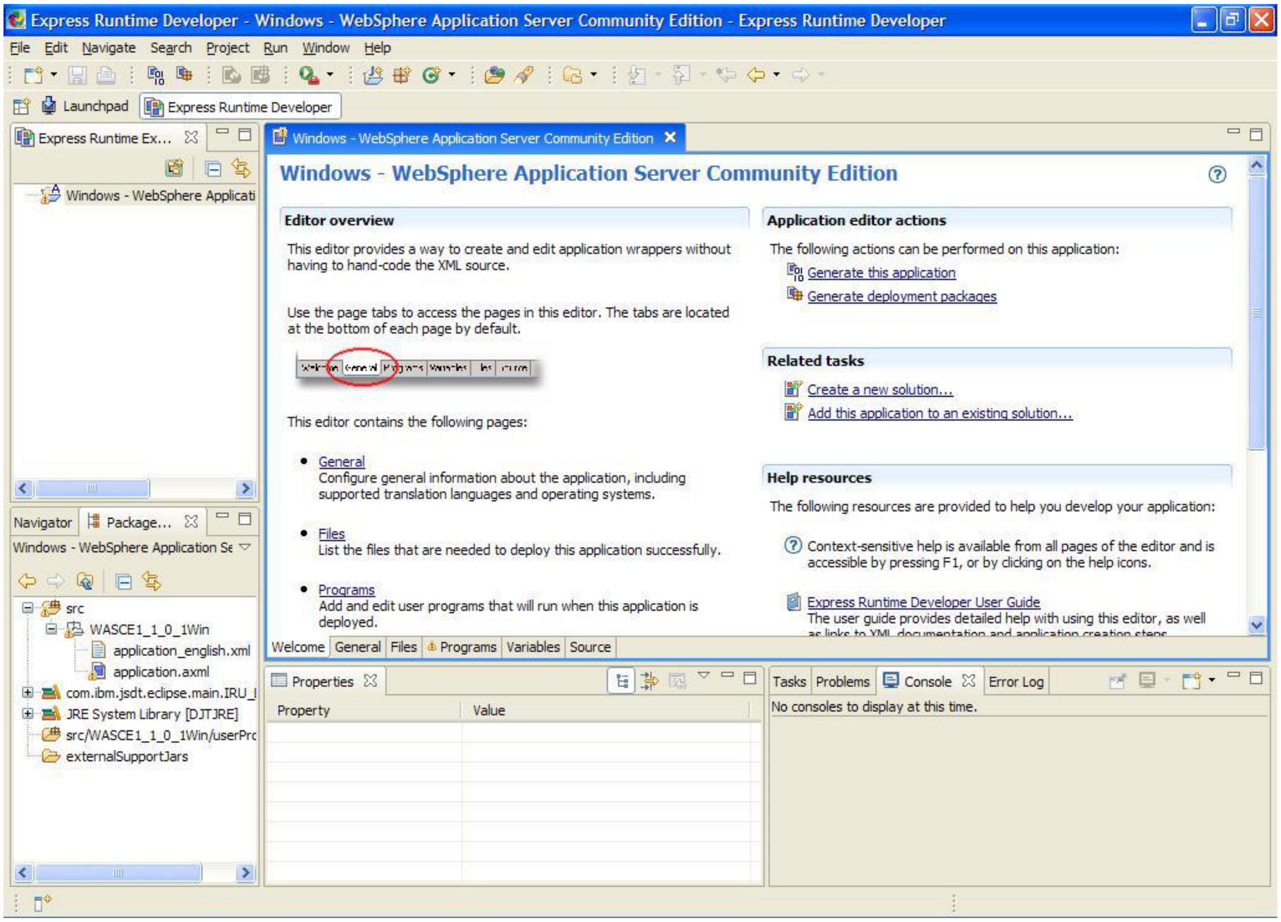
Operating systems:

- ☐ AIX
- ☐ Linux
- ☐ Linux on POWER
- ☐ OS/400 (i5/OS)
- ☑ Windows
- ☐ SunOS x86 32 Bit
- ☐ SunOS x86 64 Bit
- ☐ HP-UX PARISC

Wrapper file name: application.axml

Figure 3. New application project wizard: Second page

6. Click **Finish**. You are returned to the application project editor. The Welcome page is displayed in the right pane of the window. The Express Runtime Explorer is displayed in the upper left pane, and the Navigator is displayed in the lower left pane.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
Windows - WebSphere Application Server Community Edition
Navigator
Package...
Windows - WebSphere Application Se
src
WASCE1_1_0_1Win
application_english.xml
application.axml
com.ibm.jsdt.eclipse.main.IRU_I
JRE System Library [DJTJRE]
src/WASCE1_1_0_1Win/userPrc
externalSupportJars
Welcome
General
Files
Programs
Variables
Source
Properties
Property
Value
Tasks
Problems
Console
Error Log

Windows - WebSphere Application Server Community Edition

Editor overview

This editor provides a way to create and edit application wrappers without having to hand-code the XML source.

Use the page tabs to access the pages in this editor. The tabs are located at the bottom of each page by default.

This editor contains the following pages:

- General
  Configure general information about the application, including supported translation languages and operating systems.
- Files
  List the files that are needed to deploy this application successfully.
- Programs
  Add and edit user programs that will run when this application is deployed.

Application editor actions

The following actions can be performed on this application:

Generate this application

Generate deployment packages

Related tasks

Create a new solution...

Add this application to an existing solution...

Help resources

The following resources are provided to help you develop your application:

Context-sensitive help is available from all pages of the editor and is accessible by pressing F1, or by clicking on the help icons.

Express Runtime Developer User Guide
The user guide provides detailed help with using this editor, as well as links to XML documentation and application creation steps

No consoles to display at this time.

Figure 4. Application project editor: Welcome tab

7. Click the **General** tab. The general application information is displayed. Many of the fields contain information you entered when you created the project.

Next, you will import existing resources into the application project.

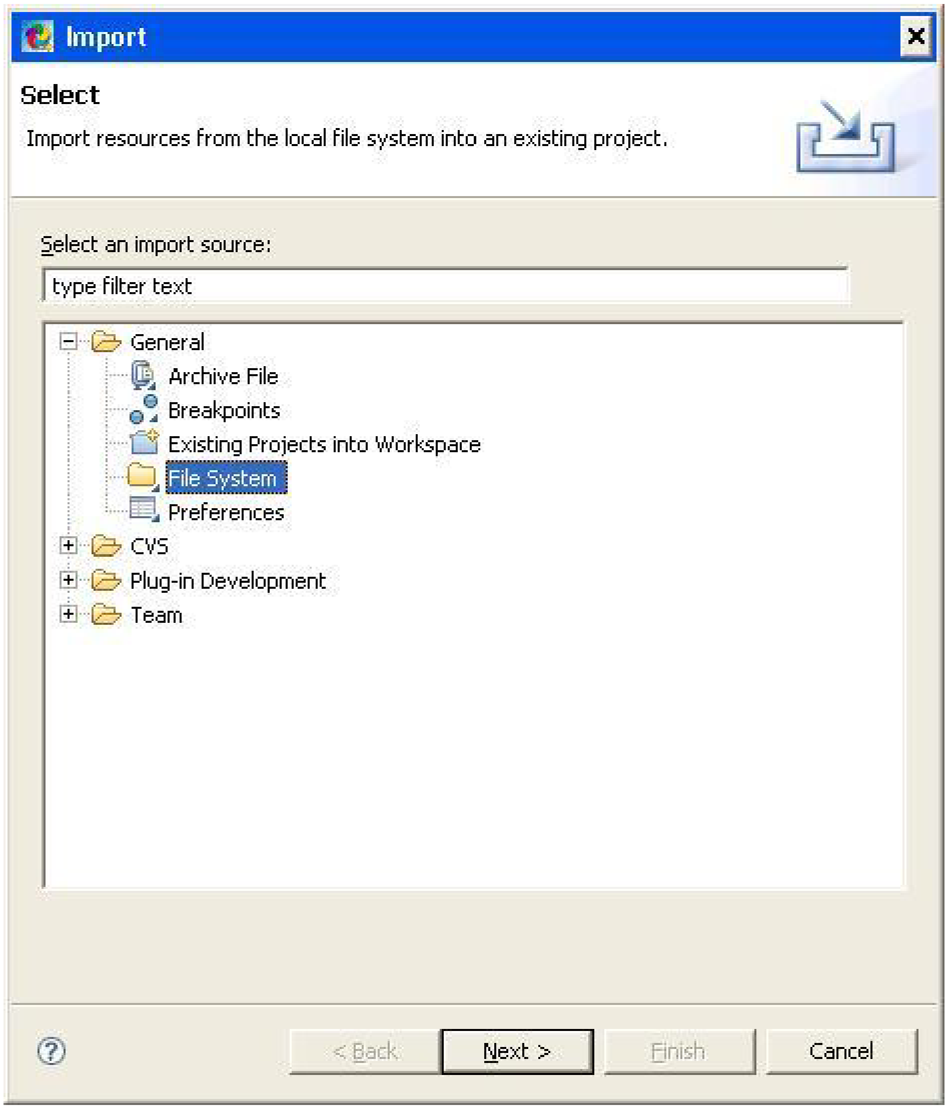
For more information about wrappers, see "What is a wrapper?" on page 2 and "Creating an application wrapper" on page 85.

Importing additional application resources

You can import existing application or solution resources so that they can be used in your new project. You will import the Windows - WebSphere Application Server Community Edition resources to simplify some of the work in the following sections of the tutorial.

To import the Windows - WebSphere Application Server Community Edition application project resources, complete the following steps:

1. Click the Windows - WebSphere Application Server Community Edition application project in the Express Runtime Explorer to select it.
2. Right-click the Express Runtime Explorer pane and click **Import**. The Import wizard: Select import source page opens.
3. Click **General** → **File System** and then click **Next**.



Select

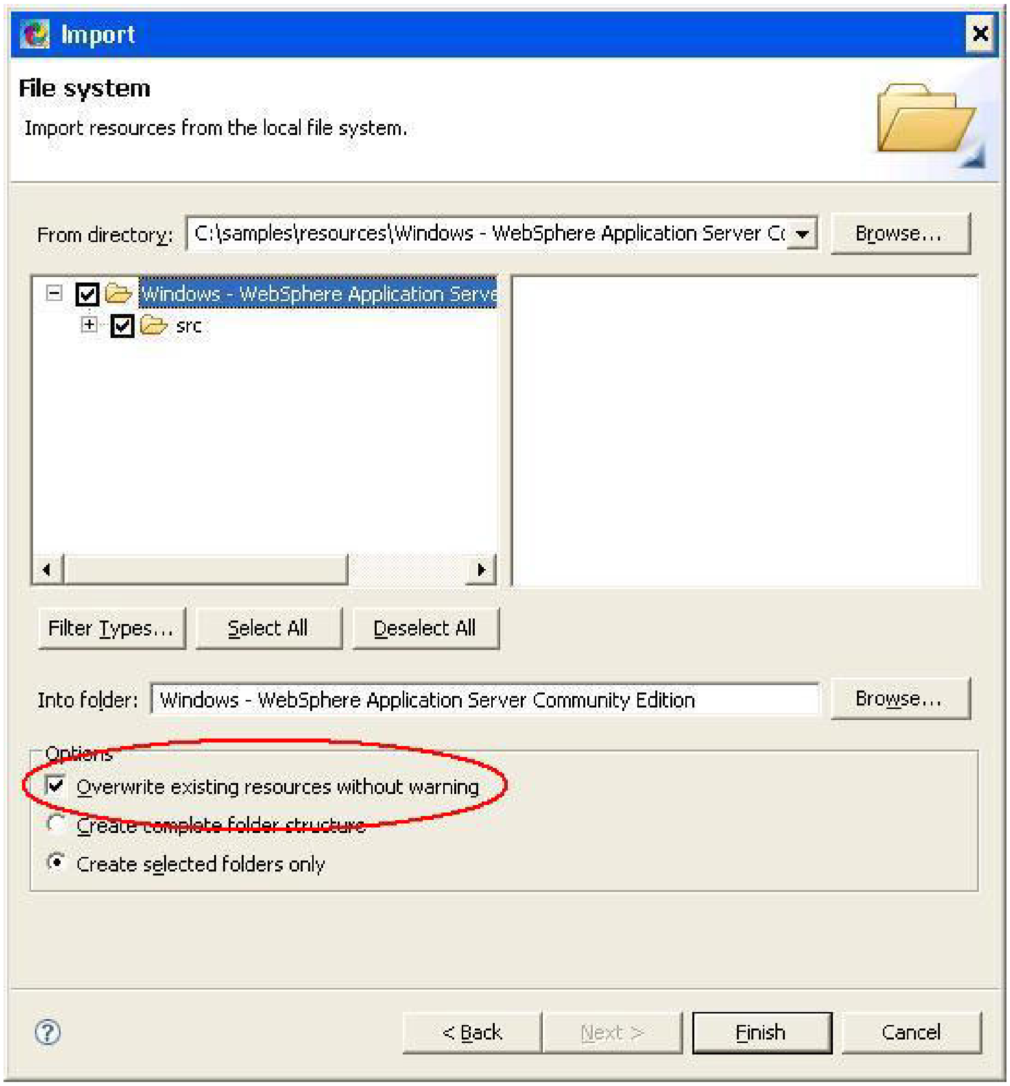
Import resources from the local file system into an existing project.

Select an import source:

type filter text

Figure 5. Import wizard: Select import source page

4. Browse to the directory to which you extracted the contents of the samples.zip file. Locate and expand the **resources** folder.
5. Click the Windows - WebSphere Application Server Community Edition folder, and click **OK**. The Import resources from the local file system window opens.



File system

Import resources from the local file system.

From directory: C:\samples\resources\Windows - WebSphere Application Server C(

☑ Windows - WebSphere Application Serve
☑ src

Into folder: Windows - WebSphere Application Server Community Edition

Options

☑ Overwrite existing resources without warning
○ Create complete folder structure
● Create selected folders only

Figure 6. Import wizard: Select import source page

6. Select the check box next to **Windows - WebSphere Application Server Community Edition**.
7. Ensure that **Overwrite existing resources without warning** is selected, and then click **Finish**.

The resources from the CD are now part of your application project.

Next, you will enable the application name for translation.

Enabling the application name for translation

To translate application information, you must populate the default-language XML file with the translatable strings for the wrapper, giving each string a meaningful key name. Keys must consist of alphanumeric characters only (no white space), and must begin with a letter.

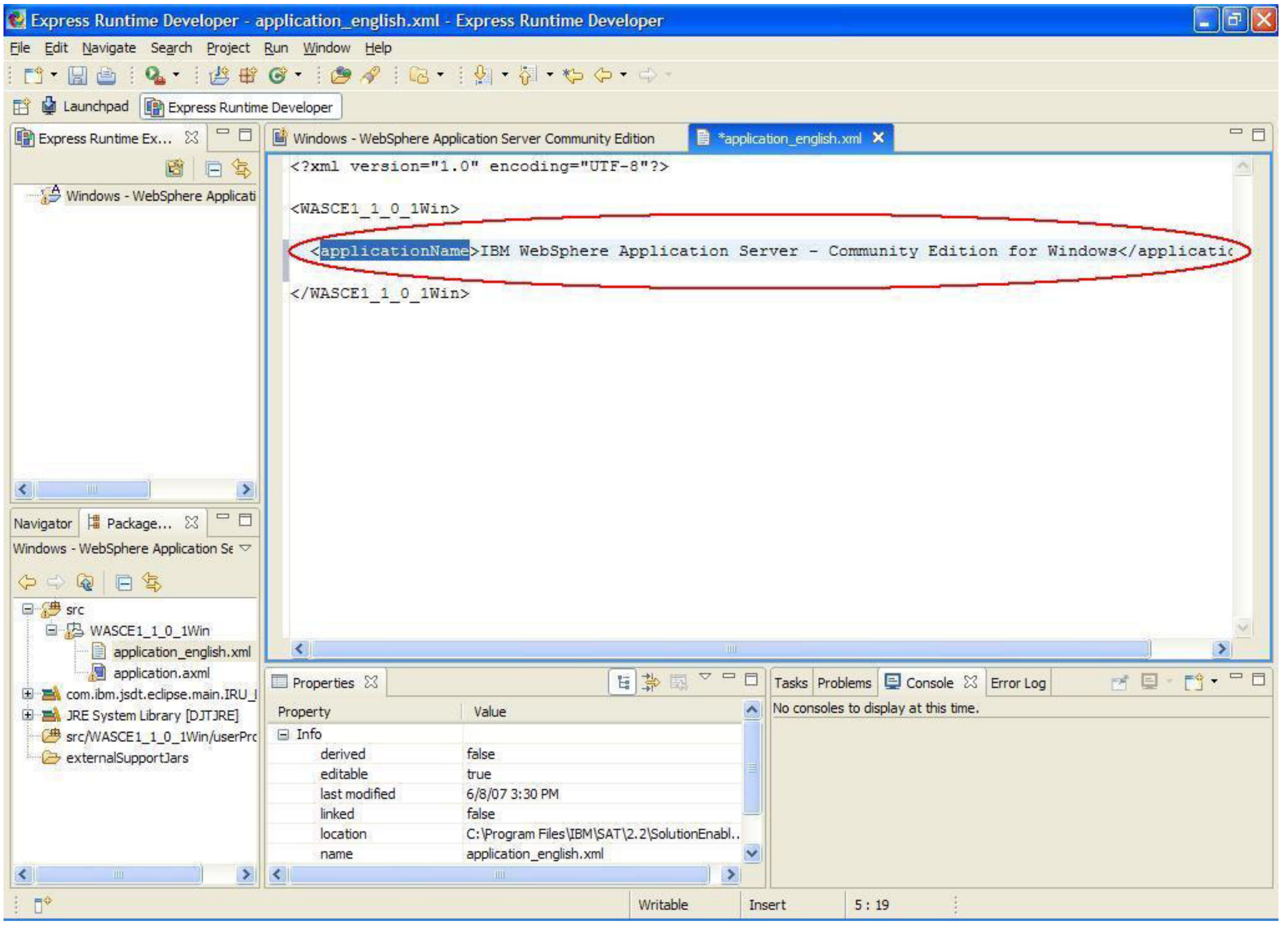
To enable the Application name for translation, perform the following steps:

1. Locate the WASCE1_1_0_1Win directory in the Navigator pane of Express Runtime developer. Expand the directory to view the contents.

2. Right-click the application_english.xml file and click **Open With** → **Text Editor** .
3. Locate the following line, as shown in Figure 7:

   ```
   <applicationName>WebSphere Application Server Community Edition 1.1.0.1</applicationName>
   ```

   Note: If you were supporting additional languages, you would need to add this tag to each language-specific XML file.


Express Runtime Developer - application_english.xml - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
Windows - WebSphere Application Server Community Edition
*application_english.xml
Navigator
Package...
Windows - WebSphere Application Se
src
WASCE1_1_0_1Win
application_english.xml
application.axml
com.ibm.jsdt.eclipse.main.IRU_l
JRE System Library [DJTJRE]
src/WASCE1_1_0_1Win/userPrc
externalSupportJars
Properties
Tasks
Problems
Console
Error Log
Writable
Insert
5 : 19

```
<?xml version="1.0" encoding="UTF-8"?>

<WASCE1_1_0_1Win>

  <applicationName>IBM WebSphere Application Server - Community Edition for Windows</applicatic

</WASCE1_1_0_1Win>
```

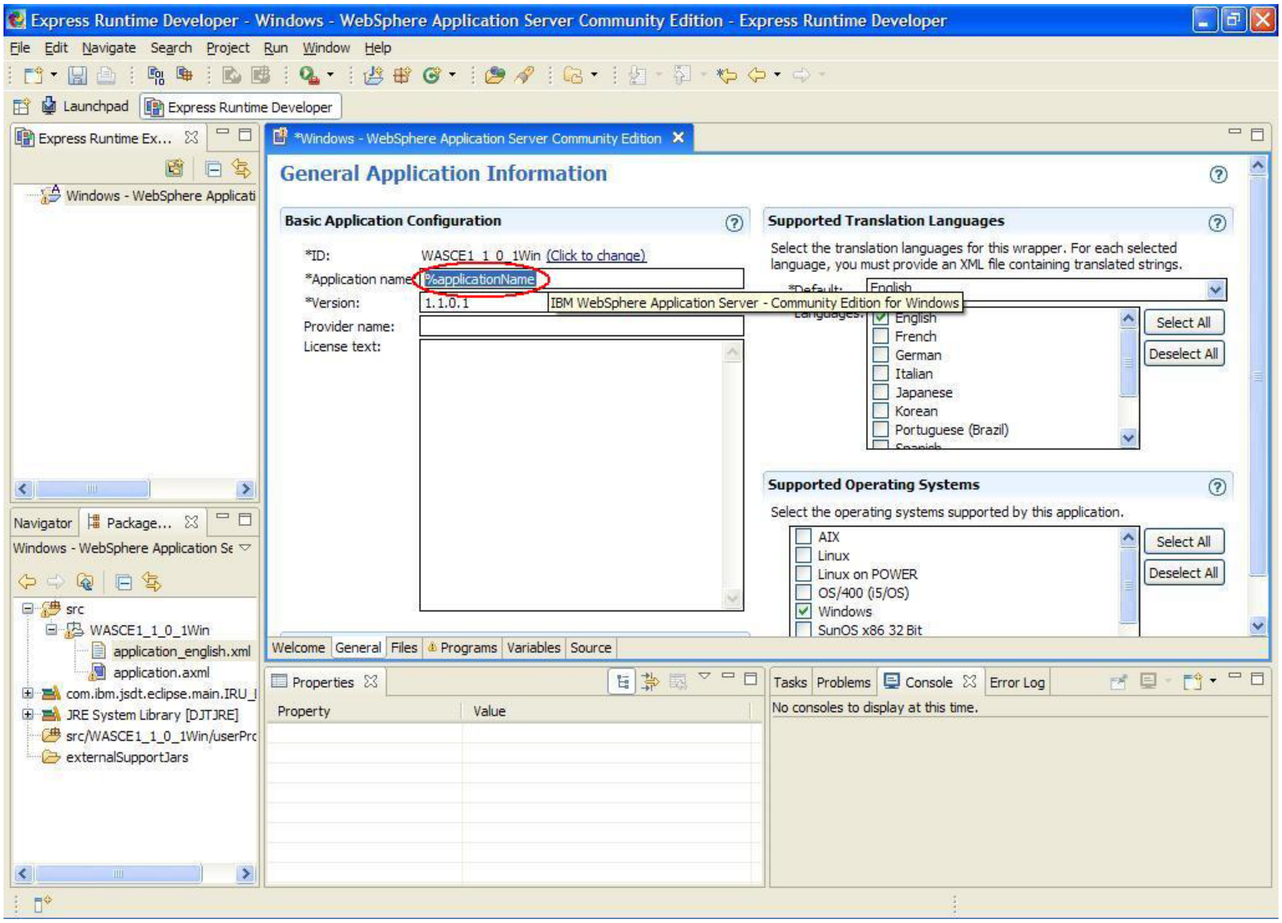
Property	Value
Info	
derived	false
editable	true
last modified	6/8/07 3:30 PM
linked	false
location	C:\Program Files\IBM\SAT\2.2\SolutionEnabl..
name	application_english.xml

No consoles to display at this time.

Figure 7. Application wrapper XML source showing <applicationName> tag

4. Click the **Windows - WebSphere Application Server Community Edition** tab.

   To represent a translated string in Eclipse, you must precede a unique, user-defined key with the percent sign (%). This key (without the percent sign) must then be included in the language-specific XML file for each language you choose to support with your solution.
5. Click the **General** tab.
6. Change the **Application name** field to `%applicationName`.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
Navigator
Package...
Windows - WebSphere Application Se
src
WASCE1_1_0_1Win
application_english.xml
application.axml
com.ibm.jsdt.eclipse.main.IRU_l
JRE System Library [DJTJRE]
src/WASCE1_1_0_1Win/userPrc
externalSupportJars
*Windows - WebSphere Application Server Community Edition
Select All
Deselect All
Select All
Deselect All
Welcome
General
Files
Programs
Variables
Source
Properties
Property
Value
Tasks
Problems
Console
Error Log

General Application Information

Basic Application Configuration

*ID: WASCE1_1_0_1Win (Click to change)

*Application name: %applicationName

*Version: 1.1.0.1

IBM WebSphere Application Server - Community Edition for Windows

Provider name:

License text:

Supported Translation Languages

Select the translation languages for this wrapper. For each selected language, you must provide an XML file containing translated strings.

*Default: English

Languages:
☑ English
☐ French
☐ German
☐ Italian
☐ Japanese
☐ Korean
☐ Portuguese (Brazil)
☐ Spanish

Supported Operating Systems

Select the operating systems supported by this application.

☐ AIX
☐ Linux
☐ Linux on POWER
☐ OS/400 (i5/OS)
☑ Windows
☐ SunOS x86 32 Bit

No consoles to display at this time.

Figure 8. General tab of the application project editor showing the new Application name value

7. Select **File** → **Save** to save your changes.

The application name will now be translated during deployment. The name in the XML file should also appear when you hover over the **Application name** field in the application project editor.

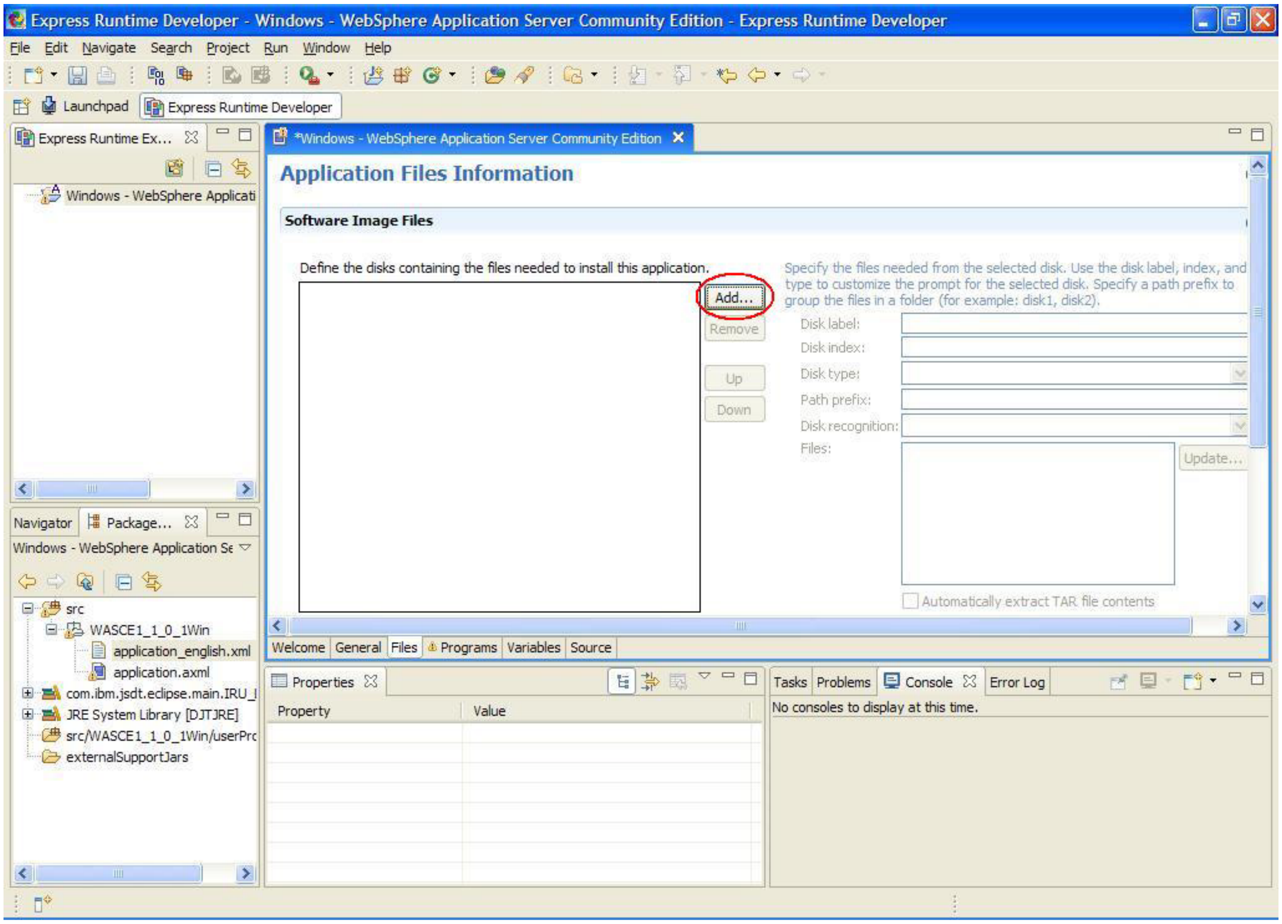
For more information, see “Providing general application information” on page 87, “Enabling an application wrapper for globalization” on page 98, and “Configuring translation languages” on page 86.

Adding software image files to the sample application wrapper

An application wrapper must list all of the files that need to be transferred to a target computer to run the application successfully during a deployment.

To add software image files to the sample application wrapper, complete the following steps:

1. Click the **Files** tab.
2. In the Software Image Files section, click **Add**, as shown in Figure 9 on page 16.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
*Windows - WebSphere Application Server Community Edition
Application Files Information
Software Image Files
Add...
Remove
Up
Down
Disk label:
Disk index:
Disk type:
Path prefix:
Disk recognition:
Files:
Update...
Automatically extract TAR file contents
Welcome
General
Files
Programs
Variables
Source
Navigator
Package...
Windows - WebSphere Application Se
src
WASCE1_1_0_1Win
application_english.xml
application.axml
com.ibm.jsdt.eclipse.main.IRU_I
JRE System Library [DJTJRE]
src/WASCE1_1_0_1Win/userPrc
externalSupportJars
Properties
Property
Value
Tasks
Problems
Console
Error Log

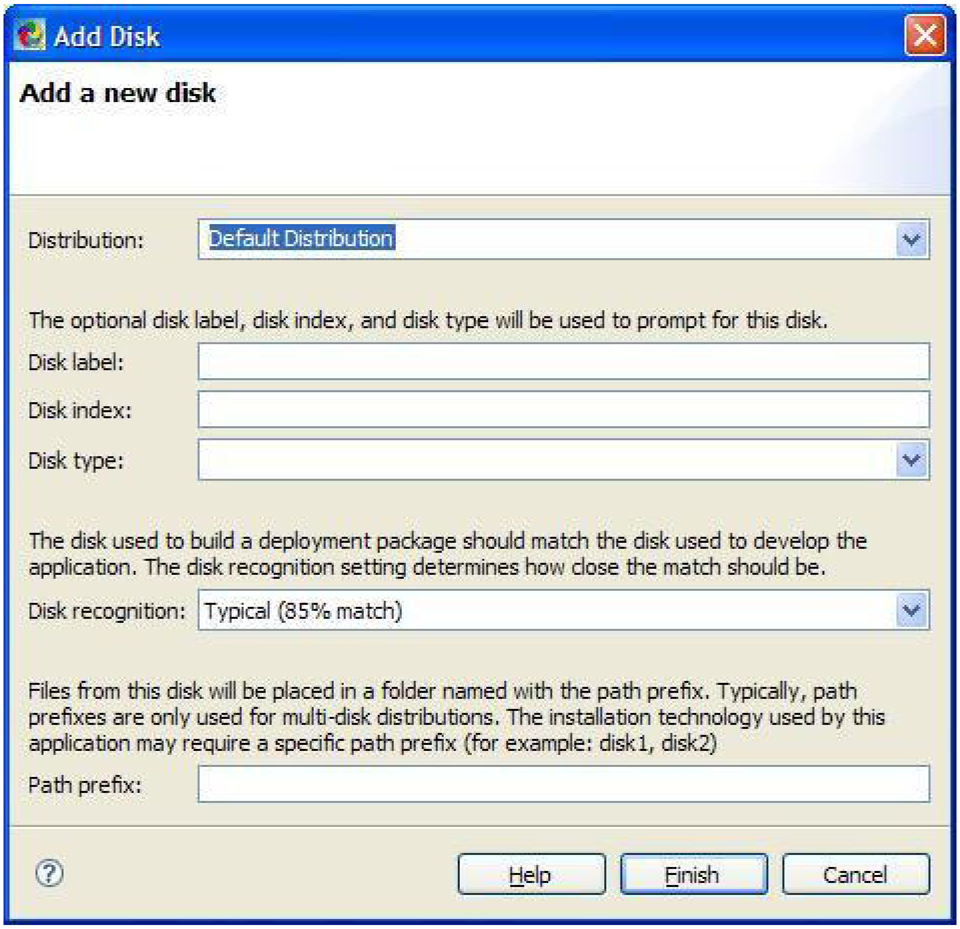
Define the disks containing the files needed to install this application.

Specify the files needed from the selected disk. Use the disk label, index, and type to customize the prompt for the selected disk. Specify a path prefix to group the files in a folder (for example: disk1, disk2).

No consoles to display at this time.

Figure 9. Software Image Files section of Files tab with Add button highlighted

The Add disk window opens. You use this window to provide distribution, prompting, and path prefix information for each disk containing the software image files.



Add a new disk

Distribution: Default Distribution

The optional disk label, disk index, and disk type will be used to prompt for this disk.

Disk label:

Disk index:

Disk type:

The disk used to build a deployment package should match the disk used to develop the application. The disk recognition setting determines how close the match should be.

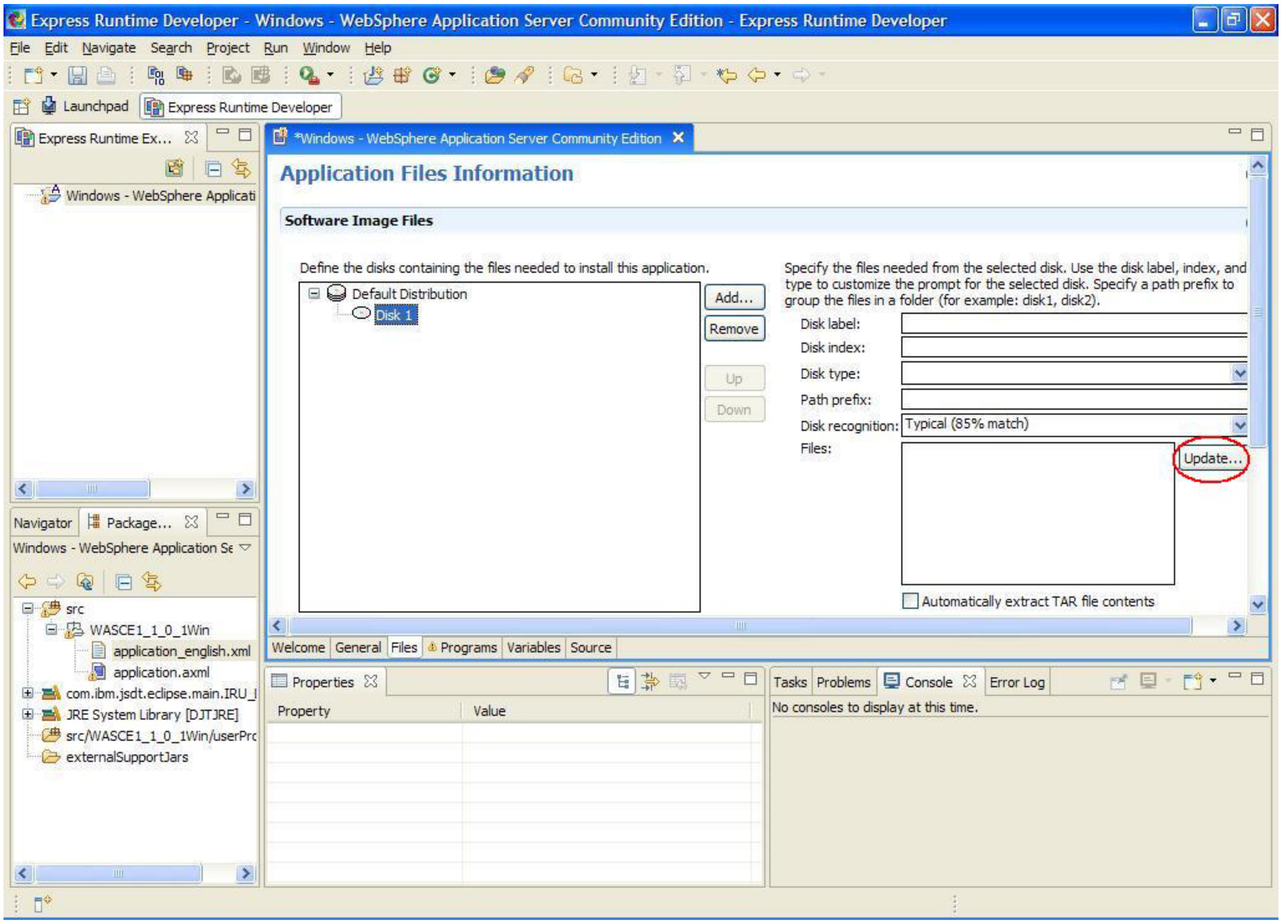
Disk recognition: Typical (85% match)

Files from this disk will be placed in a folder named with the path prefix. Typically, path prefixes are only used for multi-disk distributions. The installation technology used by this application may require a specific path prefix (for example: disk1, disk2)

Path prefix:

Figure 10. Add software image files: Add a new disk window

3. Leave the information in this window as is, and click **Finish**. You return to the **Files** tab of the application editor.
4. Click **Update**, as shown in Figure 11 on page 18.



Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer

Application Files Information

Software Image Files

Define the disks containing the files needed to install this application.

Specify the files needed from the selected disk. Use the disk label, index, and type to customize the prompt for the selected disk. Specify a path prefix to group the files in a folder (for example: disk1, disk2).

Disk recognition: Typical (85% match)

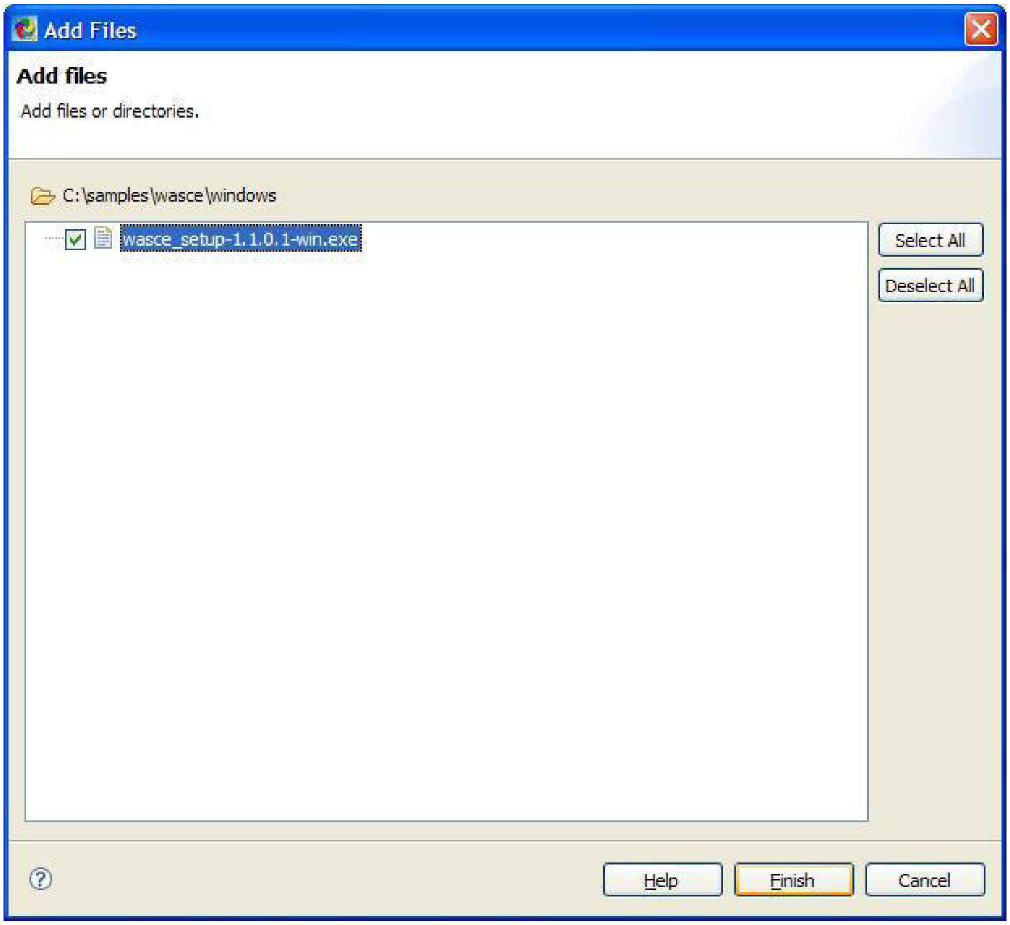
☐ Automatically extract TAR file contents

No consoles to display at this time.

Figure 11. Software Image Files section of Files tab, with Update button highlighted

The Add Files From window opens.

5. Browse to the directory to which you extracted the contents of the samples.zip file. Locate and select the samples\wasce\windows folder and click **OK**.
6. In the Add files window, select the wasce_setup-1.1.0.1-win.exe file and click **Finish**.


Add Files

Add files

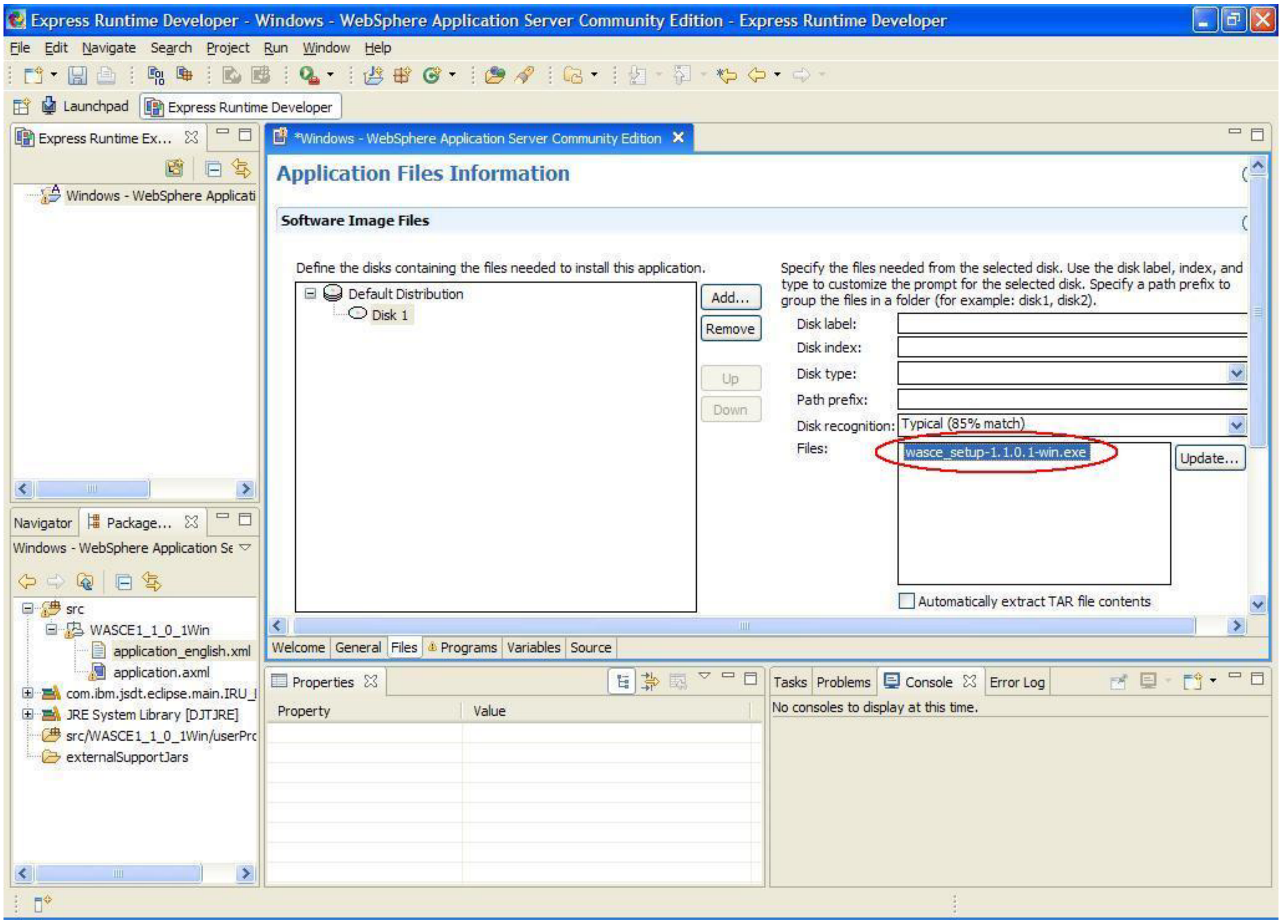
Add files or directories.

C:\samples\wasce\windows

☑ wasce_setup-1.1.0.1-win.exe

Figure 12. Add Files window

The wasce_setup-1.1.0.1-win.exe file is now listed in the **Files** field of the Software Image Files section, as shown in Figure 13 on page 20.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
*Windows - WebSphere Application Server Community Edition
Application Files Information
Software Image Files
Default Distribution
Disk 1
Add...
Remove
Up
Down
Disk label:
Disk index:
Disk type:
Path prefix:
Update...
Automatically extract TAR file contents
Welcome General Files Programs Variables Source
Navigator Package...
Windows - WebSphere Application Se
src
WASCE1_1_0_1Win
application_english.xml
application.axml
com.ibm.jsdt.eclipse.main.IRU_I
JRE System Library [DJTJRE]
src/WASCE1_1_0_1Win/userPrc
externalSupportJars
Properties
Property
Value
Tasks Problems Console Error Log

Define the disks containing the files needed to install this application.

Specify the files needed from the selected disk. Use the disk label, index, and type to customize the prompt for the selected disk. Specify a path prefix to group the files in a folder (for example: disk1, disk2).

Disk recognition: Typical (85% match)

Files: wasce_setup-1.1.0.1-win.exe

No consoles to display at this time.

Figure 13. Software Image Files section of Files tab with wasce_setup-1.1.0.1-win.exe file highlighted

For more information, see "Providing application file information" on page 95.

Specifying the main program for the sample application

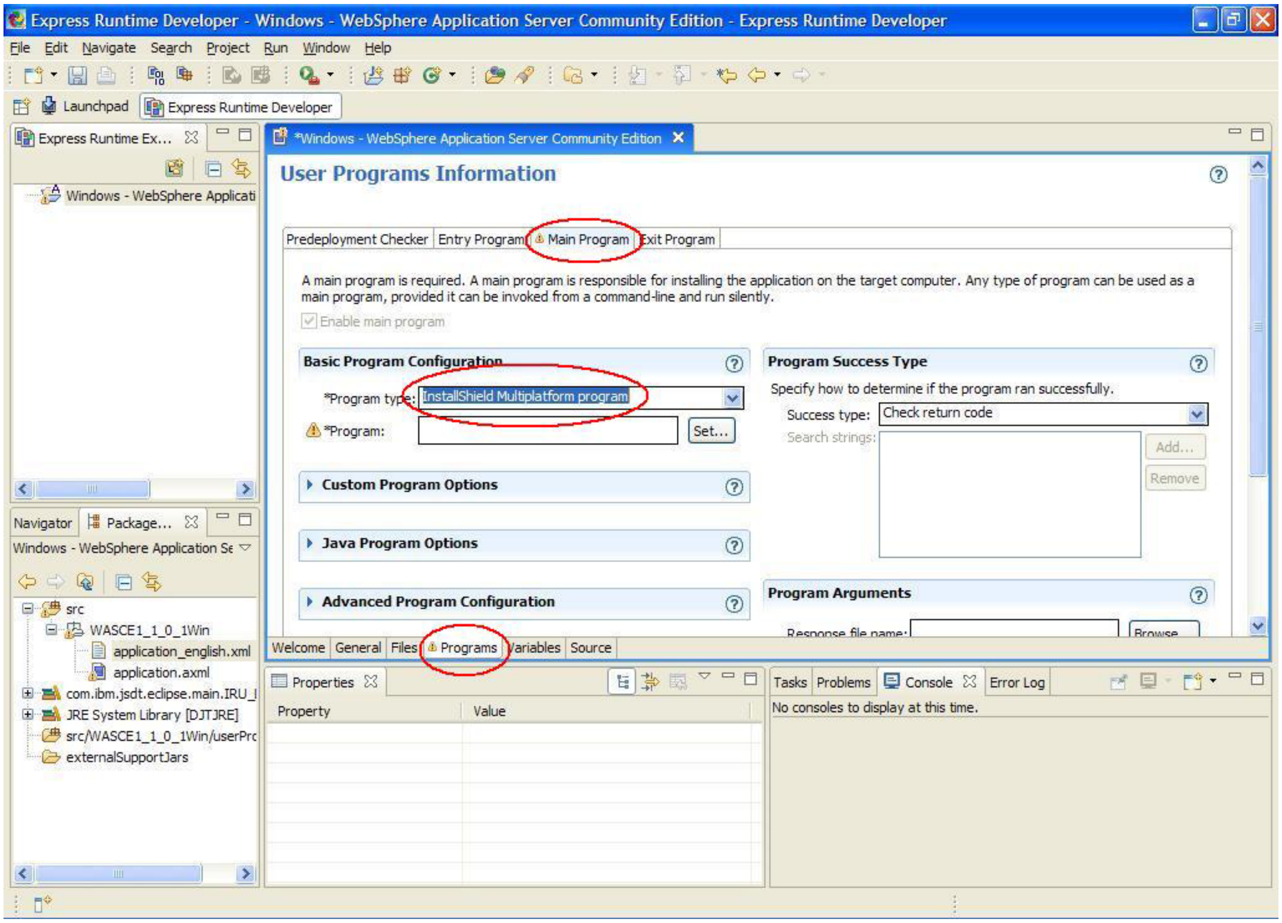
User programs run during the application deployment to perform the main installation, and pre- or post-installation operations. A user program can be a Java™ program, a stand-alone executable, or a script, but it must be able to run silently. A main user program is required in an application wrapper. Other user programs, such as a predeployment checker, an entry program, and an exit program, can be included in an application wrapper.

A main program is required for every application. The main user program is the installation program that is run on the target computer. Any type of installation program is acceptable as a main program if it can be invoked from a command-line and run silently. If there is no main program specified in an application, a warning icon is displayed and the application will not be able to be generated.

To specify the main program for the sample application, complete the following steps:

1. Click the **Programs** tab.
2. Click **Main Program**.

3. In the **Program type** field, select **InstallShield Multiplatform program**.



Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer

*Windows - WebSphere Application Server Community Edition

User Programs Information

A main program is required. A main program is responsible for installing the application on the target computer. Any type of program can be used as a main program, provided it can be invoked from a command-line and run silently.

☑ Enable main program

Basic Program Configuration

*Program type: InstallShield Multiplatform program

*Program:

Program Success Type

Specify how to determine if the program ran successfully.

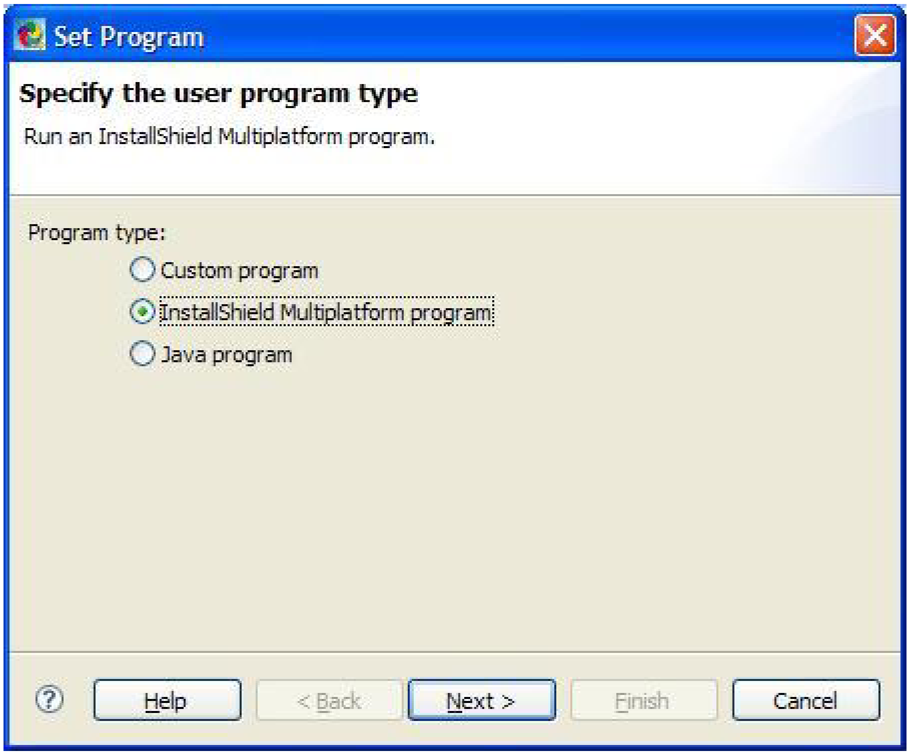
Success type: Check return code

Search strings:

No consoles to display at this time.

Figure 14. Express Runtime developer Files tab: Main Program

4. Click **Set**. The Set Program wizard: Specify the user program type page opens.



Set Program

Specify the user program type

Run an InstallShield Multiplatform program.

Program type:

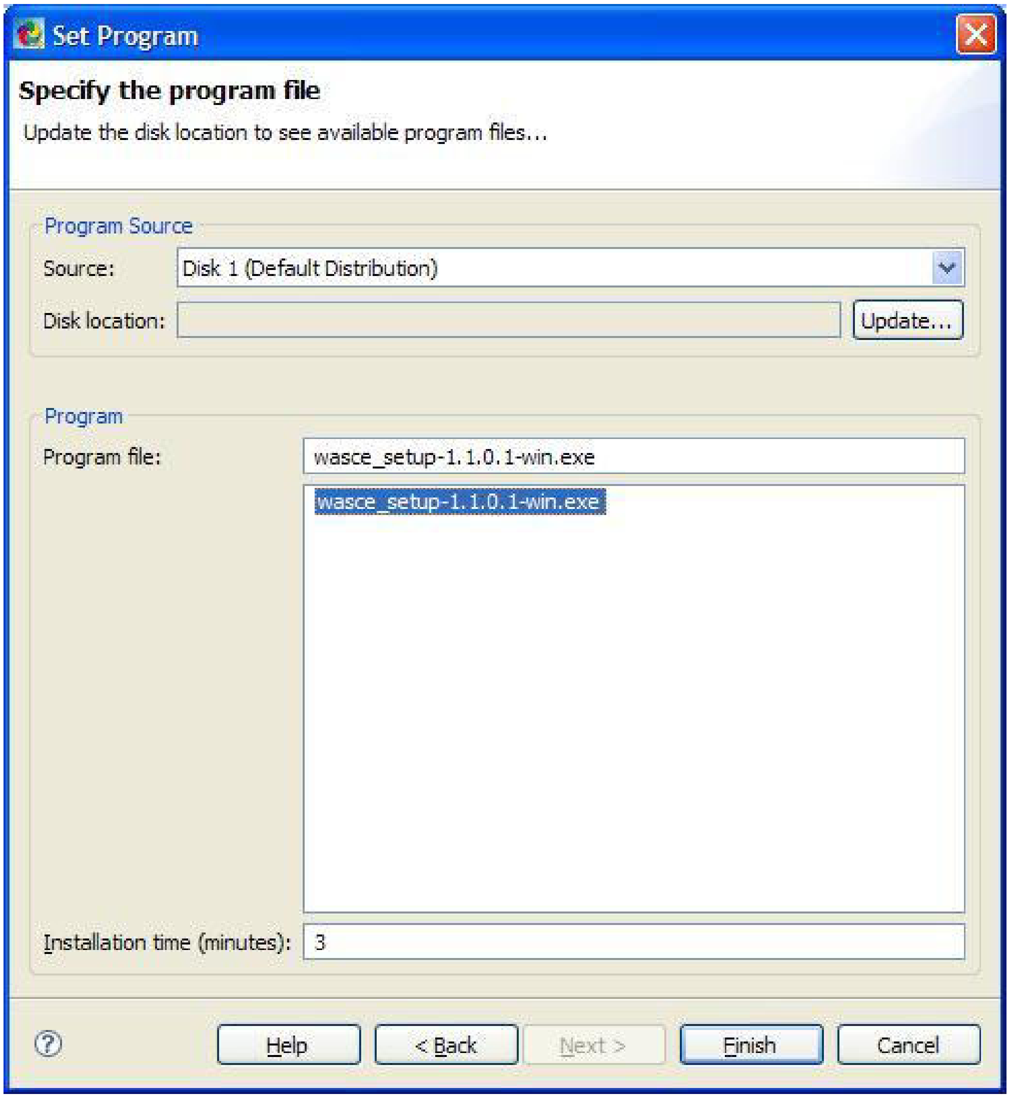
○ Custom program

● InstallShield Multiplatform program

○ Java program

Figure 15. Set Program wizard: Specify the user program type page

5. Ensure that **InstallShield Multiplatform program** is selected, as shown in Figure 15, and click **Next**. The Set Program wizard: Specify the program file page opens.
6. In the **Source** field, select **Disk 1 (Default Distribution)** (see Figure 16 on page 23).
7. In the **Program file** field, select the wasce_setup-1.1.0.1-win.exe file and click **Finish**. You are returned to the application project editor.


Set Program

Specify the program file

Update the disk location to see available program files...

Program Source

Source: Disk 1 (Default Distribution)

Disk location:

Program

Program file: wasce_setup-1.1.0.1-win.exe

wasce_setup-1.1.0.1-win.exe

Installation time (minutes): 3

Figure 16. Set Program wizard: Specify the program file page

8. In the Program Arguments section of the Programs window, click **Browse** (next to the **Response file name** field).
9. Click the wasce1101.iss file and click **Open**.
10. In the Program Arguments section of the Programs window, type `wasce1101.log` in the **Log file name** field.
11. Save the application project.

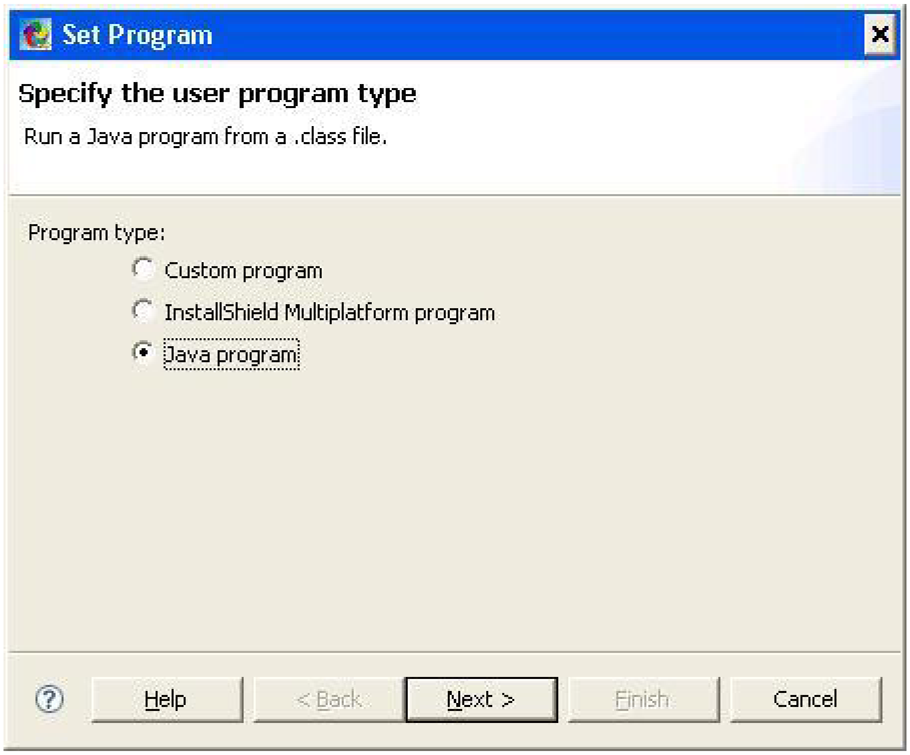
For more information about user programs, see “Providing user programs information” on page 88 and “Specifying program arguments” on page 92.

Specifying the predeployment checker for the sample application

Before an entry, exit, or main program is run, the deployment package for the application must be transferred to the target computer. A predeployment checker runs before the application entry, main and exit programs, and before the deployment package is transferred to the target computer. It can ensure that the target computer is ready for the deployment.

To specify the predeployment checker for the Windows - WebSphere Application Server Community Edition application, complete the following steps:

1. Click the **Programs** tab.
2. Click **Predeployment Checker**.
3. Select **Enable predeployment checker**. The Set Program wizard: Specify the user program type page opens.



Set Program

Specify the user program type

Run a Java program from a .class file.

Program type:

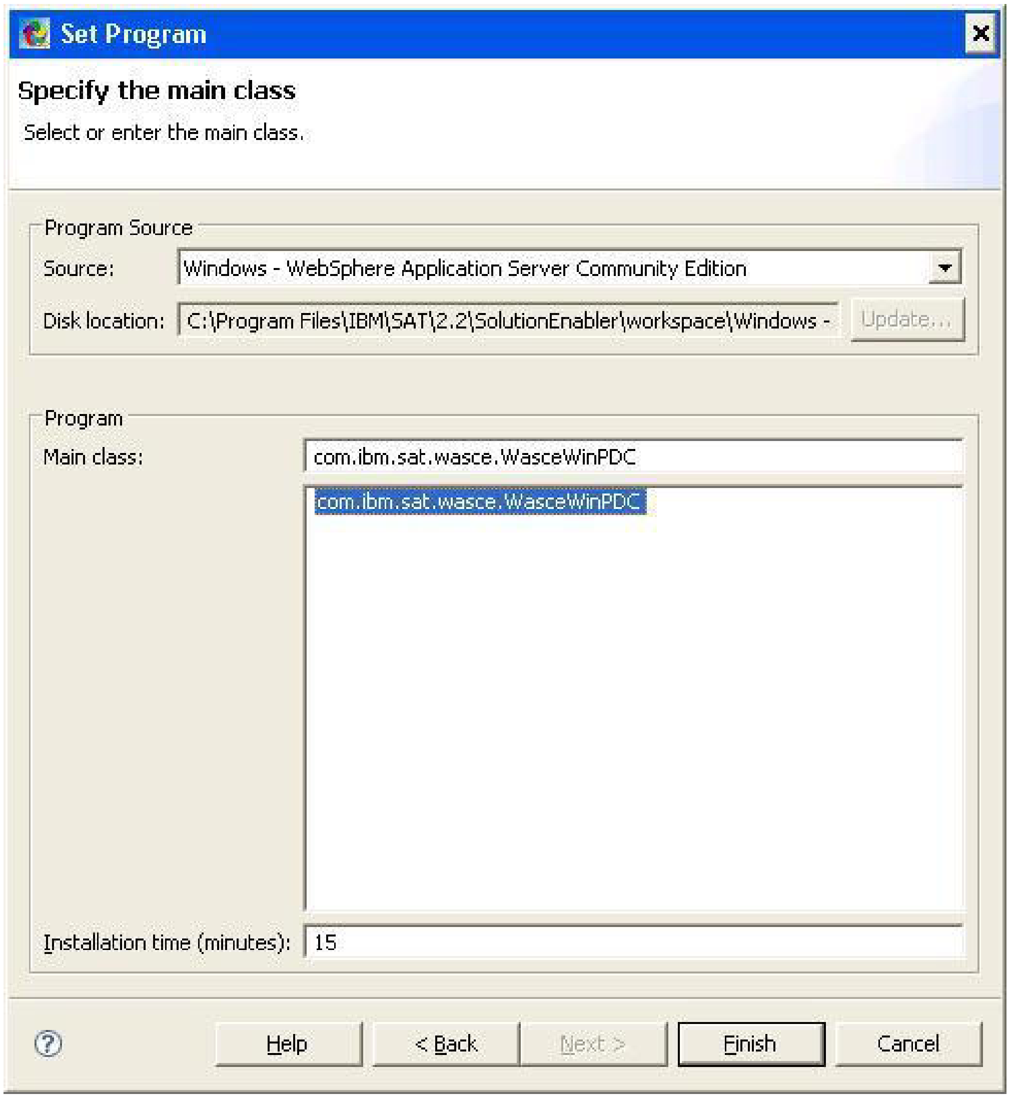
○ Custom program

○ InstallShield Multiplatform program

● Java program

Figure 17. Set Program wizard: Specify the user program type page

4. Ensure that **Java program** is selected and click **Next**. The Set Program wizard: Specify the main class page opens.
5. In the **Source** field, select **Windows - WebSphere Application Server Community Edition** (see Figure 18 on page 25).
6. In the **Main class** field, select the com.ibm.sat.wasce.WasceWinPDC class.



Specify the main class

Select or enter the main class.

Program Source

Source: Windows - WebSphere Application Server Community Edition

Disk location: C:\Program Files\IBM\SAT\2.2\SolutionEnabler\workspace\Windows -

Program

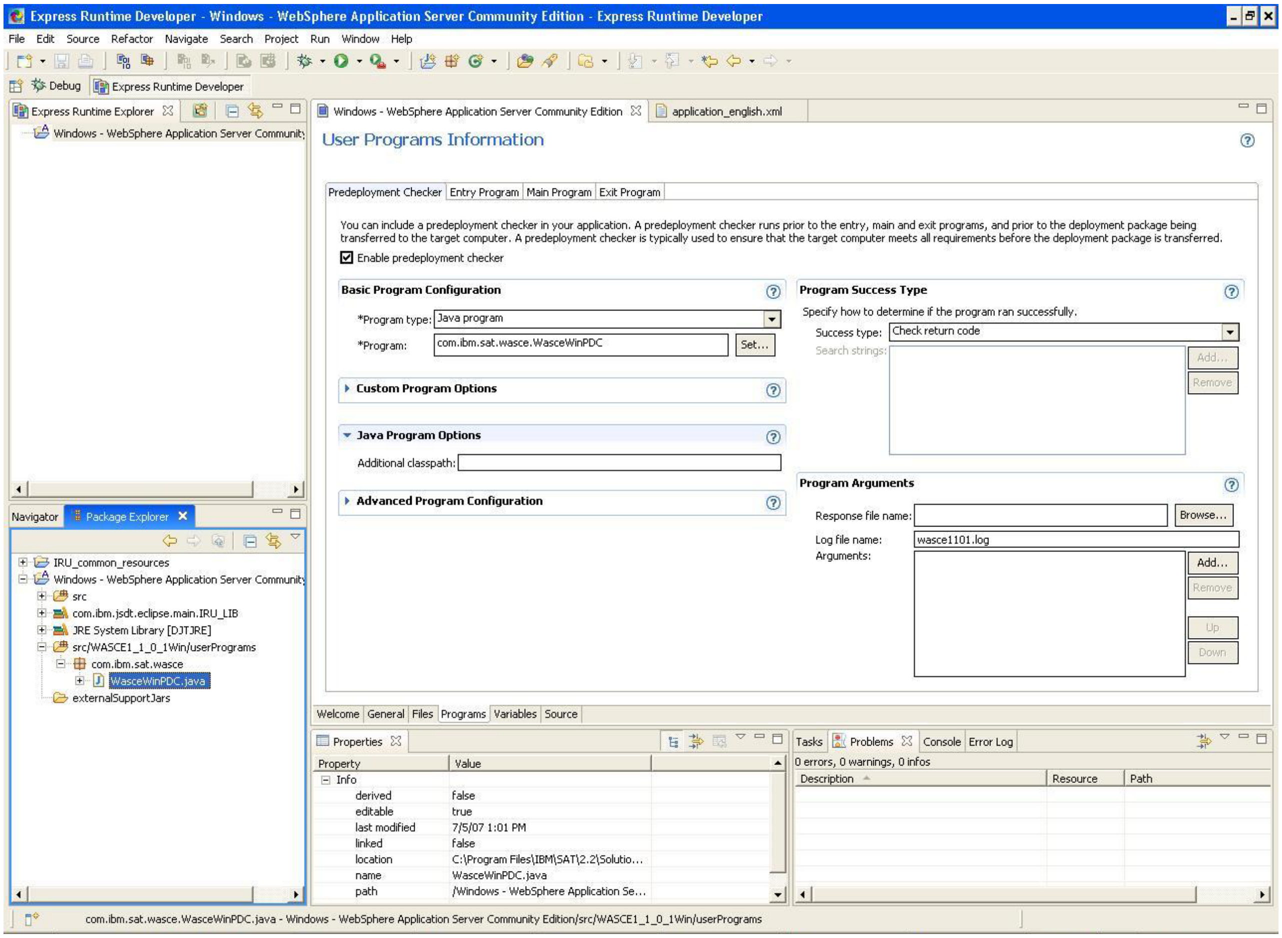
Main class: com.ibm.sat.wasce.WasceWinPDC

com.ibm.sat.wasce.WasceWinPDC

Installation time (minutes): 15

Figure 18. Set Program wizard: Specify the program file page

7. Click **Finish**. You are returned to the application project editor.



Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer

User Programs Information

Predeployment Checker | Entry Program | Main Program | Exit Program

You can include a predeployment checker in your application. A predeployment checker runs prior to the entry, main and exit programs, and prior to the deployment package being transferred to the target computer. A predeployment checker is typically used to ensure that the target computer meets all requirements before the deployment package is transferred.

☑ Enable predeployment checker

Basic Program Configuration

*Program type: Java program

*Program: com.ibm.sat.wasce.WasceWinPDC

Custom Program Options

Java Program Options

Additional classpath:

Advanced Program Configuration

Program Success Type

Specify how to determine if the program ran successfully.

Success type: Check return code

Search strings:

Program Arguments

Response file name:

Log file name: wasce1101.log

Arguments:

Package Explorer:

- IRU_common_resources
- Windows - WebSphere Application Server Community
  - src
  - com.ibm.jsdt.eclipse.main.IRU_LIB
  - JRE System Library [DJTJRE]
  - src/WASCE1_1_0_1Win/userPrograms
    - com.ibm.sat.wasce
      - WasceWinPDC.java
  - externalSupportJars

Problems: 0 errors, 0 warnings, 0 infos

Property	Value
Info	
derived	false
editable	true
last modified	7/5/07 1:01 PM
linked	false
location	C:\Program Files\IBM\SAT\2.2\Solutio…
name	WasceWinPDC.java
path	/Windows - WebSphere Application Se…

com.ibm.sat.wasce.WasceWinPDC.java - Windows - WebSphere Application Server Community Edition/src/WASCE1_1_0_1Win/userPrograms

Figure 19. User programs page of application project editor showing the predeployment checker configuration

8. Save the application project.

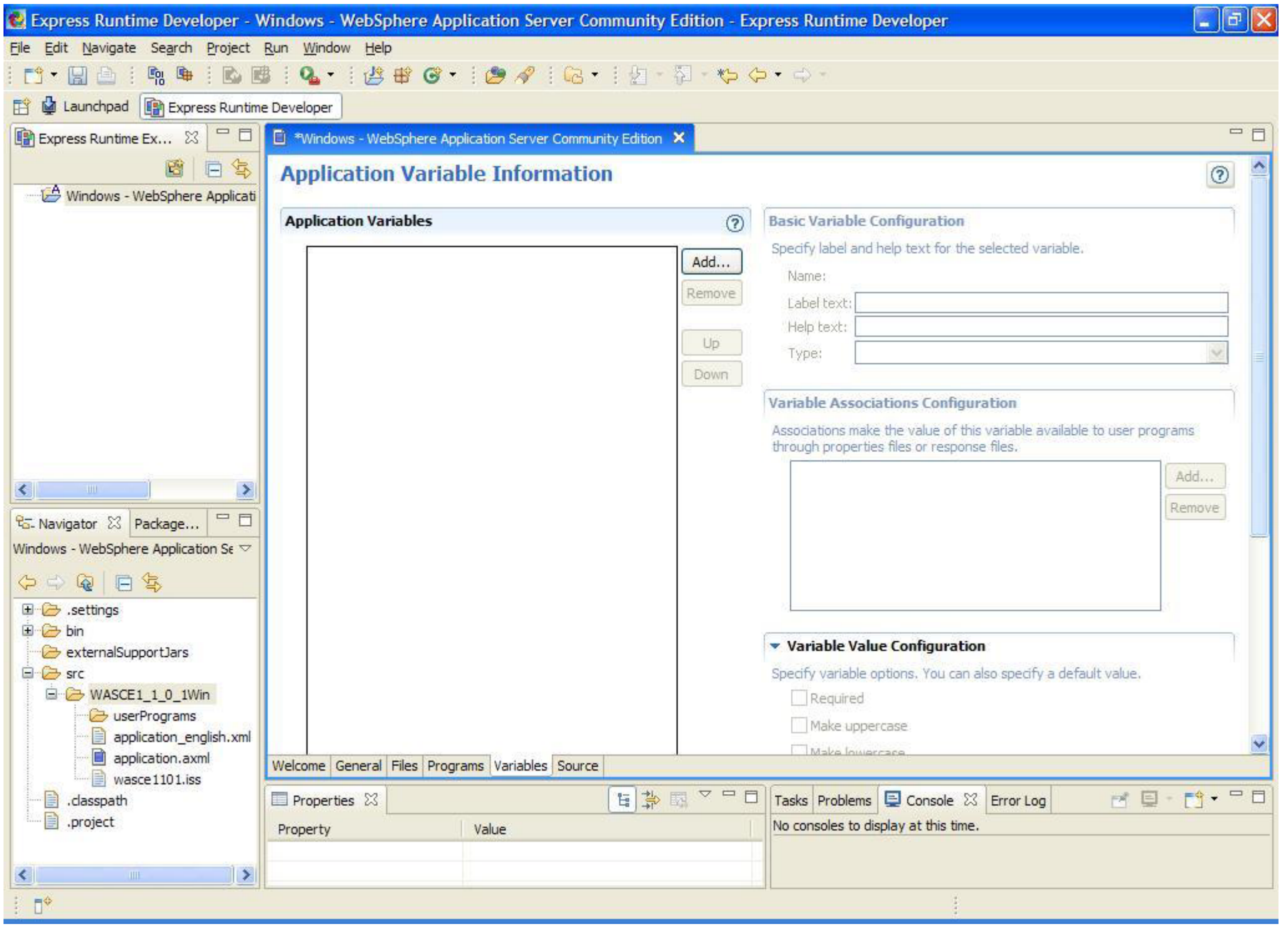
For more information about user programs, see “Providing user programs information” on page 88.

Adding variables to the application wrapper

You can add variables to an application wrapper to display configuration parameters to the end user during deployment.

To add a variable to the sample application wrapper, complete the following steps:

1. Select the **Variables** tab of the application project editor. The Application Variable Information window is displayed.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
*Windows - WebSphere Application Server Community Edition
Application Variable Information
Application Variables
Add...
Remove
Up
Down
Basic Variable Configuration
Name:
Label text:
Help text:
Type:
Variable Associations Configuration
Add...
Remove
Variable Value Configuration
Required
Make uppercase
Make lowercase
Welcome General Files Programs Variables Source
Navigator Package...
Windows - WebSphere Application Se
.settings
bin
externalSupportJars
src
WASCE1_1_0_1Win
userPrograms
application_english.xml
application.axml
wasce1101.iss
.classpath
.project
Properties
Property Value
Tasks Problems Console Error Log

Specify label and help text for the selected variable.

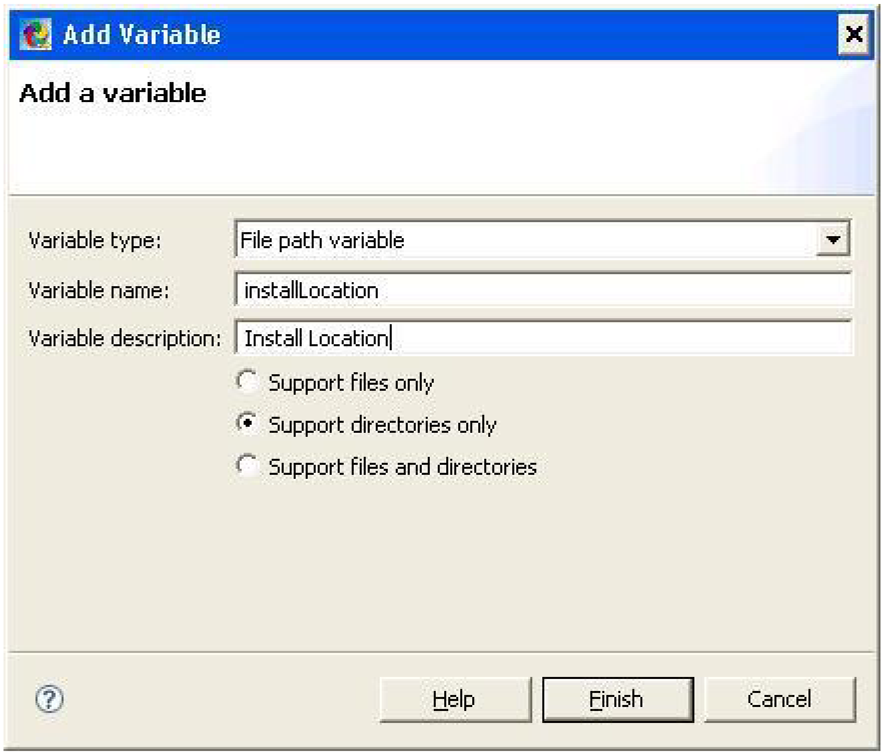
Associations make the value of this variable available to user programs through properties files or response files.

Specify variable options. You can also specify a default value.

No consoles to display at this time.

Figure 20. Application Variable Information window

2. Click **Add** in the Application Variables section. The Add Variable window opens.
3. Select or type the following values, as shown in Figure 21 on page 28.
   - In the **Variable type** field, select **File path variable**.
   - In the **Variable name** field, type `installLocation`.
   - In the **Variable description** field, type `Install Location`.
   - Leave **Support directories only** selected.



Variable type: File path variable

Variable name: installLocation

Variable description: Install Location

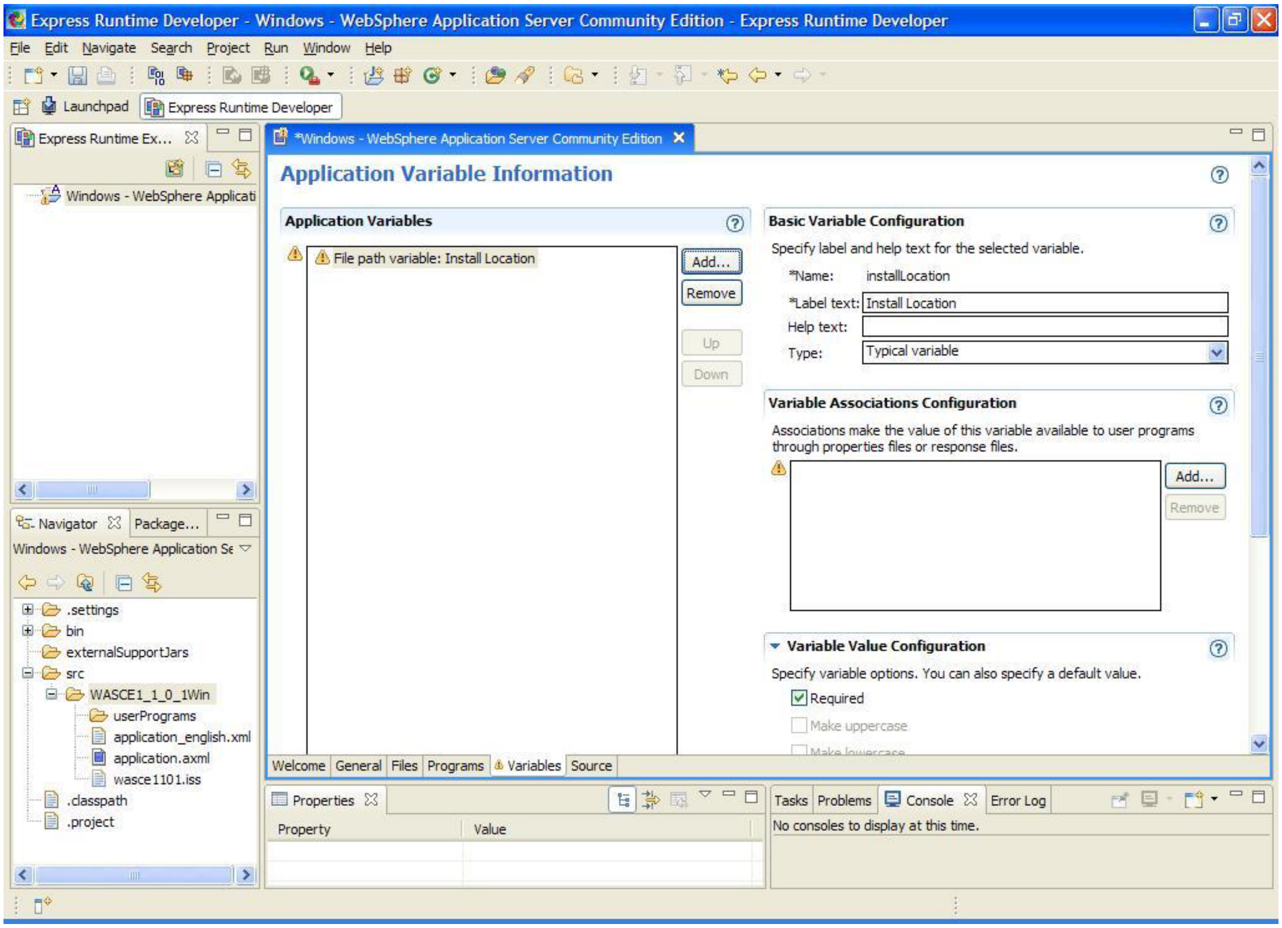
○ Support files only

● Support directories only

○ Support files and directories

Figure 21. Add Variable window

4. Click **Finish**. You return to the Variables tab of the application project editor.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
Navigator
Package...
Windows - WebSphere Application Se
.settings
bin
externalSupportJars
src
WASCE1_1_0_1Win
userPrograms
application_english.xml
application.axml
wasce1101.iss
.classpath
.project
*Windows - WebSphere Application Server Community Edition
Add...
Remove
Up
Down
Add...
Remove
Welcome
General
Files
Programs
Variables
Source
Properties
Property
Value
Tasks
Problems
Console
Error Log

Application Variable Information

Application Variables

⚠ File path variable: Install Location

Basic Variable Configuration

Specify label and help text for the selected variable.

*Name: installLocation

*Label text: Install Location

Help text:

Type: Typical variable

Variable Associations Configuration

Associations make the value of this variable available to user programs through properties files or response files.

Variable Value Configuration

Specify variable options. You can also specify a default value.

☑ Required

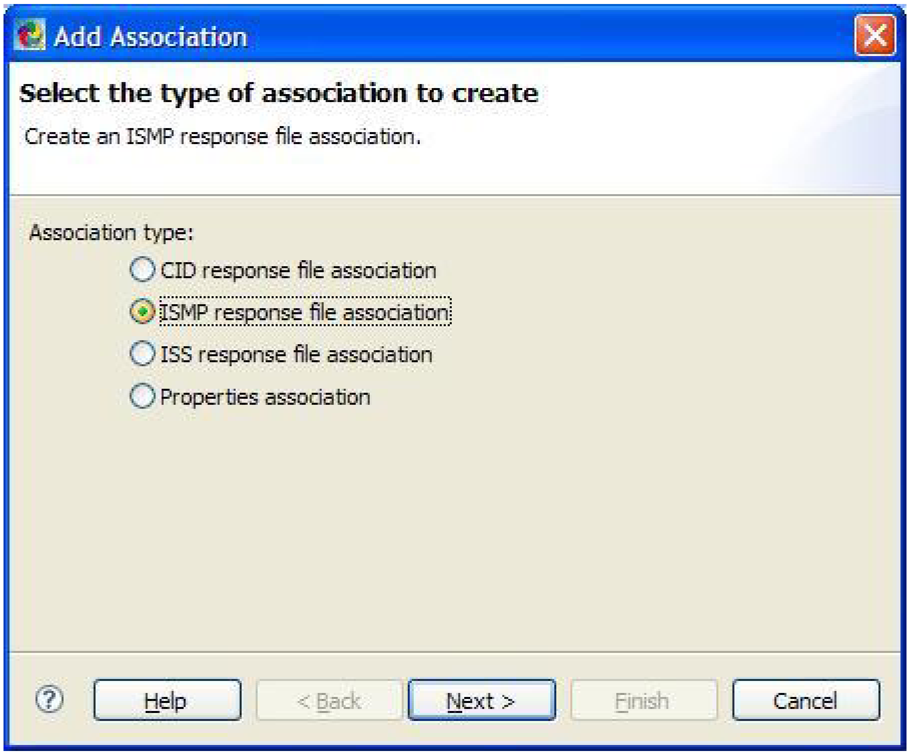
□ Make uppercase

□ Make lowercase

No consoles to display at this time.

Figure 22. Application Variables Information window

5. To configure the application to install silently using a response file, click **Add** in the Variable Associations Configuration section. The first page of the Add Association wizard is displayed.
6. Click **ISMP response file association** and click **Next**.



Add Association

Select the type of association to create

Create an ISMP response file association.

Association type:

○ CID response file association

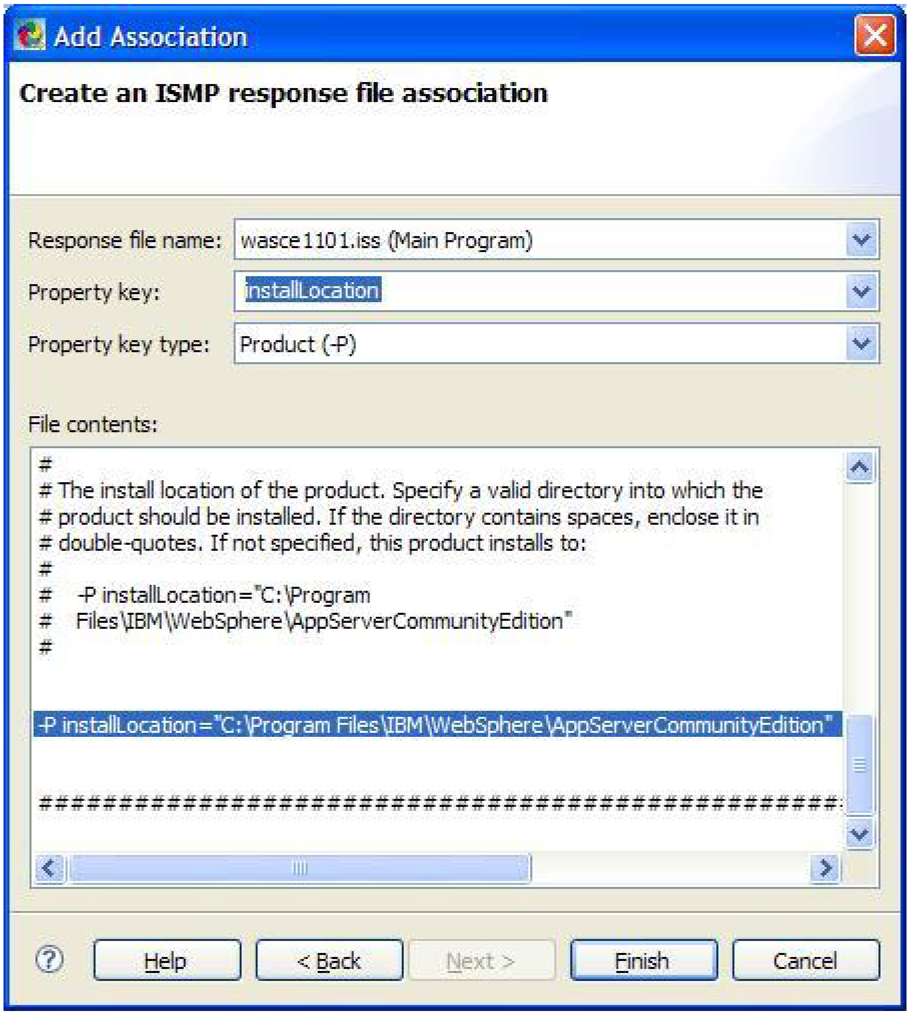
● ISMP response file association

○ ISS response file association

○ Properties association

Figure 23. Add Association wizard: Association type

7. On the next page of the wizard, select the following values, as shown in Figure 24 on page 31:
   - **Response file name**: `wasce1101.iss (Main Program)`
   - **Property key**: `installLocation`
   - **Property key type**: `Product (-P)`



Create an ISMP response file association

Response file name: wasce1101.iss (Main Program)

Property key: installLocation

Property key type: Product (-P)

File contents:

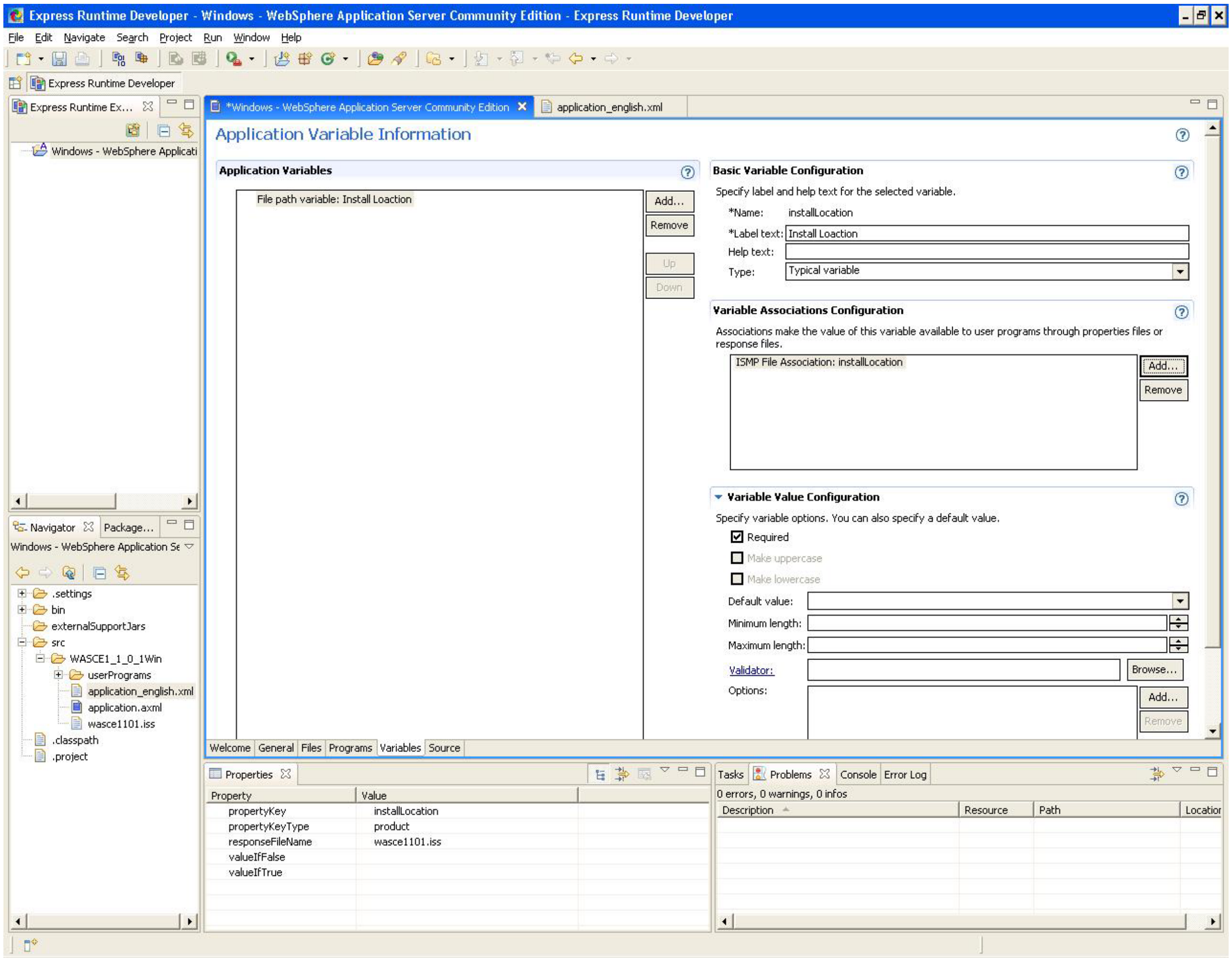
```
#
# The install location of the product. Specify a valid directory into which the
# product should be installed. If the directory contains spaces, enclose it in
# double-quotes. If not specified, this product installs to:
#
#    -P installLocation="C:\Program
#    Files\IBM\WebSphere\AppServerCommunityEdition"
#


-P installLocation="C:\Program Files\IBM\WebSphere\AppServerCommunityEdition"


#####################################################
```

Figure 24. Add Association wizard: Create an ISMP response file association

8. Click **Finish**. The association you just created is displayed in the Variable Associations Configuration section of the Application Variables Information window.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
*Windows - WebSphere Application Server Community Edition
application_english.xml
Navigator
Package...
Windows - WebSphere Application Se
.settings
bin
externalSupportJars
src
WASCE1_1_0_1Win
userPrograms
application_english.xml
application.axml
wasce1101.iss
.classpath
.project
Add...
Remove
Up
Down
Browse...
Welcome General Files Programs Variables Source
Tasks Problems Console Error Log
Description Resource Path Locatior

Application Variable Information

Application Variables

File path variable: Install Loaction

Basic Variable Configuration

Specify label and help text for the selected variable.

*Name: installLocation

*Label text: Install Loaction

Help text:

Type: Typical variable

Variable Associations Configuration

Associations make the value of this variable available to user programs through properties files or response files.

ISMP File Association: installLocation

Variable Value Configuration

Specify variable options. You can also specify a default value.

☑ Required

□ Make uppercase

□ Make lowercase

Default value:

Minimum length:

Maximum length:

Validator:

Options:

Properties

Property	Value
propertyKey	installLocation
propertyKeyType	product
responseFileName	wasce1101.iss
valueIfFalse	
valueIfTrue	

0 errors, 0 warnings, 0 infos

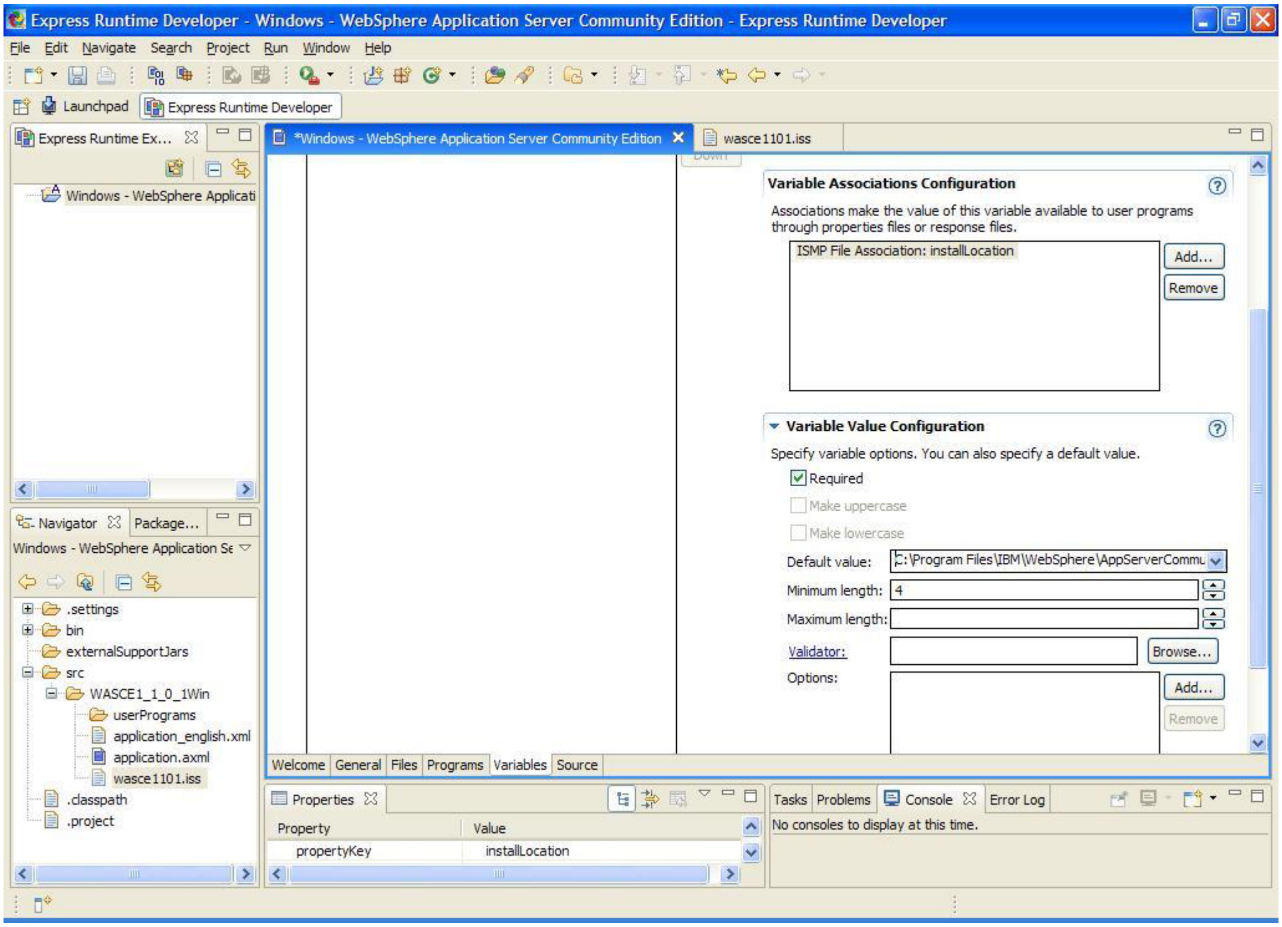
Figure 25. Application Variables Information window showing variable association

9. In the Variable Value Configuration section of the editor, ensure that the **Required** check box is selected. When this check box is selected, a value must be entered for this variable before the application can be deployed. See Figure 26 on page 33.

10. In the **Default value** field, type `C:\Program Files\IBM\WebSphere\AppServerCommunityEdition`. See Figure 26 on page 33.

    If the end user does not change the default installation location during deployment, this is the directory to which the application will be installed.

11. In the **Minimum length** field, type `4`. This entry ensures that no value for this variable less than four characters long is accepted.



Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer

Variable Associations Configuration

Associations make the value of this variable available to user programs through properties files or response files.

ISMP File Association: installLocation

Variable Value Configuration

Specify variable options. You can also specify a default value.

☑ Required

□ Make uppercase

□ Make lowercase

Default value: C:\Program Files\IBM\WebSphere\AppServerCommu

Minimum length: 4

Maximum length:

Validator:

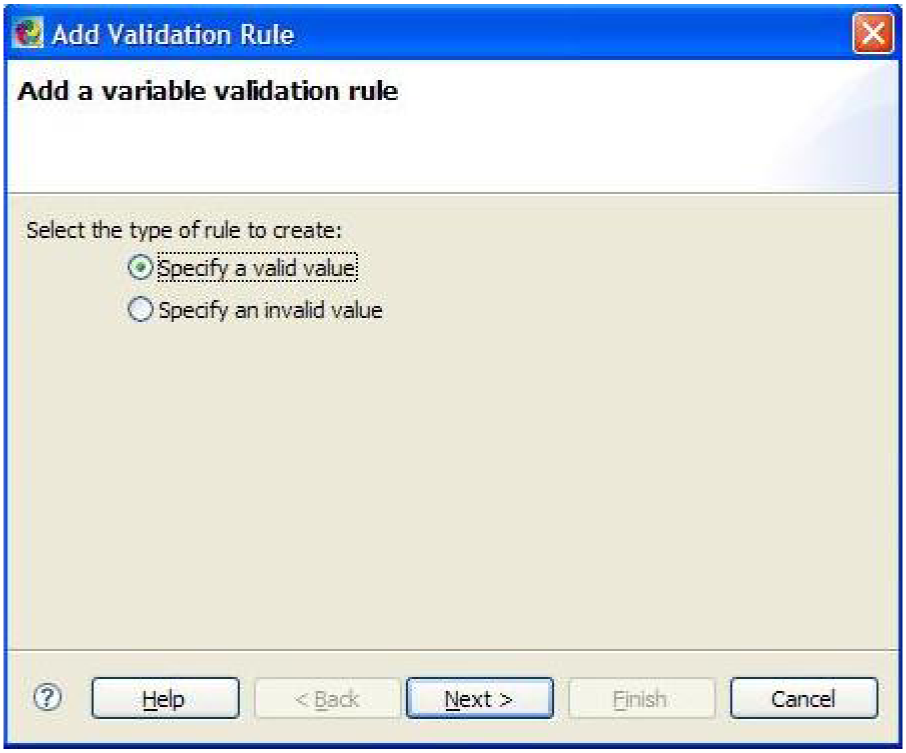
Options:

Property	Value
propertyKey	installLocation

No consoles to display at this time.

Figure 26. Application Variables Information window: Variable Value Configuration

12. To create a rule restricting the installation of the application to the C: drive, click **Add** (next to the **Options** field). The Add Validation Rule wizard opens.
13. Select **Specify a valid value** and click **Next**.


Add Validation Rule

Add a variable validation rule

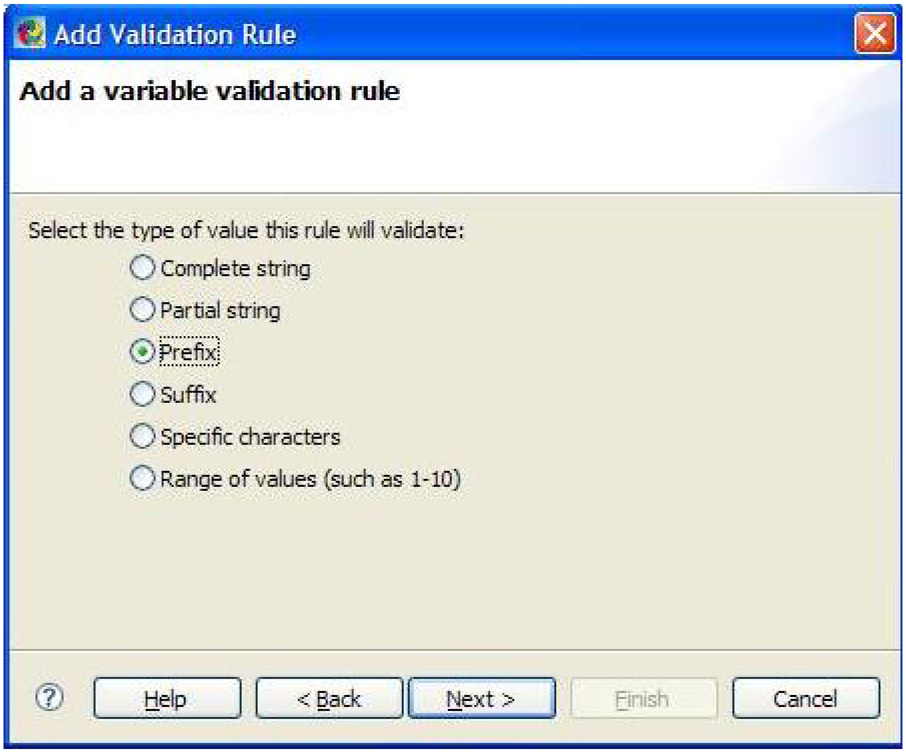
Select the type of rule to create:

● Specify a valid value

○ Specify an invalid value

Figure 27. Add Validation Rule wizard: Select type of rule for variable validation

14. On the next page of the wizard, select **Prefix** and click **Next**.


Add Validation Rule

Add a variable validation rule

Select the type of value this rule will validate:

○ Complete string

○ Partial string

● Prefix

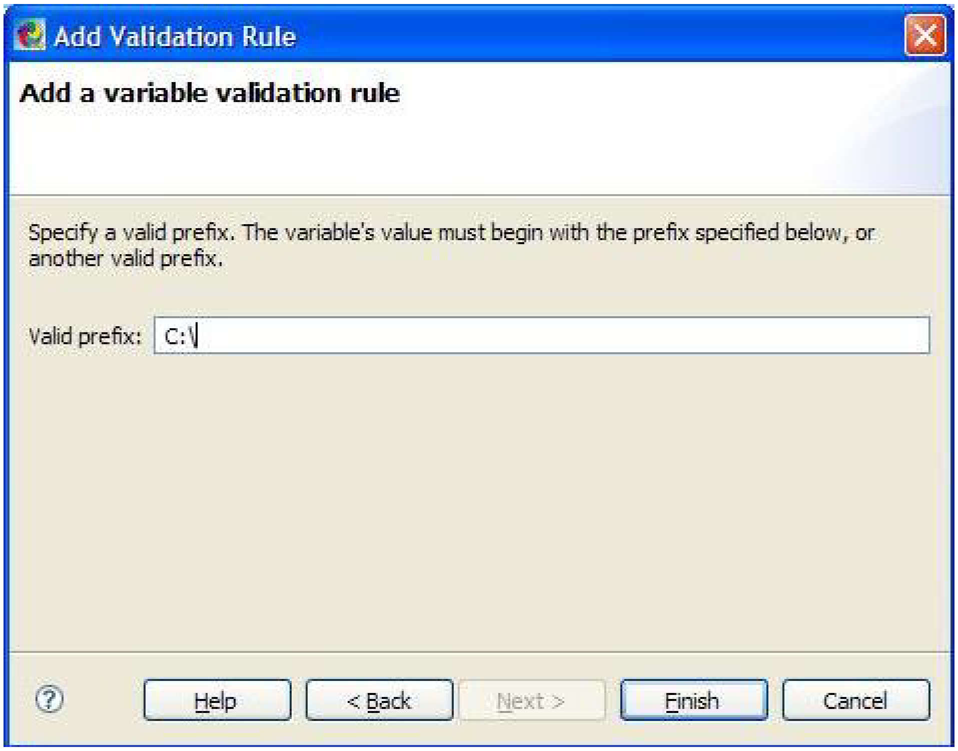
○ Suffix

○ Specific characters

○ Range of values (such as 1-10)

Figure 28. Add Validation Rule wizard: Select type of value to be validated by rule

15. Type `C:\` in the **Valid prefix** field and click **Finish**.



Add Validation Rule

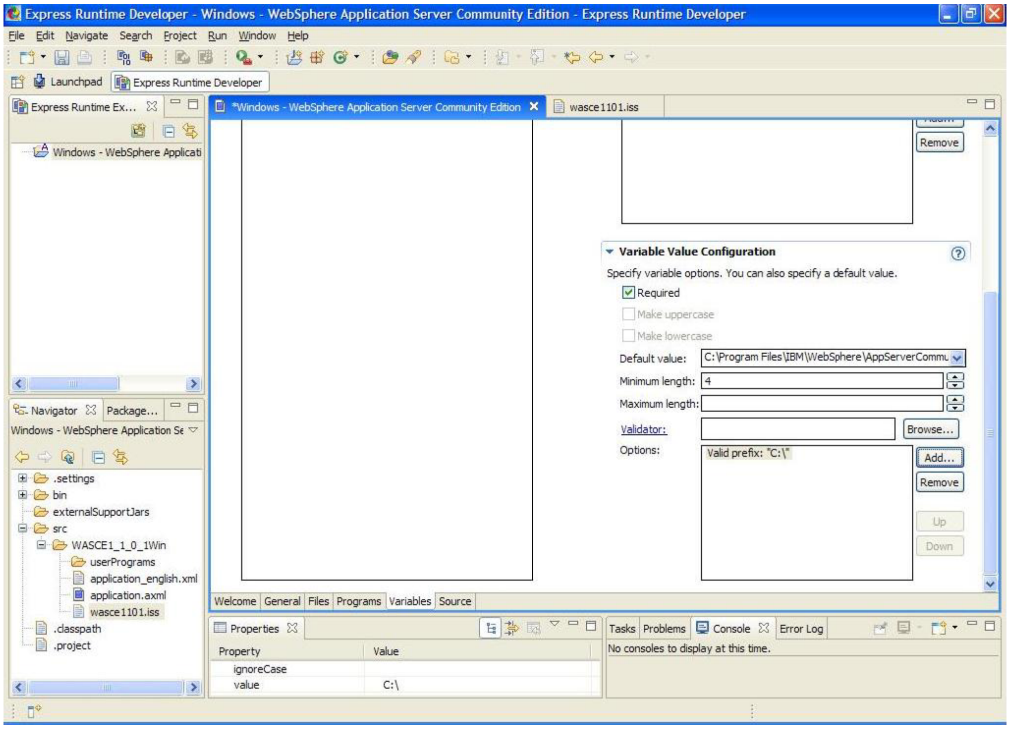
Add a variable validation rule

Specify a valid prefix. The variable's value must begin with the prefix specified below, or another valid prefix.

Valid prefix: C:\

Figure 29. Add Validation Rule wizard: Specify valid value

You return to the Application Variables Information window. The validation rule is displayed in the **Options** field of the Variable Value Configuration section. The application can only be installed to the C: drive of the target computer.



Variable Value Configuration

Specify variable options. You can also specify a default value.

☑ Required

□ Make uppercase

□ Make lowercase

Default value: C:\Program Files\IBM\WebSphere\AppServerCommu

Minimum length: 4

Maximum length:

Validator:

Options: Valid prefix: "C:\"

Property	Value
ignoreCase	
value	C:\

No consoles to display at this time.

Figure 30. The newly created validation rule, now displayed in the Variable Value Configuration section of the Application Variables Information window

16. Save the application project.

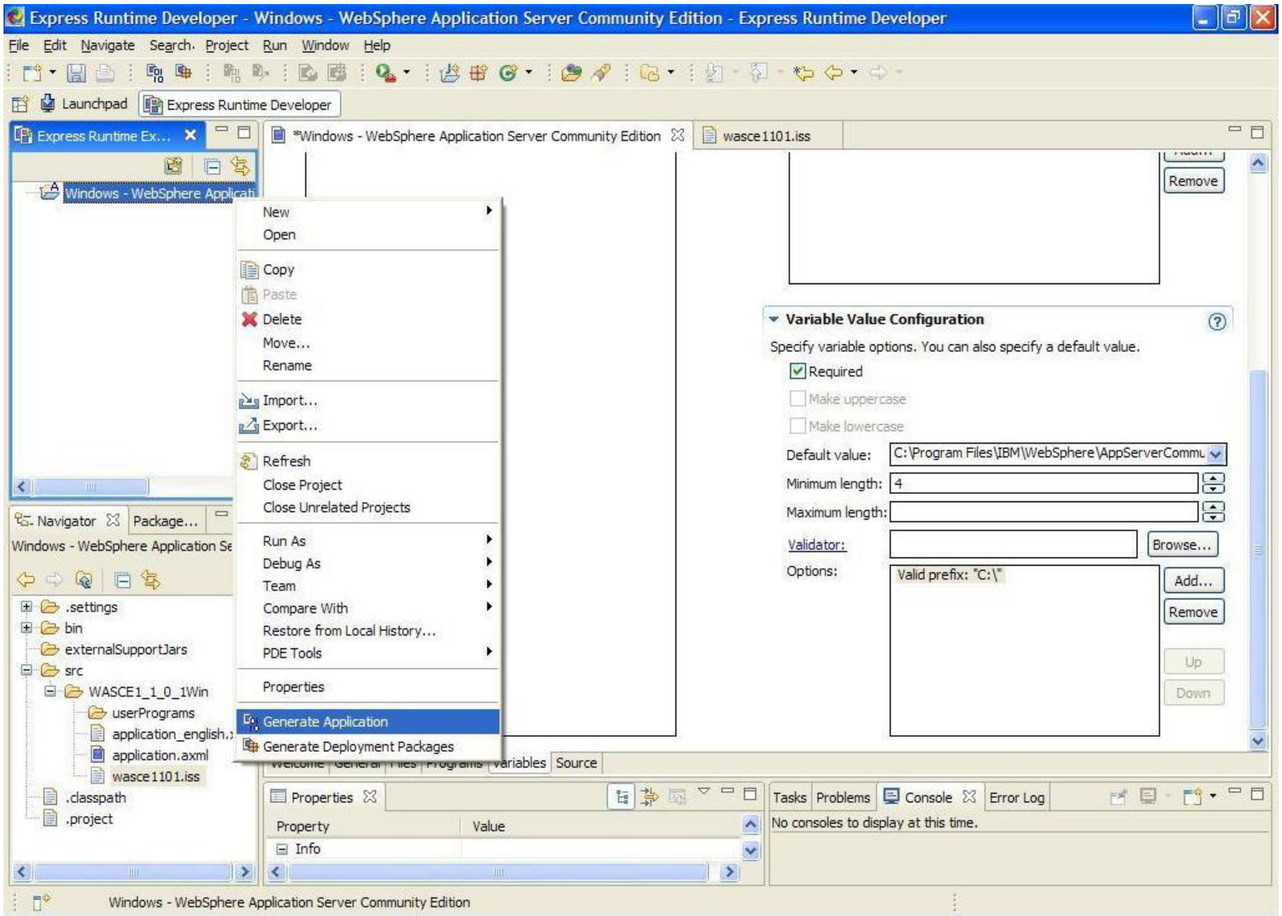
For more information about variables, see “Providing application variable information” on page 93.

Generating the application

In preparation for creating a deployment package, you must generate the application. The application generator produces one or more binary application files having an extension of .ser. A binary application file is created and stored in the project under the bin/<*applicationID*> folder, where <*applicationID*> is the ID specific to your application wrapper.

To generate the application .ser files, complete the following steps:

1. Right-click the application project and select **Generate Application**.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Launchpad
Express Runtime Developer
Express Runtime Ex...
Windows - WebSphere Applicati
*Windows - WebSphere Application Server Community Edition
wasce1101.iss
Remove
New
Open
Copy
Paste
Delete
Move...
Rename
Import...
Export...
Refresh
Close Project
Close Unrelated Projects
Run As
Debug As
Team
Compare With
Restore from Local History...
PDE Tools
Properties
Generate Application
Generate Deployment Packages
Navigator
Package...
Windows - WebSphere Application Se
.settings
bin
externalSupportJars
src
WASCE1_1_0_1Win
userPrograms
application_english.:
application.axml
wasce1101.iss
.classpath
.project
Variable Value Configuration
Validator:
Browse...
Add...
Remove
Up
Down
Welcome General Files Programs Variables Source
Properties
Property
Value
Info
Tasks Problems Console Error Log
Windows - WebSphere Application Server Community Edition

Specify variable options. You can also specify a default value.

☑ Required

☐ Make uppercase

☐ Make lowercase

Default value: C:\Program Files\IBM\WebSphere\AppServerCommu

Minimum length: 4

Maximum length:

Options: Valid prefix: "C:\"

No consoles to display at this time.

Figure 31. Generating the application

2. Click **OK** when the application generation is complete.

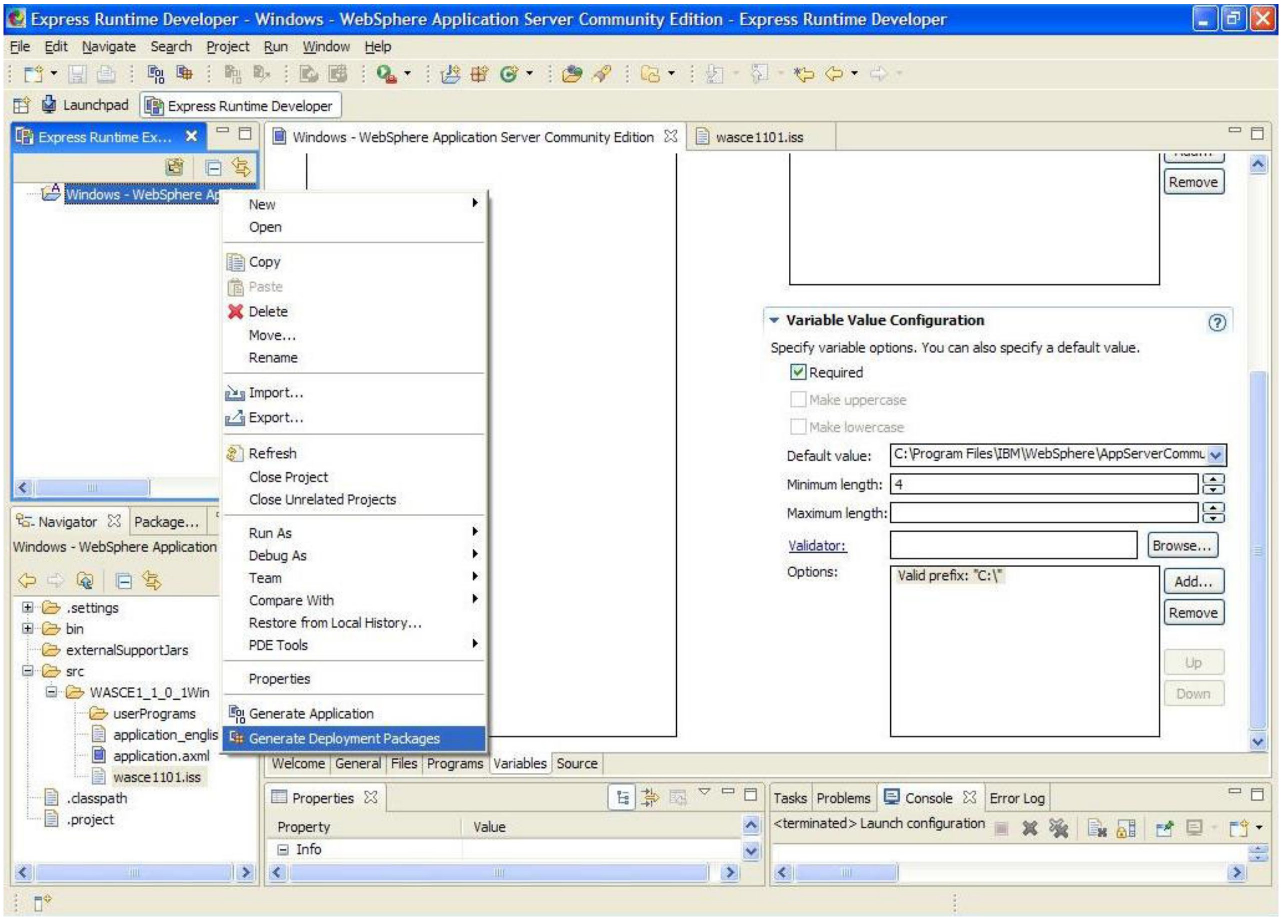
Generating the deployment package

The deployment package generator creates a deployment package JAR file. This package contains the software image files specified on the files tab of the application wrapper. This package is not stored within the application wrapper's project. Instead, it is placed in the IRU_common_resources project, which is intended to be a repository for common resources that multiple projects might need to access. Express Runtime developer creates the IRU_common_resources project as needed. The package is placed in the mediaJars folder of this project.

Note: You must generate deployment packages for each application you want to deploy with your solution.

To generate the deployment package for the application, complete the following steps:

1. Right-click the application project and select **Generate Deployment Package**.



Variable Value Configuration

Specify variable options. You can also specify a default value.

☑ Required

□ Make uppercase

□ Make lowercase

Default value: C:\Program Files\IBM\WebSphere\AppServerCommu

Minimum length: 4

Maximum length:

Validator:

Options: Valid prefix: "C:\"

Figure 32. Generating the deployment package

2. Click **OK**.

Importing additional application projects into the workspace

The final sample solution will also include a Linux[®] version of the WebSphere Application Server Community Edition application project and an additional application project called MidwestTrucking. For expediency, you will import these completed application projects, rather than creating them yourself.

To import these application projects, complete the following steps:

1. From Express Runtime developer, select **File** → **Import**. The Select page of the Import wizard opens.
2. Select **General** → **Existing Projects into Workspace**.

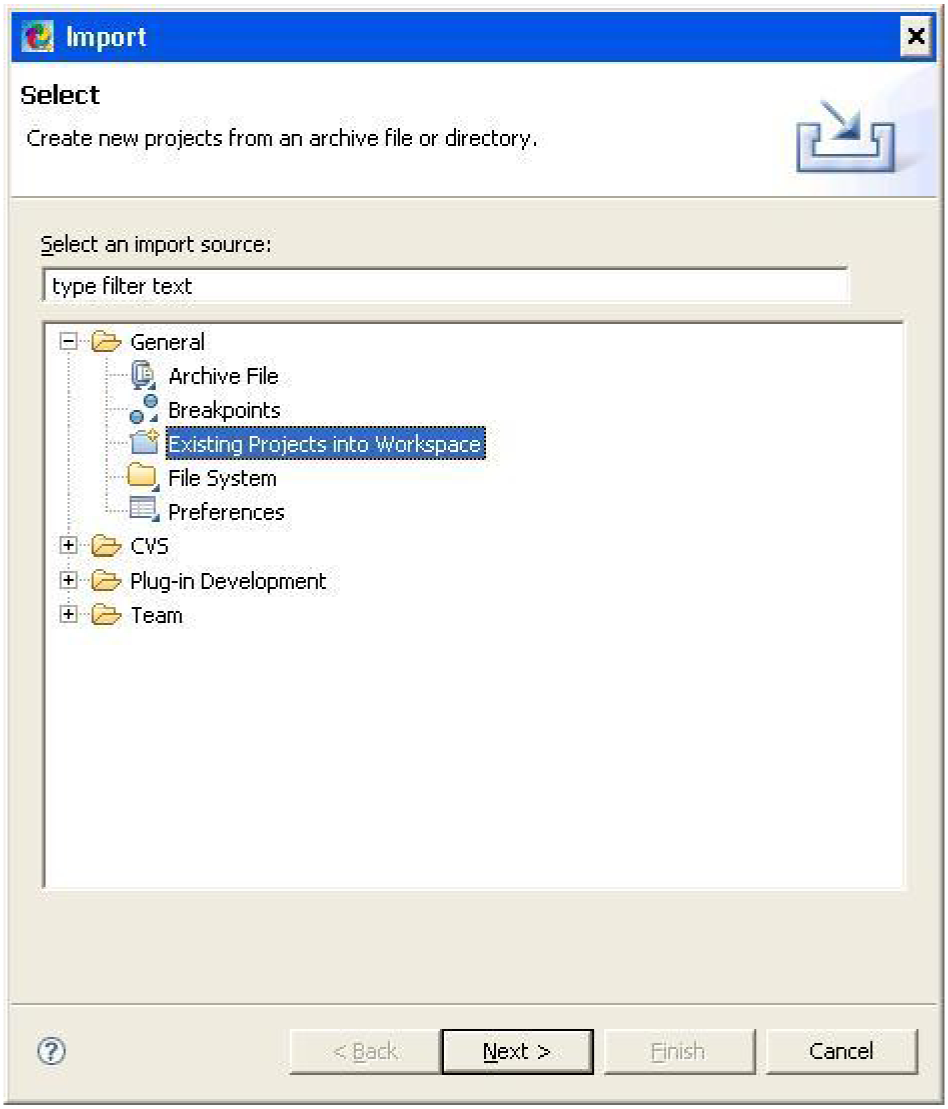
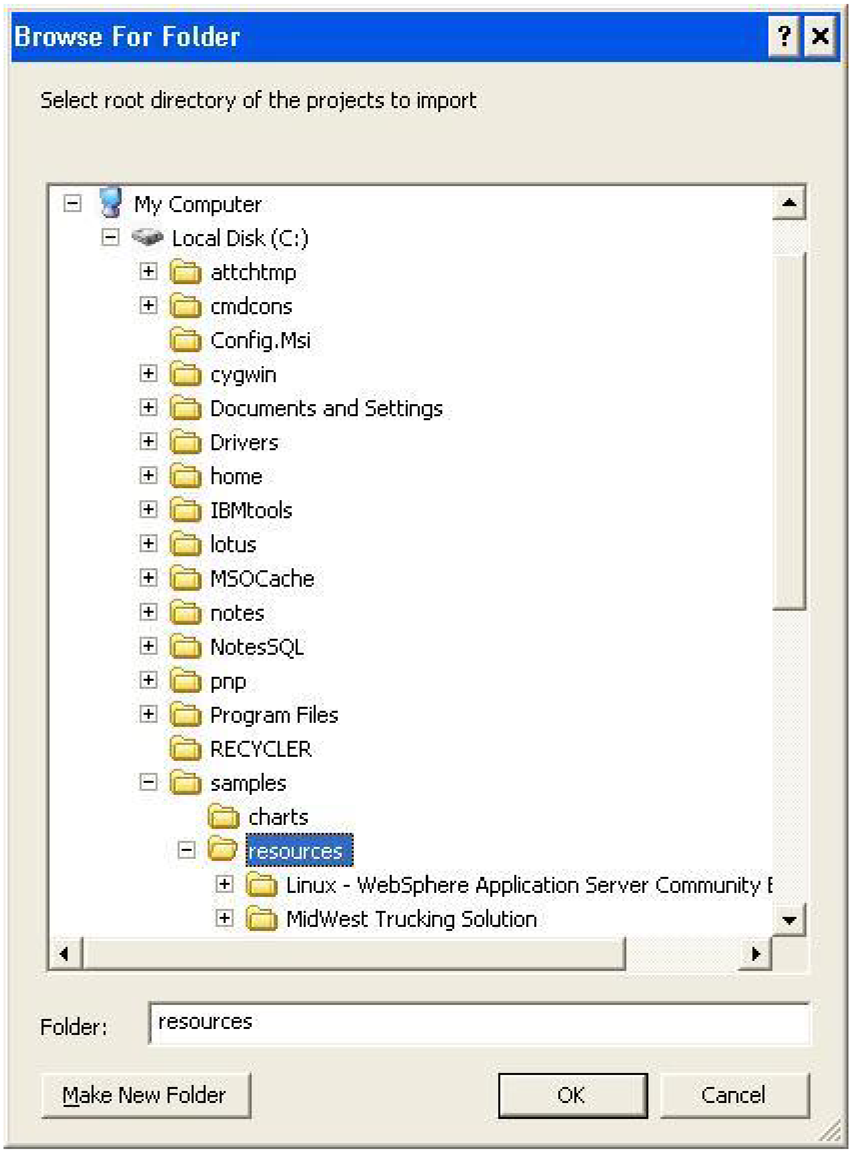


Figure 33. Import wizard: Select page

3. Click **Next**. The Import Projects window opens.
4. Click **Browse** next to the **Select root directory** field and navigate to the location to which you extracted the contents of the samples.zip file. Select the **resources** folder and click **OK**.



Browse For Folder

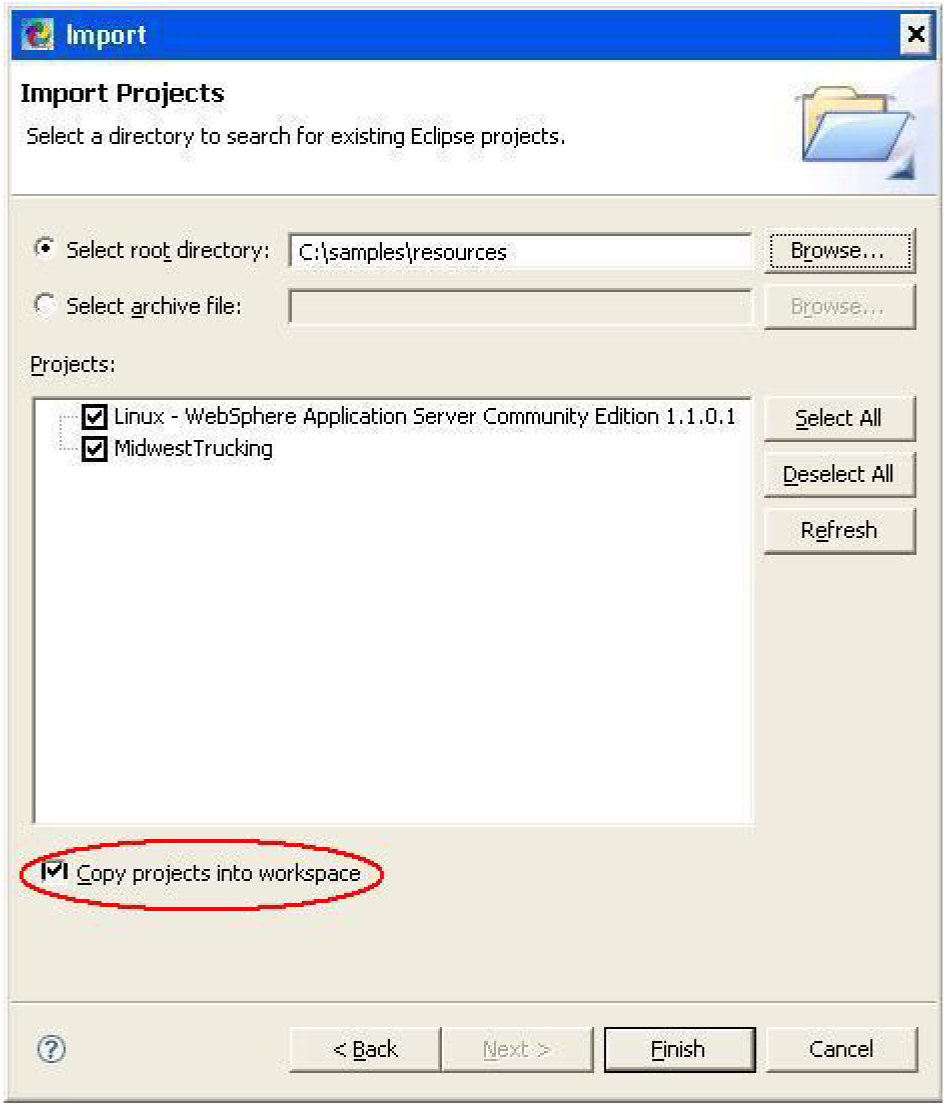
Select root directory of the projects to import

Folder: resources

Figure 34. Import wizard: Select root directory

You are returned to the Import Projects window.

5. If they are not selected already, select the Linux - WebSphere Application Server Community Edition 1.1.0.1 and MidwestTrucking projects in the **Projects** field.



Import

Import Projects

Select a directory to search for existing Eclipse projects.

● Select root directory: C:\samples\resources

○ Select archive file:

Projects:

☑ Linux - WebSphere Application Server Community Edition 1.1.0.1

☑ MidwestTrucking

☑ Copy projects into workspace

Figure 35. Import wizard: Import Projects page with projects selected

6. If it is not selected already, select the **Copy projects into workspace** check box.
7. Click **Finish**.
8. Double-click the Linux - WebSphere Application Server Community Edition 1.1.0.1 project to open it.
9. Click the **Files** tab.
10. Click **Disk 1** to select it.
11. Click **Update** (next to the **Files** field).
12. Browse to the location to which you extracted the contents of the samples.zip file. Select the samples\wasce\linux folder and click **OK**.
13. Select the wasce_setup-1.1.0.1-unix.bin file and click **Finish**.
14. Right-click the Linux - WebSphere Application Server Community Edition 1.1.0.1 application project in the explorer pane and click **Generate the Application**. Click **OK**.

15. Right-click the Linux - WebSphere Application Server Community Edition 1.1.0.1 application project in the explorer pane and click **Generate Deployment Packages**. Click **OK**.
16. Double-click the MidwestTrucking application project to open it.
17. Click the **Files** tab.
18. Click **Disk 1** to select it.
19. Click **Update** (next to the **Files** field).
20. Browse to the location to which you extracted the contents of the samples.zip file and select the **samples** folder. Click **OK**.
21. If it is not selected already, select **charts** and click **Finish**.
22. Right-click the MidWestTrucking application project in the explorer pane and click **Generate the Application**. Click **OK**.
23. Right-click in the MidWestTrucking application project in the explorer pane and click **Generate Deployment Packages**.

The MidwestTrucking and Linux - WebSphere Application Server Community Edition 1.1.0.1 projects you imported from the samples/resources directory are complete and ready to use. Both application wrappers use Java-based programs that are included in this tutorial as a reference for using the Solution Assembly Toolkit support framework APIs.

These examples perform the following functions:

- The MidwestTrucking application wrapper includes the following programs:
  - The MidwestTruckingPDC.java program is a predeployment checker. It verifies that WebSphere Application Server Community Edition 1.1.0.1 is installed on the computer and that enough hard drive space (1MB) is available on the specified target directory.
  - The MidwestTrucking.java program is the main installation program. It copies the Solution Assembly Toolkit - Overview chart deck to the specified directory.
- The Linux - WebSphere Application Server Community Edition 1.1.0.1 application wrapper includes the WasceLinuxPDC.java program, which is a predeployment checker. It verifies that WebSphere Application Server Community Edition 1.1.0.1 is installed on the computer and logs messages to indicate the status.

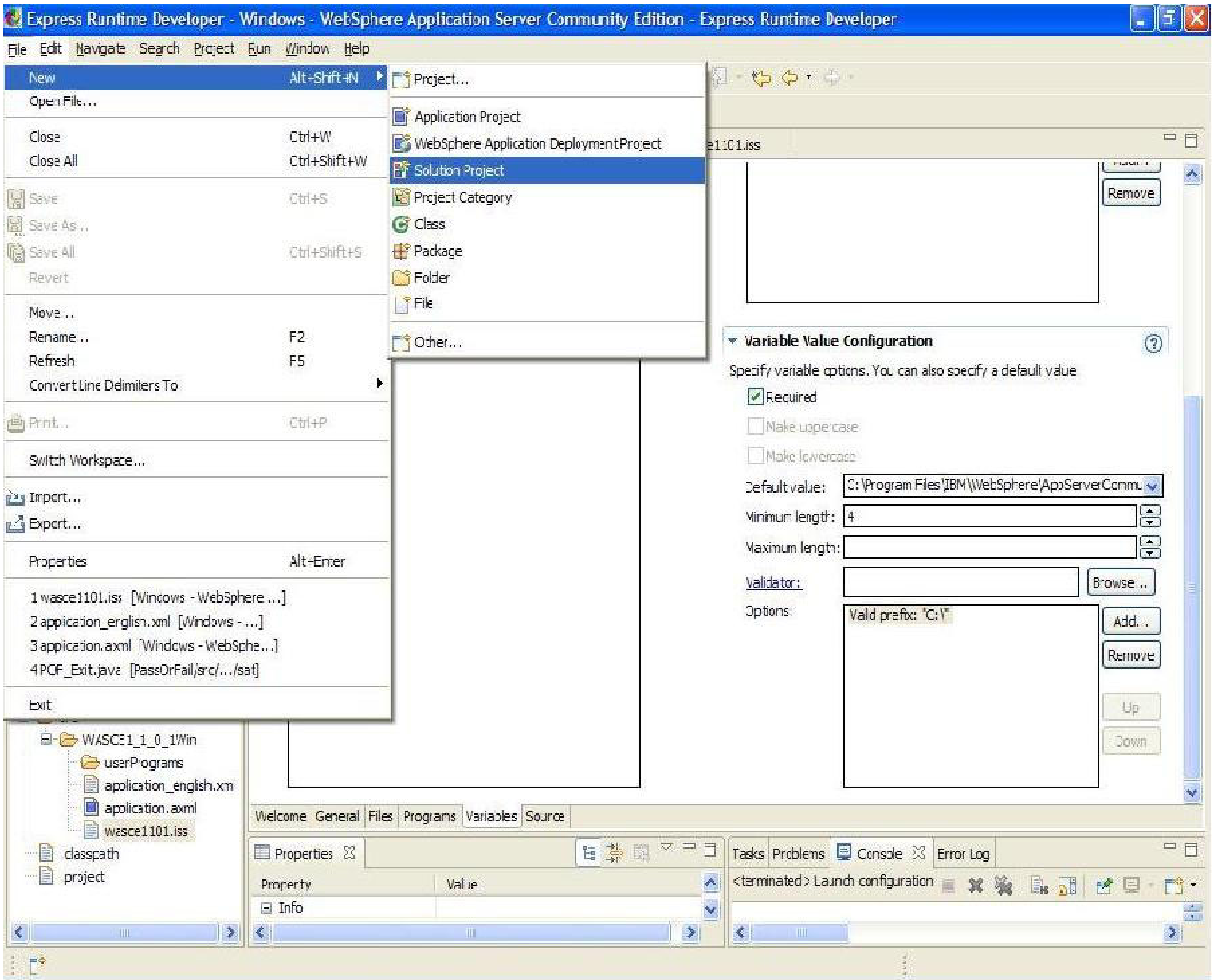
Each of these programs utilize common functions provided by the support framework. The full API documentation is available at C:\Program Files\IBM\SAT\2.2\SolutionEnabler\javadoc\index.html. If you modified the default installation directory for Solution Assembly Toolkit, modify the path as necessary.

Creating the sample solution wrapper

You can use the Express Runtime developer to package an application or applications into a solution. This is done with a solution project, or *solution wrapper*, that references existing applications and provides additional configuration information.

To create a new solution project, complete the following steps:

1. Select **File** → **New** → **Solution Project**.


Express Runtime Developer - Windows - WebSphere Application Server Community Edition - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
New Alt+Shift+N
Open File...
Close Ctrl+W
Close All Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To
Print... Ctrl+P
Switch Workspace...
Import...
Export...
Properties Alt+Enter
1 wasce1101.iss [Windows - WebSphere ...]
2 application_english.xml [Windows - ...]
3 application.axml [Windows - WebSphe...]
4 POF_Exit.java [PassOrFail/src/.../sat]
Exit
Project...
Application Project
WebSphere Application Deployment Project
Solution Project
Project Category
Class
Package
Folder
File
Other...
Variable Value Configuration
Specify variable options. You can also specify a default value
Remove
Browse...
Add...
Up
Down
WASCE1_1_0_1Win
userPrograms
application_english.xml
application.axml
wasce1101.iss
classpath
project
Welcome General Files Programs Variables Source
Properties
Property Value
Info
Tasks Problems Console Error Log
<terminated> Launch configuration

☑ Required

□ Make uppercase

□ Make lowercase

Default value: C:\Program Files\IBM\WebSphere\AppServerCommu

Minimum length: 4

Maximum length:

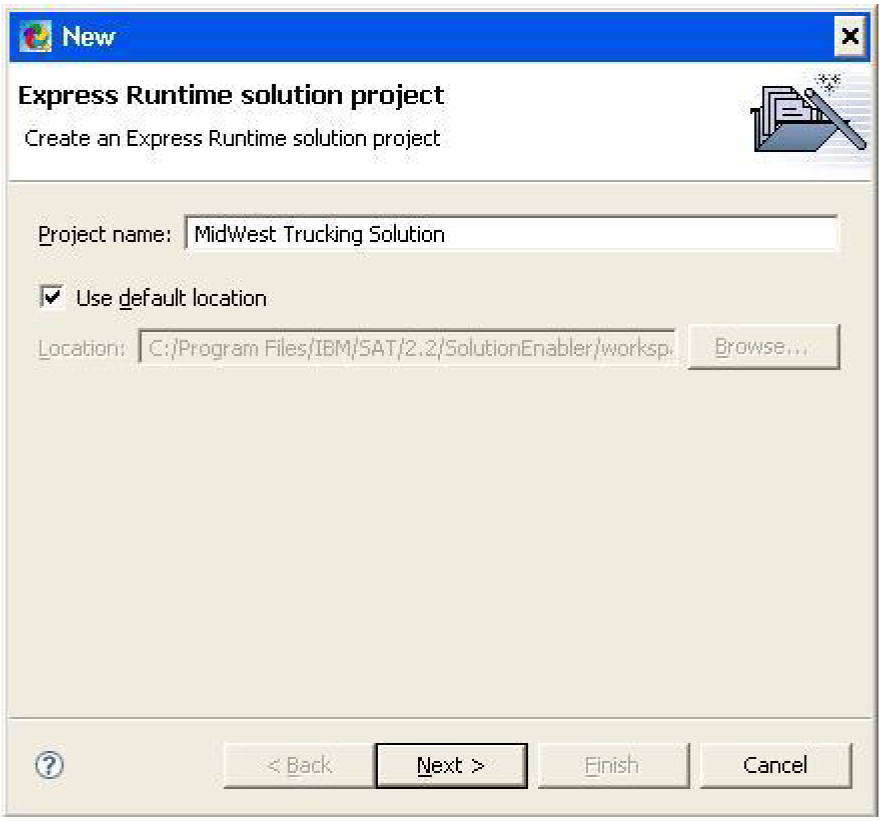
Validator:

Options: Valid prefix: "C:\"

Figure 36. Creating a new solution project

The New solution project wizard opens.

2. In the **Project name** field, type `MidWest Trucking Solution`. Leave the **Use default location** check box selected. See Figure 37 on page 44.


New
Express Runtime solution project
Create an Express Runtime solution project
< Back
Next >
Finish
Cancel
Browse...

Project name: MidWest Trucking Solution

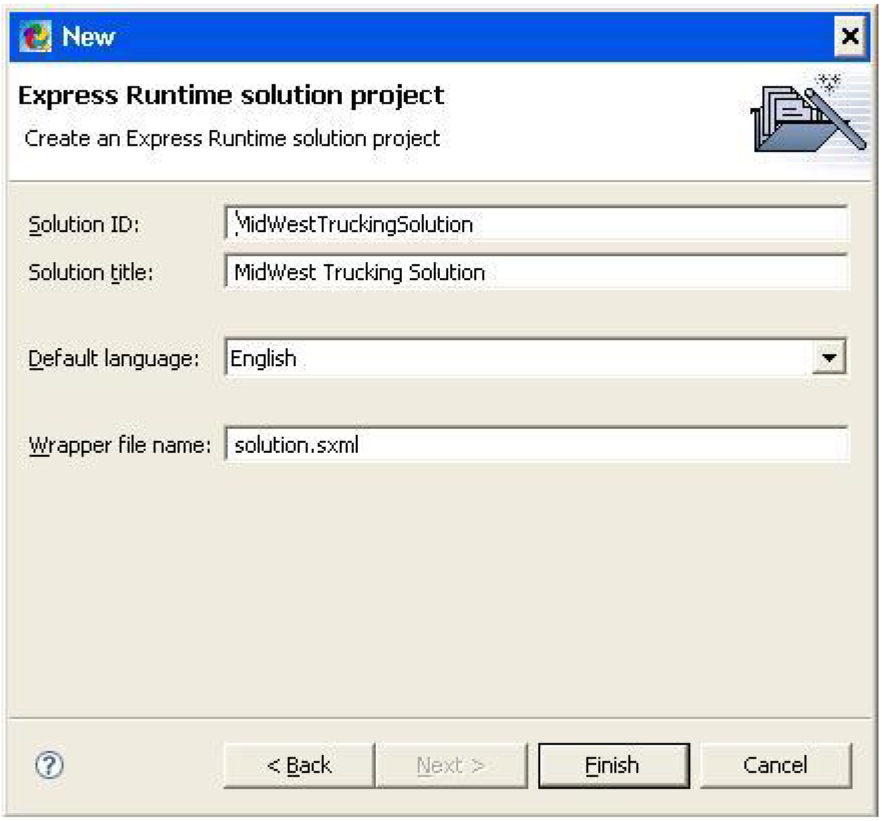
☑ Use default location

Location: C:/Program Files/IBM/SAT/2.2/SolutionEnabler/worksp

Figure 37. New solution project wizard: First page

Click **Next**.

3. On the next page of the wizard, type or select the following values, as shown in Figure 38 on page 45:
   - **Solution ID**: `MidWestTruckingSolution`
   - **Solution title**: `MidWest Trucking Solution`
   - **Default language**: English
   - **Wrapper file name**: solution.sxml (default value)



Express Runtime solution project

Create an Express Runtime solution project

Solution ID: MidWestTruckingSolution

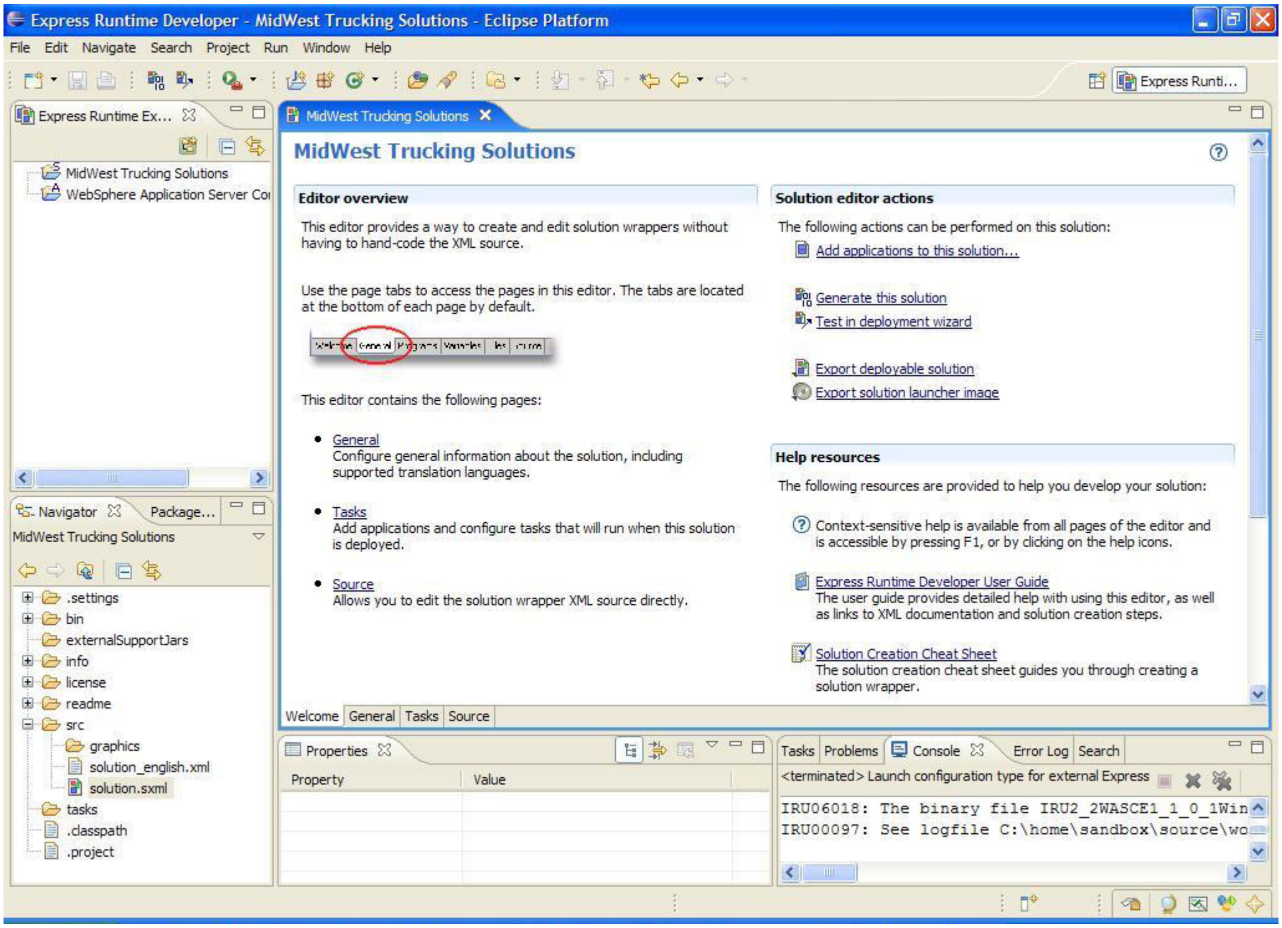
Solution title: MidWest Trucking Solution

Default language: English

Wrapper file name: solution.sxml

Figure 38. New solution project wizard: Second page

Click **Finish**. You are returned to the solution project editor. The Welcome page is displayed in the right pane of the window. The Express Runtime Explorer is displayed in the upper left pane, and the Navigator is displayed in the lower left pane.



MidWest Trucking Solutions

Editor overview

This editor provides a way to create and edit solution wrappers without having to hand-code the XML source.

Use the page tabs to access the pages in this editor. The tabs are located at the bottom of each page by default.

This editor contains the following pages:

- General
  Configure general information about the solution, including supported translation languages.
- Tasks
  Add applications and configure tasks that will run when this solution is deployed.
- Source
  Allows you to edit the solution wrapper XML source directly.

Solution editor actions

The following actions can be performed on this solution:

Add applications to this solution...

Generate this solution

Test in deployment wizard

Export deployable solution

Export solution launcher image

Help resources

The following resources are provided to help you develop your solution:

Context-sensitive help is available from all pages of the editor and is accessible by pressing F1, or by clicking on the help icons.

Express Runtime Developer User Guide
The user guide provides detailed help with using this editor, as well as links to XML documentation and solution creation steps.

Solution Creation Cheat Sheet
The solution creation cheat sheet guides you through creating a solution wrapper.

```
<terminated> Launch configuration type for external Express
IRU06018: The binary file IRU2_2WASCE1_1_0_1Win
IRU00097: See logfile C:\home\sandbox\source\wo
```

Figure 39. Solution project editor: Welcome page

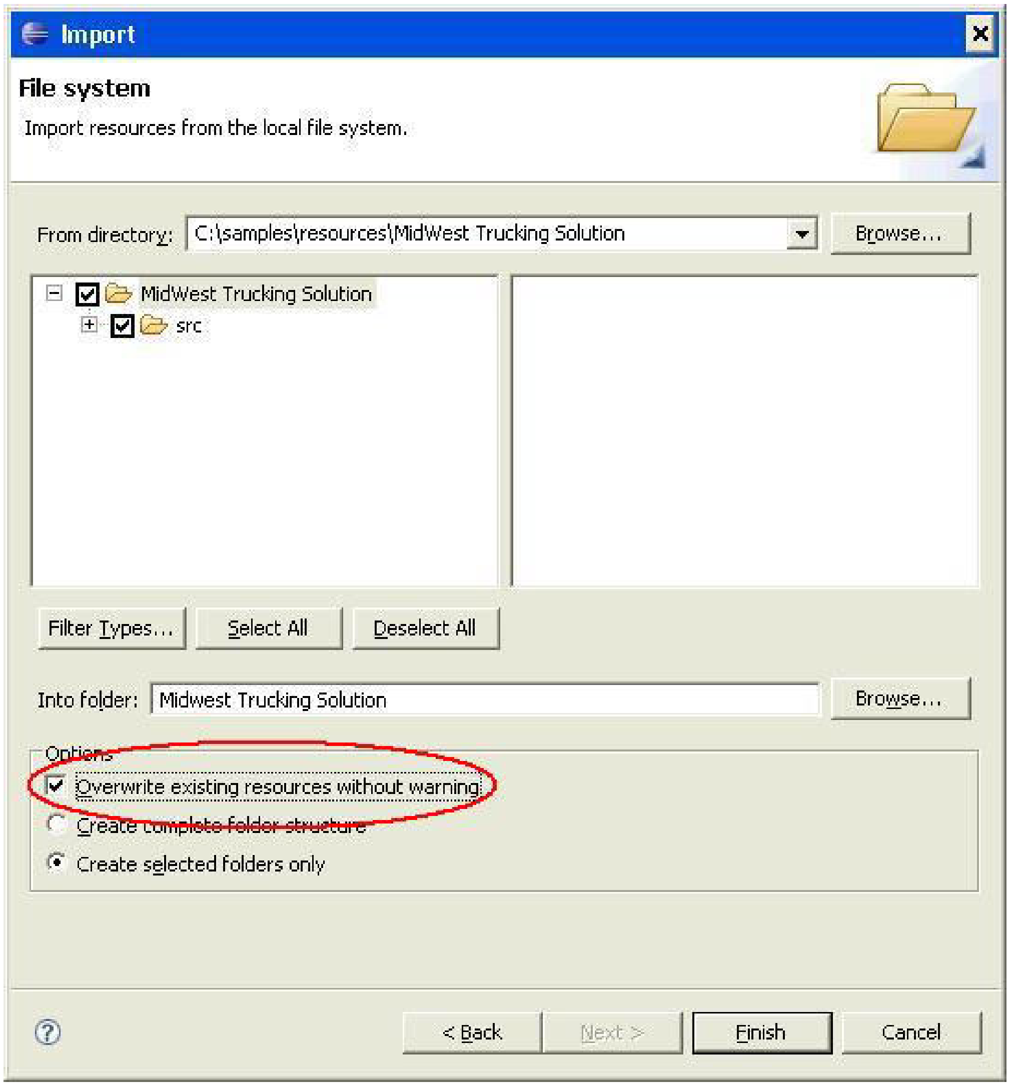
Importing additional solution resources

For the purposes of this tutorial, a completed solution_english.xml file has been provided for you. This file ensures that you do not have to add a lot of text manually to the XML file created when you created the solution project.

To incorporate the content into the solution_english.xml file for the solution, complete the following steps:

1. Click the MidWest Trucking Solution solution project in the Express Runtime Explorer pane to select it.
2. Right-click the Express Runtime Explorer pane and click **Import**. The Import wizard: Select import source page opens.
3. Click **General** → **File System** and then click **Next**.
4. Browse to the directory to which you extracted the contents of the samples.zip file. Locate and expand the **resources** folder.
5. Click the MidWest Trucking Solution folder, and click **OK**. The Import resources from the local file system window opens.
6. Select the check box next to **MidWest Trucking Solution**.

7. Ensure that **Overwrite existing resources without warning** is selected, and then click **Finish**.



Import

File system

Import resources from the local file system.

From directory: C:\samples\resources\MidWest Trucking Solution

- ☑ MidWest Trucking Solution
  - ☑ src

Into folder: Midwest Trucking Solution

Options

☑ Overwrite existing resources without warning

○ Create complete folder structure

● Create selected folders only

Figure 40. Import wizard: Select import source page

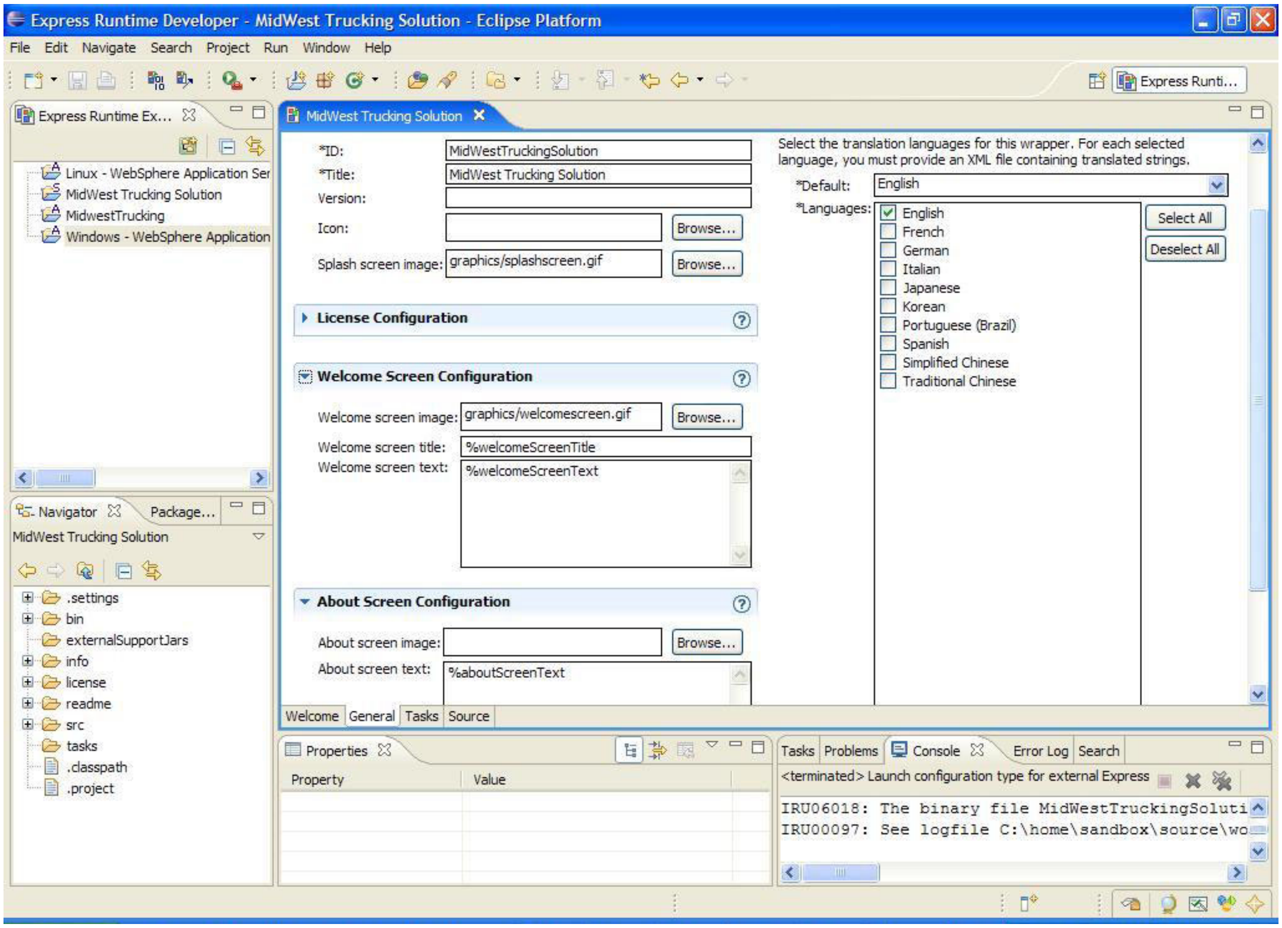
8. Save the solution project.

Customizing the solution with graphics and text

To customize your solution, you can add brand-specific text and images that will be displayed during deployment.

To add text and images to the solution, complete the following steps:

1. With the solution project opened, click the **General** tab. The general solution information is displayed. Many of the fields contain information you entered when you created the project.


Express Runtime Developer - MidWest Trucking Solution - Eclipse Platform
File Edit Navigate Search Project Run Window Help
Express Runti...
Express Runtime Ex...
Linux - WebSphere Application Ser
MidWest Trucking Solution
MidwestTrucking
Windows - WebSphere Application
Navigator
Package...
MidWest Trucking Solution
.settings
bin
externalSupportJars
info
license
readme
src
tasks
.classpath
.project
MidWest Trucking Solution
Browse...
License Configuration
Welcome Screen Configuration
About Screen Configuration
Select All
Deselect All
Welcome General Tasks Source
Properties
Property
Value
Tasks Problems Console Error Log Search

*ID: MidWestTruckingSolution

*Title: MidWest Trucking Solution

Version:

Icon:

Splash screen image: graphics/splashscreen.gif

Welcome screen image: graphics/welcomescreen.gif

Welcome screen title: %welcomeScreenTitle

Welcome screen text: %welcomeScreenText

About screen image:

About screen text: %aboutScreenText

Select the translation languages for this wrapper. For each selected language, you must provide an XML file containing translated strings.

*Default: English

*Languages:
☑ English
☐ French
☐ German
☐ Italian
☐ Japanese
☐ Korean
☐ Portuguese (Brazil)
☐ Spanish
☐ Simplified Chinese
☐ Traditional Chinese

```
<terminated> Launch configuration type for external Express
IRU06018: The binary file MidWestTruckingSoluti
IRU00097: See logfile C:\home\sandbox\source\wo
```

Figure 41. Solution project editor: General page

2. To include an image on the splash screen during deployment, click **Browse** next to the **Splash screen image** field. This field is in the Basic Solution Configuration section of the General page. Browse to the splashscreen.gif file, located in the C:\Program Files\IBM\SAT\2.2\SolutionEnabler\workspace\ Midwest Trucking Solutions\src\graphics folder. Select the file, and click **Open**.
3. To include an image on the Welcome screen during deployment, click **Browse** next to the **Welcome screen image** field. This field is in the Welcome Screen Configuration section of the General page. Browse to the welcomescreen.gif file, located in the C:\Program Files\IBM\SAT\2.2\SolutionEnabler\workspace\ Midwest Trucking Solution\src\graphics folder. Select the file, and click **Open**.
4. To include a title on the Welcome screen, type `%welcomeScreenTitle` in the **Welcome screen title** field.
5. To include text on the Welcome screen, type `%welcomeScreenText` in the **Welcome screen text** field.
6. To include text about your solution on the About screen accessed through the Help menu, type `%aboutScreenText` in the **About screen text** field. This field is in the About Screen Configuration section of the General page.
7. Save the project.

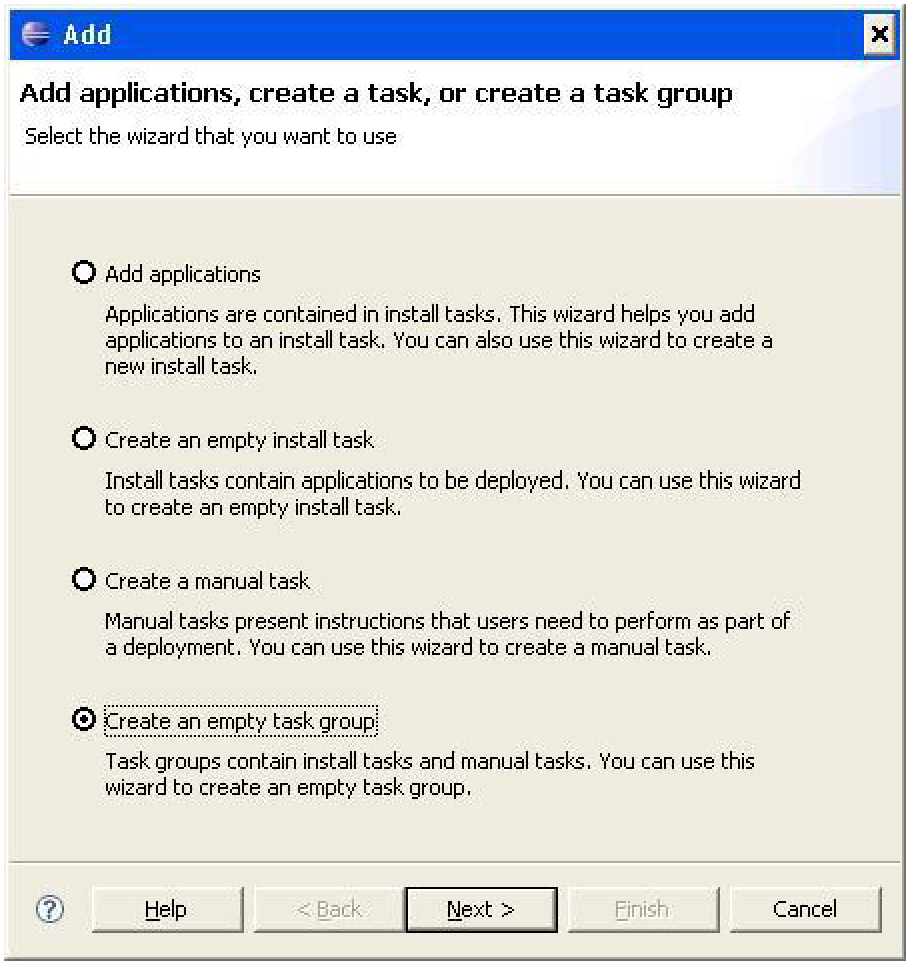
Next we will add task groups to the solution. The task groups will enable the user to deploy either a Windows or a Linux version of the solution.

Adding a Windows task group to the solution

A task group is used to group related tasks. The deployment wizard presents the end user with a list of all the task groups in the solution. Users select the tasks to deploy.

To add the Windows task group to the sample solution, complete the following steps:

1. Click the **Tasks** tab in the solution project editor.
2. Click **Add** in the **Solution Tasks** section of the window. The list of Add wizards opens.
3. Select **Create an empty task group**.



Add

Add applications, create a task, or create a task group

Select the wizard that you want to use

○ Add applications

Applications are contained in install tasks. This wizard helps you add applications to an install task. You can also use this wizard to create a new install task.

○ Create an empty install task

Install tasks contain applications to be deployed. You can use this wizard to create an empty install task.

○ Create a manual task

Manual tasks present instructions that users need to perform as part of a deployment. You can use this wizard to create a manual task.

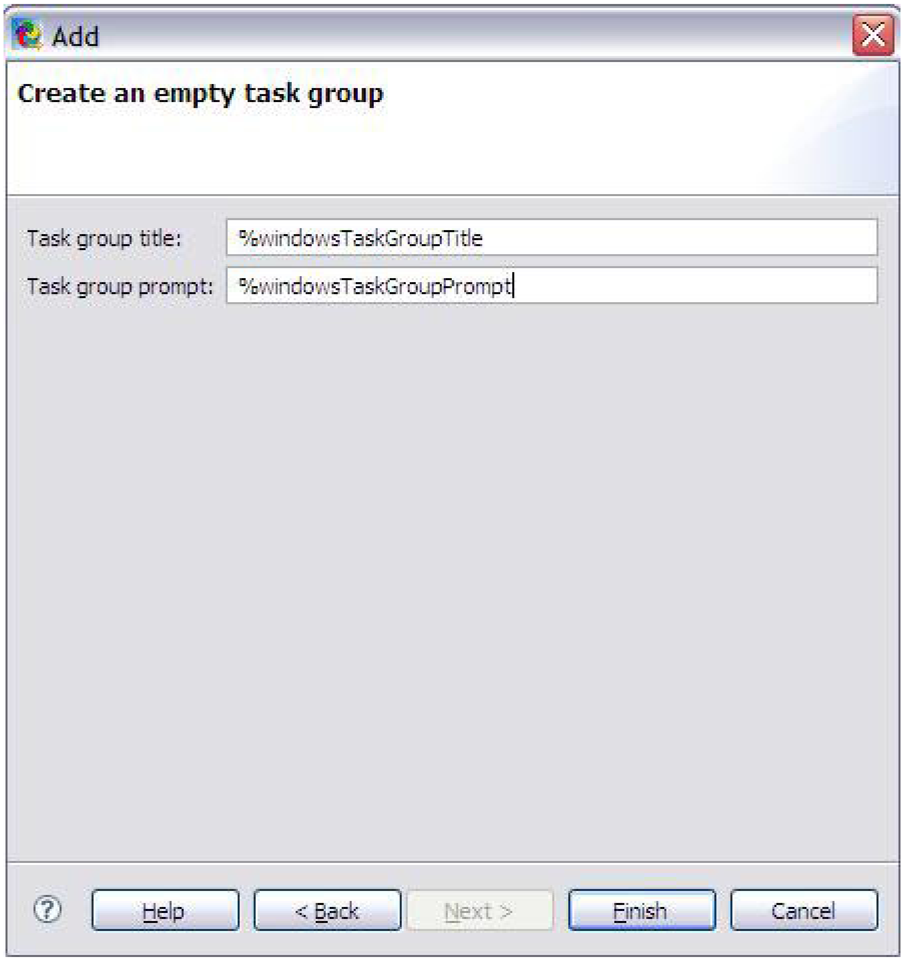
● Create an empty task group

Task groups contain install tasks and manual tasks. You can use this wizard to create an empty task group.

Figure 42. Add wizards; Create an empty task group selected

4. Click **Next**. The Create an empty task group wizard opens.
5. Type translatable keys for the Task group title and Task group prompt as follows:
   - **Task group title**: `%windowsTaskGroupTitle`
   - **Task group prompt**: `%windowsTaskGroupPrompt`

   See Figure 43 on page 50.



Task group title: %windowsTaskGroupTitle

Task group prompt: %windowsTaskGroupPrompt

Figure 43. Create an empty task group for Windows

6. Click **Finish**. You are returned to the Tasks page of the solution project editor.
7. Save the project.

For more information about task groups, see "Adding task groups to a solution" on page 111.

Adding a Linux task group to the solution

A task group is used to group related tasks. The deployment wizard presents the end user with a list of all the task groups in the solution. Users select the tasks to deploy.

To add the Linux task group to the sample solution, complete the following steps:

1. Click the **Tasks** tab in the solution project editor.
2. Click **Add** in the **Solution Tasks** section of the window. The list of Add wizards opens.
3. Select **Create an empty task group**.
4. Click **Next**. The Create an empty task group wizard opens.
5. Type translatable keys for the Task group title and Task group prompt as follows:
   - **Task group title**: `%linuxTaskGroupTitle`

- **Task group prompt**: `%linuxTaskGroupPrompt`

6. Click **Finish**. You are returned to the Tasks page of the solution project editor.
7. Save the project.

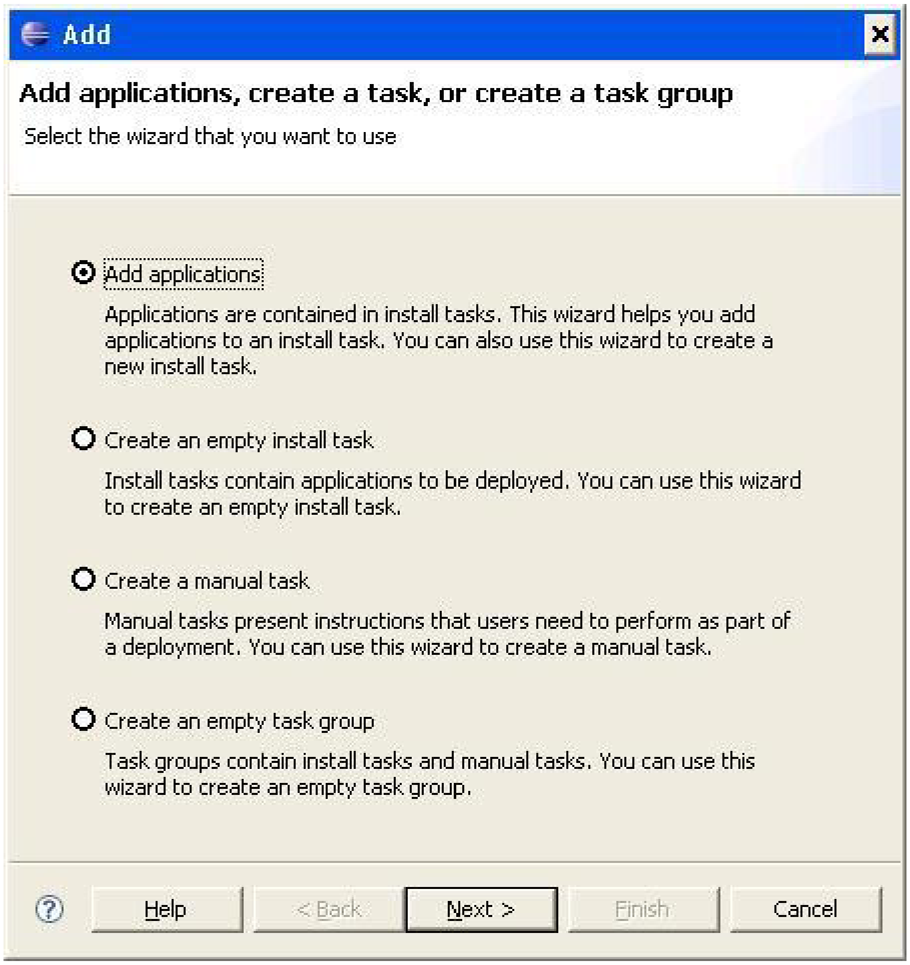
For more information about task groups, see "Adding task groups to a solution" on page 111.

Adding a Windows install task to the solution

An install task consists of a set of applications that are to be deployed to one or more target computers. Each of the install tasks in a solution is displayed in the deployment wizard. From the deployment wizard, the user can select any combination of install tasks for deployment to one or more target computers.

To add a Windows install task to the sample solution, complete the following steps:

1. Select the **Tasks** tab in the solution project editor.
2. In the **Solution tasks** section, click **Add**.
3. Select **Add applications**. In addition to allowing you to add applications to an existing task, this selection can also be used to create a new install task.



Select the wizard that you want to use

● Add applications

Applications are contained in install tasks. This wizard helps you add applications to an install task. You can also use this wizard to create a new install task.

○ Create an empty install task

Install tasks contain applications to be deployed. You can use this wizard to create an empty install task.

○ Create a manual task

Manual tasks present instructions that users need to perform as part of a deployment. You can use this wizard to create a manual task.

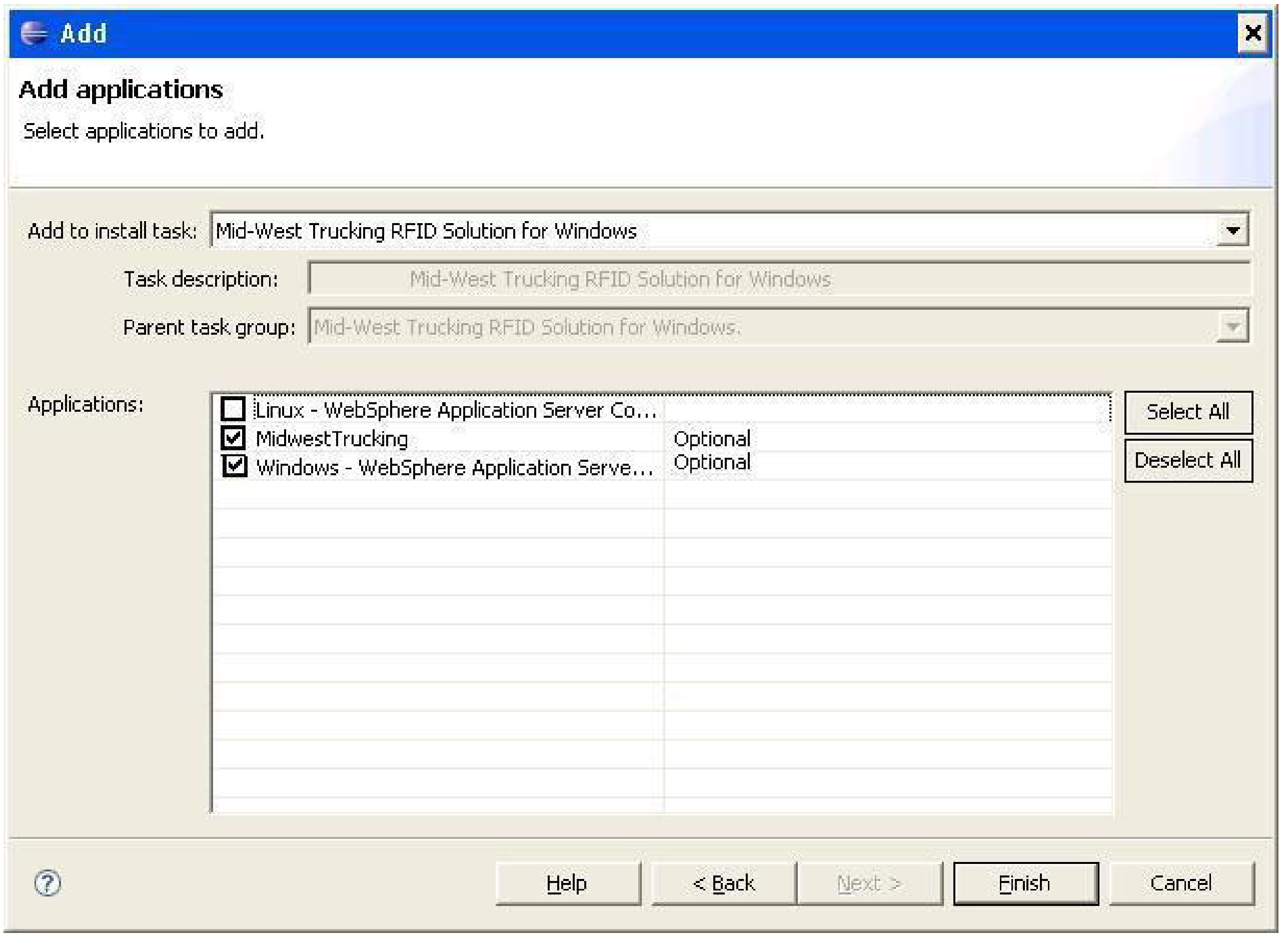
○ Create an empty task group

Task groups contain install tasks and manual tasks. You can use this wizard to create an empty task group.

Figure 44. Add wizards window; Add applications option selected

4. Click **Next**. The **Add applications** window opens.

5. Refer to Figure 45 and make the following entries and selections:
   a. In the **Add to install task** field, select **Create a new install task below**.
   b. In the **Task description** field, type `%windowsTaskDescription`.
   c. In the **Parent task group** field, select **Mid-West Trucking RFID Solution for Windows**.
   d. In the **Applications** section, select **Windows - WebSphere Application Server Community Edition** and **MidwestTrucking**.


Add

Add applications

Select applications to add.

Add to install task: Mid-West Trucking RFID Solution for Windows

Task description: Mid-West Trucking RFID Solution for Windows

Parent task group: Mid-West Trucking RFID Solution for Windows.

Applications:

□ Linux - WebSphere Application Server Co...	
☑ MidwestTrucking	Optional
☑ Windows - WebSphere Application Serve...	Optional

Figure 45. Add applications window with Windows - WebSphere Application Server Community Edition and MidwestTrucking applications selected

6. Click **Finish**.

For more information about adding install tasks to a solution, see "Adding install tasks to a solution" on page 109.

Adding a Linux install task to the solution

An install task consists of a set of applications that are to be deployed to one or more target computers. Each of the install tasks in a solution is displayed in the deployment wizard. From the deployment wizard, select any combination of install tasks for deployment to one or more target computers.

To add a Linux install task to the sample solution, complete the following steps:

1. Select the **Tasks** tab in the solution project editor.

2. In the **Solution tasks** section, click **Add**.
3. Select **Add applications**. In addition to allowing you to add applications to an existing task, this selection also creates a new install task.
4. Click **Next**. The **Add applications** window opens.
5. In the **Add to install task** field, select **Create a new install task below**.
6. Refer to Figure 45 on page 52 and make the following entries and selections in the remaining fields, changing instances of Windows to Linux:
   a. In the **Task description** field, type `%linuxTaskDescription`.
   b. In the **Parent task group** field, select **Mid-West Trucking RFID Solution for Linux**.
   c. In the **Applications** section, select **Linux - WebSphere Application Server Community Edition** and **MidwestTrucking**.
7. Click **Finish**.

See "Adding a Windows install task to the solution" on page 51 for images.

For more information about adding install tasks to a solution, see "Adding install tasks to a solution" on page 109.

Configuring solution tasks

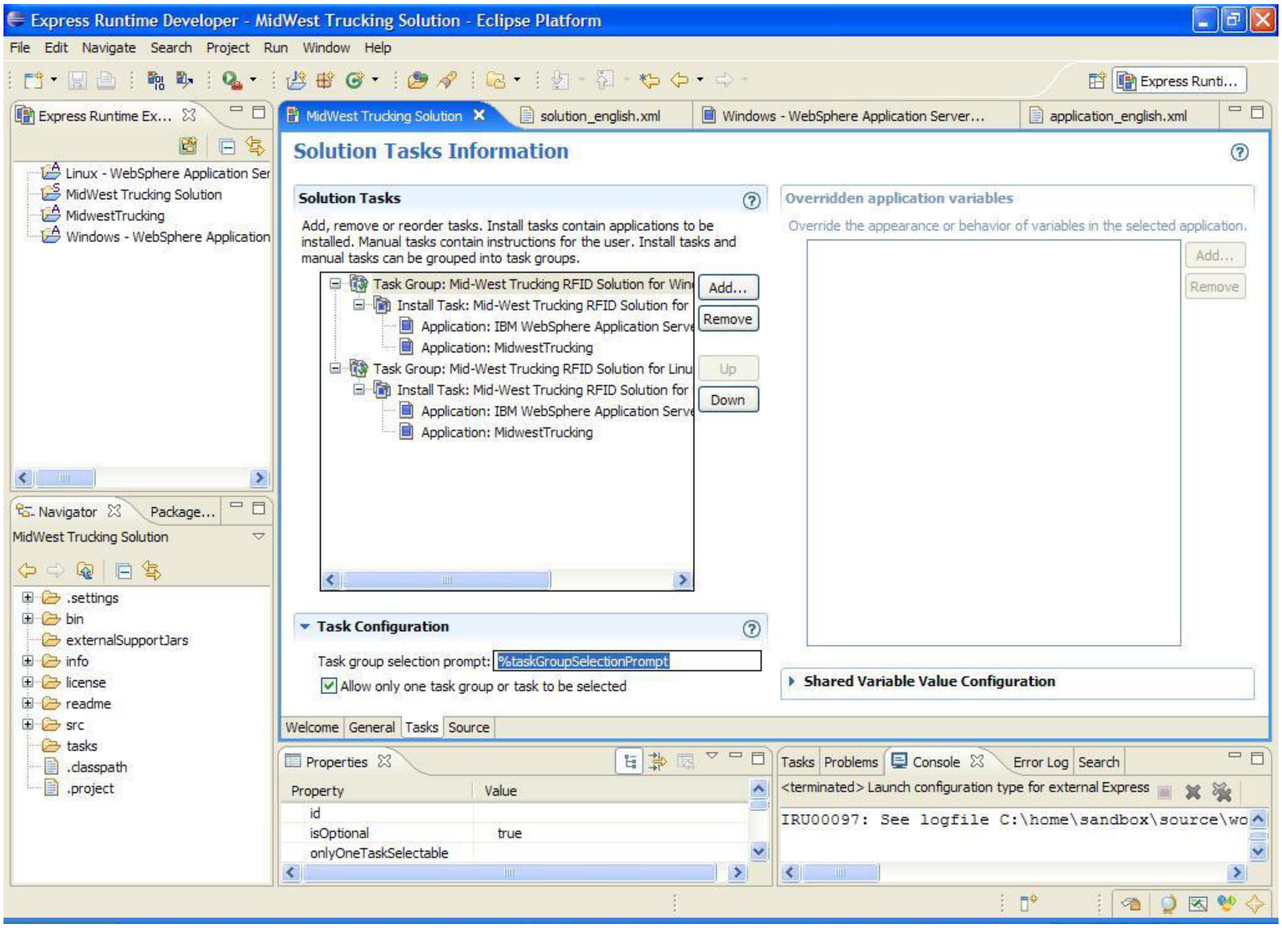
You can include task- and task group-specific information that will be displayed to the user during deployment.

To include task- and task group-specific information in the solution, complete the following steps:

Note:

Ensure that the install tasks and applications in your solution are in the same order as the install tasks in Figure 46 on page 54. To reorder a task, select it and click **Up** or **Down** as necessary.

1. To allow the end user to choose only one task group during deployment, in the Task Configuration section of the Tasks page, select the **Allow only one task group or task to be selected** check box.


Express Runtime Developer - MidWest Trucking Solution - Eclipse Platform
File Edit Navigate Search Project Run Window Help
Express Runti...
Express Runtime Ex...
Linux - WebSphere Application Ser
MidWest Trucking Solution
MidwestTrucking
Windows - WebSphere Application
MidWest Trucking Solution
solution_english.xml
Windows - WebSphere Application Server...
application_english.xml
Add...
Remove
Up
Down
Add...
Remove
Navigator
Package...
MidWest Trucking Solution
.settings
bin
externalSupportJars
info
license
readme
src
tasks
.classpath
.project
Welcome General Tasks Source
Properties
Tasks Problems Console Error Log Search

Solution Tasks Information

Solution Tasks

Add, remove or reorder tasks. Install tasks contain applications to be installed. Manual tasks contain instructions for the user. Install tasks and manual tasks can be grouped into task groups.

- Task Group: Mid-West Trucking RFID Solution for Win
  - Install Task: Mid-West Trucking RFID Solution for
    - Application: IBM WebSphere Application Serve
    - Application: MidwestTrucking
- Task Group: Mid-West Trucking RFID Solution for Linu
  - Install Task: Mid-West Trucking RFID Solution for
    - Application: IBM WebSphere Application Serve
    - Application: MidwestTrucking

Overridden application variables

Override the appearance or behavior of variables in the selected application.

Task Configuration

Task group selection prompt: %taskGroupSelectionPrompt

☑ Allow only one task group or task to be selected

Shared Variable Value Configuration

Property	Value
id	
isOptional	true
onlyOneTaskSelectable	

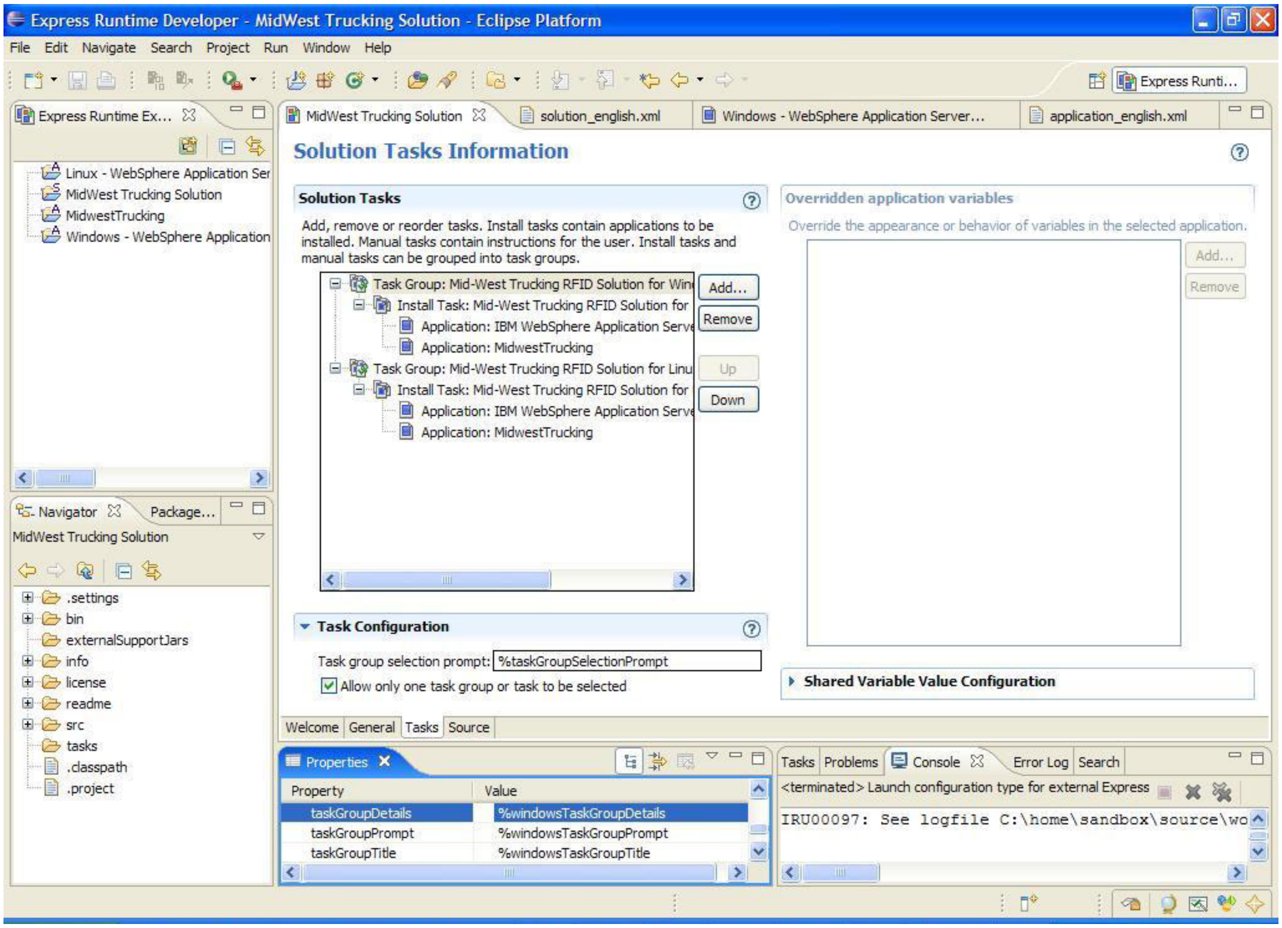
<terminated> Launch configuration type for external Express

```
IRU00097: See logfile C:\home\sandbox\source\wo
```

Figure 46. Tasks page showing selection of Allow only one task group or task to be selected check box

2. To include details for the Windows task group that will appear during deployment, complete the following steps:
   a. Select **Task Group: Mid-West Trucking RFID Solution for Windows** in the Solution Tasks section of the window.
   b. Click the **Properties** tab at the bottom of the page.
   c. In the **taskGroupDetails** field, type `%windowsTaskGroupDetails`.

   See Figure 47 on page 55.



Solution Tasks Information

Solution Tasks

Add, remove or reorder tasks. Install tasks contain applications to be installed. Manual tasks contain instructions for the user. Install tasks and manual tasks can be grouped into task groups.

- Task Group: Mid-West Trucking RFID Solution for Win
  - Install Task: Mid-West Trucking RFID Solution for
    - Application: IBM WebSphere Application Serve
    - Application: MidwestTrucking
- Task Group: Mid-West Trucking RFID Solution for Linu
  - Install Task: Mid-West Trucking RFID Solution for
    - Application: IBM WebSphere Application Serve
    - Application: MidwestTrucking

Overridden application variables

Override the appearance or behavior of variables in the selected application.

Task Configuration

Task group selection prompt: %taskGroupSelectionPrompt

☑ Allow only one task group or task to be selected

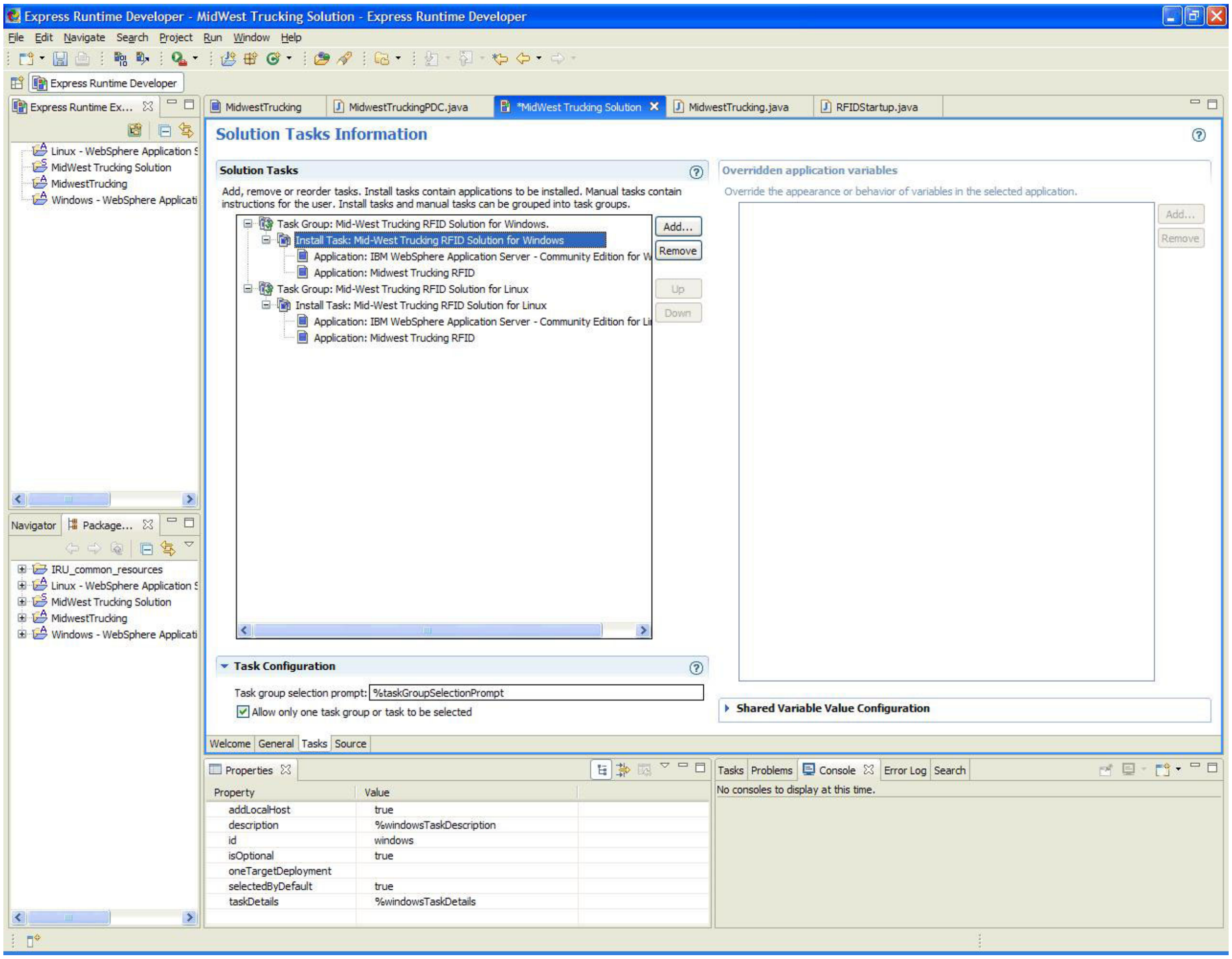
Property	Value
taskGroupDetails	%windowsTaskGroupDetails
taskGroupPrompt	%windowsTaskGroupPrompt
taskGroupTitle	%windowsTaskGroupTitle

```
<terminated> Launch configuration type for external Express
IRU00097: See logfile C:\home\sandbox\source\wo
```

Figure 47. Adding details, a prompt, and a title to the Mid-West Trucking RFID Solution for Windows task group

3. Repeat step 2 on page 54 to include details for the Linux task group. Select **Task Group: Mid-West Trucking RFID Solution for Linux** and change the key value as follows:
   - **taskGroupDetails**: `%linuxTaskGroupDetails`
4. To add details about the Mid-West Trucking RFID Solution for Windows install task, complete the following steps:
   a. Select **Install Task: Mid-West Trucking RFID Solution for Windows** in the Solution Tasks section of the Tasks page.
   b. Click the **Properties** tab at the bottom of the page.
   c. Type `windows` in the **id** field.
   d. Type `%windowsTaskDetails` in the **taskDetails** field.

   See Figure 48 on page 56.


Express Runtime Developer - MidWest Trucking Solution - Express Runtime Developer
File Edit Navigate Search Project Run Window Help
Express Runtime Developer
Express Runtime Ex...
Linux - WebSphere Application S
MidWest Trucking Solution
MidwestTrucking
Windows - WebSphere Applicati
Navigator
Package...
IRU_common_resources
MidwestTrucking
MidwestTruckingPDC.java
*MidWest Trucking Solution
MidwestTrucking.java
RFIDStartup.java
Add...
Remove
Up
Down
Shared Variable Value Configuration
Welcome
General
Tasks
Source
Properties
Tasks
Problems
Console
Error Log
Search

Solution Tasks Information

Solution Tasks

Add, remove or reorder tasks. Install tasks contain applications to be installed. Manual tasks contain instructions for the user. Install tasks and manual tasks can be grouped into task groups.

- Task Group: Mid-West Trucking RFID Solution for Windows.
  - Install Task: Mid-West Trucking RFID Solution for Windows
    - Application: IBM WebSphere Application Server - Community Edition for W
    - Application: Midwest Trucking RFID
- Task Group: Mid-West Trucking RFID Solution for Linux
  - Install Task: Mid-West Trucking RFID Solution for Linux
    - Application: IBM WebSphere Application Server - Community Edition for Li
    - Application: Midwest Trucking RFID

Overridden application variables

Override the appearance or behavior of variables in the selected application.

Task Configuration

Task group selection prompt: %taskGroupSelectionPrompt

☑ Allow only one task group or task to be selected

Property	Value
addLocalHost	true
description	%windowsTaskDescription
id	windows
isOptional	true
oneTargetDeployment	
selectedByDefault	true
taskDetails	%windowsTaskDetails

No consoles to display at this time.

Figure 48. Adding descriptive text to the Mid-West Trucking RFID Solution for Windows install task

5. To add details about the Mid-West Trucking RFID Solution for Linux install task, complete the following steps:
   a. Select **Install Task: Mid-West Trucking RFID Solution for Linux** in the Solution Tasks section of the Tasks page.
   b. Click the **Properties** tab at the bottom of the page.
   c. Type `linux` in the **id** field.
   d. Type `%linuxTaskDetails` in the **taskDetails** field.
6. Save the project.

For more details about configuring tasks, see “Providing task configuration information” on page 114.

Adding a startup checkpoint program to the solution

You can configure a startup checkpoint program that runs before the solution is loaded in the deployment wizard.

When you imported additional solution resources into the workspace (see “Importing additional solution resources” on page 46), a startup checkpoint program was added to your project. A startup checkpoint program is a custom program you can develop to enable the modification of values in your solution depending on the environment of the staging server. In this example, the startup checkpoint program detects and displays the operating system on the staging server to the user during deployment. It accomplishes this by modifying the value of a shared variable in the MidwestTrucking application.

To activate the startup checkpoint program, complete the following steps:

1. From the **General** page of the solution project editor, click **Advanced Solution Configuration**.
2. Click the **Browse** button next to the **Startup checkpoint** field.
3. Select the **RFIDStartup - com.midwest.rfid** class and click **OK**.
4. Save the solution project.

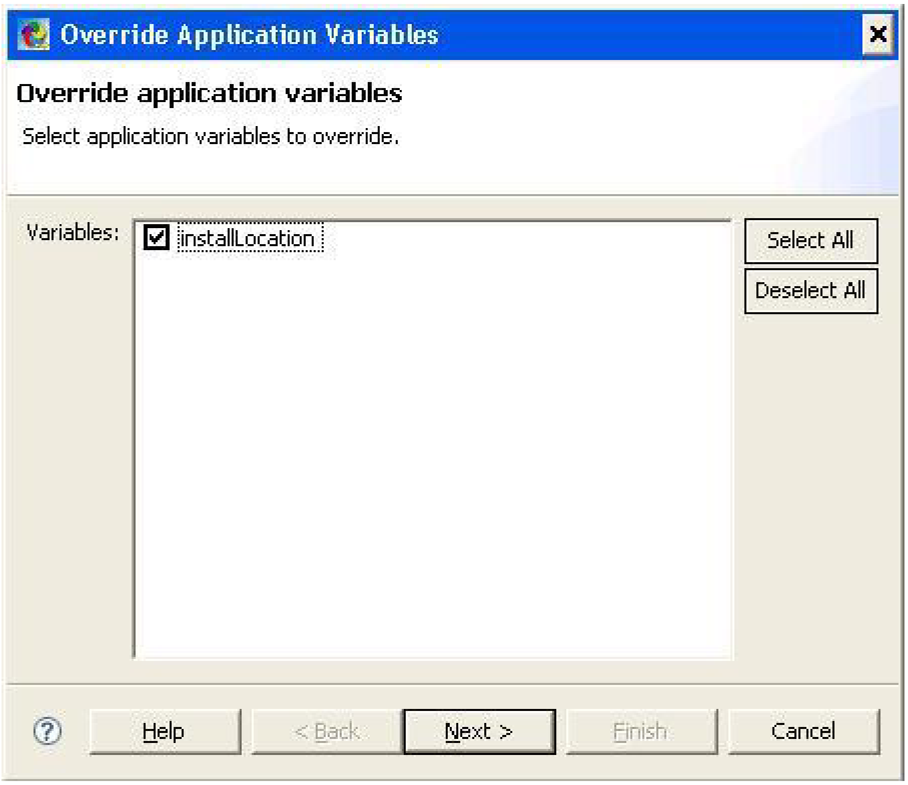
Sharing variables between applications

You can share information between applications and modify the appearance of variables in the deployment wizard. To do this, you share variables between application projects.

In this tutorial, we will share the installation location of WebSphere Application Server Community Edition with the MidwestTrucking application. By accessing these variables using the support framework APIs, the MidwestTrucking application predeployment checker can determine the installation location entered by the user regardless of whether the user selected the Linux task or the Windows task. In addition, the variables defined in the MidwestTrucking application are hidden from the user because the information is for programming purposes only.

To share variables between the applications, complete the following steps:

1. From the **Tasks** page of the solution project editor, expand the Mid-West Trucking RFID Solution for Windows task group and install task.
2. Select the **IBM WebSphere Application Server - Community Edition for Windows** application.
3. In the Overridden application variables section of the page, click **Add**. The Override Application Variables wizard is displayed.
4. Select the **installLocation** variable and click **Next**.



Override Application Variables

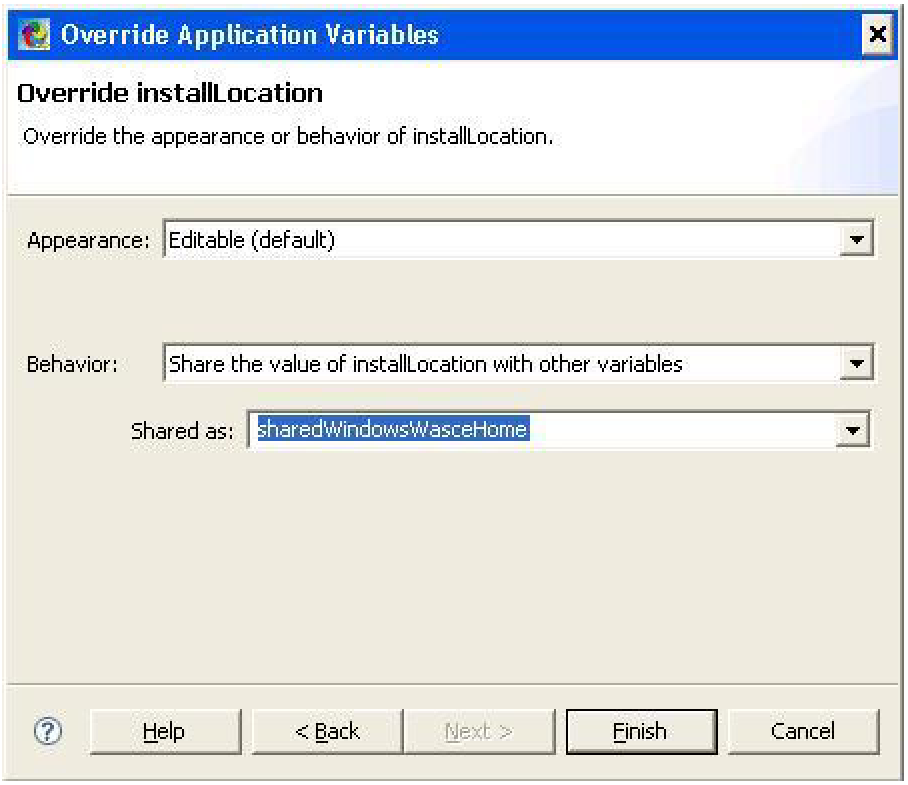
Override application variables

Select application variables to override.

Variables: ☑ installLocation

Figure 49. Override Application Variables wizard: Select application variables to override page

5. On the next page of the wizard, leave the **Appearance** field as is.
6. In the **Behavior** field, select **Share the value of installLocation with other variables**.
7. In the **Shared as** field, type `sharedWindowsWasceHome`.



Override installLocation

Override the appearance or behavior of installLocation.

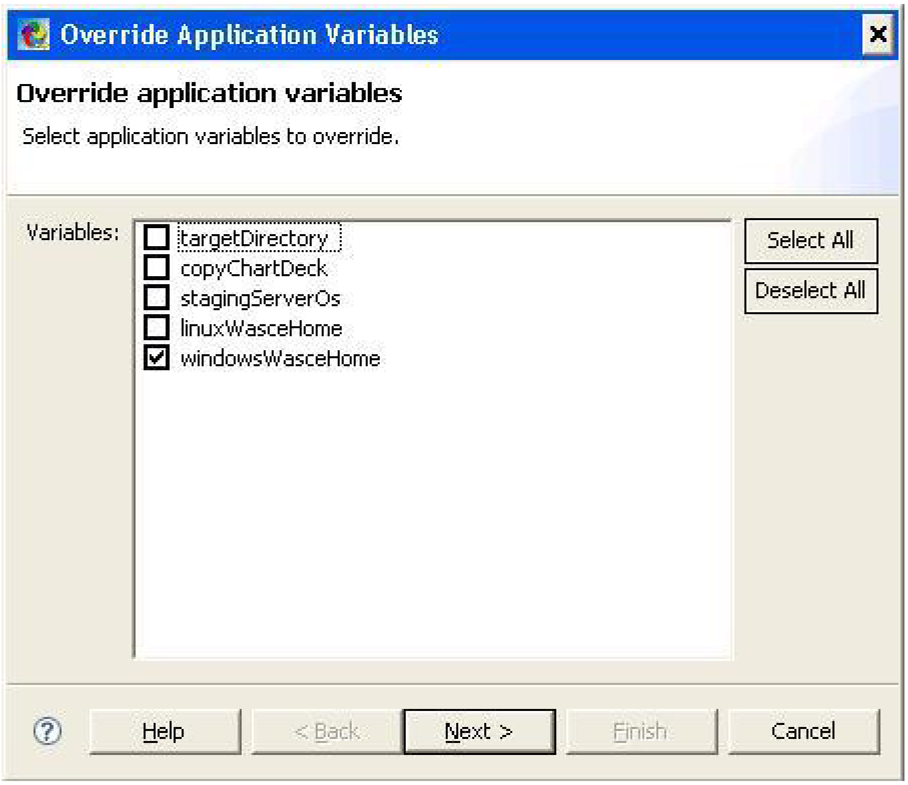
Appearance: Editable (default)

Behavior: Share the value of installLocation with other variables

Shared as: sharedWindowsWasceHome

Figure 50. Override Application Variables wizard: Override the appearance of behavior page

8. Click **Finish**. You are returned to the Tasks page of the solution project editor.
9. Select the **Midwest Trucking RFID** application from the Windows install task.
10. In the Overridden application variables section of the page, click **Add**. The Override Application Variables wizard is displayed.
11. Select the **windowsWasceHome** variable and click **Next**.



Override Application Variables

Override application variables

Select application variables to override.

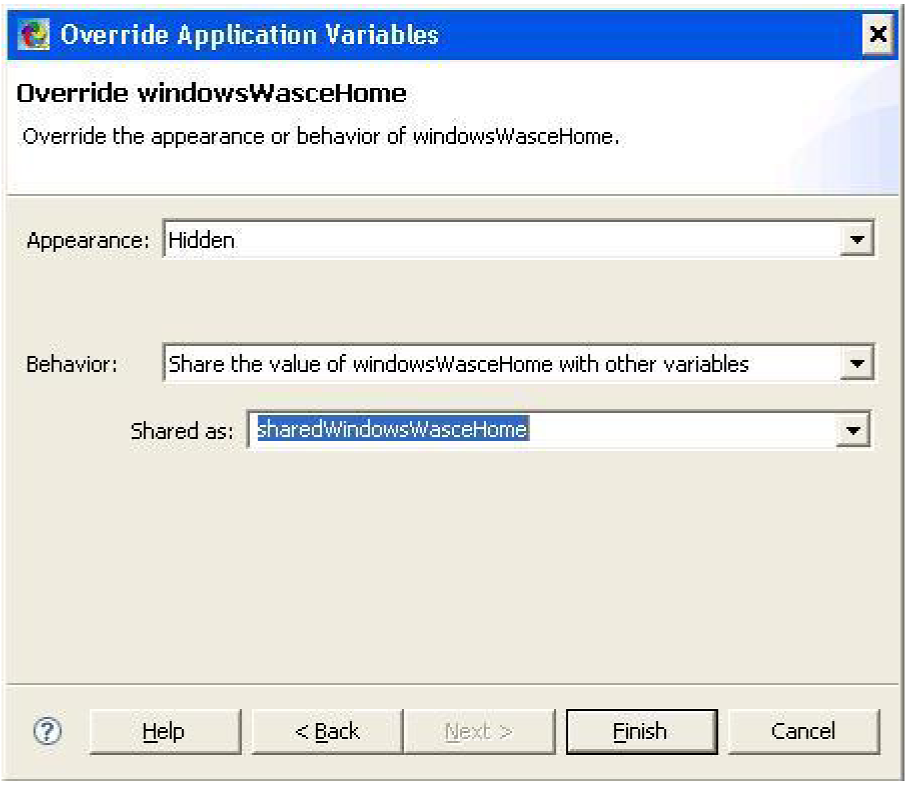
Variables:

- □ targetDirectory
- □ copyChartDeck
- □ stagingServerOs
- □ linuxWasceHome
- ☑ windowsWasceHome

Figure 51. Override Application Variables wizard: Select application variables to override page

The Override windowsWasceHome page of the wizard opens.

12. In the **Appearance** field, select **Hidden**.
13. In the **Behavior** field, select **Share the value of windowsWasceHome with other variables**.
14. In the **Shared as** field, select **sharedWindowsWasceHome**.



Override Application Variables

Override windowsWasceHome

Override the appearance or behavior of windowsWasceHome.

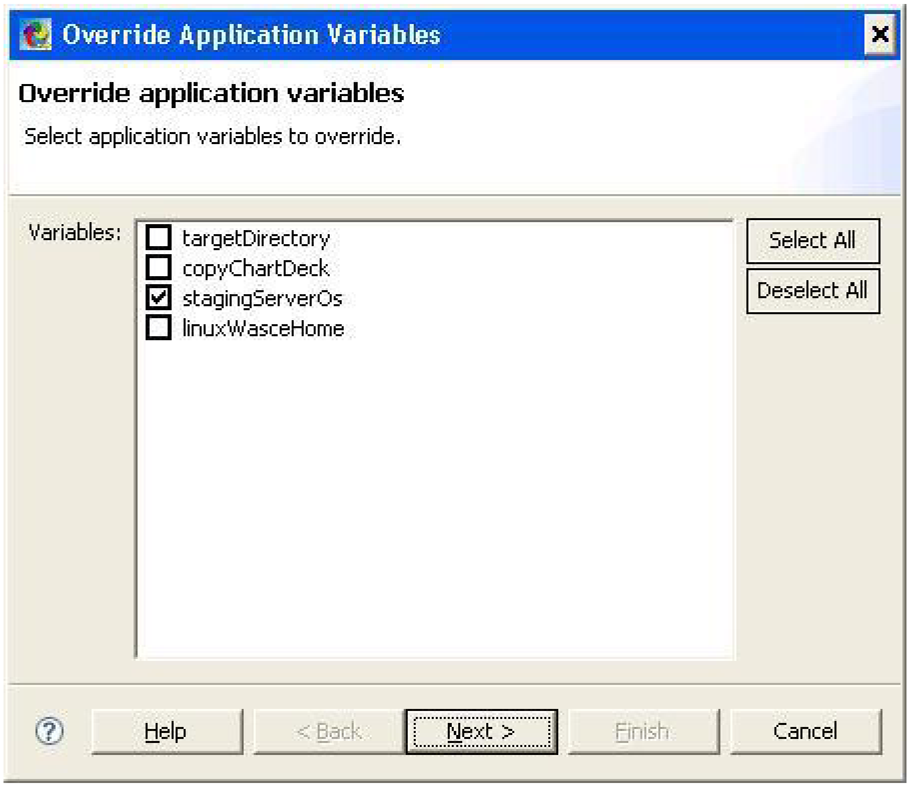
Appearance: Hidden

Behavior: Share the value of windowsWasceHome with other variables

Shared as: sharedWindowsWasceHome

Figure 52. Override Application Variables wizard: Override windowsWasceHome page

15. Click **Finish**. You are returned to the Tasks page of the solution project editor.
16. Ensure that the Midwest Trucking RFID application is still selected.
17. In the Overridden application variables section of the page, click **Add**. The Override Application Variables wizard is displayed.
18. Select the **stagingServerOS** variable and click **Next**.



Override Application Variables

Override application variables

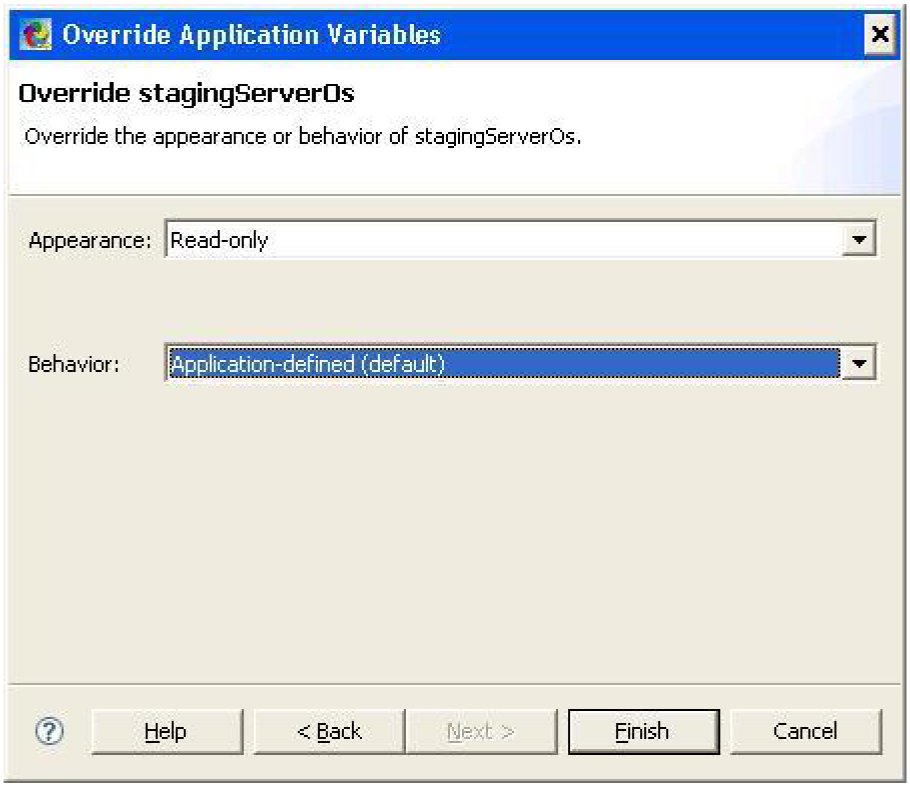
Select application variables to override.

Variables:

□ targetDirectory
□ copyChartDeck
☑ stagingServerOs
□ linuxWasceHome

Figure 53. Override Application Variables wizard: Select application variables to override page

19. On the next page of the wizard, in the **Appearance** field, select **Read-only**.
20. In the **Behavior** field, select **Application-defined (default)**.



Override Application Variables

Override stagingServerOs

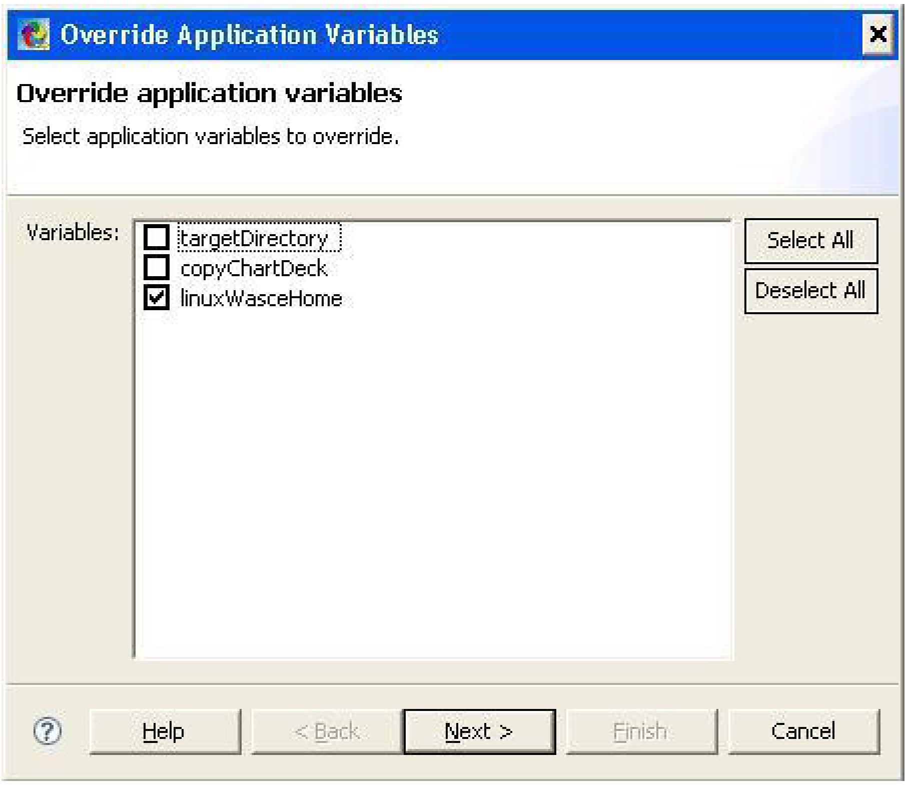
Override the appearance or behavior of stagingServerOs.

Appearance: Read-only

Behavior: Application-defined (default)

Figure 54. Override Application Variables wizard: Override the appearance or behavior page

21. Click **Finish**. You are returned to the Tasks page of the solution project editor.
22. In the Overridden application variables section of the page, click **Add**. The Override Application Variables wizard is displayed.
23. Select the **linuxWasceHome** variable and click **Next**.



Override application variables

Select application variables to override.

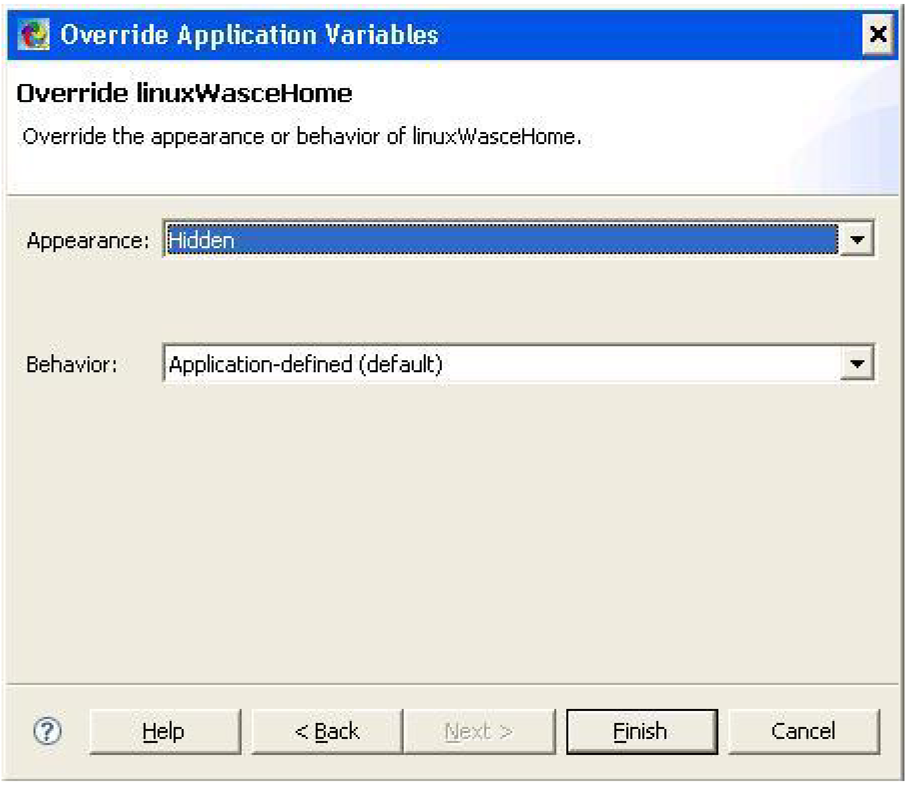
Variables:

□ targetDirectory
□ copyChartDeck
☑ linuxWasceHome

Figure 55. Override Application Variables wizard: Select application variables to override page

The Override linuxWasceHome page of the wizard opens.

24. In the **Appearance** field, select **Hidden**.
25. In the **Behavior** field, select **Application-defined (default)**.



Override Application Variables

Override linuxWasceHome

Override the appearance or behavior of linuxWasceHome.

Appearance: Hidden

Behavior: Application-defined (default)

Figure 56. Override Application Variables wizard: Override the appearance of behavior page

26. Click **Finish**. You are returned to the Tasks page of the solution project editor. The variable sharing for the Windows task group is complete.
27. From the **Tasks** page of the solution project editor, expand the Mid-West Trucking RFID Solution for Linux task group and install task.
28. Select the **IBM WebSphere Application Server - Community Edition for Linux** application.
29. Repeat steps 3 on page 57 through 26 for the Linux variables, selecting the Linux install tasks and making the following modifications:

    Note:

    a. In step 7 on page 58, type `sharedLinuxWasceHome` instead of `sharedWindowsWasceHome`.
    b. In step 11 on page 59, select the **linuxWasceHome** variable instead of the **windowsWasceHome** variable.
    c. In step 14 on page 60, select **sharedLinuxWasceHome** for the Shared as name.
    d. In step 23 on page 63, select the **windowsWasceHome** variable instead of the **linuxWasceHome** variable.
30. Save the solution project.

For more information, see "Modifying the default value of a variable" on page 106, "Changing the appearance of a variable" on page 106, "Sharing variables between applications" on page 107 and "Sharing variables between applications and install tasks" on page 108.

Exporting the sample solution to a solution launcher image

The Express Runtime developer provides an export wizard that you can use to export a solution project to one or more CDs or DVDs for distribution. The CDs or DVDs contain a complete image of the solution that you have created, and when end users insert the media, the deployment wizard is displayed and enables them to deploy the solution. When the solution deployment is complete, the deployment wizard is uninstalled. Media that contains the combination of solutions and the deployment wizard are referred to as solution launcher images.

To export the solution you just created to CDS or DVDs for distribution, complete the following steps:

1. Right-click in the project editor and click **Generate Solution**.
2. Click **OK**.
3. Right-click the solution file in the Express Runtime Explorer section of the editor and select **Export**.

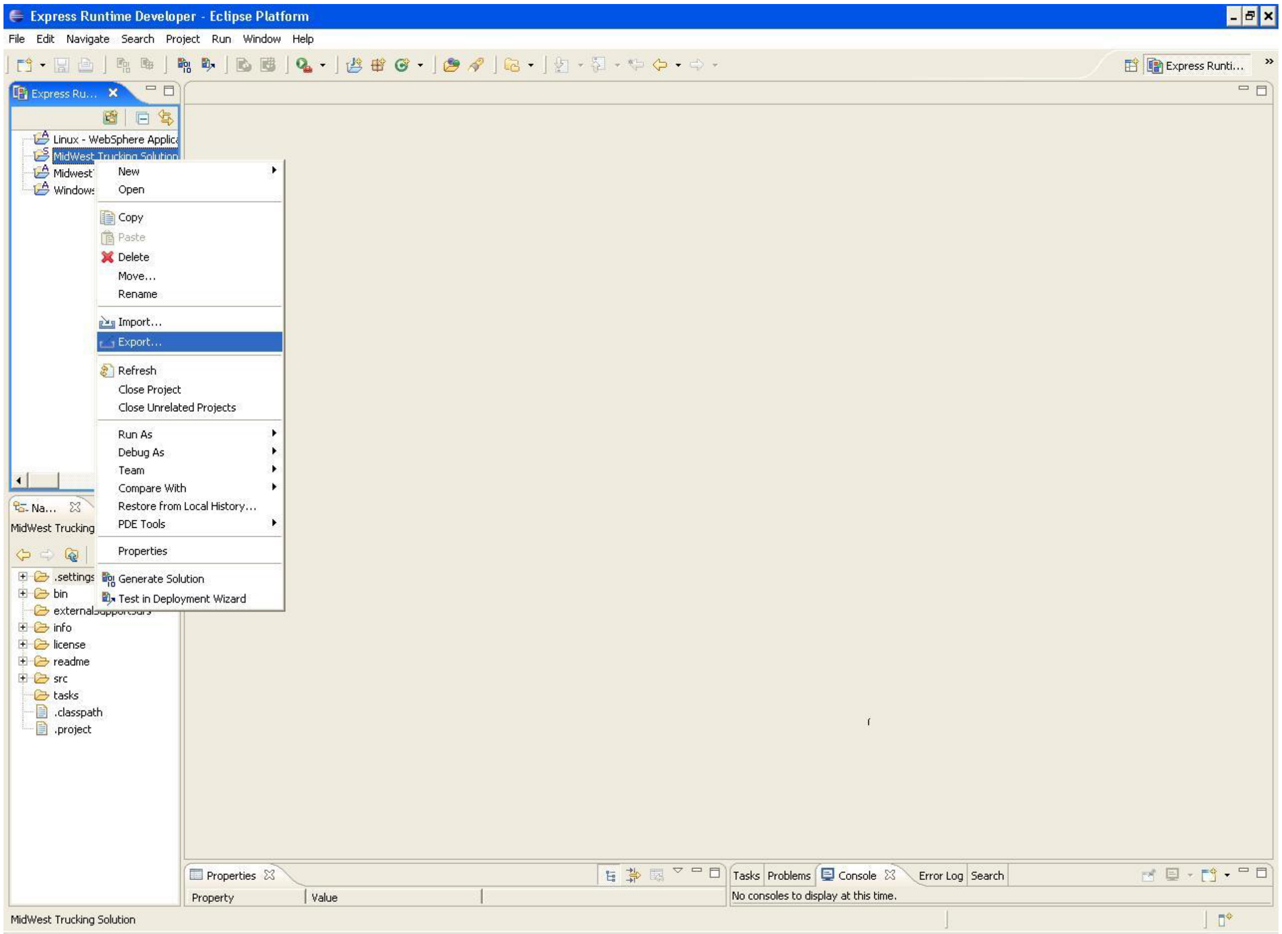
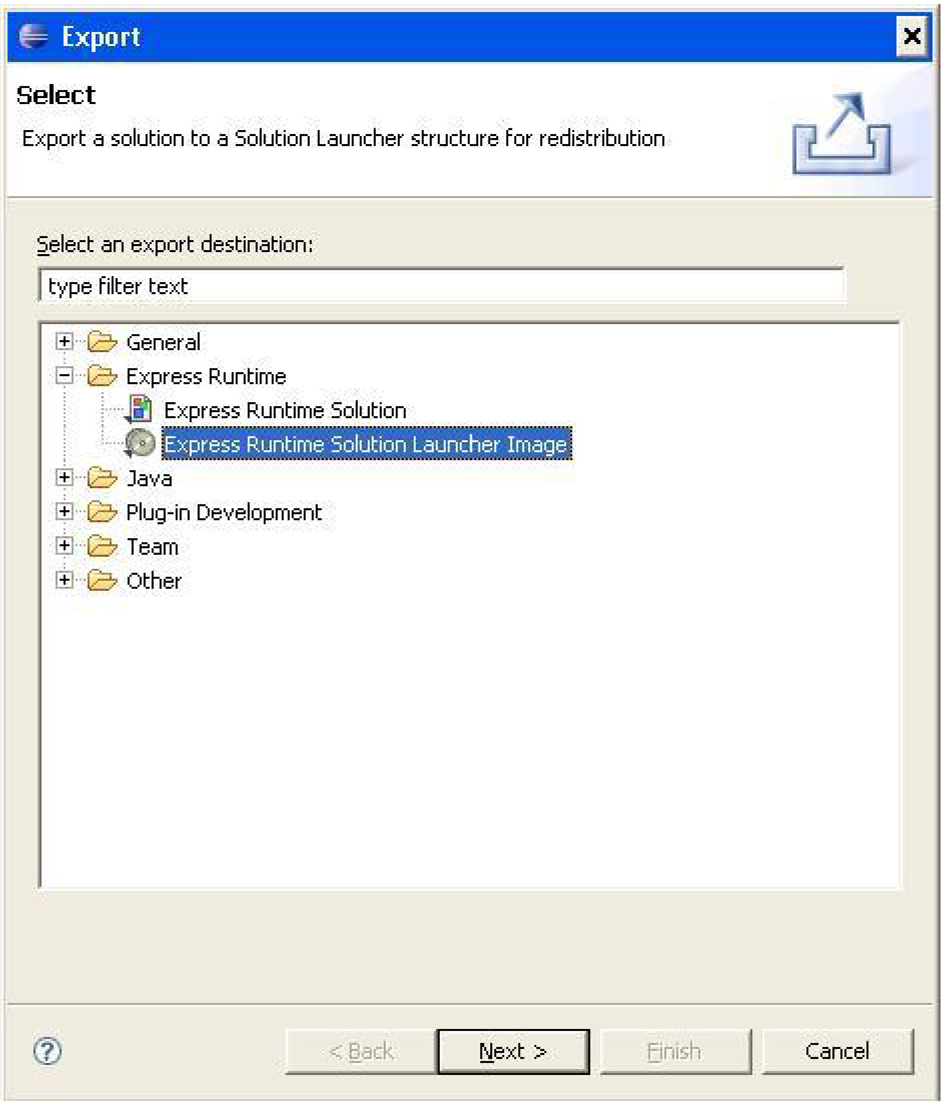


Figure 57. Exporting the Midwest Trucking Solution to a solution launcher image

The export wizard opens.

4. Click **Express Runtime** → **Express Runtime Solution Launcher Image**. See Figure 58 on page 67.



Export

Select

Export a solution to a Solution Launcher structure for redistribution

Select an export destination:

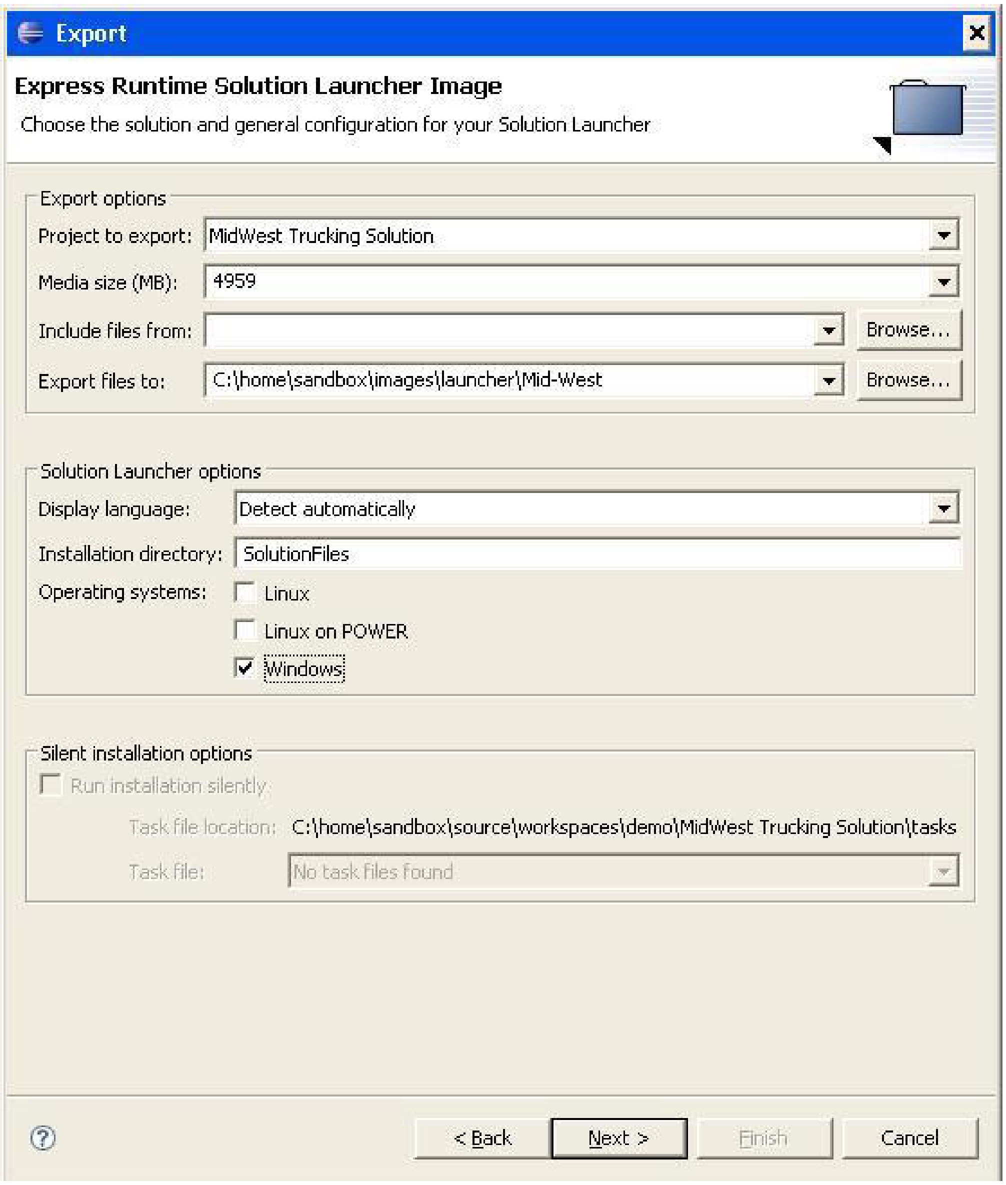
type filter text

- General
- Express Runtime
  - Express Runtime Solution
  - Express Runtime Solution Launcher Image
- Java
- Plug-in Development
- Team
- Other

Figure 58. Export wizard; Express Runtime Solution Launcher Image selected

5. Click **Next**. The Express Runtime Solution Launcher Image export options window opens.
6. If you right-clicked the MidWest Trucking Solution in step 3 on page 66, it is selected in the **Project to export** field. If it is not already selected, select it from the list. Make the following entries or selections in the remainder of the fields on this page, as shown in Figure 59 on page 69:
   - **Media size (MB)**: Use this field to define the capacity per disk, in MB, of the media that the solution launcher image will be distributed on. Select **4959**.
   - **Include files from**: Leave this field blank. You would use this field if you wanted to include external files in the image. For example, you might want to include documentation, images, or other files that accompany the image. In that case, you would enter the path, or click **Browse** to select the root directory of the files to include.
   - **Export files to**: Type the path to which you want to export the solution launcher files, or click **Browse** to select the destination directory.

- **Display language**: Use this field to select the language in which you want the deployment wizard panels to be displayed. Select **Detect automatically**. This allows the installation to detect and use the language for the default locale on the computer running the solution launcher.
- **Installation directory**: Use this field to specify the location of the solution launcher logs and solution files that contain the configuration information for the solution. The installation uses the directory name SolutionFiles by default. Keep the default value.
- **Operating systems**: Use the check boxes to select the operating systems on which the solution launcher will run. For the purposes of this tutorial, select **Windows**.
- **Run installation silently**: If you wanted to install the solution launcher image silently, you would select this check box and provide the location of a task file in the **Task file location** field and the name of the file in the Task file field. For the purposes of the tutorial, ignore this section of the window. For more information about creating a task file and installing a solution launcher image silently, see "Silent installation of a solution launcher image" on page 121.



Express Runtime Solution Launcher Image

Choose the solution and general configuration for your Solution Launcher

Export options

Project to export: MidWest Trucking Solution

Media size (MB): 4959

Include files from:

Export files to: C:\home\sandbox\images\launcher\Mid-West

Solution Launcher options

Display language: Detect automatically

Installation directory: SolutionFiles

Operating systems: □ Linux □ Linux on POWER ☑ Windows

Silent installation options

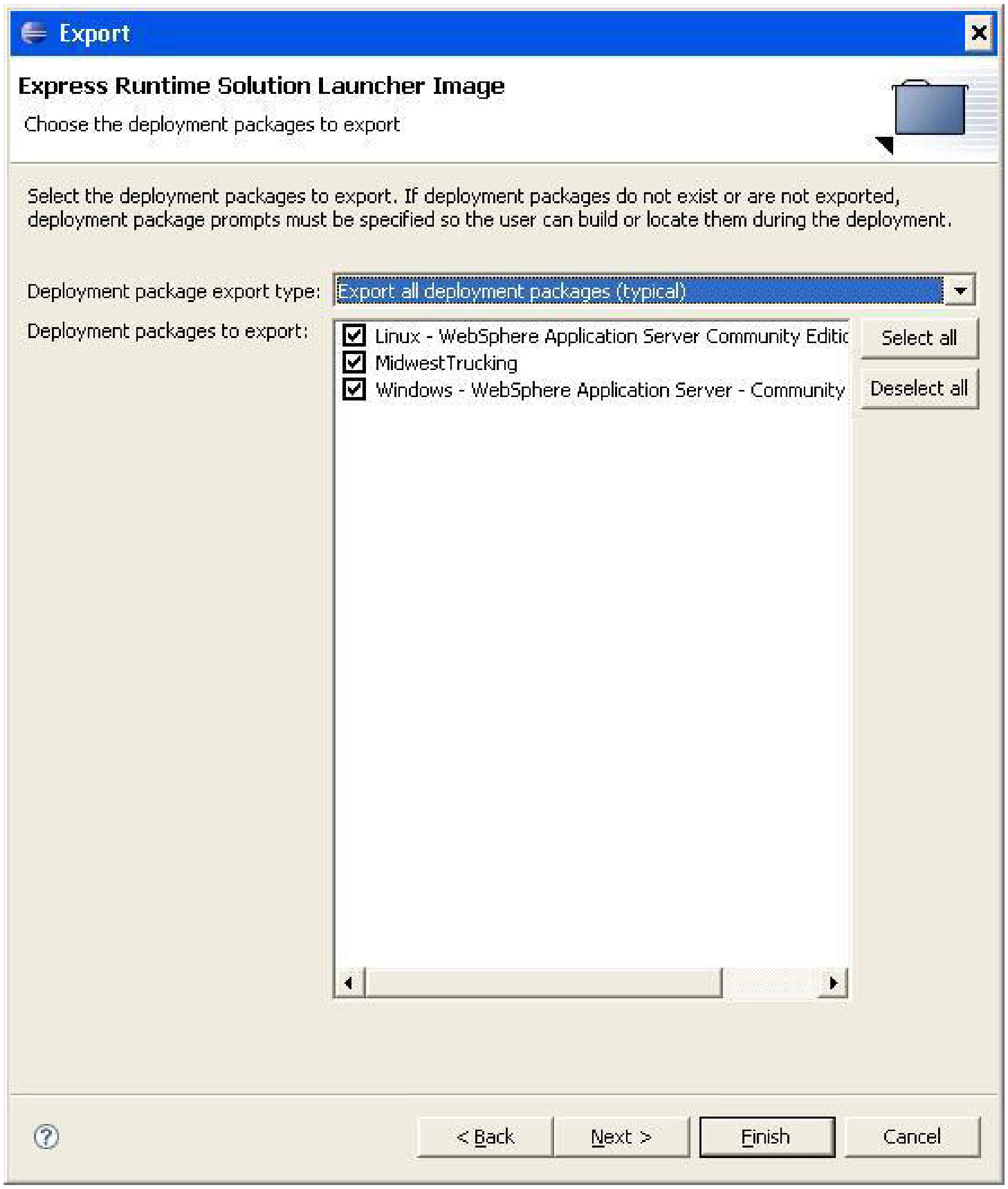
□ Run installation silently

Task file location: C:\home\sandbox\source\workspaces\demo\MidWest Trucking Solution\tasks

Task file: No task files found

Figure 59. Export Express Runtime Solution Launcher Image wizard export options window

7. Click **Next**.
8. In the **Deployment package export type** field, select **Export all deployment packages (typical)**. This option exports all of the deployment packages for all of the applications that are included as a part of the solution. All deployment packages must be generated to use this option. See Figure 60 on page 70.



Export

Express Runtime Solution Launcher Image

Choose the deployment packages to export

Select the deployment packages to export. If deployment packages do not exist or are not exported, deployment package prompts must be specified so the user can build or locate them during the deployment.

Deployment package export type: Export all deployment packages (typical)

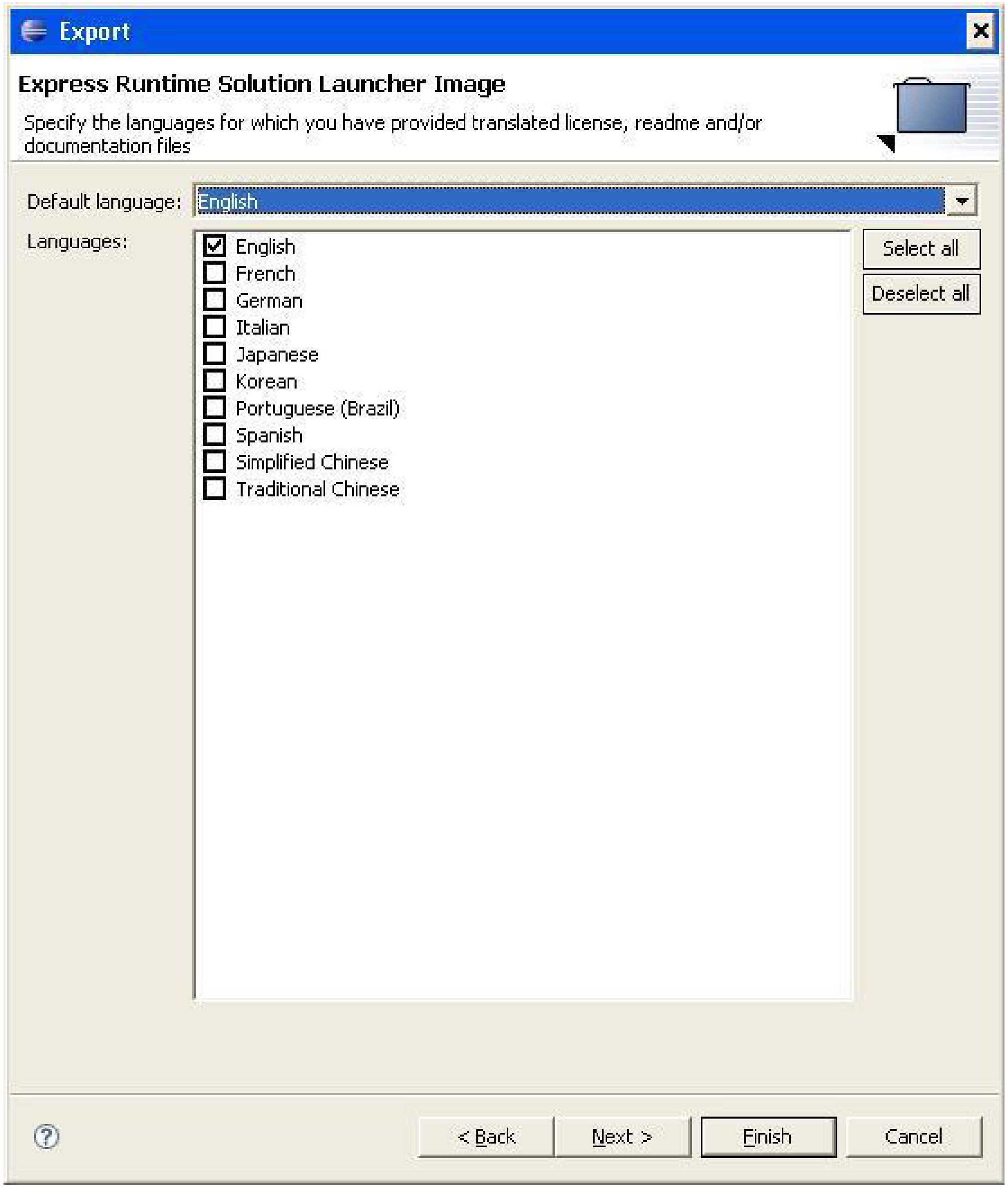
Deployment packages to export:

☑ Linux - WebSphere Application Server Community Editic
☑ MidwestTrucking
☑ Windows - WebSphere Application Server - Community

Figure 60. Export Express Runtime Solution Launcher Image wizard deployment packages selection window

9. Click **Next**.
10. In the **Default language** field, select English. This field sets the default language for the solution on the target computer.
11. In the **Languages** list, select English. This list is used to select the languages for which you provide translated license, readme, or documentation files so that if the installation detects one of those languages as the locale of the target computer, the appropriate translated documentation will be displayed. If the locale is not one of the specified languages, or if you did not provide the translated file, the file will be displayed in the specified default language.

    Note: The default language must also be selected in the **Languages** list, as shown in Figure 61 on page 71.


Export

Express Runtime Solution Launcher Image

Specify the languages for which you have provided translated license, readme and/or documentation files

Default language: English

Languages:

☑ English
☐ French
☐ German
☐ Italian
☐ Japanese
☐ Korean
☐ Portuguese (Brazil)
☐ Spanish
☐ Simplified Chinese
☐ Traditional Chinese

Figure 61. Export Express Runtime Solution Launcher Image wizard specify languages window

12. Click **Finish**. A window stating that the export was successful opens.
13. The Solution Export Successful window also gives you the option to save the current export settings for use later with a command-line export. For the purposes of this tutorial, click **No**.

For more information about exporting solutions, see Appendix C, “Exporting,” on page 117.

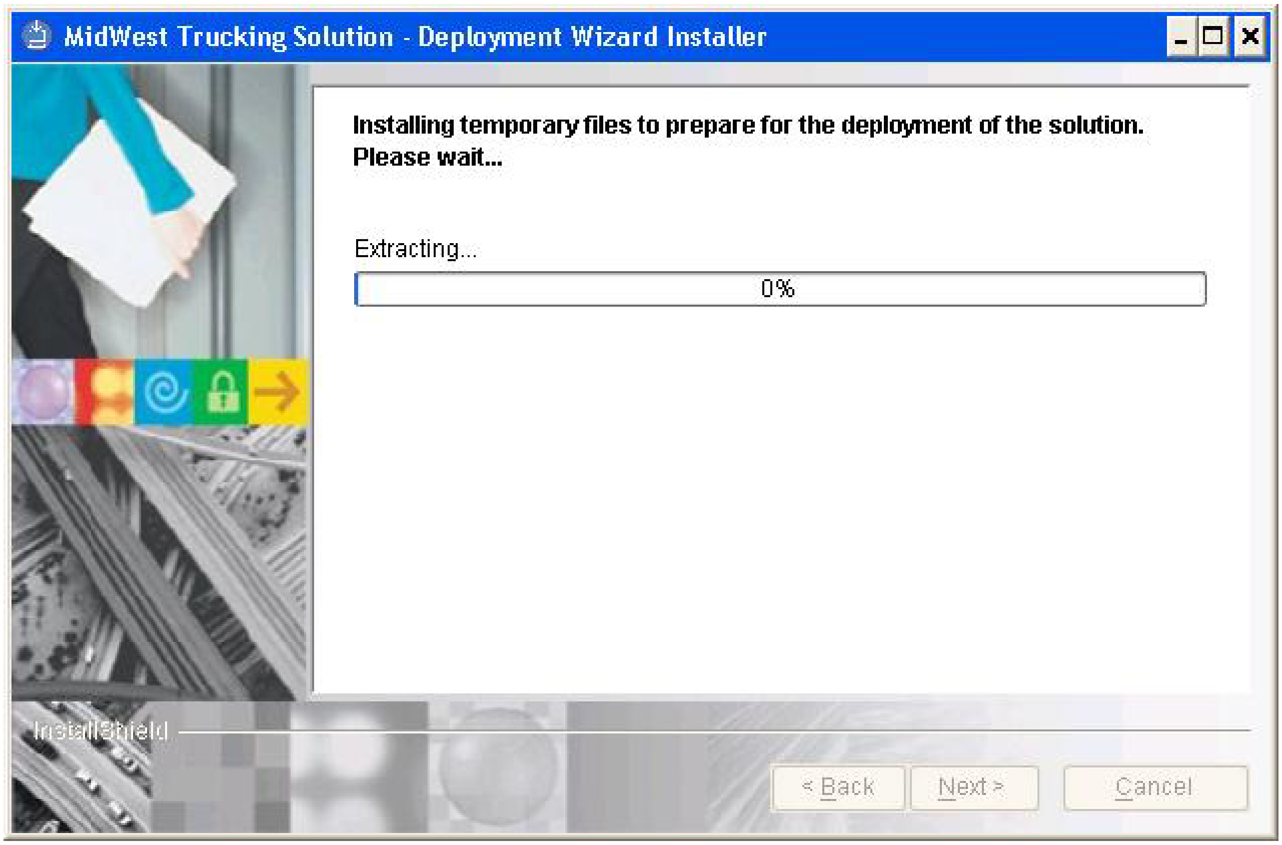
Deploying the sample solution

Now that you have created your solution, you can deploy it and see how the pieces all fit together.

Note: For the purposes of this tutorial, you will deploy the sample solution from the solution launcher image that you exported previously. Alternatively, you could deploy the solution from a staging server. To deploy the solution from a staging server, the deployment wizard and all of the files required by your solution must be installed on that server.

To deploy your solution, browse to the location of the WindowsSetup.exe file that you exported in "Exporting the sample solution to a solution launcher image" on page 66. The file will be in a folder called disk1. Complete the following steps:

1. Double-click the WindowsSetup.exe file to start deployment. The following window opens:



MidWest Trucking Solution - Deployment Wizard Installer

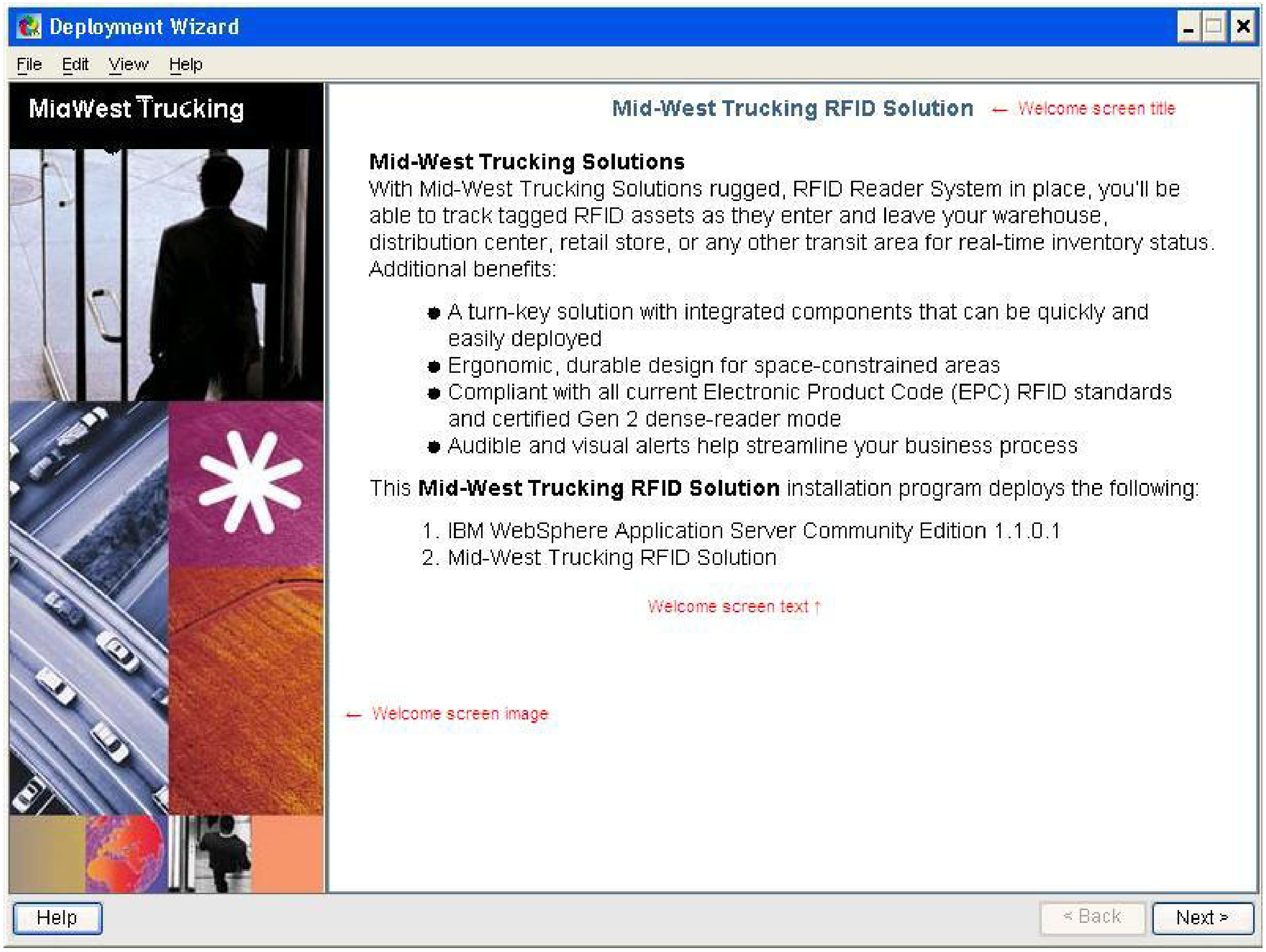
Installing temporary files to prepare for the deployment of the solution. Please wait...

Extracting...

0%

Figure 62. Deployment wizard: Installing temporary files window

2. After the temporary files have been extracted and installed, the Welcome page for the solution opens.



Mid-West Trucking Solutions

With Mid-West Trucking Solutions rugged, RFID Reader System in place, you'll be able to track tagged RFID assets as they enter and leave your warehouse, distribution center, retail store, or any other transit area for real-time inventory status. Additional benefits:

- A turn-key solution with integrated components that can be quickly and easily deployed
- Ergonomic, durable design for space-constrained areas
- Compliant with all current Electronic Product Code (EPC) RFID standards and certified Gen 2 dense-reader mode
- Audible and visual alerts help streamline your business process

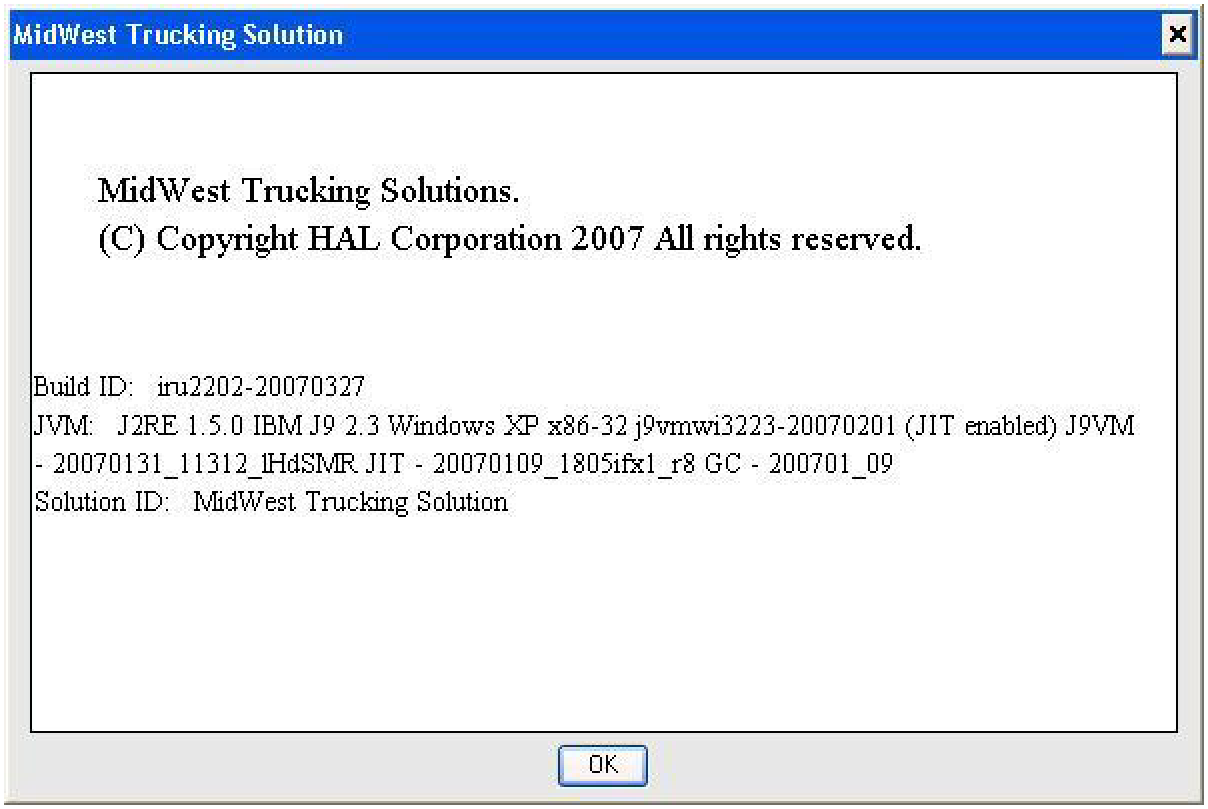
This **Mid-West Trucking RFID Solution** installation program deploys the following:

1. IBM WebSphere Application Server Community Edition 1.1.0.1
2. Mid-West Trucking RFID Solution

Figure 63. Deployment wizard: Mid-West Trucking RFID Solution Welcome page

The image has been modified to point out items that you configured when you created the solution. You configured the Welcome screen image in step 3 on page 48, the Welcome screen title in step 4 on page 48, and the Welcome screen text in step 5 on page 48.

3. Click **Help** → **About**. The About screen for the solution opens. You configured the text to appear in this window in step 6 on page 48.



MidWest Trucking Solutions.

(C) Copyright HAL Corporation 2007 All rights reserved.

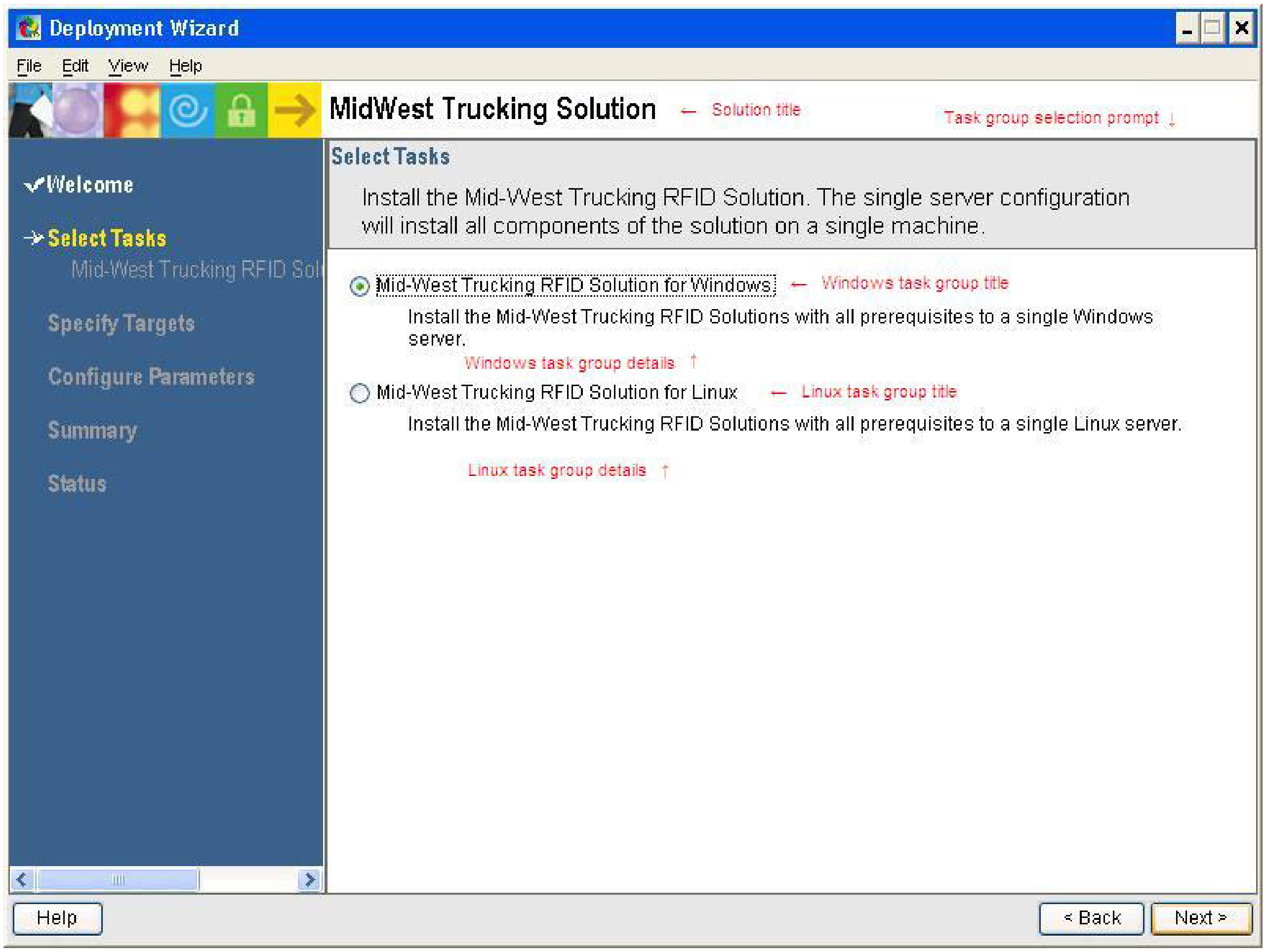
Build ID: iru2202-20070327

JVM: J2RE 1.5.0 IBM J9 2.3 Windows XP x86-32 j9vmwi3223-20070201 (JIT enabled) J9VM - 20070131_11312_lHdSMR JIT - 20070109_1805ifx1_r8 GC - 200701_09

Solution ID: MidWest Trucking Solution

Figure 64. About screen for the Mid-West Trucking RFID Solution for Windows

4. Click **OK** to return to the Welcome window.
5. Click **Next**. The Select Tasks window opens.



Install the Mid-West Trucking RFID Solution. The single server configuration will install all components of the solution on a single machine.

● Mid-West Trucking RFID Solution for Windows

Install the Mid-West Trucking RFID Solutions with all prerequisites to a single Windows server.

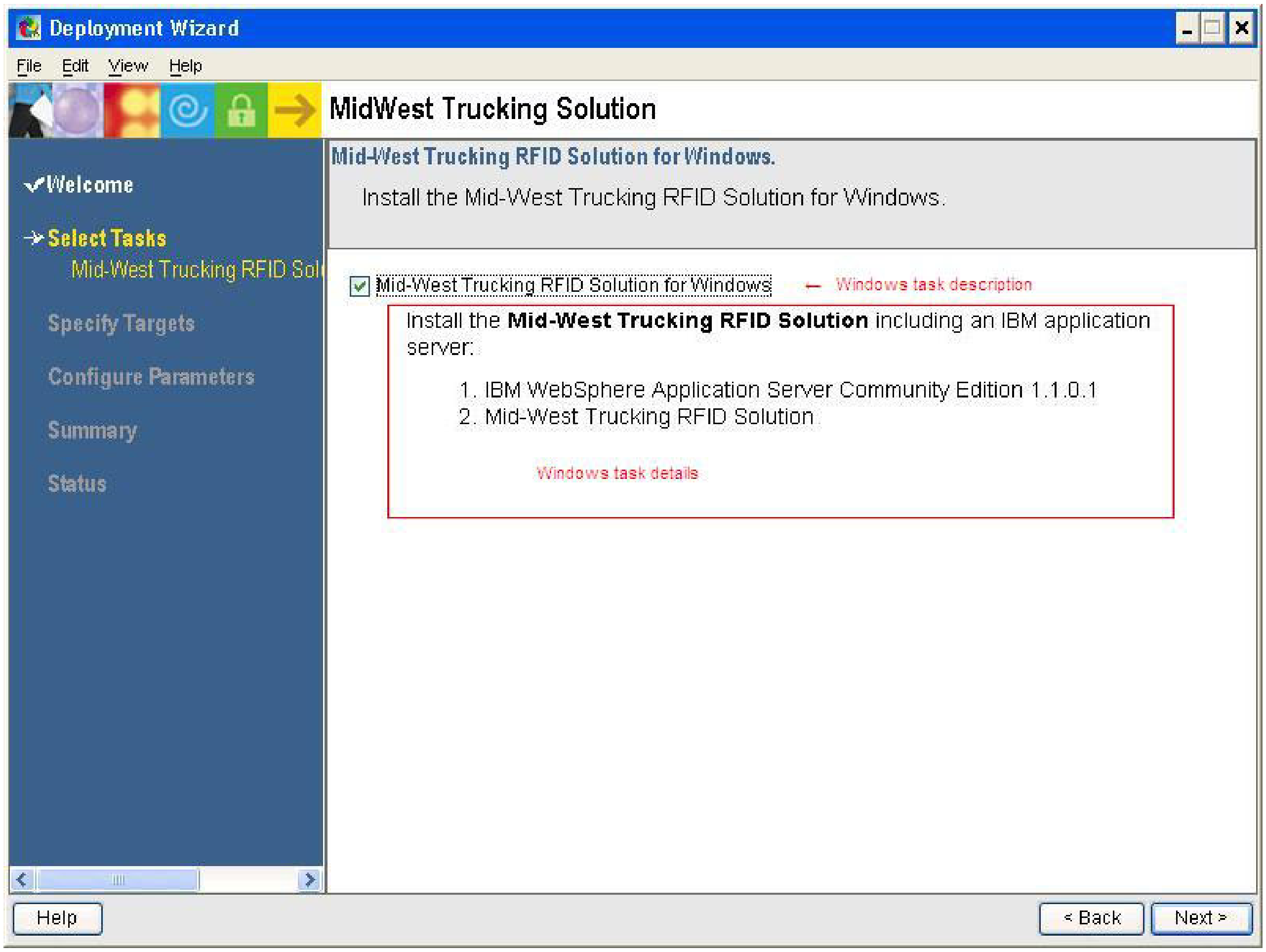
○ Mid-West Trucking RFID Solution for Linux

Install the Mid-West Trucking RFID Solutions with all prerequisites to a single Linux server.

Figure 65. Deployment wizard: MidWest Trucking Solution Select Tasks window

The image has been modified to point out items that you configured when you created the solution. You configured the Task group selection prompt in step 1 on page 53. You configured the Windows task group title and Windows task group details in step 2 on page 54. You configured the Linux task group title and Linux task group details in step 3 on page 55.

6. Select **Mid-West Trucking RFID Solution for Windows** and click **Next**. The Mid-West Trucking RFID Solution for Windows task window opens.



Mid-West Trucking RFID Solution for Windows.

Install the Mid-West Trucking RFID Solution for Windows.

☑ Mid-West Trucking RFID Solution for Windows ← Windows task description

Install the **Mid-West Trucking RFID Solution** including an IBM application server:

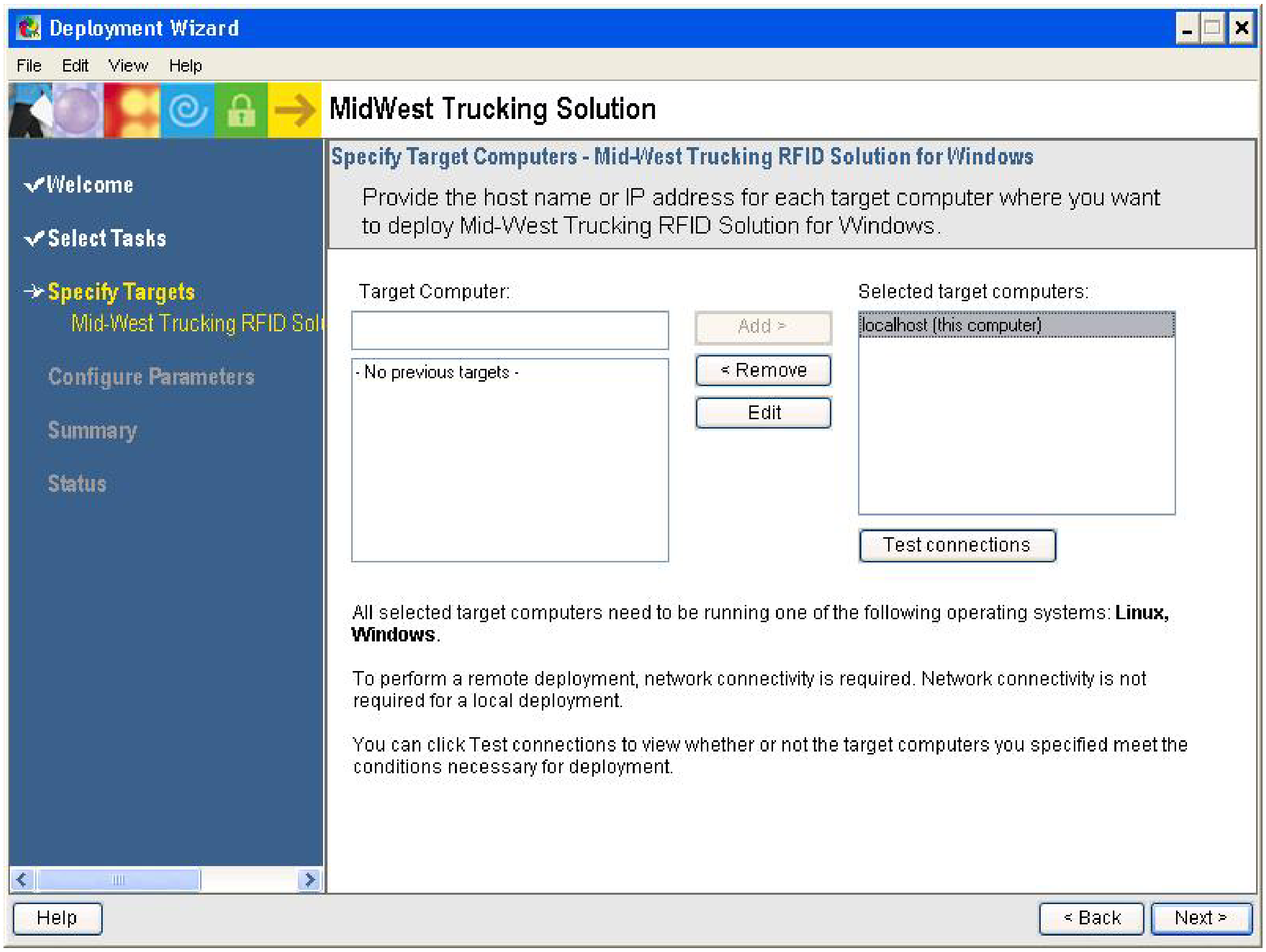
1. IBM WebSphere Application Server Community Edition 1.1.0.1
2. Mid-West Trucking RFID Solution

Windows task details

Figure 66. Deployment wizard: Mid-West Trucking RFID Solution for Windows task window

The image has been modified to point out items that you configured when you created the solution. You configured the Windows install task description and details in step 4 on page 55.

7. Select the **Mid-West Trucking RFID Solution for Windows** check box and click **Next**. The Specify Targets window opens.



MidWest Trucking Solution

Specify Target Computers - Mid-West Trucking RFID Solution for Windows

Provide the host name or IP address for each target computer where you want to deploy Mid-West Trucking RFID Solution for Windows.

Target Computer:

- No previous targets -

Selected target computers:

localhost (this computer)

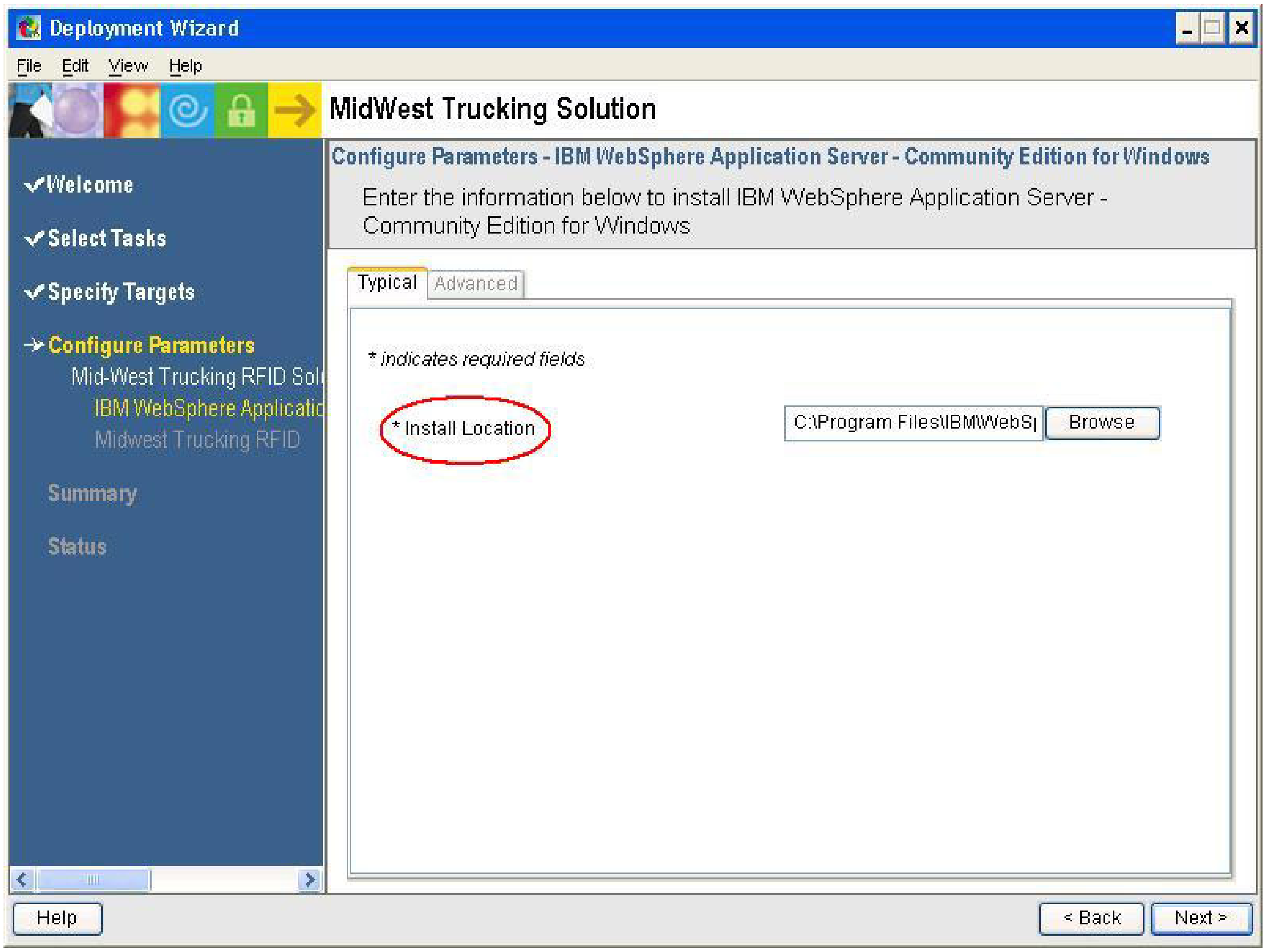
All selected target computers need to be running one of the following operating systems: **Linux, Windows**.

To perform a remote deployment, network connectivity is required. Network connectivity is not required for a local deployment.

You can click Test connections to view whether or not the target computers you specified meet the conditions necessary for deployment.

Figure 67. Specify targets to which to deploy the Mid-West Trucking RFID Solution for Windows

8. In the **Target Computer** field, type `localhost` and click **Add**. This setting deploys the solution to your local computer.
9. Click **Next**. The Configure Parameters window for IBM WebSphere Application Server Community Edition opens.



Configure Parameters - IBM WebSphere Application Server - Community Edition for Windows

Enter the information below to install IBM WebSphere Application Server - Community Edition for Windows

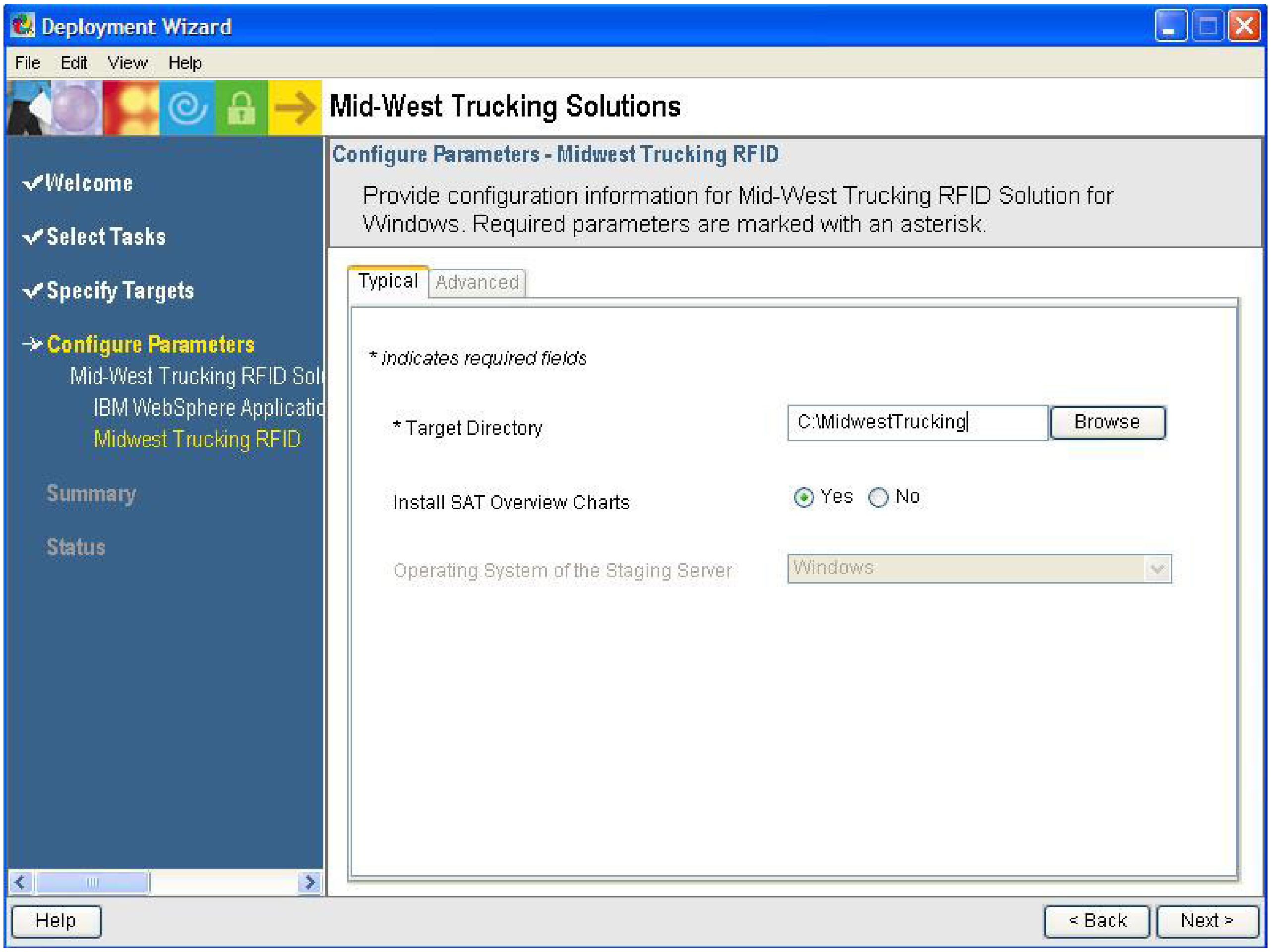
* indicates required fields

* Install Location: C:\Program Files\IBM\WebS|

Figure 68. Deployment wizard: Configure Parameters (IBM WebSphere Application Server Community Edition) window

Figure 68 shows the variable you configured in “Adding variables to the application wrapper” on page 26.

10. Accept the default installation location for IBM WebSphere Application Server Community Edition and click **Next**. The Configure Parameters window for the Midwest Trucking RFID application opens.



Mid-West Trucking Solutions

Configure Parameters - Midwest Trucking RFID

→ Configure Parameters
Mid-West Trucking RFID Sol
IBM WebSphere Applicatio
Midwest Trucking RFID

Provide configuration information for Mid-West Trucking RFID Solution for Windows. Required parameters are marked with an asterisk.

** indicates required fields*

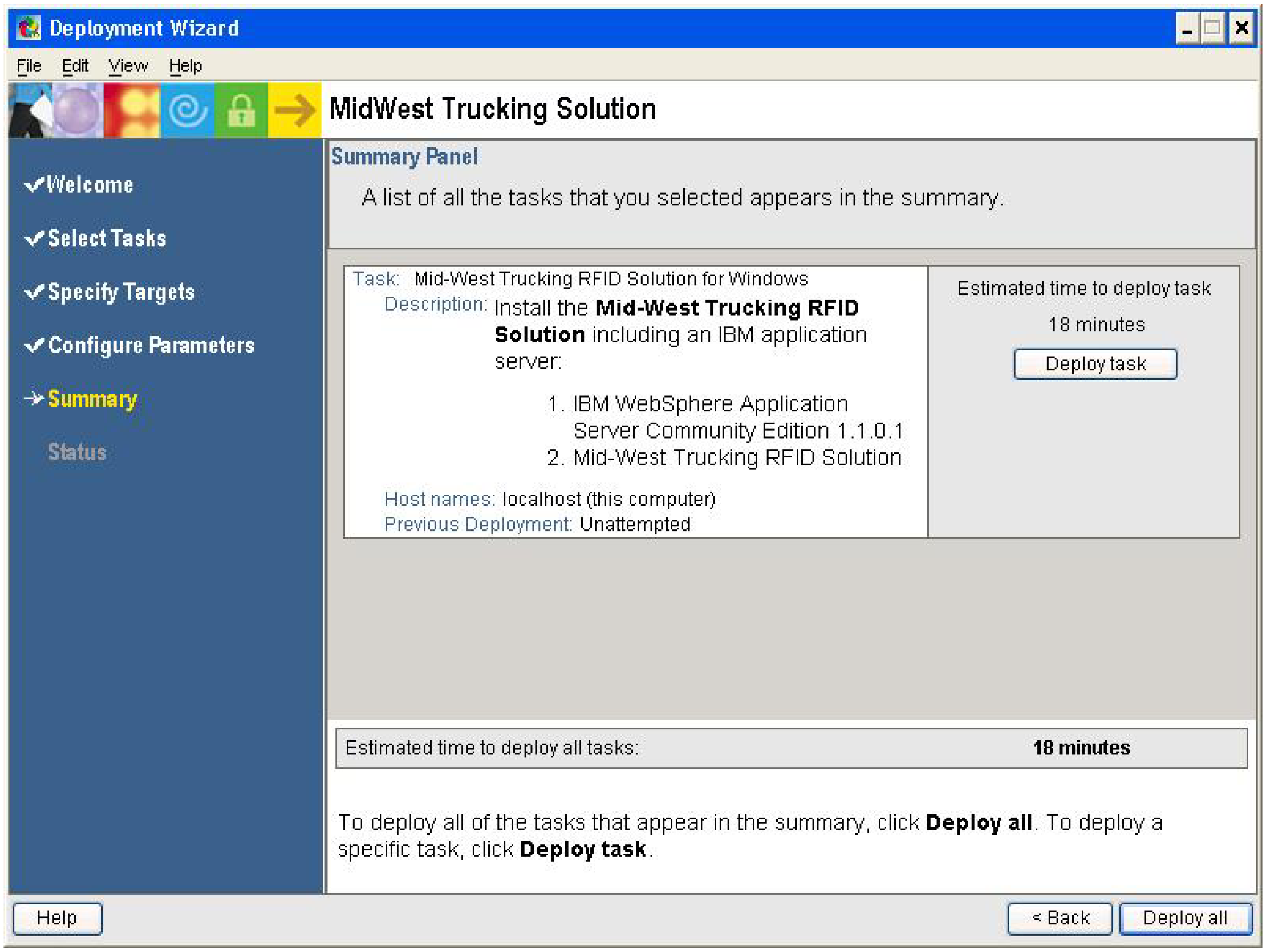
* Target Directory: C:\MidwestTrucking

Install SAT Overview Charts: ● Yes ○ No

Operating System of the Staging Server: Windows

Figure 69. Deployment wizard: Configure Parameters (Midwest Trucking RFID application) window

11. In the **Target Directory** field, type `C:\MidwestTrucking` and click **Next**. The Summary window opens.



Summary Panel

A list of all the tasks that you selected appears in the summary.

Task: Mid-West Trucking RFID Solution for Windows

Description: Install the **Mid-West Trucking RFID Solution** including an IBM application server:

1. IBM WebSphere Application Server Community Edition 1.1.0.1
2. Mid-West Trucking RFID Solution

Host names: localhost (this computer)

Previous Deployment: Unattempted

Estimated time to deploy task: 18 minutes

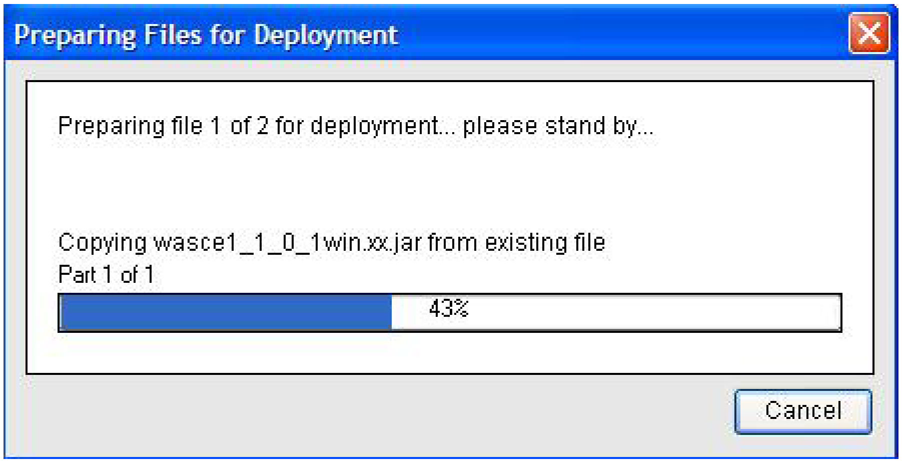
Estimated time to deploy all tasks: **18 minutes**

To deploy all of the tasks that appear in the summary, click **Deploy all**. To deploy a specific task, click **Deploy task**.

Figure 70. Deployment wizard: Summary window

The Summary window includes the tasks you have chosen to install, the target to which the tasks will be installed, and an estimated time to deploy the solution.

12. Click **Deploy all**. The progress bar opens while the necessary files for deployment are being prepared.



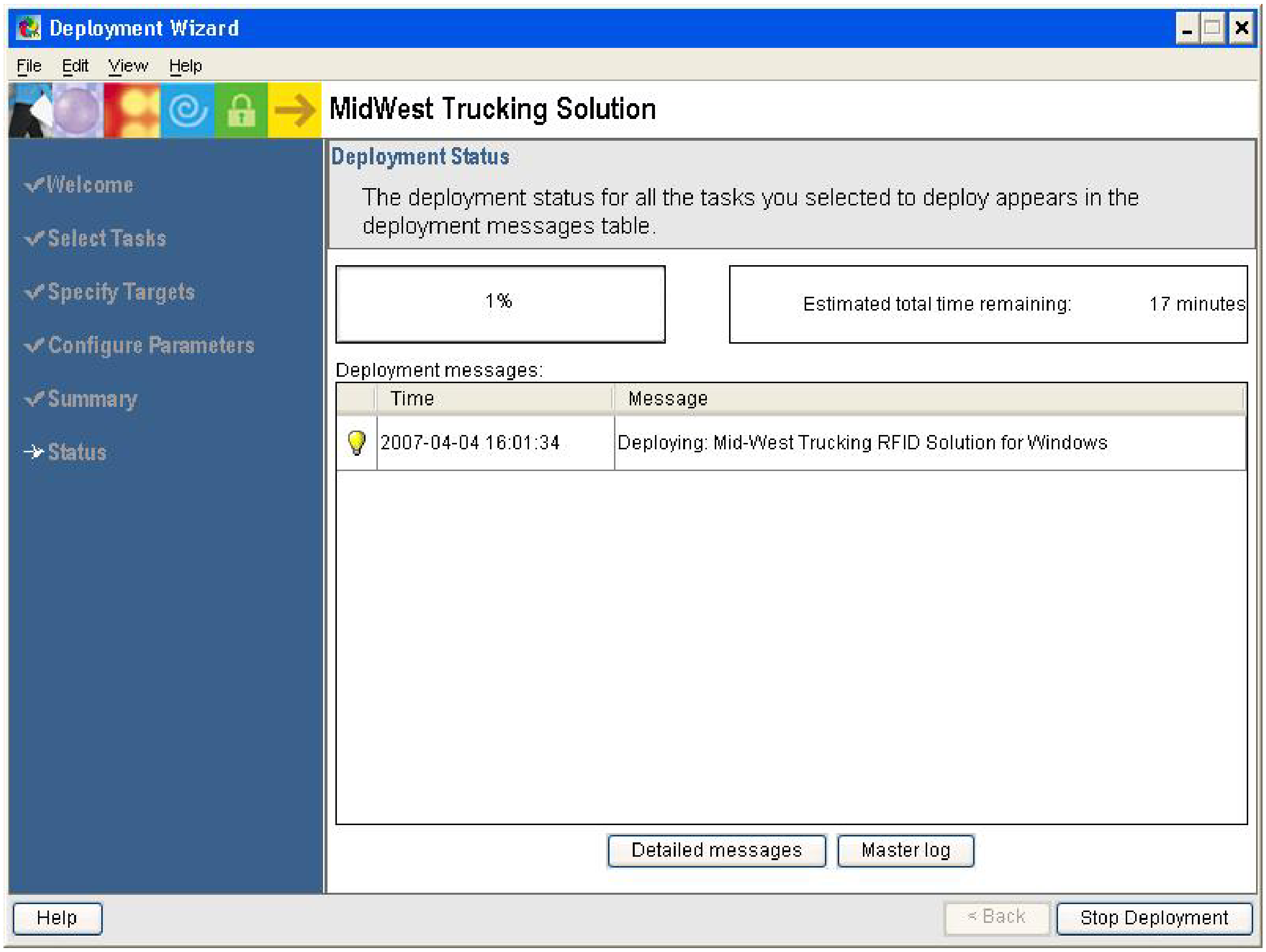
Preparing Files for Deployment

Preparing file 1 of 2 for deployment... please stand by...

Copying wasce1_1_0_1win.xx.jar from existing file
Part 1 of 1

Figure 71. Deployment wizard: Preparing Files for Deployment progress

13. After the files have been prepared, the Status window opens.



Deployment Wizard

MidWest Trucking Solution

Deployment Status

The deployment status for all the tasks you selected to deploy appears in the deployment messages table.

1%

Estimated total time remaining: 17 minutes

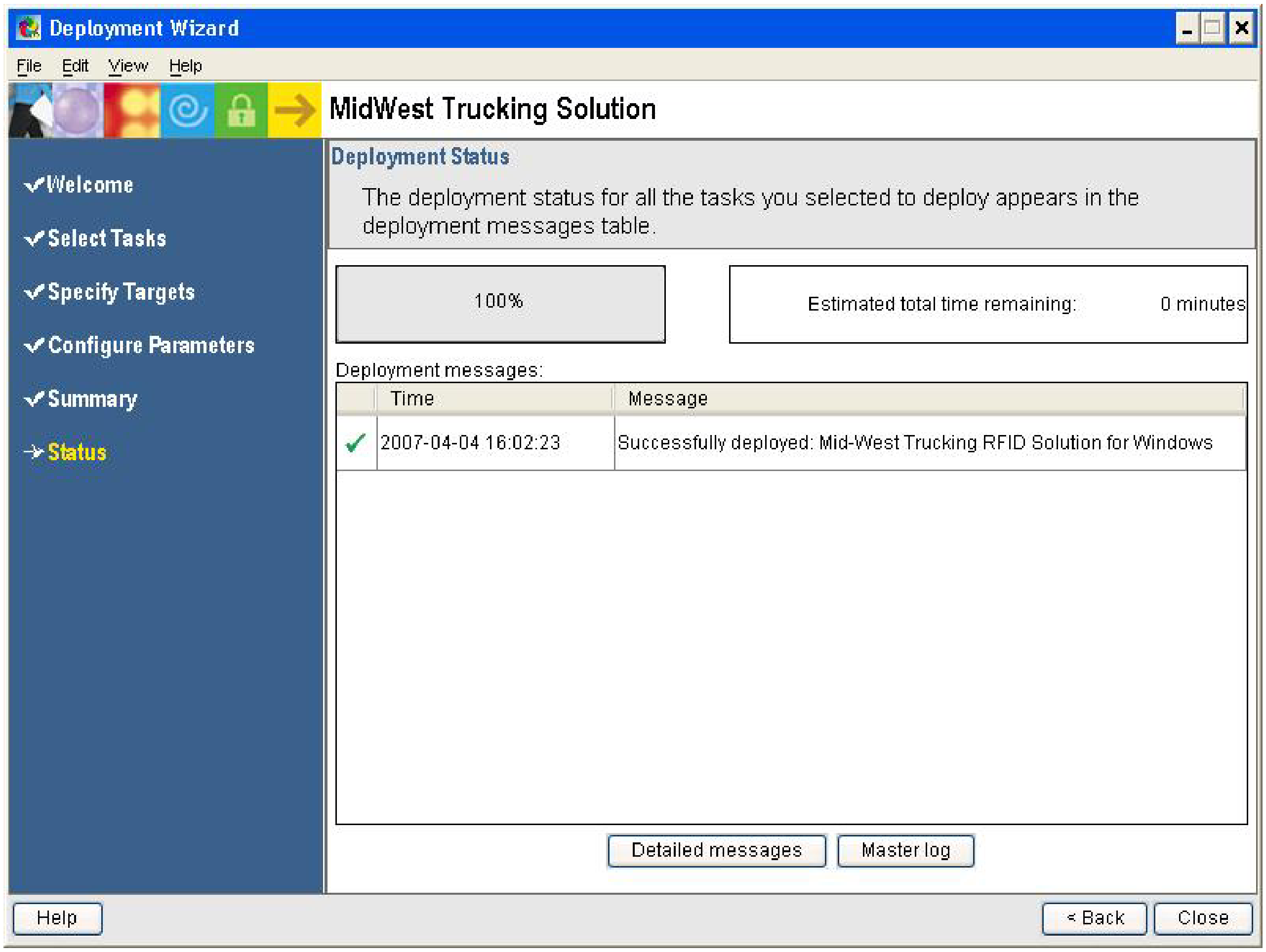
Deployment messages:

Time	Message
2007-04-04 16:01:34	Deploying: Mid-West Trucking RFID Solution for Windows

Figure 72. Deployment wizard: Status window

The percent of deployment completed, estimated time remaining, and deployment messages are displayed in the Status window.

14. When the deployment is complete, the Status window should appear as in Figure 73.



Deployment Status

The deployment status for all the tasks you selected to deploy appears in the deployment messages table.

100%

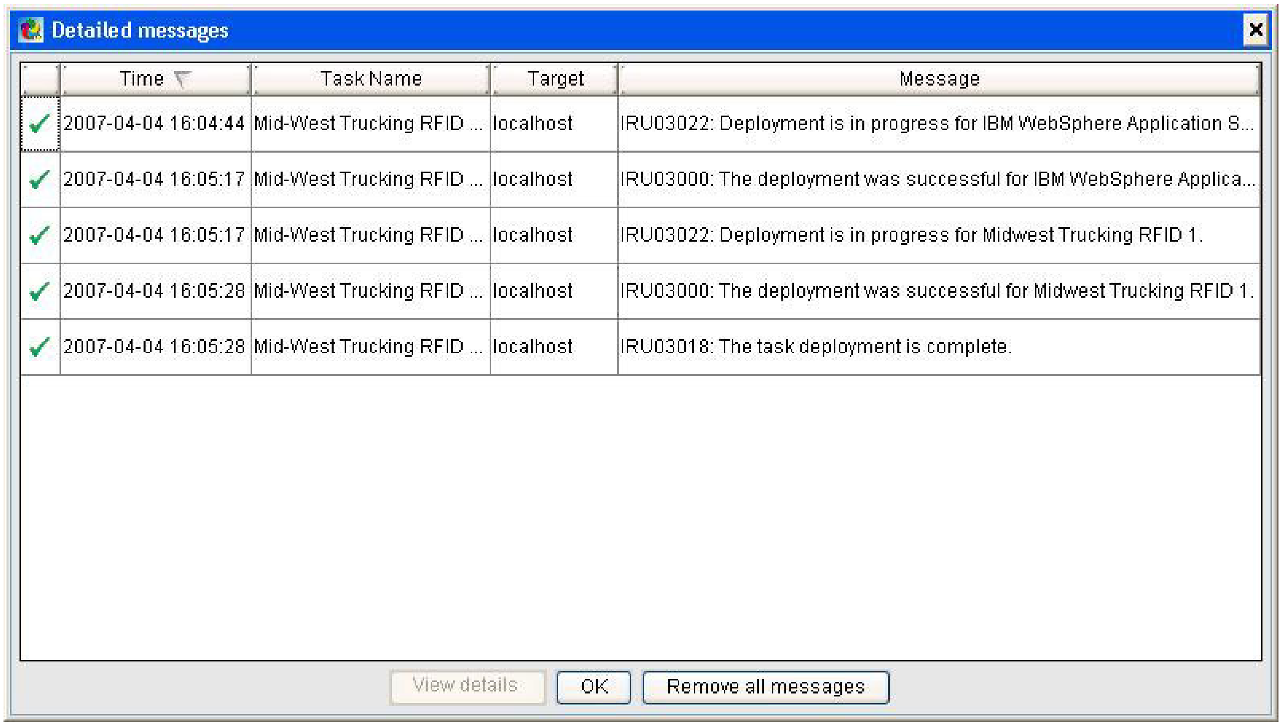
Estimated total time remaining: 0 minutes

Deployment messages:

	Time	Message
✓	2007-04-04 16:02:23	Successfully deployed: Mid-West Trucking RFID Solution for Windows

Figure 73. Deployment wizard: Status window; deployment complete

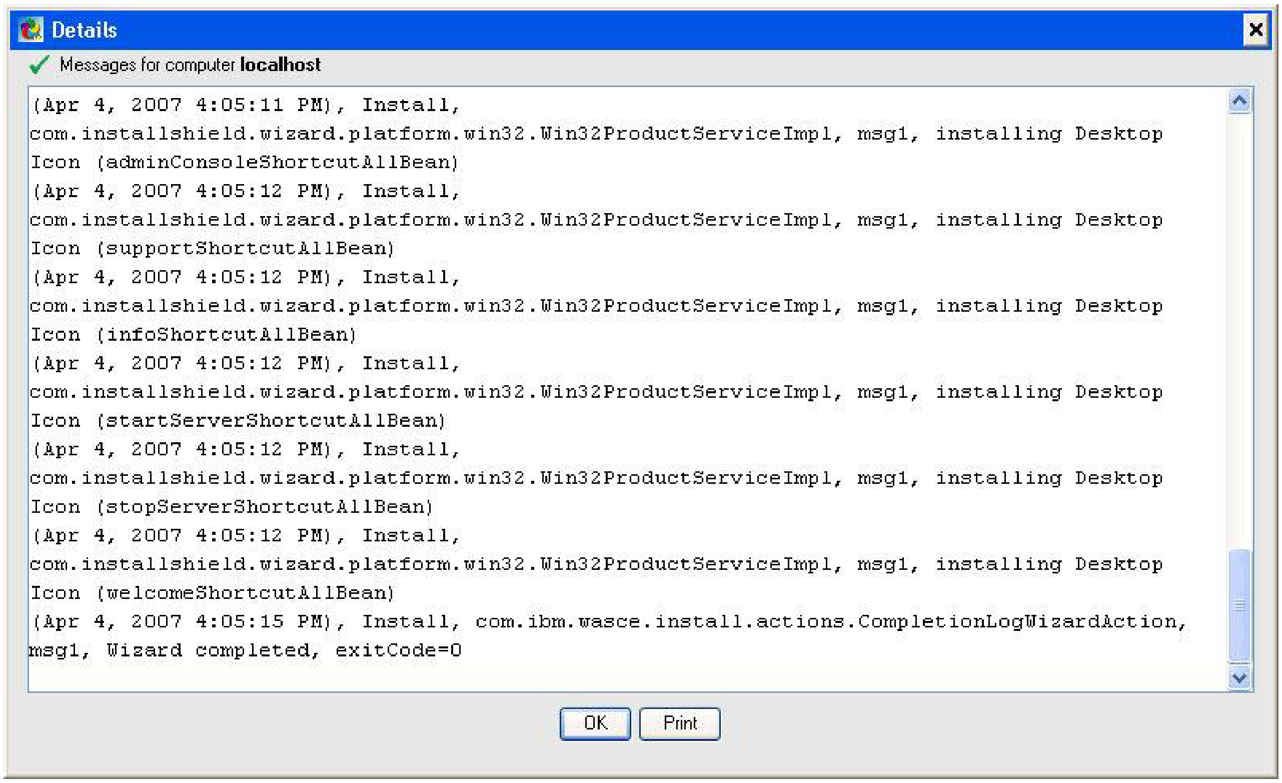
15. To review the messages generated during deployment in detail, click **Detailed messages**. The Detailed messages window opens.



	Time	Task Name	Target	Message
✓	2007-04-04 16:04:44	Mid-West Trucking RFID ...	localhost	IRU03022: Deployment is in progress for IBM WebSphere Application S...
✓	2007-04-04 16:05:17	Mid-West Trucking RFID ...	localhost	IRU03000: The deployment was successful for IBM WebSphere Applica...
✓	2007-04-04 16:05:17	Mid-West Trucking RFID ...	localhost	IRU03022: Deployment is in progress for Midwest Trucking RFID 1.
✓	2007-04-04 16:05:28	Mid-West Trucking RFID ...	localhost	IRU03000: The deployment was successful for Midwest Trucking RFID 1.
✓	2007-04-04 16:05:28	Mid-West Trucking RFID ...	localhost	IRU03018: The task deployment is complete.

Figure 74. Deployment wizard: Detailed messages window

16. To view the log for the WebSphere Application Server Community Edition deployment, select the IRU03000 message and click **View details**. The log file opens.



```
(Apr 4, 2007 4:05:11 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (adminConsoleShortcutAllBean)
(Apr 4, 2007 4:05:12 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (supportShortcutAllBean)
(Apr 4, 2007 4:05:12 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (infoShortcutAllBean)
(Apr 4, 2007 4:05:12 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (startServerShortcutAllBean)
(Apr 4, 2007 4:05:12 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (stopServerShortcutAllBean)
(Apr 4, 2007 4:05:12 PM), Install,
com.installshield.wizard.platform.win32.Win32ProductServiceImpl, msg1, installing Desktop
Icon (welcomeShortcutAllBean)
(Apr 4, 2007 4:05:15 PM), Install, com.ibm.wasce.install.actions.CompletionLogWizardAction,
msg1, Wizard completed, exitCode=0
```

Figure 75. WebSphere Application Server Community Edition log file

17. Click **OK** to exit the log file and return to the Status window.
18. Click **Close**.

For more information about the deployment wizard, see Appendix D, “Using the deployment wizard,” on page 123.

Appendix A. Using the application editor

The application editor provides a way to create and edit application wrappers without having to hand-code the XML source. The editor contains the following pages:

General
Configure general information about the application, including supported translation languages and operating systems.

Files
Define the disks containing the files needed to install the application, and specify any external JAR files that your user programs need in their classpaths.

Programs
Add and edit user programs that run when you deploy the application.

Variables
Define and associate variables that the end-user can configure from the deployment wizard.

Source
Edit the application wrapper XML source directly.

Creating an application project

To create a Solution Assembly Toolkit application project, perform the following steps:

1. From the Express Runtime developer perspective, click **File > New > Application Project**.
2. Type the project name. Click **Next**.
3. Type a version number and estimated installation time for the application.
4. Select the default language and operating system for the application.
5. Click **Finish** to generate the application project.

The project includes the standard folder structure, including the externalSupportJars folders populated with all default resources. It also includes a basic application .axml file that is created in the `src/<applicationID>` folder.

Notice that there is a warning icon on the Programs tab. Before a new application can be generated successfully, you must enter additional information on the Programs page. See "Providing user programs information" on page 88 for more information.

Creating an application wrapper

In the planning process for creating an application wrapper, identify the tasks necessary during implementation:

- Define the application ID.
- Define the general application information.
- Define the main deployment program, along with any other user programs needed.
- Define variables for any configuration parameters.

- Specify the names of all files required to run the application's user programs.

After the planning phase for the application wrapper, implementation can begin. To begin developing an application wrapper with the Express Runtime developer, click **Start > Programs > IBM Solution Assembly Toolkit 2.2 > Express Runtime Developer** on Windows, or **Main Menu > IBM Solution Assembly Toolkit 2.2 > Solution Assembly Toolkit Developer** on Linux.

When the application wrapper is created, the next step is to generate a binary application from the application wrapper.

Configuring translation languages

Select the translation languages that you want to provide for the application wrapper. For every language that you select, you must provide an XML file that provides translated strings for that language. The following table shows the languages that you can select to use in an application wrapper, and the corresponding XML file names that you need to provide for each language that you use. The XML file name depends on the file name of the application wrapper. The following file names are based on an application wrapper file name of application.axml

Table 1. Supported languages and corresponding xml file names

Language	**XML file name**
Brazilian Portuguese	application_brazilianportuguese.xml
English	application_english.xml
French	application_french.xml
German	application_german.xml
Italian	application_italian.xml
Japanese	application_japanese.xml
Korean	application_korean.xml
Simplified Chinese	application_simplifiedchinese.xml
Spanish	application_spanish.xml
Traditional Chinese	application_traditionalchinese.xml

To specify the languages you want to use for an application wrapper, perform the following steps:

1. Open the Express Runtime developer. Create a new application project, or open an existing application project.
2. Click the General tab of the application editor.
3. Select a default language for the application wrapper.
4. Select the check boxes that correspond with the languages you want to use for the application wrapper.
5. For each language selected, create a corresponding XML file from the table above.

For more information about the behavior of wrapper translation, and the steps necessary to use the translation mechanism, refer to "Enabling an application wrapper for globalization" on page 98.

Configuring operating systems

You can specify the operating systems that the application runs on. Choose from the following operating systems:

- AIX

  Note: The Express Runtime console and included middleware does not support AIX®.
- Linux
- Linux on POWER™
- OS/400® (i5/OS®)
- Windows

To select one or more operating systems that the application runs on, perform the following steps:

1. Open the Express Runtime developer. Create a new application project, or open an existing application wrapper.
2. Click the General tab of the application editor.
3. Select the check boxes that correspond with the operating systems your application supports.

Click **Select All** to select all of the available operating systems, or click **Deselect all** to clear all the selections you have made.

If any of your application's user programs are custom programs, or InstallShield executables, it is likely that your application supports only one operating system.

Providing general application information

When a new application project is created, the application XML file is displayed in the application editor of Express Runtime developer. You can customize the application project by providing information to the editor. To start the editor, select **Start > Programs > IBM Solution Assembly Toolkit 2.2 > Express Runtime Developer** on Windows, or **Main Menu > IBM Solution Assembly Toolkit 2.2 > Solution Assembly Toolkit Developer** on Linux. You can create a new application project, or work with an existing application project. To work with an existing application project, double-click the project you want to work with in the Express Runtime Explorer view.

To provide general application information to the editor, perform the following steps:

1. Click the **General** tab.
2. The application name, application ID, version number, operating systems and default language provided during the project creation are displayed on this tab. An asterisk (*) denotes a required field in the editor. The following information is optional:

   Provider name
   The display name for the application provider. For example, *My Company*.

   License text
   Contains the text for the application license. The license is displayed in

the Application Properties License Acceptance window. If license text is also specified in the solution XML file, the application license text is not displayed.

Deployment package protected
Used to prevent regeneration of the deployment package.

Deployment package name
Determines the name of the deployment package .jar file. If nothing is specified, the application ID is used.

Configuration instructions
Provides the text that is displayed in the application deployment parameters panel which describes, in general terms, what the user is to do. For example, "Provide the following information."

3. Save the application wrapper.

Providing user programs information

User programs run during the application deployment to perform the main installation, and pre- or post-installation operations. A user program can be a Java program, a stand-alone executable, or a script, but it must be able to run silently. A main user program is required in an application wrapper. Other user programs, such as a predeployment checker, an entry program, and an exit program, can be included in an application wrapper.

The support framework APIs are included in the product information center to help you write Java-based user programs. You can also access the APIs installed with the product in the <install location>\SolutionEnabler\javadoc directory by opening index.html.

Predeployment checker

Before an entry, exit, or main program is run, the application's deployment package must be transferred to the target computer. A predeployment checker runs before the application entry, main and exit programs, and before the deployment package is transferred to the target computer. It can ensure that the target computer is ready for the deployment. For example, a predeployment checker might ensure the following criteria are met:

- Adequate disk space is available on the target computer.
- System requirements are sufficient.
- There is no conflicting software.
- No previous version of the application exists.

When the predeployment checker exits, it returns one of the following return codes:

- **0** indicates that the application deployment continues.
- **1** indicates that deployment will not occur because the application is already installed. A message is displayed. The deployment wizard continues to the next application.
- **A negative value** indicates that the application deployment will not occur. An error message is displayed. The deployment wizard continues to the next application.

To display customized messages in the deployment wizard user interface, refer to the support framework Java documentation to see methods available in SupportHelper.java, such as postCompletionMessageToDeployerUI.

Entry program

Entry programs are typically used to perform pre-configuration or setup required before the main installation program runs.

Main program

The main user program is the installation program that is run on the target computer. Any type of installation program is acceptable as a main program if it can be invoked from a command-line and run silently. A main program is required for every application. If there is no main program specified in an application, a warning icon is displayed and the application will not be able to be generated.

Exit program

Exit programs are typically used for post-configuration or cleanup required following the product installation. Exit programs can also be used to determine if a main installation program installed successfully.

Note: The return code values for all predeployment checkers, entry, exit, and main programs running on Linux or OS/400 (i5/OS)[1] platforms must be between -128 and +127.

Specifying basic user program configuration

Specify basic information about a user program by performing the following steps:

1. Open the Express Runtime developer. Create a new application project, or open an existing application project.
2. Click the Programs tab of the application editor.
3. Select an existing user program from the tabs across the top to edit, or click the **Enable program** check box to add a new user program.
4. Select a program type. The main user program is the installation program that is run on the target computer. Any type of installation program is acceptable as a main program if it can be invoked from a command-line and run silently. A main program is required for every application. The following types of programs are available:

   Custom program
   Use to run a custom script, executable, or system command. ISMP executables should be specified as custom programs.

   InstallShield program for Windows
   Use to run an InstallShield installation on a Windows platform. Ensure that the program is an InstallShield executable and not an ISMP executable.

   Java program
   Use to run a Java program from a .class file.

[1] The OS/400 operating system is known as the i5/OS operating system beginning with V5R3.

5. Specify a program name. If you have already specified the files needed to run this user program, and you know the name of the program, type it directly into the **Program** field. Alternatively, you can click **Set...** to use the Set Program wizard.

Using the Set Program wizard

To use the Set Program wizard to add user programs, or configure existing user programs, perform the following steps:

1. Click the **Enable program** check box. The wizard is displayed and prompts you to select a type of user program that you want to add. Select a type of user program, and click **Next**. If you are configuring an existing user program, select **Set...** beside the **Program** field. Verify that the program type as displayed in the wizard is correct. Click **Next**.
2. Specify any software image files needed to run the user program on the **Add software image files** page.
3. Click **Next**. A list of detected files (or classes, in the case of Java programs) is provided on the last panel of the wizard. If the program you want this user program to run is not displayed in the list, ensure that you have added the program files correctly (as explained in step 2). After you specify a valid main program, the informational exclamation mark icon is no longer displayed.

Specifying custom program options

Custom programs and Java programs support additional options that are unique to these types of user programs, and are not required by other program types. You can use the following additional options:

Custom programs

Identify custom programs that use the shell interpreter of the operating system to run the program by selecting **System command** in the Custom program options section. For example, the command ″ping″ would be a system command since it is located in the system path, and not provided to the target computer by the application. If you provide a ″ping″ executable (by including it in the application project or listing it in the software image files list), **System command** should not be checked.

Specifying Java program options

Custom programs and Java programs allow additional options that are unique to these types of user programs, and are not required by other program types. You can use the following additional options:

Java programs

Provide the classpath of any additional resources a Java program uses in the Java Program options section.

Specifying advanced user program configuration

Provide advanced configuration information for a user program by performing the following steps:

1. Open the Express Runtime developer. Create a new application project, or open an existing application project.
2. Click the Programs tab of the application editor.
3. Select an existing user program from the tabs across the top to edit, or click the **Enable program** check box to add a new user program.

4. After providing the required Basic Program Configuration information, expand the Advanced Program Configuration information section and provide the following information as needed:

   Wait for completion

   Set by default, deployment of the next program does not begin until this program has completed. If set to false, the success type for the user program must be ″Assume Success″. This option must be set as true for predeployment checkers.

   Program reboots

   Select to indicate that the user program reboots the target computer after it is finished running. If set to true, the success type for the user program must be ″Assume Success″.

   Note: This option is not available for predeployment checkers.

   Force reboot

   Select to force an automatic reboot of the target computer on successful completion of the user program.

   Include Ant

   Select to include the Apache Ant 1.6.5 runtime with an application wrapper.

   Note: Selecting the Include Ant check box causes all the files required for the Ant runtime to be added to the user programs .jar file for the application. Additionally, this causes the JAVA_HOME environment variable to point to the JRE that is installed on the target computer during deployment. If you want to use a different JRE for this application, you must add a JAVA_HOME environment setting for the application.
   To invoke Ant on Windows, you must specify `iru_ant\bin\ant.bat` as the program. To invoke Ant on Linux, i5/OS, or AIX, you must specify `iru/bin/ant` as the program. The program must be defined in the **Basic Program Configuration** section of the **Programs** tab. When you include Ant, and one of the above invocations is used, the --noconfig argument is added to the invocation as the first argument. This ensures that the version that is deployed with Solution Assembly Toolkit is used.

   Timeout

   Specify the number of minutes the deployment wizard waits for a program to complete. If this time limit is exceeded, the program times out, and the deployment wizard indicates failure. The default value is 90 minutes. Be sure to calculate the installation time for each user program when specifying the estimated installation time. To set the estimated installation time for the application, see **Providing general application information** in this information center.

   Environment variables

   Specify any environment variables that you want set while the user program runs. You can override environment variables, or specify new ones to be defined. To add an environment variable Click **Add**. Provide a variable name and variable value. Click **Finish**.

Specifying success type and log file search strings

The success type of a user program defines how the deployment wizard determines if it ran successfully. In the Success type section, select a success type and provide any additional required configuration. The following success types are available:

Check return code
Installation is considered successful if a return code of 0 is returned.

Check log file for string
Success is determined by searching the user program's log file for specific strings.

Assume success
Installation is always considered successful.

If you use the success type of **Search log file for string**, you can add a string that the Express Runtime developer searches for to evaluate if the program runs successfully. For example, you can establish a rule that if the log file that is generated by the program contains the word ″exception″, the program encountered an error when it ran.

To add a search string, perform the following steps:

1. Click **Add**. The Add Log File Search String dialog is displayed.
2. Provide a search sting, and select whether the presence of that search string indicates success or failure.
3. Click **Finish**.

To remove a search string, select the search string that you want to remove, and click **Remove**.

Specifying program arguments

In the Arguments field of the Program Arguments section, click **Add** to add program arguments. Program arguments that are available depend on the way the user programs are configured. For example, if the application project does not use variables, variable values cannot be used as program arguments. The following types of arguments are available:

Log file name
Passes the fully-qualified path of the log file to the user program.

Response file name
Passes the fully-qualified path of the response file to the user program. The file name of the response file might be different then the original file name when it is created on the target computer. Always use the response file name argument type instead of entering the response file name directly as a string argument.

String argument
Passes the provided string to the user program.

Variable value
Passes the value of an application variable to the user program.

Select the radio button that corresponds to the type of argument that you use in the Add Arguments dialog. Click **Next**. Provide the string, response file, log file, or variable, and click **Finish**.

By default, arguments are passed to the user program individually. In some cases, multiple arguments need to be grouped together. For example, a user program might require a single argument in the form

-installLocation=<PATH>

where -installLocation= is a simple string argument, and <PATH> is a variable for the desired installation path an end user can change.

To combine two arguments into a single argument, perform the following steps:

1. Right-click the first argument.
2. Select Properties from the context menu. The properties dialog is displayed.
3. Set the **Concatenate With Next Argument** property to true.
4. Click **OK**.

Providing application variable information

Define variables to display configuration parameters to the end user during deployment.

Specifying the variable type

Define variables to display the deployment parameters to the user at deployment. The variables are displayed in the deployment wizard's configuration parameters panel. There are five types of variables:

Boolean variable
To request a true or false value from the user. The field is displayed as **Yes** and **No** radio buttons on the deployment wizard interface.

Drop-down list variable
To request a value from a list of three or more choices shown by clicking an arrow. The field is displayed with either a default value or blank on the deployment wizard interface. A drop-down list displays the list of possible choices when clicked. This type of variable requires variable value configuration.

File path variable
To enable browsing to request a path from the target computer. The field is displayed as a text field with a Browse button to the right on the deployment wizard interface.

Password variable
To request data input that masks the entered value for security reasons. The field is displayed as an asterisk-masked text field on the deployment wizard interface.

String variable
To request a standard text field for data input. The field is displayed as a text field on the deployment wizard interface.

To add a variable, perform the following steps:

1. Click the **Variables** tab.
2. In the Application Variables section, click **Add**.
3. From the Add Variable dialog, select one of the following variable types:
   - Boolean variable
   - Drop-down list variable

- File path variable
- Password variable
- String variable

4. Provide a variable name and description. Click **Finish**.

Adding label and help text to an application variable

After a variable is added, you can add or change basic configuration information by performing the following steps:

1. Click the Variables tab.
2. In the Application Variables section, select the variable you want to change. When you select the variable, notice that the fields on the right populate with the variable's information.
3. In the Basic Variable Configuration section, change any of the following fields:

   Label text
   This field specifies the text that labels an input field on a task's application customization parameters panel in the deployment wizard.

   Help text
   This optional field specifies help text for the variable. Help text is displayed in a tooltip that is shown when hovering the mouse over an input field, or when Ctrl + F1 is pressed while the cursor is in an input field.

   Type Choose whether this variable is a ″Typical Variable″ or an ″Advanced Variable″. In the deployment wizard interface, typical variables are displayed on a separate tab than advanced variables. Expose commonly changed variables (such as installation locations) as typical variables, and rarely changed variables (such as port numbers) as advanced variables. Additionally, assign acceptable default values to advanced variables so the end user is not required to change any value on the advanced configuration tab in the deployment wizard to deploy a solution.

Creating variable associations

After you complete the basic configuration of a variable, configure the variable associations. Variable associations make the variable values available to the user programs by including these values in response files or properties files.

To pass the value of a variable to a user program as an argument, see "Specifying program arguments" on page 92.

To configure variable associations, perform the following steps:

1. In the Variable Associations Configuration section, click **Add**.
2. In the Add Association dialog, select the type of association that you want to add. You can create the following types of associations:

   CID response file association
   The Configuration, Installation, and Distribution response file format supported by DB2® applications.

   ISMP response file association
   The response file format for InstallShield MultiPlatform Edition installations.

ISS response file association
The response file format for InstallShield installations.

Properties association
Standard properties file format. Java-based user programs can access the value of a variable through this type of association, using the support framework.

3. Click **Next**
4. Provide the information needed to insert the value of the variable into the file type specified:
   - CID response file association
     - Choose which response file to insert the value of the variable into.
     - Choose the keyword to store the value of the variable under. If the keyword exists in the response file, the value of the variable is substituted for the existing value. Otherwise, the keyword is created at the end of the response file.
   - ISMP response file association
     - Choose which response file to insert the value of the variable into.
     - Choose the property key and the property key type to store the value of the variable under. If the property key and the property key type exist in the response file, the value of the variable is substituted for the existing value. Otherwise, the property key and the property key type are created at the end of the response file.
   - ISS response file association
     - Choose which response file to insert the value of the variable into.
     - Choose the response file section to store the value of the variable under. If the section type exists in the response file, the value of the variable is substituted for the existing value. Otherwise, the section is created at the end of the response file.
     - Choose the keyword to store the value of the variable under. If the keyword exists in the response file, the value of the variable is substituted for the existing value. Otherwise, the keyword is created at the end of the specified section.
   - Properties file association
     - Choose the keyword under which to store the value of the variable. The variable's value is stored in a standard Java properties file. Java-based user programs can access the value using the support framework provided with Solution Assembly Toolkit.
5. Click **Finish**.
6. If the variable you are adding an association to is a Boolean variable and you want different values for the association other than the standard ″true″ and ″false″ values, you can configure the valueIfTrue and valueIfFalse properties of the association. To configure these values, right-click the variable association, select Properties, then edit the valueIfTrue and valueIfFalse properties.

Providing application file information

An application wrapper must list all of the files that need to be transferred to a target computer to run the application successfully during a deployment.

Two categories of files are required:

- Software image files

- External JAR files

Providing software image file information

Software image files are the files needed to run the application installation. The deployment package is transferred to the target computer after the application's predeployment checker determines the deployment should proceed.

There are two ways a customer can obtain the deployment package needed to deploy an application:

- A pre-built deployment package can be provided with the application
- The customer can be prompted for the disks needed to create the deployment package

The disk information you provide is used to create the deployment package for your application. If your application has a deployment package, you need to define at least one disk containing the files for that deployment package. You can also customize the disk prompts that appear while creating the deployment package.

The definition of the media on which the files exist for a deployment package is called a distribution. You can use distributions to define the media from which you will accept the files to create a deployment package. For example, you can define one distribution to accept PrereqApp application's source files from four CD-ROMs, and you can define a second distribution accepting the same PrereqApp application's source files from two DVDs. Defining multiple distributions is not necessary if a pre-built deployment package will be provided for your application.

After you determine which disks and distributions you want to define, perform the following steps to set up the disks in the deployment package:

1. Click the **Files** tab.
2. Click the **Add...** button under Software Image Files. The Add disk wizard launches to guide you through distribution, prompting, and path prefix information for each disk.
3. Indicate the distribution name to which this disk will be added. Either leave the Default Distribution value, select another existing distribution, or specify a new distribution. If you do not specify information in the Add disk wizard, the default distribution for a single disk is assumed, and the user is prompted with generic ″Insert disk 1″ messages if building deployment packages. If you are providing the deployment packages, the user is not prompted for disks defined in a distribution. (He might still be prompted for disks if the deployment package is large enough to require multiple disks for size reasons.)
4. Type the disk label, index and media type of the disk. These values are used to prompt for this disk, following the pattern ″Insert {Disk type} {Disk index} labeled {Disk label}.″ You can change the values later in the editor. These fields are optional.
5. Type the path prefix. The installation technology for your application might require a specific disk path prefix to keep files from different disks grouped together appropriately during deployment. For example, if there is more than one setup.exe file, the prefix, such as disk1, distinguishes the files, such as /disk1/setup.exe.
6. Click **Finish** to exit the Add disk wizard.

7. Click the **Update** button beside the Files field to define the source files included from this disk. The Add files from... wizard appears.
8. Browse to the location of the disk, on either your hard drive or an external drive, and click **OK**. The wizard prompts you for which files to include from the disk.
9. Select the appropriate files, and click **OK**.
10. To include more files, click the **Update...** button again. Confirm the location of the disk, or specify a different location for the disk. The Add files from... wizard verifies that the location contains the files already specified for the disk. If the existing files cannot be found, you are prompted with a message. Click **Change location...** to find the correct disk location, or click **Continue** to replace the previous selection of source files with the files from this new directory.
11. Repeat for additional disks until you define all the disks for your deployment packages.
12. Create new distributions for each possible media combination. For the customer to distinguish distributions, the first disk of each distribution should be unique.
13. Save the application project.

When the customer uses the deployment wizard to create his deployment package, he is prompted for the first disk as you defined it in each of your distributions. When he inserts a disk, the deployment wizard checks to make sure the disk is the correct disk for at least one of your defined distributions. If it matches the first disk of multiple distributions, the user is prompted for which distribution is desired. The deployment wizard also checks that the disk has the required contents you defined. If the content requirements are not met, the deployment wizard prompts for a valid disk until the customer provides a valid disk.

Examples

For example, Customer A might have your application on two CD-ROMs while Customer B might have it on a single DVD. Define two distributions (one for CD-ROMs and one for DVDs) using the following steps:

1. Click the **Add...** button and define a new distribution called `MyApp CD-ROM`.
2. For Disk label, enter `MyApp`.
3. For Disk index, enter `1`.
4. For Disk type, select `CD-ROM`.
5. For Disk prefix, enter `disk1`.
6. Click **Update...**, browse to the first CD-ROM disk that contains the MyApp files, select all the files for your customer to install, and click **OK**.
7. Follow the same process for the second CD-ROM, indicating a Disk label of `MyApp`, a Disk index of `2`, a Disk type of `CD-ROM`, and a Disk prefix of `disk2`.
8. Click the **Add...** button again and define a new distribution called `MyApp DVD`.
9. For Disk label, enter `MyApp`.
10. For Disk index, enter `1`.
11. For Disk type, select `DVD`.
12. For Disk prefix, enter `disk1`.
13. Click **Update**, browse to the DVD disk that contains the files for your application, select all the files for your customer to install, and click **OK**.

When Customer A builds his own deployment package, he is prompted with ″Insert CD-ROM 1 labeled MyApp or insert DVD 1 labeled MyApp″. He inserts CD-ROM 1 labeled MyApp. All the MyApp files are pulled into the deployment package with a path prefix of /disk1/, such as /disk1/setup.exe. The deployment wizard continues prompting him for each disk in the order of the distribution definition. All the files from CD-ROM 2 labeled MyApp are pulled into the deployment package with a path prefix of /disk2/, such as /disk2/setup.exe.

When Customer B builds his own deployment package, he is also prompted with ″Insert CD-ROM 1 labeled MyApp or insert DVD 1 labeled MyApp″. He inserts DVD 1 labeled MyApp. All the MyApp files are pulled into the deployment package with a path prefix of /disk1/, such as /disk1/setup.exe.

Providing required libraries information

If an application's user programs require any external JAR files to run, you can specify them on the Files tab of the application editor. To add external JAR files, perform the following steps:

1. Click the **Files** Tab.
2. Click **Add** in the External JAR Files section to add external JAR files. The Add External JAR Files dialog is displayed.
3. Browse to and select JAR files to add.
4. Click **Add**.
5. Repeat this step for each folder or file you want to add.
6. Save the application project.

Specify any external JAR files an application's user programs require to run. Any external JAR files are automatically added to the classpath of any Java user programs. You do not need to add the support framework JAR files, because those files are already included during the deployment.

Enabling an application wrapper for globalization

There are many strings in application and solution wrappers that are displayed to end users in the deployment wizard interface. These strings can be specified directly in the wrapper files, or they can be placed in translatable XML files. To enable translation for an application or solution wrapper, perform the following steps:

1. Decide which translation languages your application or solution will support, and create the necessary translatable XML files, as described in the topic Translation Language Configuration
2. Populate the default-language XML file with all the translatable strings for the wrapper, giving each string a meaningful key name. If you have already created a wrapper, you can copy the translatable strings from your wrapper into the default-language XML file. Keys must consist of alphanumeric characters only (no white-space), and must begin with a letter. The following example shows what a default-language XML file for a simple solution might look like.

```
<?xml version="1.0" ?>
<SolutionID>
    <solutionTitle>Solution Title</solutionTitle>
    <task1Details>Details for task #1</task1Details>
    <task1Description>Description for task #1</task1Description>
</SolutionID>
```

3. Using the application or solution editor, replace all the translatable strings with references to their keys in the default-language XML file.

   If using the GUI editor, simply enter the key name, prefixed with a percent sign (%).

   Following the example, the solution title field would be set to: `%solutionTitle`

   If editing the XML source directly, use the `translatedKey` attribute. Following the example, the `<title>` element in the solution wrapper would be specified as:

   ```
   <title translatedKey="solutionTitle" />
   ```
4. Place the translations of the strings in the default-language XML file in the appropriate language XML files. This step can be performed separately, if translations are not immediately available. If a string does not exist for a particular translation language, the string from the default-language XML file is used.
5. Generate the application or solution wrapper. Any missing translatable strings are detected, and the appropriate error messages are displayed in the console.

Appendix B. Using the solution editor

The solution editor provides a way to create and edit solution wrappers without having to hand-code the XML source. The editor contains the following pages:

General
Configure general information about the solution, including supported translation languages.

Tasks
Add applications and configure tasks that run when the solution is deployed.

Source
Helps you to edit the solution wrapper XML source directly.

Creating a solution wrapper

You can use the Express Runtime developer to package an application or applications into a solution. This is done with an .sxml file that references existing applications and provides additional configuration information. This .sxml file is called a solution wrapper.

Creating a solution project

To create a solution project, perform the following steps:

1. From the Solution Assembly Toolkit perspective, click **File > New > Solution Project**.
2. Type a project name. Click **Next**.
3. Type a solution ID and a solution title.
4. Select the default language for the solution.
5. Click **Finish**.

The project includes the standard folder structure, populated with all default resources. It also includes a basic solution .sxml file in the `src` folder.

Providing general solution information

When a new solution project is created, the solution XML file is displayed in the solution editor of Express Runtime developer. You can customize the solution project by providing information to the editor. To start the editor, select **Start > Programs > IBM Solution Assembly Toolkit 2.2 > Express Runtime Developer** on Windows, or **Main Menu > IBM Solution Assembly Toolkit 2.2 > Express Runtime Developer** on Linux. You can create a new solution project, or work with an existing solution project. To work with an existing solution project, double click the project you want to work with in the Express Runtime Explorer view.

To enter or edit the general solution information, perform the following steps:

1. Select the **General** tab in the right pane of the Express Runtime developer.
2. Enter or modify the following information for the basic solution configuration. Fields designated with an asterisk (*) are required. All other fields are optional.

   ID
   Enter an ID for the solution. This ID must begin with an alphabetic character, be less than 200 characters in length, and consist only of alphanumeric characters, underscores, hyphens, and periods.

Title Enter a descriptive title for the solution. This title is displayed in the title bar of the deployment wizard main window and in the Product Information window.

Version
Enter the version number of the solution.

Icon Select an image file to be displayed next to the title of the solution. However, the icon is not displayed on Red Hat Enterprise Linux platforms where Bluecurve is the desktop theme. Bluecurve is the default desktop style for GNOME and KDE environments. Supported image formats for icons are the following:

- .gif
- .jpg
- .png

Icon images should be 48x48 pixels in dimension.

Splash screen image
Select an image file to use in the splash screen for the solution. The image is displayed when the solution is opened in the deployment wizard. Supported image formats for splash screen images are the following:

- .gif
- .jpg
- .png

Splash screen images should be 516x390 pixels in dimension.

3. Specify information for the license. License configuration information can include the following:

License prompt
Enter text to display on the license agreement dialog.

License text
Enter the text of the solution license agreement. The license agreement is displayed when a deployment to a target computer is started. If license text is specified here, license text from the application is not displayed.

4. Specify information for the welcome screen. Welcome screen configuration information can include the following:

Welcome screen image
Select an image file to use in the welcome screen for the solution. The image is displayed when the solution is opened in the deployment wizard. Supported image formats for welcome screen images are the following:

- .gif
- .jpg
- .png

Welcome screen images should be 202x518 pixels in dimension.

Welcome screen title
Enter a title for the welcome screen.

Welcome screen text
Enter text to be displayed on the welcome screen for the solution.

5. Specify information for the ″About″ screen. ″About″ screen configuration information can include the following:

   About screen image
   Select an image file to use in the About screen for the solution. The image is displayed when the solution is opened in the deployment wizard. Supported image formats for about screen images are the following:

   - .gif
   - .jpg
   - .png

   About screen images should be 400x300 pixels in dimension. 300 pixels is the maximum width for an about screen image.

   About screen text
   Provide legal information about the solution in an accessible format.

6. Specify advanced solution configuration. Advanced solution configuration information can include the following:

   Deploy only to localhost
   Indicate that the solution is deployed only to the computer that is running the solution media (the local computer). In the deployment wizard, the end user is not prompted to specify target computers, and localhost is assumed to be the only target computer.

   Disable individual task deployment
   Indicate that the end user is not allowed to deploy tasks individually. If selected, the deployment wizard deploys all selected tasks together.

   Deployment package path
   Enter a location for the deployment packages.

   Startup checkpoint
   Enter a program that runs before loading the solution in the deployment wizard. For example, you can use the solution startup checkpoint to call a Java class that checks for prerequisites on the target computer and cancels launching the deployment wizard if the prerequisites are not met. In the Startup checkpoint field, type in the package-qualified name of the Java class to run, or click the **Startup checkpoint** field name to launch the New Java Class wizard. For more information about creating a checkpoint, see the Startup Checkpoint Java documentation.

   Maximum number of target connections
   Enter the maximum number of target computers that can be connected simultaneously with the staging server. The valid range is 1 to 100 target computers.

   Maximum number of data connections
   Enter the maximum number of active data connections that a staging server can have at one time. The valid range is 1 to 100 data connections.

   Data port
   Specify the TCP/IP port number the deployment wizard uses to send data such as deployment packages to target computers. If this port number is not available, specify a different port number between 0 to 65535. Note that ports 1 to 1023 are usually reserved for use by other programs such as FTP and Telnet.

Communication port

Specify the TCP/IP port number the deployment wizard uses to communicate with the target computers. If this port number is not available, specify a different port number between 0 to 65535. Note that ports 1 to 1023 are usually reserved for use by other programs such as FTP and Telnet.

7. Specify supported language configuration. Supported language configuration information can include the following:

   Default

   Select the default language for the solution from the drop-down list. You must select a language in the Languages field for that language to appear in the drop-down list.

   Languages

   Select each language into which the solution is translated. You cannot deselect a language that you selected for the Default field.

8. Save the solution wrapper.

Adding applications and tasks

A solution defines a list of tasks which are performed at deployment time. You can add the following applications and tasks to solutions.

Applications

Programs that are deployed to the target computers that the end user specifies. Applications are contained in install tasks

Install tasks

Install tasks consist of an application or set of applications to be deployed to one or more target computers. All of the applications in an install task are deployed to all of the target computers with the corresponding operating systems that the end user specifies for the task.

For each install task, you can specify whether it can be deployed to one or multiple target computers. For example, you might want to have a database client application installed to several target computers, and a Web application server and portal server installed to a different target computer. Each of these would be a separate install task, with the database client application task allowed to target multiple computers, and the Web application server and portal server task only allowed to target a single computer. If a solution contains install tasks that are not required for all customers or in all situations, you can specify the tasks as optional so that the customer can decide whether to deploy them or not.

For each install task, you can specify whether it can be deployed to one or multiple operating systems, and for which operating systems the application is required. For example, you might want to have a Web application server installed to target computers running Windows or Linux, and a security application required to be installed to only target computers running Windows. Each of these would be a separate install task. If a solution contains install tasks that are not required for all customers or in all situations, you can specify the tasks as optional so that the customer can decide whether to deploy them or not.

Manual tasks

Manual tasks display instructions that you can customize during deployment of the solution, and wait for the user to indicate the instructions have been followed. For example, a manual task might instruct the end user to copy a script generated on one system during installation to another system and run that script.

Task groups
Task groups are collections of install tasks and manual tasks. Similar tasks (both install tasks and manual tasks) can be grouped into task groups, if desired. For example, one solution could have three sets of install tasks, each configured for a different target computer operating system. These install tasks could be placed into three task groups, allowing the user to easily select and configure install tasks for a particular operating system. If all of the tasks in a task group are optional, the task group becomes optional.

You can also override hidden variables and configure shared variable values in this section.

Adding applications to solutions

To deploy an application using the deployment wizard, you must first add the application to an install task in a solution. To add an application to an install task, perform the following steps:

1. Select the **Tasks** tab in the solution editor.
2. In the solution tasks section, click **Add**.
3. Select **Add applications**. Click **Next**.
4. Select the install task you want to add applications to, or specify the information for a new install task.
5. From the list of compatible applications, select the applications you want to add to this install task. If you are adding applications to an existing install task, the supported operating systems of the applications already in the install task determine which applications you can select. Applications are deployed only to target computers with compatible operating systems. When you select an application to add, the application defaults to deploy optionally. To make this application a requirement for deployment, click the Optional text, click the arrow to the right of the text, and select **Required** from the drop-down list. Repeat the sequence to change the Required value to Optional, if needed. A required application is always deployed as part of its install task, so target computers specified for an install task must have operating systems that are compatible with all the required applications in the install task. All required applications for an install task must have at least one operating system in common. An optional application is deployed only if the target computer has an operating system that the application supports.
6. Click **Finish**.

Use the Properties view to enter optional information about the application. Right-click the application in the editor and select Properties from the context menu. Specify information for the following optional attributes:

stopDeploymentOnFail
Set to true to stop solution deployment if the installation of this particular application fails on a target computer.

required
Set to true to require this application to be installed on the target computer. All target computers specified for the install task containing this application must have an operating system that this application supports. If set to false, this application is deployed only if the target computer has an operating system that this application supports.

Modifying the default value of a variable

Applications can provide default values for any variables they define. Sometimes, it is necessary to set or override those default values at the solution level, as in the following cases:

- If a solution changes the appearance of a variable to make it hidden or read only, and the variable does not have a valid default value, then the solution must provide a valid default.
- If a solution includes applications that, if deployed with their default values, would not work correctly (for example, the default installation locations would conflict), the solution must override the necessary default values of the variables.

To change the default value of an application variable, perform the following steps:

1. Click the **Tasks** tab in the solution editor.
2. In the Solution Tasks section, expand the install task and select the application that contains variables you want to change.
3. In the Overridden Application Variables section, click **Add** to select the variables to change.

   Notes:

   a. If the variables you want to change are already listed, modify them without going through the Override Application Variables wizard. Skip to step 6.

   b. If all variables defined in the application are already listed, skip to step 6.
4. In the Override Application Variables wizard, select the variables you want to change from the checklist. Click **Next**.
5. For each variable selected in step 4, you are presented with a wizard page that helps you to change the default value of that variable. In the Behavior drop down, select Modify the default value, then enter the new default value in the Default value text field. You can also change the appearance of this variable from this page. Once you have finished setting the default value, click **Next** or **Finish**.
6. To change the default value of a variable that is already in the Overridden Application Variables list, right-click on the variable, choose Properties, then set the following read-only and hidden properties as desired:
   - hidden
   - readonly
   - sharedAs
   - value

Changing the appearance of a variable

All of the variables defined within an application are displayed as editable fields on the configuration parameters panel in the deployment wizard. When an application is added to a solution, variables within the application can be hidden or made read-only for a particular configuration parameters panel.

To change the appearance of variables, perform the following steps:

1. Click the **Tasks** tab in the solution editor.
2. In the Solution Tasks section, expand the install task and select the application that contains variables you want to change.
3. In the Overridden Application Variables section, click **Add** to select the variables to change.

   Notes:

a. If the variables you want to change are already listed, modify them without going through the Override Application Variables wizard. Skip to step 6.
b. If all the variables defined in the application are already listed, skip to step 6.

4. In the Override Application Variables wizard, select the variables you want to change from the checklist. Click **Next**.

   Note: If the application project is not available in the workspace, no list of variables is provided. Instead, you must enter the name of the variable you want to change.
5. For each variable selected in step 4, you are presented with a wizard page that helps you change the appearance of a variable. Choose the appearance from the list, then click **Next** or **Finish** for each variable. While on this panel you can modify the default value of the variable, share the value of the variable with other applications, or share the value of the variable with an install task variable. The following options are available for the appearance of a variable:

   Editable
      This is the default type of variable.

   Read-only
      Makes the variable read-only, but still visible in the deployment wizard. Read-only variables must have valid default values, since the end-user cannot change their values.

   Hidden
      Hides the variable in the deployment wizard. Hidden variables must have valid default values, since the end-user cannot change their values.
6. To change the appearance of a variable that is already in the Overridden Application Variables list, right-click on the variable, choose Properties, then set the following properties as desired:
   - hidden
   - readonly
   - sharedAs
   - value

Sharing variables between applications

Sometimes, applications need to access the variable values of another application in the same solution. An example would be a solution which contained DB2 and a DB2 fix pack, in which the fix pack would need to have access to the main DB2 installation location. The solution editor provides a way to “link” or “share” variables between applications. Doing so ensures that any value entered for one application’s variable is stored in any other variables that are shared with it.

To share variables between applications, perform the following steps:

1. Click the **Tasks** tab in the solution editor.
2. In the Solution Tasks section, expand the install task and select the application that contains variables that you want to share.
3. In the Overridden Application Variables section, click **Add** to select the variables to share.

   Note: If the variables you want to share are already listed, modify them without going through the Override Application Variables wizard. Skip to step 7 on page 108.

4. In the Override Application Variables wizard, select the variables you want to share from the checklist. Click **Next**.

   Note: If the application project is not available in the workspace, no list of variables is provided. Instead, you must enter the name of the variable you want to share.
5. For each variable selected in step 4, you see a wizard page that allows you to specify how that variable is shared.
   a. In the Behavior drop-down list, select Share the value with other variables.
   b. If you have already shared variables from other applications in the solution, the sharedAs names of those variables are listed in the Shared as drop-down list. Select one of the existing sharedAs names from the list in order to link this value with all other variables with the same sharedAs name. If this is the first application whose variables you are sharing, or you want to share this value with variables that are not listed yet, then type in a unique name to share this variable as.
6. Once you finish configuring each variable, click **Next** or **Finish**.
7. To share a variable that is already in the Overridden Application Variables list, right-click on the variable, choose Properties, then set the sharedAs property as desired.
   - hidden
   - readonly
   - sharedAs
   - value
8. Change the appearance of this variable in the Overridden Application Variables section.
9. Configure any validation for shared variables in the Shared Variable Value Configuration section.

Sharing variables between applications and install tasks

Sometimes, applications need values provided while targeting install tasks. For example, you can indicate that the application variable WAS_Password gets its value from what the user enters as the password of the target computer for the WebSphere Application Server - Express install task. The solution editor provides a way to ″link″ variable values to install task target attributes. Doing so ensures that the value entered for an install task’s target is stored in any variables that are linked to it.

To have a variable obtain the value from an install task, perform the following steps:

1. Click the **Tasks** tab in the solution editor.
2. In the Solution Tasks section, expand the install task and select the application that contains variables that you want to share.
3. In the Overridden Application Variables section, click **Add** to select the variables to share.

   Note: If the variables you want to share are already listed, modify them without going through the Override Application Variables wizard. Skip to step 7 on page 109.
4. In the Override Application Variables wizard, select the variables you want to share from the checklist. Click **Next**.

Note: If the application project is not available in the workspace, no list of variables is provided. Instead, you must enter the name of the variable you want to share.

5. For each variable selected in step 4 on page 108, you see a wizard page that allows you to specify how that variable is shared.
   a. In the Behavior drop-down list, select **Get the value for <variable> from an install task's target attribute**.
   b. In the Install task drop-down list, specify which install task provides the information needed for this variable.
   c. In the Target attribute drop-down list, specify which variable attribute to use from the specified install task. Valid attributes include:
      - Fully-qualified hostname
      - Short hostname
      - IP address
      - Operating system
      - Selected for deployment
      - Username
      - Password (valid for password application variables only)
6. Once you finish configuring each variable, click **Next** or **Finish**.
7. To share a variable that is already in the Overridden Application Variables list, right-click on the variable, choose Properties, then set the following properties as needed:
   - hidden
   - id
   - overriddenValue
   - overrideValue
   - readonly
   - sharedAs
   - tieToAttribute
   - tieToTask

Adding install tasks to a solution

Install tasks can be parts of a parent task group.

An install task consists of a set of applications that are to be deployed to one or more target computers. Each of the install tasks in a solution is displayed in the deployment wizard. From the deployment wizard, select any combination of install tasks for deployment to one or more target computers.

To add an install task to a solution, perform the following steps:

1. Select the **Tasks** tab in the solution editor.
2. In the Solution tasks section, click **Add**.
3. Select **Create an empty install task**.
4. Click **Next**.
5. Select a **Parent task group** (if applicable).
6. Enter a **Task description**.
7. Click **Finish**.

Use the Properties view to specify optional information about the install task. Right-click the task in the editor and choose Properties from the context menu. Specify information for the following optional attributes:

id Enter text to uniquely identify your task. The id can only contain alphanumeric characters. This id can be used to reference task information from the support framework.

description
Enter label text to display on the task selection panel of the deployment wizard.

addLocalHost
Set to true to initialize the install task with localhost as the target.

isOptional
Set to true to make the selected task optional.

oneTargetDeployment
Set to true to restrict deployment of the selected task to only one computer.

selectedByDefault
Set to true to deploy the selected task by default.

Note: If the `isOptional` attribute is set to false, `selectedByDefault` is returned as **true**.

taskDetails
Enter a text description to display a brief description of the task on the task selection panel of the deployment wizard.

Adding manual tasks to a solution

A manual task is an instruction that is displayed during the deployment of the solution. The user must complete the manual task before continuing with the solution deployment. For example, a manual task might instruct the user to verify that the user ID specified has the appropriate administrative authority.

To add a manual task to the solution, perform the following steps:

1. Select the **Tasks** tab in the solution editor.
2. In the solution section, Click **Add**.
3. Select **Create a manual task**. Click **Next**.
4. Select a **Parent task group** (if applicable).
5. Enter a **Task description**.
6. Provide a brief description of the task and the instructions needed to complete it in the **Task instructions** field.
7. Click **Finish**.

Use the Properties view to specify optional information about the manual task. Right-click the task in the editor and choose Properties from the context menu. Specify information for the following optional attributes

id Enter text to uniquely identify your task. The id can only contain alphanumeric characters. This id can be used to reference task information from the support framework.

description
Enter label text to display on the task selection panel of the deployment wizard.

instructions
Enter the text for the instructions that are displayed in the deployment wizard that prompt a user to perform an action.

isOptional
Set to `true` to make the selected task optional.

selectedByDefault
Set to true to deploy the selected task by default.

Note: If the `isOptional` attribute is set to false, `selectedByDefault` is returned as **true**.

taskDetails
Enter a text description to display a brief description of the task on the task selection panel of the deployment wizard.

Adding task groups to a solution

A task group is used to group related tasks. The deployment wizard presents the end user with a list of all the task groups in the solution. Users select the tasks to deploy.

To add a task group to a solution, follow these steps:

1. Select the **Tasks** tab in the right pane of solution editor.
2. Click **Add**.
3. Select **Create an empty task group**. Click **Next**.
4. Enter a **Task group title**.
5. Enter a **Task group prompt.** This prompt is displayed on the task selection panel in the deployment wizard.
6. Click **Finish**.

Use the Properties view to specify optional information about the task group. Right-click a task group in the editor and choose **Properties** from the context menu. Specify information for the following optional attributes in the corresponding Value field:

isOptional
Set to `true` to make the selected task group optional.

onlyOneTaskSelectable
Set to `true` to specify that the user can only select one task.

Note: Setting the `onlyOneTaskSelectable` attribute to true is not valid if multiple tasks have the `selectedByDefault` attribute set to true or the `isOptional` attribute set to false.

selectedByDefault
Set to `true` to specify that this task group is selected by default in the deployment wizard.

Note: If the `isOptional` attribute is set to false, `selectedByDefault` is set to `true`.

taskGroupDetails
This text is displayed along with the task group title on the task group selection panel in the deployment wizard.

taskGroupPrompt

This text is displayed at the top of the task selection panel for this task group in the deployment wizard.

taskGroupTitle

This text is displayed along with the task group details on the task group selection panel in the deployment wizard.

Reordering tasks within a solution

To reorder tasks in a solution, perform the following steps:

1. From the left pane of the Express Runtime developer, select the **Package Explorer** view.
2. Select the appropriate solution project file.
3. Open the src/ directory and open the *.sxml file.
4. Select the **Tasks** tab in the right pane of the Express Runtime developer.
5. Select the task you want to move.
6. Click **Up** to place the task earlier in the solution, or **Down** to place it later in the solution.

You can also drag and drop tasks from one group to another in the Navigator and Package Explorer views of the Solution Assembly Toolkit solution editor.

Providing variables validation and values information

To provide validation and value information for solution variables, perform the following steps:

1. Select the **Tasks** tab in the solution editor.
2. If there are shared solution variables to add to the validation list, select the variable in the Overridden Application Variables section and click **Add**.
3. If there are shared solution variables to delete from the validation list, select the variable in the Overridden Application Variables section and click **Remove**.
4. To add or modify validation information for a variable, select the variable in the Overridden Application Variables section.
5. In the Shared Variable Value Configuration section, you can specify the following information:

   Required

   If selected, indicates that a value must be entered for this variable. If no default value is provided, the end user needs to enter or select a value before deploying the application. Boolean variables are always required.

   Make uppercase

   If selected, transforms any value entered for the variable to be uppercase only.

   Make lowercase

   If selected, transforms any value entered for the variable to be lowercase only.

   Make editable at least once

   Specifies that the shared variable is displayed the first time the deployment wizard encounters it, so that the user can input a value at least one time. Use this setting for variables that might not be displayed based on other selections, but for which the user must specify a value.

Default value
Specifies the default value for a variable. Any variable specified here must comply with any other validation restrictions placed on this variable. For a drop-down list variable, the default must also be displayed in the Options field.

Minimum length
Specifies the minimum number of characters allowed for the value of the variable.

Maximum length
Specifies the maximum number of characters allowed for the value of the variable.

Validator
Browse to a custom program to validate the variable in the deployment wizard. To create a validator program, click on **Validator** and enter the name of your new validator program. Implement the validate() method of the new class to perform your custom validation by following the description in the Javadoc documentation of the validate() method.

The Java class specified as the validator must exist in the solution project. That class can refer to any other Java classes that exist in the solution project, the included application projects, or in the external support JAR files for the application projects.

If the validator refers to a class that exists in more than one referenced application project, the validator might use any of the projects' classes. For this reason, if you plan to use solution-level validators, which will refer to application-level Java classes, and your application projects contain different Java classes with the same name, consider package-qualifying the application-level Java classes with the application ID.

Note: If validation is not specified for a shared variable, the validation and default data from the first editable appearance of the source application variable in the deployment wizard configuration panels is used.

Options
Offers more restrictions or values for the variable. To specify values that are must be part of the value of the variable, create a valid validation rule. To specify values that are not allowed to be part of the value of a variable, create an invalid validation rule. To specify choices for a drop-down variable, create possible selections. You can specify one of the following pieces of information for each validation rule. Not all restrictions are available for all variable types.

Complete string
a sequence of characters that either must match or is not allowed to match the complete variable value

Partial string
a sequence of characters that either must match or is not allowed to match a sequence of characters in the variable value

Prefix a sequence of characters that either must occur or is not allowed to occur at the beginning of the variable value

Suffix a sequence of characters that either must occur or is not allowed to occur at the end of the variable value

Specific characters
a list of characters, where sequence does not matter, that either must occur or is not allowed to occur in the variable value

Range of values
a minimum and maximum numerical value that represents either a valid or invalid range of values for the variable

Drop-down list option
an exact value that a user can select from the drop-down list variable. Create separate options for each possible value in the drop-down list. You must define the default value in the Default value field and in the Options field.

6. Save the solution.

Providing task configuration information

The Task Configuration section contains the task group selection prompt and check box.

To specify task configuration information, follow these steps:

1. Select the **Tasks** tab in the right pane of solution editor.
2. Enter a **Task group selection prompt.** This prompt is displayed at the top of the task group selection panel in the deployment wizard.
3. Use the **Allow only one task group or task to be selected** check box to specify whether only one task group or top-level task can be selected.

   Note: Specifying that only one task group or top-level task can be selected is not valid if multiple task groups or top-level tasks are selected by default or if multiple task groups or top-level tasks are required.

Enabling a solution for globalization

There are many strings in application and solution wrappers that are displayed to end users in the deployment wizard interface. These strings can be specified directly in the wrapper files, or they can be placed in translatable XML files. To enable translation for an application or solution wrapper, perform the following steps:

1. Decide which translation languages your application or solution will support, and create the necessary translatable XML files, as described in the topic Translation Language Configuration
2. Populate the default-language XML file with all the translatable strings for the wrapper, giving each string a meaningful key name. If you have already created a wrapper, you can copy the translatable strings from your wrapper into the default-language XML file. Keys must consist of alphanumeric characters only (no white-space), and must begin with a letter. The following example shows what a default-language XML file for a simple solution might look like.

   ```
   <?xml version="1.0" ?>
   <SolutionID>
       <solutionTitle>Solution Title</solutionTitle>
       <task1Details>Details for task #1</task1Details>
       <task1Description>Description for task #1</task1Description>
   </SolutionID>
   ```
3. Using the application or solution editor, replace all the translatable strings with references to their keys in the default-language XML file.

If using the GUI editor, simply enter the key name, prefixed with a percent sign (%).

Following the example, the solution title field would be set to: `%solutionTitle`

If editing the XML source directly, use the `translatedKey` attribute. Following the example, the `<title>` element in the solution wrapper would be specified as:

```
<title translatedKey="solutionTitle" />
```

4. Place the translations of the strings in the default-language XML file in the appropriate language XML files. This step can be performed separately, if translations are not immediately available. If a string does not exist for a particular translation language, the string from the default-language XML file is used.
5. Generate the application or solution wrapper. Any missing translatable strings are detected, and the appropriate error messages are displayed in the console.

Appendix C. Exporting

You can export files and folders from the Express Runtime developer either by using the Eclipse Export wizard, or dragging or copying them to the file system.

To use the Eclipse Export wizard to export files or folders to the file system, perform the following steps:

1. Right-click the project that you want to export in the Navigator view.
2. Select **Export** from the context menu. The Eclipse Export wizard is displayed.
3. In the Export wizard, select **File system**.
4. Click **Next**.

To use the operating system's file system explorer to export a copy of a folder or file from the Express Runtime developer to the file system, perform the following steps:

1. Drag the files or folders from the Navigator view to the file system explorer.
2. Hold down the Ctrl key while dragging to ensure the file is copied.
3. You can also export by selecting the file in the Navigator and choosing **Edit > Copy**, then pasting it in the file system explorer.

To export an application or solution project's source files, use the standard Eclipse file system export wizard (**File > Export File System**).

Exporting solution projects

A solution project has two distinct characteristics. One characteristic is that of a solution definition consisting of various source files (such as XML wrappers and Java files). The other characteristic is the deployable parts of a solution (such as user programs packages and the solution .ser file) that are generated when you build the solution.

The Express Runtime developer provides a wizard to export deployable solution parts. To invoke the export wizard, select a solution project in the Navigator or Package Explorer view and right-click it to display the context menu.

- Select **Export > Solution Assembly Toolkit Solution**.
- Select the solution to export
- Enter the destination directory

When this export wizard is run, the binary solution file and the user program packages are exported to the selected directory.

Exporting to CD or DVD images

The Express Runtime developer provides an export wizard that you can use to export a solution project to one or more CDs or DVDs for distribution. The CDs or DVDs contain a complete image of the solution that you have created, and when end users insert the media, the deployment wizard is displayed and enables them to deploy the solution. When the solution deployment is complete, the deployment wizard is uninstalled. Media that contains the combination of solutions and the deployment wizard are referred to as solution launcher images.

The solution launcher image is an easy way to distribute and deploy a solution. The solution you export becomes a solution launcher image that can then be deployed by using the deployment wizard. The deployment wizard is installed by an installation program that is packaged with your solution launcher image. The deployment wizard can then be used to deploy the solution to either the computer on which the deployment wizard is installed or to remote target computers.

There is also the option to silently install the deployment wizard and silently deploy the solution contained in the solution launcher image. For instructions for silent installation, see "“Silent installation of a solution launcher image” on page 121."

The solution launcher image you export can display one or more license agreements as part of an installation. If you want to include licenses, place all the license files in the license directory of the solution project, in the appropriate language folder. If you have translated license files, place them in the other language folders. For the solution launcher to detect translated license files correctly, the filenames of the different translations should be placed in language specific subfolders. In the following example, if the default language is English, the translated versions of license1.txt are displayed for Spanish and German computers, and the English version of license1.txt is displayed for all other languages. Additionally, the English version of license2.txt is displayed for all languages, since no translations were provided.

```
Solution/
 license/
  en/
   license1.txt
   license2.txt
  es/
   license1.txt
  de/
   license1.txt
```

The solution launcher image can also include a launch pad with links to a readme file and documentation for the solution.

If you choose to include either a readme file or documentation, you must include the files in the solution. The readme file must be placed in the `readme` folder in the solution, and the documentation files must go in the `info` folder in the solution. The files must be placed in the appropriate language folder. Place any translated files in the appropriate language folders.

Solution launcher export: Step 1 of 5

To export a deployable solution as a solution launcher image, complete the following steps:

1. From the left pane of the Express Runtime Developer, select the **Express Runtime Explorer** view from the top pane, or the **Navigator** or **Package Explorer** view from the bottom pane.
2. Select the appropriate solution project. You might have to expand the solution category to select the solution.
3. Click **File > Export** from the main menu or from the pop-up menu.
4. Select **Express Runtime > Express Runtime Solution Launcher Image** from the list of available wizards. This selection is under the Express Runtime folder.
5. Click **Next**.

6. Select the solution project that you want to export.
7. In the **Media size (MB)** field, enter the capacity, in MB, of the media that the solution launcher image will be distributed on. You can enter a value or select one from the list.
8. Use the **Include files from** field to include external files in the image. For example, you might want to include documentation, images, or other files that accompany the image. Enter the path, or click **Browse** to select the root directory of the files to include.
9. In the **Export files to** field, enter a destination directory or **Browse** to an export destination directory.
10. In the **Display language** field, select the language in which you want the deployment wizard panels to be displayed. If you want the installation to detect and use the language for the default locale on the computer running the solution launcher, select **Detect automatically**.
11. Enter the **Installation directory** for the solution launcher logs and solution files that contain the configuration information for the solution. The installation uses the directory name SolutionFiles by default. For Windows this is `<`*`Drive where temp directory is located`*`>\<`*`directory specified in export wizard`*`>` and on Linux it is `/<`*`directory specified in export wizard`*`>`.
12. Select the operating systems on which the solution launcher will run.
13. Select the **Run installation silently** field to include a task file in the image. A task file is required to install the solution launcher image silently. Use the **Task file** field to enter the path to the task file. For information on creating a task file, refer to the “Creating a task file for silent deployment of a solution launcher image” on page 121 topic of the Solution Assembly Toolkit information center.
14. Click **Next**.

Note: The deployment packages are located in the launcher CD in the bin\com\ibm\jsdt\webserver\tree folder.

Solution launcher export: Step 2 of 5

1. In the Deployment package export type field, specify how the export wizard should export deployment packages. You can select one of the following options:

   Export all deployment packages
   This option exports all of the deployment packages for all of the applications that are included as a part of the solution. All deployment packages must be generated to use this option.

   Export available deployment packages
   This option exports all of the generated deployment packages for the applications that are included as part of the solution. If an application does not have a generated deployment package, the end user is prompted to provide the required source disks at deployment time. For information about how to customize disk prompts, refer to the topic “Adding applications to solutions” on page 105.

   Do not export any deployment packages
   This option prompts the end user to provide the location of all of the deployment packages that are included as part of the solution at deployment time. For information about how to customize disk prompts, refer to the topic “Adding applications to solutions” on page 105.

Customize which deployment packages to export

You can use this option to select which deployment packages to export as part of the solution. You can only export generated deployment packages. If a deployment package is not generated, or you select not to export a deployment package, you can provide prompts for the end user to supply the required source disks. For information about how to customize disk prompts, refer to the topic “Adding applications to solutions” on page 105.

2. If the type of deployment package export type you select requires you to select individual deployment packages for export, you can select them in the **Deployment packages to export** field.
3. Click **Next**.

Solution launcher export: Step 3 of 5

1. In the **Default language** field, select the default language for the solution on the target computer. The default language must be selected in the **Languages** list.
2. In the **Languages** list, select the languages for which you provide translated license, readme, or documentation files so that if the installation detects one of those languages as the locale of the target computer, the appropriate translated documentation will be displayed. If the locale is not one of the specified languages, or if you did not provide the translated file, the file will be displayed in the specified default language.
3. Click **Next**.

Note: The deployment packages are located in the launcher CD in the bin\com\ibm\jsdt\webserver\tree folder.

Solution launcher export: Step 4 of 5

1. Use the **Install license files** options to specify the license files included with the image. If you select **Text/HTML files** as the license type, you must choose at least one file from the **Text/HTML files** field. Ensure each license file is encoded properly. A license file that is not encoded properly is not displayed correctly during installation. To ensure that a license file is encoded properly, refer to the list of supported encodings. A link to this information is provided in the next step.
2. For a list of the supported encoding, see ″Supported Encodings″ on the Java Web site.
3. Click **Next**.

Solution launcher export: Step 5 of 5

1. Use the **Install readme file** options to include a readme file in the image.
2. Use the **Install documentation files** options to include product documentation with the image.
3. Use the **Display launch pad as part of solution launcher installation** to include the launch pad and its options with the image. The banner image is 800 x 66 pixels and must be in .gif or .jpg format. The logo image is 100 x 66 pixels and must be in .gif or .jpg format. The icon image is 16 x 16 pixels and must be in .ico format. If you do not provide custom images that match the default dimensions, the appearance of the launch pad is not displayed correctly. If you do not provide custom images, the launch pad uses the default images. If you provide incorrect paths to images, or image types other then .ico, .gif, or .jpg,

the export wizard displays an error and requires you to provide a valid image and path. This option is only available if you have readme and documentation files in the solution project.

4. Specify whether you want the launch pad to link to the readme or documentation. If you include a readme file and product documentation, but do not include the launch pad or do not include a link to the documentation and readme file from the launch pad, you must provide another way for customers to access the readme file and product documentation. No shortcuts to the documentation or readme file are included in the solution launcher image. This option is only available if you have readme and documentation files in the solution project.
5. Click **Finish**.

Silent installation of a solution launcher image

You can perform a silent installation of the solutions provided by Solution Assembly Toolkit as solution launcher images. The solution launcher export wizard creates a response file for a silent installation named IRU_install.iss.

To complete a silent installation of a solution launcher image, you need a task file. A task file is an .xml file that stores information about the tasks that are part of a solution, and target computers that each task in a solution is deployed to. By default, the task file is named *solution name*_Task.xml where *solution name* corresponds to the name of the solution that the task file is associated with. By using a task file, you can specify tasks to deploy, target computers, and other information, such as the privileged user ID and password for each computer to which the silent deployment is targeted.

When you have created a task file, you can initiate a silent installation.

To initiate a silent installation, perform the following steps:

1. Copy the contents of the disk1 folder, and the bin subdirectory of all folders for disk2 and above from the output folder where the solution launcher image resides, to the computer that you are deploying from.
2. Create the disk1, disk2, etc. structure on the machine and copy the contents of each disk to the corresponding disk folder.
3. Navigate to the disk1 folder, and double-click the <*OS*>Setup executable program for the operating system you use. Installation progresses without any panels being displayed.
4. View log files to confirm a successful installation. The log files are in the location that is specified in the User files default location section of IRU_install.iss.

Creating a task file for silent deployment of a solution launcher image

To silently deploy a solution launcher image, you must first create a task file for the solution, then export the solution to the solution launcher image for silent deployment.

To create a task file, perform the following steps:

1. In the Express Runtime developer, expand the solution and right-click the tasks folder. Select **New > File**.
2. In the New File dialog, ensure the tasks folder is displayed in the Enter or select the parent folder field and type in the task file name you want to create (for example, MyTaskFile.xml). Click **Finish**.

3. In the Solution Assembly Toolkit installation CD1 or DVD1, locate the sample task file in \IRU2_2MiddlewareAll\tasks\IRU2_2MiddlewareAll_Task.xml. You can use this file as a template.
4. Copy the header and taskSet tags and paste them into the task file.
5. Update the fields in your task.xml file as follows:
   - Search for the <targetHostname> </targetHostname> tag and specify the target computer name here.
   - If the target computer is not localhost, search for the <add Credentials userId=″user″ password=″password″ hostname=″hostname″> element nested within a <credentialsSet> block and provide the privileged user ID and password and target computer name.
   - Search for <taskSet solutionFileName=″SolutionFileLocation/ IRU2_2MiddlewareAll.ser″>tag. Replace the ″IRU2_2MiddelwareAll.ser″ with your solution file name, followed by ″.ser″. For example, if the solution name is ″MySolution″, the value will be ″MySolution.ser″. Note: Do not remove the SolutionFileLocation tag.
   - Search for <deploy taskNumber=″″> tag. Fill in value for ″taskNumber″. The taskNumber should map to the sequence number of the same task in your solution.sxml.
   - Search for <application id=″″> tag. Fill in the value for ″id″. Use the Application ID from the Application.axml file.
   - Search for <variable id=″″> For each required variable for the application, fill in the variable ID and the variable value. You can also specify optional variables.

Sample task file

The following sample task file shows which tags to update.

```
<iru:tasks
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"Â
  xmlns:iru="http://www.ibm.com/xmlns/prod/iru/task"
  xsi:schemaLocation="http://www.ibm.com/xmlns/prod/iru/task DJT_task.xsd"Â
  silent="false">
  <credentialsSet>
       <addCredentials hostname="isiis03.rtp.raleigh.ibm.com" userId="administrator" password="password"/>
       <addCredentials hostname="rtpi153a.rtp.raleigh.ibm.com" userId="administrator" password="password"/>
  </credentialsSet>
  <taskSet solutionFileName="SolutionFileLocation/TestSuite.ser"
   logFileName="IRU_DeploymentWizard.log">
  <createDeploymentPackage applicationId="sampleApplication"
   language="english"
   softwareImageRoot="D:\sandbox\testsuite">
  </createDeploymentPackage>
 <deploy taskNumber="12" />
</taskSet>
</iru:tasks>
```

Appendix D. Using the deployment wizard

The deployment wizard provides a simple graphical user interface for deploying a solution (.ser file) to target computers that you specify.

Navigate the steps of the deployment wizard by completing them in order and clicking **Back** and **Next** to switch to the previous or next step.

If no solution is open in the deployment wizard, or the next button is not enabled, you can open a solution to begin the deployment process.

To open a solution, perform the following steps:

1. Open the deployment wizard. Select **Start > Programs > IBM Solution Assembly Toolkit > Deployment Wizard** on computers running Windows operating systems, and **Main Menu > Programs > IBM Solution Assembly Toolkit > Deployment Wizard** on computers running Linux operating systems.
2. Click **File > Open**.
3. Click the solution with the .ser file extension that you want to deploy and click **Open**. The deployment wizard Welcome dialog is displayed.
4. Click **Next**.

Note: If you are deploying from a solution launcher image, the default solution automatically opens in the deployment wizard.

Selecting tasks to deploy

The deployment wizard provides a list of all of the tasks that you can deploy. A task is a set of actions that are performed together. Select any combination of tasks listed in the wizard for deployment.

To select tasks for deployment, perform the following steps:

1. Select at least one task from the list that is displayed in the deployment wizard, by selecting the corresponding check box.
2. Click **Next**.
3. Tasks might contain one or more sets of subtasks. Select at least one subtask for each task.
4. Click **Next**.

Specifying target computers

Specify one or more target computers for each task that you select for deployment.

Specify one or more target computers for each task that you select for deployment. Before you deploy middleware to a target computer, ensure that installing middleware on that computer does not violate the terms of your license for the middleware.

To specify a target computer, you must specify three things:

- the fully-qualified domain name or the IP address of that computer.
- a user ID and password for that computer if it is not `localhost`.

- the operating system of the target computer.

A fully qualified domain name includes all higher level domain names up to the top-level domain name. Use the following guidelines to ensure the correct format of a domain name:

- `localhost` represents the local computer.
- An alphanumeric text string up to 24 characters in length, containing any of the letters A-Z, digits between 0-9, and the minus sign (-) and period (.) characters
- The first character must be an alphabetical.
- You can use upper and lowercase letters.
- The last character cannot be a minus sign (-) or a period (.).
- Only use periods to delimit components of a domain name.
- Do not use blank or space characters.

Use the following guidelines to ensure the correct format for an IP Address:

- A 32-bit numeric address written containing four numbers.
- Each of these four numbers can range from 0 to 255.
- Each of the four numbers is separated by periods (.).

To specify a target computer, follow these steps:

Note: If you are using one of the predefined solutions, you can only specify a single host computer. The parts of these instructions related to specifying multiple computers are not needed.

1. In the Specify Target dialog, follow these steps:
   a. In the **Target Computer** field, specify either `localhost`, or the fully-qualified domain name or the IP address of the target computer.

      Note: When you enter the IP address or hostname, the format is checked to ensure the entry is valid. In addition to verifying the correct format, a check is made for duplicate entries.

   b. In the **User ID** field, specify the user name to use when connecting to this target computer. This is only valid if the target computer is not `localhost`.
   c. In the **Password** field, specify the password to use when connecting to this target computer. This is only valid if the target computer is not `localhost`.
   d. In the **Operating system** drop-down, select the operating system of the target computer. This is only valid if the target computer is not `localhost`.
   e. If you want to save the User ID, Password, and operating system information that you specified for this target computer, click the **Save this login information** check box. If you have a need to repeatedly deploy software to a given target, saving the login information prevents you from reentering the information between sessions. Based on the target name, the ID, password and OS are saved locally. The password is stored encrypted in the local key store.
2. Click **Test connections** to verify that the deployment wizard can connect to the specified target computer. The Target Computer Data window is displayed and shows the status for this target computer. If the deployment wizard cannot connect to a target computer, verify that the fully-qualified domain name or IP address is correct. If this information is correct, contact the computer owner to determine if the target computer is running and connected to a network. If a firewall exists between the staging server and the target computer, ensure that the firewall permits communication on the specified port. If no firewall is

running, you should use the default port 0. Using port 0 causes the deployment wizard to allocate an open port. You should only use ports other then 0 if a firewall is in use. Ensure that communication is open from both sides of the firewall.

3. Click **Next**.

Target Computer Data panel

The Target Computer Data panel displays information for all target computers associated with a task, and indicates whether the deployment wizard can connect to the target computer. A green check mark to the left of the Target column indicates the deployment wizard can connect to the IBM Installation Agent on the target computer. A black question mark to the left of the Target column indicates that the deployment wizard is collecting data or was unable to determine the operating system of the target computer because the target computer is running a down-level version of the Installation Agent. A red circle with an X to the left of the Target column indicates the deployment wizard cannot connect to the Installation Agent on the target computer. Possible reasons for failure to connect include:

- The computer does not exist on the network.
- The computer is not running the IBM Installation Agent.
-

See the *Details* column for more information on the target computer's status.

Configuring deployment parameters

The deployment parameters dialog contains the deployment parameters for an application associated with a selected task. Use deployment parameters to configure an installation during deployment. Examples of deployment parameters include user IDs, passwords, and target directories.

In many cases, the deployment parameters can have default values. You can use or modify the default values. The values are shared with all target computers associated with a task.

Complete all the fields on the deployment parameters dialog denoted with an asterisk. Provide deployment parameters for every task in the solution selected for deployment, and click **Next**.

Modify parameters listed on the **Typical** tab only. The parameters on the **Advanced** tab are for advanced users only. Do not modify advanced parameters unless you understand the potential results of changing the parameter.

When the deployment wizard encounters an error with variable validation, and additional information is available for the validation of that variable, a link is displayed. To learn more about the validation rules for a specific variable, and the input that is required, you can click the link below the variable.

Deploying tasks to target computers

The summary dialog of the deployment wizard displays a summary of the tasks that have been selected to deploy and the target computers selected for deployment. An estimate of the time to install each task is provided, along with the summary of the task. This dialog also shows the status of the last attempted deployment for each task.

There are two methods for deploying tasks. Click **Deploy Task** to deploy each task individually. Click **Deploy All** to deploy all of the tasks that are displayed on the summary dialog sequentially.

If the files required for deployment span more than one disk or cannot be located, a dialog prompts you for the correct location of each disk or deployment package JAR file that cannot be found. A dialog prompts you for each item needed to complete the deployment.

Specify location

The Specify Location panel is displayed when you attempt to deploy a task and the deployment wizard either cannot locate the deployment package, or cannot locate the files needed to create the deployment package.

When the deployment wizard cannot locate the deployment package in the deployment package path, it attempts to locate the source software image files from the root installation directory. If the deployment wizard can locate the files, it automatically creates the deployment package using these files. If the deployment wizard cannot locate the files, the Specify Location panel is displayed for you to create the deployment package or locate a pre-built deployment package.

To create a deployment package during deployment, enter the location of the indicated disk. If multiple disks are listed, locate one of them to create the deployment package. Either enter the directory at the prompt, or click **Browse** to locate the directory.

Select disk

If you need to build a deployment package during deployment, you are prompted to locate the disk containing the required source files. If multiple disks are listed, you can locate one of them to create the deployment package. If the location you specify matches more than one of the listed disks, you might be prompted to identify the disk you located.

Viewing deployment status

The status dialog of the deployment wizard provides detailed information on the deployment progress. It contains deployment status and messages that explain the reason for any errors that occurred during deployment. To view a list of all the messages that are displayed during deployment, click **Detailed Messages**. Detailed messages provides the name of the solution, time, and the target computer for the deployment of each solution.

To view a log of the tasks that were performed during deployment, click **Master Log**. The Master log provides a list of every action that occurred as part of deployment. You might need to refer to this information when debugging unsuccessful deployments.

Deployment options

This topic contains information regarding the options you can provide to the deployment wizard when starting a deployment from the command line.

Deployment wizard options

You can provide specific deployment options when you start the deployment wizard from the command line.

To use deployment options when you start the deployment wizard from the command line, you can apply the following arguments:

- **-leaveFiles** - Leaves the files on the local computer.
- **-remoteLeaveFiles** - Leaves the files on the remote computer.
- **-deploymentPackagePath** - The location on the local computer where the deployment package is created.
- **-skipStartupCheckpoint** - This option will bypass the execution of the solution's startup checkpoint.
- **-skipWelcomePanel** - Skips welcome panel.
- **-enableSolutionDeployerTrace** - Enables the deployment wizard trace.
- **-enableSupportFrameworkTrace** - Enables the support framework trace.
- **-solutionDeployerTraceFile** - Sets the deployment wizard trace file.
- **-supportFrameworkTraceFile** - Sets the support framework log file.
- **-maxSolutionDeployerTraceFile** - Sets the maximum size of the deployment trace log file.
- **-maxSupportFrameworkTraceFile** - Sets the maximum size of the support framework log file.

The following command provides an example of a command line invocation of a solution that uses all of the available arguments:

```
DJTJRE\bin\java -jar DJT_ibmnsit.jar -leaveFiles -remoteLeaveFiles
-deploymentPackagePath -skipStartupCheckpoint -skipWelcomePanel
-enableSolutionDeployerTrace -enableSupportFrameworkTrace
-solutionDeployerTraceFile deployer.log -supportFrameworkTraceFile
framework.log -maxSolutionDeployerTraceFile 10
-maxSupportFrameworkTraceFile 10
```

Logging options

You can use specific logging options in the deployment wizard to generate log files that provide information that is useful to have when resolving problems that you might encounter during deployment. You can enable logging to provide diagnostic information that is related to the support framework, the deployment wizard, or both. Support framework diagnostic information (to display user-defined messages that are created during solution development), and deployment wizard diagnostic information that might be helpful to debug a deployment is logged in separate files. You can specify the name that you want to use for each log file, and the maximum amount of space that each log file can physically occupy.

Use diagnostic logging only when it is necessary for troubleshooting a deployment.

To use deployment logging, perform the following steps:

1. From the deployment wizard select **Edit > Preferences**.
2. Click **Target Computer Settings**.
3. In the Diagnostic Trace dialog, select the check box that corresponds to the type of diagnostic information you want logged. You can select support framework information, deployment wizard information, or both.
4. Provide a name for the file to which diagnostic information is recorded.
5. Provide a value for the maximum amount of space that each log file can physically occupy.
6. From the Diagnostic Trace dialog, click **OK**.
7. From the Deployment Preference dialog, click **OK**.

Do not remove setup files when deployment is complete must be selected for log files to remain after deployment is complete on the target computer. If you do not select **Do not remove setup files when deployment is complete** the log files are not saved after the deployment completes. The **Do not remove setup files** setting applies to the specified target computer. The log files are useful when debugging deployments.

The log files that the deployment wizard generates are saved in the following locations:

- `*Installation_Directory*/SolutionEnabler/logs/`
- `*Installation_Directory*/IIA/logs/`

You can also use deployment diagnostic logging when you start the deployment wizard from the command line.

To use deployment diagnostic logging when you start the deployment wizard from the command line, you can apply the following arguments:

- **-enableSolutionDeployerTrace** - Enables the deployment wizard diagnostic logging.
- **-solutionDeployerTraceFile** - Sets the name of the deployment wizard log file.
- **-maxSolutionDeployerTraceFileSize** - Sets the maximum size of the deployment wizard log file.
- **-enableSupportFrameworkTrace** - Enables the support framework diagnostic logging.
- **-supportFrameworkTraceFile** - Sets the name of the support framework log file.
- **-maxSupportFrameworkTraceFileSize** - Sets the maximum size of the support framework log file.

The following command provides an example of a command line invocation of a solution that uses all of the available arguments to perform diagnostic values:

```
DJTJRE\bin\java -jar DJT_ibmnsit.jar -enableSolutionDeployerTrace
-solutionDeployerTraceFile myTrace.log -maxSolutionDeployerTraceFileSize 10
-enableSupportFrameworkTrace -supportFrameworkTraceFile mySupportTrace.log
-maxSupportFrameworkTraceFileSize 2
```

Note: If you enable the support framework trace and perform a remote installation, the support framework trace information is not logged.

Appendix E. Debugging and problem determination

If you encounter errors and problems while using Solution Assembly Toolkit, there are a number of debugging and problem determination options available to you.

Using log files

Solution Assembly Toolkit creates several log files. You can use these log files to determine the cause of a problem. These files contain information that you might need to supply to the IBM Support Center if there is a problem with the Solution Assembly Toolkit product itself. There are three phases during which Solution Assembly Toolkit can generate these logs: installation, development, and deployment.

The Solution Assembly Toolkit logs are located in the <install dir>\Runtime\2.2\SolutionEnabler\logs folder, including those created during installation. Logs that are generated at deployment time are placed in the /deployment/logs folder on the target computer.

Installation log files

The Solution Assembly Toolkit installation is called from the WindowsSetup program on Windows, the LinuxSetup program on Linux, or the LinuxPPCSetup program on Linux on IBM POWER. You can run these programs directly.

The launch pad starts automatically on Windows computers when you load the installation CD. If it does not start automatically, you can start it by using the launchpad.exe program on Windows or the launchpad.sh program on Linux or Linux on POWER. Once the installation CD loads, the launch pad starts.

The installation process creates a log named IRU_Install.log, which is contained in the <ER installation directory>/Runtime22/SolutionEnabler/logs folder. The installation generally completes successfully, however, if an error does occur during installation, you can consult this log file.

If you attempted a silent installation and do not see a log file generated, be sure that you have administrative privileges. Only users with administrative authority can install IBM Solution Assembly Toolkit on Windows or Linux operating systems. During a silent installation, most errors are written to a log file; however, log messages cannot be logged for non-administrative users.

Development log files

With Solution Assembly Toolkit installed, use the Solution Assembly Toolkit development environment to create and build your application wrappers and your solution wrappers. The Solution Assembly Toolkit development environment, in turn, uses three tools to build a solution:

- The application generator compiles the application wrapper and builds manifest files.
- The solution generator compiles the solution wrapper.
- The deployment package generator creates the user JAR files.

Each of the above tools also creates a log file. The tools place these log files, listed in table 2, into the workspace associated with the Express Runtime developer workbench. The location of the application generator log is dependent on whether you are building an application or a solution:

Table 2. Log files created at development time.

Log file name	Location	Description
applicationBuilder.log	When you are building an application: `<application project>\bin\<application id>\log`	The application generator writes status and error information to this log file as it creates the application .ser file. In addition, Extensible Markup Language (XML) parser information about the application wrapper is written to this file.
solutionBuilder.log	When you are building a solution: `<solution project>\bin \log`	The solution generator writes status and error information to this log file as it creates the solution .ser file.
deploymentPackageBuilder.log	`<solution project>\bin\<application id>\log` or `<application project>\bin\<application id>\log`	The deployment package generator writes status and error information to this log file as it creates the user .jar file.

Deployment log files and using a trace

There are several types of files that can aid your debugging effort during the deployment phase:

- Log files
- User program and media files
- Response files
- Property file

In general, the location of these files depends on whether the deployment is remote or local.

Performing the deployment

To perform a deployment, use the deployment wizard, which is a Solution Assembly Toolkit tool. The deployment wizard must be running on the source computer (also called the staging server) and have network connectivity to the target computer. To perform a deployment on a local computer, network connectivity is not required.

Keeping temporary deployment files

Some of the files that are used during deployment are temporary files. If you want to prevent Solution Assembly Toolkit from deleting temporary files after a deployment, follow these steps:

1. Select the **Edit > Preferences** menu option on the deployment wizard.
2. On the Deployment Preferences dialog, click **Target Computer Settings**.
3. On the Target Computer Settings dialog, select the target computer from the drop-down list.
4. Select the **Do not remove setup files when deployment is complete** checkbox.

If the **Do not remove setup files when deployment is complete** checkbox is selected, Solution Assembly Toolkit does not delete the media and user files after deploying an application. Instead, these files remain in the unpacked directory. These files can be useful for ensuring that Solution Assembly Toolkit transferred the user and media JAR files to the target computer correctly. For remote and local deployments, the unpacked directory in located in the <ER installation directory>/iru directory. If there is already a directory with this name on the target computer, a numerical suffix is added to create a unique name for the unpacked directory (for example, <ER installation directory>/iru1).

To determine what values were specified for those configuration parameters that have `fileAssociations` XML entries associated with them, examine the contents of response files. The names are controlled by the application wrappers.

Similarly, to determine what values were specified for those configuration parameters that have `propertiesAssociations` XML entries associated with them, you can examine the contents of the property file, ibmnsi.properties. To find the response files and the property files, look in the following directories:

- For remote deployments, these files are located in the <IIA installation directory>/IIA/deployment/logs
- For local deployments, these files are located in the <ER installation directory>/Runtime22/SolutionEnabler/deployment/logs

Deployment logging and requesting a trace

You can use specific logging options in the deployment wizard to generate log files that provide information that is useful to have when resolving problems that you might encounter during deployment. You can enable logging to provide diagnostic information that is related to the support framework, the deployment wizard, or both. Support framework diagnostic information (to display user-defined messages that are created during solution development), and deployment wizard diagnostic information that might be helpful to debug a deployment is logged in separate files. You can specify the name that you want to use for each log file and the maximum amount of space that each log file can physically occupy.

Logging diagnostic information is CPU intensive, and therefore slows system performance. Use diagnostic logging only when it is necessary for troubleshooting a deployment.

To learn more about these logging options and how to request a trace, refer to **Deploying solutions to target computers > Deployment logging options** in this information center.

Deployment logs

During deployment, there are four types of log files:

Solution Assembly Toolkit

The file is named `IRU_DeploymentWizard.log`.

The file is located in <ER install dir>/Runtime22/SolutionEnabler/logs.

This log file contains messages and exceptions that are stored on the staging server (for both local deployment and remote deployment).

Target computer

The file is named `IRU_IIATrace.log`.

The file is located in <ER install dir>/Runtime22/SolutionEnabler/logs if you are performing a local deployment.

The file is located in <IIA install dir>/IIA/deployment/logs if you are performing a remote deployment.

This log file allows you to add print instructions to your user programs, which can aid you when debugging a problem. Data that you write to either the System.out or System.err file is displayed in this log. In addition, exceptions that occur while a user program is running also is displayed in this log.

NSI The file is named `ibmnsi.log`.

The file is located in <ER install dir>/Runtime22/SolutionEnabler/deployment/logs if you are performing a local deployment.

The file is located in <IIA install dir>/IIA/deployment/logs if you are performing a remote deployment.

This log file can be used for debugging user programs. It contains the command line call of the most recent user program. You can run this command string from a command prompt to start the user program directly without having to go through the deployment wizard. In this way, you can run your user program with all of the scaffolding that would be there if your user program had been started by the normal process.

Application

Different types of user programs each create a log file.

The file is located in <ER install dir>/Runtime22/SolutionEnabler/deployment/logs if you are performing a local deployment.

The file is located in <IIA install dir>/IIA/deployment/logs if you are performing a remote deployment.

The creation of application logs is controlled by the application wrapper and the user programs.
Each of the different types of user programs can create a log file, if desired. The `logFile` attribute in the application wrapper controls the name of the log file. This attribute can be displayed in each of the following XML elements:

- `<preDeploymentChecker>`
- `<entryProgram>`

- `<mainProgram>`
- `<exitProgram>`

These log files are controlled by user programs. Using these logs, you can gauge the progress of your programs. You determine what data should be logged in the application logs.
Recommendation: Consider data such as logging exceptions, return codes, parameters, calls, progress, and so on, in order to determine what information should be logged to allow you to debug errors.

Launch pad log files

When a launch pad is running, the generated launch pad messages are captured in a hidden log frame. To display the log frame on the bottom of the launch pad panels during runtime, hold CTRL and click in the banner frame of the launch pad.

Messages that are generated while the launch pad is running are not automatically saved on the hard drive. You can manually save the messages from a runtime session by clicking **Save** at the bottom of the log frame and specifying where you want to save the file.

Debugging Eclipse-specific errors and exceptions

If you encounter Eclipse-specific errors when you use IBM Solution Assembly Toolkit, you can review the Eclipse log files for more information.

If you installed the Eclipse environment using Solution Assembly Toolkit, review the log files in the following locations:

- <*installationDirectory*>\SolutionEnabler\workspace\.metadata\.log
- <*installationDirectory*>\SolutionEnabler\WindowsSampleWorkspace\.metadata\.log
- <*installationDirectory*>\SolutionEnabler\LinuxSampleWorkspace\.metadata\.log

where <*installationDirectory*> is the location to which you installed Solution Assembly Toolkit.

If you connected to an existing instance of Eclipse when you installed Solution Assembly Toolkit, review the log file at <*installationDirectory*>/<*workspace name*>/.metadata/.log, where <*installationDirectory*> is the location to which you installed Solution Assembly Toolkit and <*workspace name*> is the name of the existing Eclipse instance to which you connected.

Debugging a solution during deployment

You can debug Java user programs in the Express Runtime developer while you deploy them to local and remote computers using the deployment wizard.

To debug a Java user program during deployment, perform the following steps:

1. Start Express Runtime developer on the staging server. Ensure the application project containing the Java user program is open in the Express Runtime developer workspace.
2. Open the Java user program source file you want to debug, and insert a breakpoint where you want to pause during execution.

3. Start the deployment wizard by selecting a solution that contains the application you want to debug. Right-click the solution, and select **Test in deployment wizard** from the context menu.
4. Deploy the Java user program to a local or remote Linux, Windows, or i5/OS computer.

If you set a breakpoint on a line that is executed, you have the option to switch to the debug perspective. Click **Yes**. You can debug the user program as you would a normal Java program.

Troubleshooting debugging

If a message is displayed stating that debugging cannot be enabled because port 1099 is in use when you start the deployment wizard from the Express Runtime developer, you must free port 1099. When you free port 1099, restart the deployment wizard from the Express Runtime developer to debug user programs.

If your user program does not suspend execution during deployment, enabling debugging, verify that the following conditions exist:

- You set a breakpoint in the correct Java file, and on a line that is executed.
- There is an application project in the Express Runtime developer workspace for the user program you want to deploy.
- The type of the user program is Java.
- There is no firewall between the deployment server and the target computer, or if there is, that port 1099 is open.
- Port 1099 is not in use by any other applications on the server.

If a message is displayed during deployment stating that the debugger cannot connect, ensure that there is no firewall between the deployment server and the target computer, or if there is, that port 8000-8004 is open.

Appendix F. Installing the IBM Solution Assembly Toolkit from a disk or network location

An explanation of the IBM Solution Assembly Toolkit installation

When you start the installation process of the IBM Solution Assembly Toolkit, the setup files are automatically unpacked. This can take anywhere from a few seconds to several minutes, depending on computer speed, hard disk condition (how fragmented it is), and disk access speed. Because laptop computers have slower disk access, they might require several minutes to complete this process. You can see disk and CPU activity during this process. Wait until the setup files are unpacked.

If you are installing over an existing version of the IBM Solution Assembly Toolkit, you can only install features that you did not select in a previous installation. For information on installing a new version of the IBM Solution Assembly Toolkit over an existing version, refer to “Migrating to a newer version of the IBM Solution Assembly Toolkit” on page 143.

Only users with administrative user access can install the IBM Solution Assembly Toolkit. Check that the logged-on user has administrative access to the computer prior to installing.

To install the IBM Solution Assembly Toolkit, follow these steps:

1. Start the launch pad application. The launch pad is an application from which you can initiate the installation of the IBM Solution Assembly Toolkit. On Windows computers, the launch pad starts automatically from the installation CD or DVD. If it does not start automatically, you can start it by using the launchpad.exe program. To start the launch pad on Linux or Linux on POWER computers, use the launchpad.sh program. By default, the launchpad programs are on the first CD or DVD in the installation media.
2. Click the launch pad link to interactively install the IBM Solution Assembly Toolkit. Click **Install now** to start the installation.
3. The Welcome dialog displays the name of the product to be installed, the IBM Solution Assembly Toolkit. Read the dialog text and click **Next**.
4. The License Agreement dialog is displayed. To install the IBM Solution Assembly Toolkit, you must accept the terms of the license agreement. Read the license agreement information. Select **I accept both the IBM and the non- IBM terms** and click **Next**. Otherwise, select **I do not accept the terms of the license agreement** and click **Cancel** to exit the installation.
5. The installation directory dialog prompts for a destination to install the IBM Solution Assembly Toolkit. This dialog is not displayed if the IBM Solution Assembly Toolkit is already installed, for example, if you want to install components that you did not install originally. Either accept the default destination or select a new destination. Additional subdirectories are created under the destination directory. You can either manually enter a directory or click **Browse** to navigate to a directory where you want to install the IBM Solution Assembly Toolkit. Click **Next**.
6. Select an Eclipse environment. You can install the Express Runtime developer plug-in into an existing installation of IBM® WebSphere Application Server

Toolkit or Rational® Application Developer, or you can select to install the Eclipse environment provided with the IBM Solution Assembly Toolkit.

7. The summary dialog summarizes installation information such as the installation directory, the features to be installed, and the estimated total size of the features that you selected to install. Review this installation summary before beginning the installation. To make changes, click **Back** to return to the previous dialogs and make any necessary changes. Click **Next** to begin the installation.
8. The final installation dialog indicates whether the installation was successful. You can choose to start First Steps for the IBM Solution Assembly Toolkit. First Steps provides links to information and functions to help familiarize you with the IBM Solution Assembly Toolkit and its capabilities. If you want to start First Steps, click **Launch First Steps**. Click **Finish** to exit the installation wizard.

IBM Solution Assembly Toolkit system requirements

This topic lists the hardware and software requirements for installing and using the IBM Solution Assembly Toolkit.

Operating system support

The following table shows which operating systems are supported:

Table 3. Operating system support

		Environment		
Platform	**Operating system**	**Development**	**Deployment Server**	**Deployment Target**
Windows	Windows XP Professional SP™ 2	Yes	Yes	Yes
	Windows Server 2003, Standard Edition SP 1	Yes	Yes	Yes
	Windows Server 2003, Enterprise Edition SP 1	Yes	Yes	Yes
	Windows Server 2003, Standard Edition R2	Yes	Yes	Yes
	Windows Server 2003, Enterprise Edition R2	Yes	Yes	Yes
Linux (Intel® Pentium® or equivalent platforms only)	Red Hat Enterprise Linux 4.0 WS/AS/ES	Yes	Yes	Yes
	Red Hat Enterprise Linux 3.0 WS/AS/ES	Yes	Yes	Yes
	SUSE Linux Enterprise Server 10.0	Yes	Yes	Yes
	SUSE Linux Enterprise Server 9.0	Yes	Yes	Yes

Table 3. Operating system support (continued)

Platform	Operating system	Environment		
		Development	Deployment Server	Deployment Target
Linux (IBM POWER5™ processor-based technology systems only)	SUSE Linux Enterprise Server 10.0	No	Yes	Yes
	SUSE Linux Enterprise Server 9.0	No	Yes	Yes
	Red Hat Enterprise Linux AS 4.0	No	Yes	Yes
	Red Hat Enterprise Linux AS 3.0	No	Yes	Yes
UNIX®	AIX 5.3	No	No	Yes
i5/OS	V5R4	No	No	Yes
	V5R3 and V5R3M5	No	No	Yes

Windows development server hardware and software requirements

The following table details the hardware and software requirements for installing the Solution Assembly Toolkit on a computer that is running Windows.

Table 4. Windows development server hardware and software requirements

Memory		Processor		Disk Space		
Minimum	Recommended	Minimum	Recommended	Max Total	Temporary	Other Requirements
512 MB	1.0 GB	Intel Pentium III processor with a minimum clock speed of 800 MHz	Intel Pentium 4 processor with a minimum clock speed of 1.7 GHz	1 GB **Solution Launcher installations** 10 MB	300 MB **Solution Laucher installations** 100 MB (Additional disk space might be required for the solution to be installed. Refer to the solution-specific requirements.)	• Administrative authority (required for product installation and uninstallation) • TCP/IP or DNS network support for deploying solutions to network-attached target computers. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • One of the following Web browsers: – Internet Explorer 6.0 SP1 + – Mozilla 1.7 or SeaMonkey 1.0 – Firefox 1.x • Eclipse versions supported - based on the following: – IES - Eclipse Full SDK - 3.2 – eclipse.org - Eclipse SDK - 3.2

Linux development server hardware and software requirements

The following table details the hardware and software requirements for installing the Solution Assembly Toolkit on an x86 system that is running Linux.

Table 5. Linux development server hardware and software requirements

Memory		Processor		Disk Space		
Minimum	Recommended	Minimum	Recommended	Max Total	Temporary	Other Requirements
512 MB	1.0 GB	Intel Pentium III processor with a minimum clock speed of 800 MHz	Intel Pentium 4 processor with a minimum clock speed of 1.7 GHz	1 GB **Solution Launcher installations** 10 MB	300 MB **Solution Launcher installations** 100 MB (Additional disk space might be required for the solution to be installed. Refer to the solution-specific requirements.)	• root authority (required for product installation and uninstallation) • A local area network (LAN) connection. • TCP/IP or DNS network support for deploying solutions to network-attached target computers. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • Any Linux supported video card that supports the resolution requirements. • One of the following Web browsers: – Mozilla 1.7 or SeaMonkey 1.0 – Firefox 1.x Other Web browsers, such as Konquerer and Opera, are not supported. • The 32-bit version of the compat-libstdc rpm package. • Eclipse versions supported - based on the following: – IES - Eclipse Full SDK - 3.2 – eclipse.org - Eclipse SDK - 3.2

Linux on IBM POWER hardware and software requirements

The following table details the hardware and software requirements for installing the Solution Assembly Toolkit on a computer that is running Linux on POWER.

Table 6. Linux on POWER hardware and software requirements

Memory		Processor		Disk Space		
Minimum	Recommended	Minimum	Recommended	Max Total	Temporary	Other Requirements
512 MB	1.0 GB	RS64-IV processor with a minimum clock speed of 600 MHz	RS64-V (POWER5) processor with a minimum clock speed of 1.6 GHz	1 GB **Solution Launcher installations** 10 MB	300 MB **Solution Launcher installations** 100 MB (Additional disk space might be required for the solution to be installed. Refer to the solution-specific requirements.)	• root authority (required for product installation and uninstallation) • A local area network (LAN) connection. • TCP/IP or DNS network support for deploying solutions to network-attached target computers. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • Any Linux supported video card that supports the resolution requirements. • One of the following Web browsers: – Mozilla 1.7 or SeaMonkey 1.0 – Firefox 1.x Other Web browsers, such as Konquerer and Opera, are not supported. • The 32-bit version of the compat-libstdc rpm package is required.

Windows deployment target requirements

The following table details the hardware and software requirements for deploying a solution to a target computer that is running Windows.

Table 7. Windows target computer hardware and software requirements

Memory		Processor		Disk Space		
Minimum	**Recommended**	**Minimum**	**Recommended**	**Total**	**Temporary**	**Other Requirements**
512 MB (Additional memory might be required for the solution to be installed on this computer. Refer to the solution-specific requirements.)	1.0 GB	Intel Pentium III processor with a minimum clock speed of 800 MHz	Intel Pentium 4 processor with a minimum clock speed of 1.7 GHz	Disk space requirements are specific to the solution to be installed.	130 MB	• A local area network (LAN) connection. • TCP/IP or DNS network support for solutions deployed across a network. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • For Windows XP target computers, Simple File Sharing, the Internet Connection Firewall (ICF) and the Windows Firewall must be disabled. • Remote registry administration must be enabled. • If the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 139. • Administrative authority (required for product installation and uninstallation).

Linux deployment target requirements

The following table details the hardware and software requirements for deploying a solution to an x86 target computer that is running Linux.

Table 8. Linux target computer hardware and software requirements

Memory		Processor		Disk Space		
Minimum	**Recommended**	**Minimum**	**Recommended**	**Total**	**Temporary**	**Other Requirements**
512 MB (Additional memory might be required for the solution to be installed on this computer. Refer to the solution-specific requirements.)	1.0 GB	Intel Pentium III processor with a minimum clock speed of 800 MHz	Intel Pentium 4 processor with a minimum clock speed of 1.7 GHz	Disk space requirements are specific to the solution to be installed.	130 MB	• A local area network (LAN) connection. • TCP/IP or DNS network support for solutions deployed across a network. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • Any Linux-supported video card that supports the resolution requirements • root authority (required for product installation and uninstallation). • The 32-bit version of the compat-libstdc rpm package is required. • If the deployment server is not the same as the target computer, ensure the following: – SSH is installed and enabled. – Password authentication is set to yes in the /etc/ssh/sshd_config file. After setting the password authentication, the SSH daemon must be stopped and restarted. – If the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 22. – For target computers using the REXECProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 512. Any available local port on the target computer is used to connect back to the deployment computer. – For target computers using the RSHProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 514. The first available local port between 600 and 1023, inclusive, on the target computer is used to connect back to the deployment computer.

Linux on IBM POWER deployment target requirements

The following table details the hardware and software requirements for deploying a solution to a target computer that is running Linux on IBM POWER.

Table 9. Linux on POWER target computer hardware and software requirements

Memory		Processor		Disk Space		
Minimum	Recommended	Minimum	Recommended	Total	Temporary	Other Requirements
512 MB (Additional memory might be required for the solution to be installed on this computer. Refer to the solution-specific requirements.)	1.0 GB	RS64-IV processor with a minimum clock speed of 600 MHz	RS64-V (POWER5) processor with a minimum clock speed of 1.6 GHz	Disk space requirements are specific to the solution to be installed.	130 MB	• A local area network (LAN) connection. • TCP/IP or DNS network support for solutions deployed across a network. • An SVGA monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors. • Any Linux-supported video card that supports the resolution requirements. • root authority (required for product installation and uninstallation). • The 32-bit and 64-bit versions of the compat-libstdc rpm package is required. • If the deployment server is not the same as the target computer, ensure the following: – SSH is installed and enabled. – Password authentication is set to yes in the /etc/ssh/sshd_config file. After setting the password authentication, the SSH daemon must be stopped and restarted. – If the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 22. – For target computers using the REXECProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 512. Any available local port on the target computer is used to connect back to the deployment computer. – For target computers using the RSHProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 514. The first available local port between 600 and 1023, inclusive, on the target computer is used to connect back to the deployment computer.

AIX deployment target requirements

The following table details the hardware and software requirements for deploying a solution to a target computer that is running AIX.

Table 10. AIX target computer hardware and software requirements

Memory		Processor		Disk Space		
Minimum	Recommended	Minimum	Recommended	Total	Temporary	Other Requirements
512 MB (Additional memory might be required for the solution to be installed on this computer. Refer to the solution-specific requirements.)	Memory requirements are specific to the solution to be installed.	IBM POWER3-II processor with a minimum clock speed of 400® MHz	Processor requirements are specific to the solution to be installed.	Disk space requirements are specific to the solution to be installed.	130 MB	• A local area network (LAN) connection. • TCP/IP or DNS network support for solutions deployed across a network. • A monitor with a minimum 1024 x 768 video resolution configured to display a minimum color depth of 256 colors (only required for GUI-based tasks, such as interactive installation; not needed for silent installation or command-line functions). • If the deployment server is not the same as the target computer, ensure the following: – SSH is installed and enabled. – Password authentication is set to yes in the /etc/ssh/sshd_config file. After setting the password authentication, the SSH daemon must be stopped and restarted. – If the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 22. – For target computers using the REXECProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 512. Any available local port on the target computer is used to connect back to the deployment computer. – For target computers using the RSHProtocol, if the target computer is behind a firewall, the firewall must be configured to allow inbound connections on port 514. The first available local port between 600 and 1023, inclusive, on the target computer is used to connect back to the deployment computer.

i5/OS deployment target requirements

The following table details the hardware and software requirements for deploying a solution to a target computer that is running i5/OS.

Table 11. i5/OS target computer hardware and software requirements

Memory		Processor		Disk Space		Other Requirements
Minimum	Recommended	Minimum	Recommended	Total	Temporary	
N/A		System i™ models that support V5R3 or V5R4		Disk space requirements are specific to the solution to be installed.	80 MB	• i5/OS (5722SS1) V5R3, V5R3M5, or V5R4 – For V5R3Mx, SF99530 Cumulative PTF pack and SF99269 Java Group minimum PTF level 10, and SI17529 PTF – For V5R4 Cumulative PTF pack and SF99291 Java Group minimum PTF level 2 • Host Servers (5722SS1, option 12) • QShell (5722SS1, option 30) • Java Developer Kit 1.5 (5722JV1, option 7) **Note:** Java Runtime Environment version 1.5 is shipped with i5/OS version 5 release 4. For i5/OS version 5 release 3 servers, order PTF SI17529 and refer to the cover letter for instructions. • The Open Source version of the IBM Toolbox for Java (JTOpen release 4.9 or higher) • Secure Sockets Layer (SSL) optionally installed and enabled to support SSL-secured communications. Additional software for SSL includes: – For V5R3Mx, Cryptographic Access Provider 128-bit (5722AC3) – Digital Certificate Manager (5722SS1, option 34) • A local area network (LAN) connection. • TCP/IP or DNS network support for solutions deployed across a network. • SECOFR user with the following permissions (required for product installation and uninstallation): – ALLOBJ – SAVSYS – JOBCTL – SERVICE – SPLCTL – SECADM – AUDIT – IOSYSCFG • By default IBM Remote Execution and Access (RXA) connects to port 449, 8473, 8475, or 8476 on target computers not running SSL and to port 449, 9473, 9475, or 9476 on target computers running SSL. If the target computer is behind a firewall, the firewall must be configured to allow inbound connections on the appropriate port.

Migrating to a newer version of the IBM Solution Assembly Toolkit

You can migrate projects from previous versions of the Solution Assembly Toolkit.

To migrate specific projects from a previous version, follow these steps:

1. Install the Solution Assembly Toolkit.
2. Copy the deployment packages you want to migrate from <SAT install dir>\SolutionEnabler\com\ibm\jsdt\webserver\tree to /SolutionEnabler/com/ibm/jsdt/webserver/tree.
3. Open the Express Runtime developer, and select **File > Import > Existing Projects into Workspace.**

4. Click **Next**.
5. Select the following root directory:

   <SAT install dir>\SolutionEnabler\workspace
6. Choose the projects that you want to migrate and select **Copy projects into workspace**.
7. Click **Finish**.

Printed in USA