WebSphere® Business Integrator

Solution Templates

Version 2.1

WebSphere® Business Integrator

Solution Templates

Version 2.1

Note
Before using this information and the products it supports, read the information in "Notices" on page 55

First Edition (November 2001)

This edition applies to Version 2.1 of IBM® WebSphere® Business Integrator (program number 5724-A78) and to all subsequent release and modifications until otherwise indicated in new editions.

Contents

Figures

Preface

This guide provides information about the user-registration and purchase-order management templates that are provided with IBM® WebSphere® Studio Business Integrator Extensions (referred to as *Solution Studio* throughout this documentation). It discusses the contents of the templates and provides instructions for developing them in Solution Studio and deploying them to the WebSphere Business Integrator run time.

CAUTION:
The procedures included in this document are believed to be accurate but were not formally tested and validated by the time of publication. Future editions of this document are expected to include validated procedures. For the latest information about the solution templates, see the WebSphere Business Integrator Web page at the following URL:

www.ibm.com/software/webservers/btobintegrator

Who should read this book

This document is intended for Business Integrator developers and administrators who want to gain familiarity with the development and deployment of Business Integrator solutions by using a predefined template in Solution Studio.

What you need to know

You need to have a familiarity with the Solution Studio and Business Integrator products. Information on using Solution Studio can be found in the *WebSphere Studio Business Integrator Extensions Developer's Guide* and in Solution Studio's online help system. An overview of Business Integrator can be found in the *WebSphere Business Integrator Concepts and Planning* book. See "Bibliography" on page 59 for a full listing of Business Integrator books and related libraries.

The instructions in this book assume the following:

- You are working on a Solution Studio client machine that has the full complement of supported development products installed on it, including MQSeries®, MQSeries Adapter Builder, MQSeries Integrator, MQSeries Workflow, and VisualAge® for Java™. For information on installing and configuring Solution Studio clients, including information on configuring it to work with supported development products, see the *WebSphere Studio Business Integrator Extensions Installation Guide*.

- The client machine is supported by a Solution Studio server machine. For information on installing and configuring Solution Studio servers, see the *WebSphere Studio Business Integrator Extensions Installation Guide*.
- You have a Business Integrator run-time installation to which to deploy the developed templates. For information on installing and configuring the Business Integrator run time, see the *WebSphere Business Integrator Installation Guide* and *WebSphere Business Integrator Run Time* books.

Before you implement WebSphere Business Integrator

WebSphere Business Integrator uses multiple underlying products and technologies to support the solutions that you create and run. In general, before you implement Business Integrator, you need to understand the underlying products and technologies that support your solution.

Before you implement Business Integrator, you or other members of your organization need to be generally skilled in the activities listed below for similar solutions, products, and underlying products and technologies. If you and other members of your organization do not possess these skills, you need to obtain assistance from qualified services staff, either from IBM or from third parties, to implement Business Integrator. You must be prepared to use the documentation of the underlying products and technologies. (This documentation is provided with Business Integrator or otherwise from IBM.)

When you plan, install, and configure Business Integrator, you need to understand how to install and configure the underlying products and technologies that you use in your installation. Business Integrator provides the installation of most of the underlying products and technologies into its run-time environment. However, you might need to install and configure certain underlying products separately into either the build-time or run-time environment. You might also need to diagnose and correct installation problems with underlying products and technologies.

Before you design, develop, and publish solutions, you need to be:

- Generally familiar with system-integration techniques in a business environment.
- Prepared to use the tools of the underlying products and technologies that your solution requires.
- Familiar with the run-time behavior of the underlying products and technologies that your solution requires.
- Familiar with modeling concepts and techniques such as Unified Modeling Language and related tools, with state machine concepts, and with visual flow composition-modeling concepts and techniques.
- Familiar with Internet and electronic data interchange (EDI) concepts and technologies, if required by your solution.

- Prepared to research the existing applications, systems, and networks that you integrate with Business Integrator.
  - Inside your enterprise, they can be known as legacy systems, back-end systems, enterprise applications, or endpoint applications.
  - Outside your enterprise, they can be known as trading networks, private EDI networks, or similar networks that your solution requires.

Before you deploy, run, manage, diagnose, and tune Business Integrator, you need to be prepared to use the management, trace, audit, exception handling, diagnostic, and related tools of the underlying products and technologies that support your solution. You need to be prepared to understand the solution itself to the degree needed for these tasks.

How to send your comments

IBM welcomes your comments. You can send your comments by any one of the following methods:

- Electronically to the following address:

  `idrcf@hursley.ibm.com`

  Be sure to include your network address if you want a reply.
- By FAX to the following numbers:

  `UK: 01962-842327`
  `Other countries: +44-1962-842327`
- By mail to the following address:

  User Technologies
  Mail Point 095
  IBM United Kingdom Laboratories
  Hursley Park
  Winchester
  Hampshire
  SO21 2JN
  United Kingdom

Documentation conventions

Throughout this book, the following conventions distinguish different elements of text:

bold	Menu bar and menu choices, push-button names, icons, check boxes, radio buttons, entry fields, and command names
plain text	Window, screen, and dialog box titles, file names, directories, and keyboard key names
italic	First occurrence of words with special meaning, variables, and emphasis
`monospaced`	Output to the display, and user input at the command prompt or in an entry field

Chapter 1. About the templates

Solution Studio provides two templates for use with Business Integrator: the user-registration template and the purchase-order (PO) management template. These templates are designed for the following purposes:

- To demonstrate the capabilities of Business Integrator's build-time tools and run-time managers.
- To illustrate the use of the Solution Studio build-time tools to develop and deploy solutions.
- To show complete examples of end-to-end solutions; that is, solutions that can handle all phases of a business problem, including interactions that involve multiple users and secure transactions across one or more back-end systems in one or more enterprises.
- To provide working templates for typical business problems that can be customized for optimum efficiency at individual sites.

The templates consist of a number of Business Integrator *artifacts*; that is, deployable files such as executable files and content files. Artifacts are created and modified in Solution Studio during the development phase of the solution, then deployed to the Business Integrator run time to perform the functionality of the solution. As the first step in preparing a template solution for deployment, you create a new project in Solution Studio and populate it with artifacts from one of the templates. Many of the artifacts provided with the templates can be deployed without modification; others require customization. Regardless of the amount of customization performed on them in Solution Studio, all artifacts are packaged and deployed to a run-time system as the final phase of development.

The individual templates are discussed in more detail in the following chapters:

- The user-registration template is discussed in "Chapter 2. The user-registration template" on page 3.
- The PO-management template is discussed in "Chapter 3. The purchase-order management template" on page 31.

Note: The templates provided with WebSphere Business Integrator version 2.x do not necessarily represent best programming practices. They do not use all of the components of the Business Integrator run time. They are intended to be used with the current release of Business Integrator; however, it is anticipated that templates provided with future releases

of Business Integrator will be based on more-advanced programming models and will use all run-time components. More information can be found at the following Web site:

www.ibm.com/software/webservers/btobintegrator/

Chapter 2. The user-registration template

This chapter discusses the user-registration template. It includes the following sections:

About the user-registration template

The user-registration template enables users to log on to a Business Integrator solution; after a user logs on, he or she can perform the tasks that he or she is authorized to perform. A user can also register with the solution and request to participate in a specific role or roles.

The user-registration template uses the following Business Integrator services. For more information on these services, see the *WebSphere Business Integrator Concepts and Planning* and *WebSphere Business Integrator Run Time* books.

- Business Flow Manager
- Interaction Manager
- Trust and Access Manager

Business-process model

The business-process model describes the business events that are received by the system as well as those that are generated by the system. For the purposes of the user-registration template, it is assumed that the current business process for registering a new user is analyzed to reveal that an automated process for registering to use the e-marketplace can streamline the business process. The end-to-end process for registering a new user includes subprocesses that are assumed to be in place and functioning. For instance, it is assumed that the user-logon process includes the following two subprocesses:

- For existing users of the e-marketplace, a process to log on to perform work based on the user's role.
- For new users, a process to register to use the e-marketplace (a one-time activity).

A simplified version of the end-to-end process for the user-registration solution from logging on to request registration to logging on as a user in an appropriate role is shown in Figure 1.

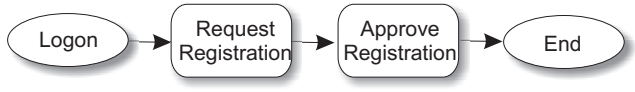

Figure 1. Process flow of the user-registration solution

The process has two subprocesses. In this example, a new user logon subprocess initiates the user-registration process.

The improved business-process model is as follows:

1. From a browser, a new user requests registration to the e-marketplace and participation with trading partners in the role of a buyer.
2. An administrator approves the new user in a guest role and provides the new user with a user ID and password.
3. The new user participates in the e-marketplace in the limited role of guest until he or she is added to the trading-partner services in the buyer role for which he or she registered.
4. The registered user logs on to use the e-marketplace and participates in the role for which he or she registered.

The user-registration solution focuses on the registration-approval process. This process is discussed in "Task model".

The improved process eliminates many of the manual steps for the e-marketplace and trading-partner administrators, and involves the new user in the e-marketplace much more quickly in the guest role. After the new user is approved in the role for which he or she registered, participation in the e-marketplace in that role is automatic at logon.

Task model

The task model focuses on the approval process for the user-registration solution.

The MQSeries Workflow build-time tool is used to model the workflow for the approval process. The result is a Flow Definition Language (FDL) script that can be used by the MQSeries Workflow run-time engine to act as a template for instances of the process flow.

The approval process has one start point and two endpoints. The process also has a subprocess for adding the registering user to services before the

endpoints. The flow for the approval process as modeled in the MQSeries Workflow build-time tool is illustrated in Figure 2.

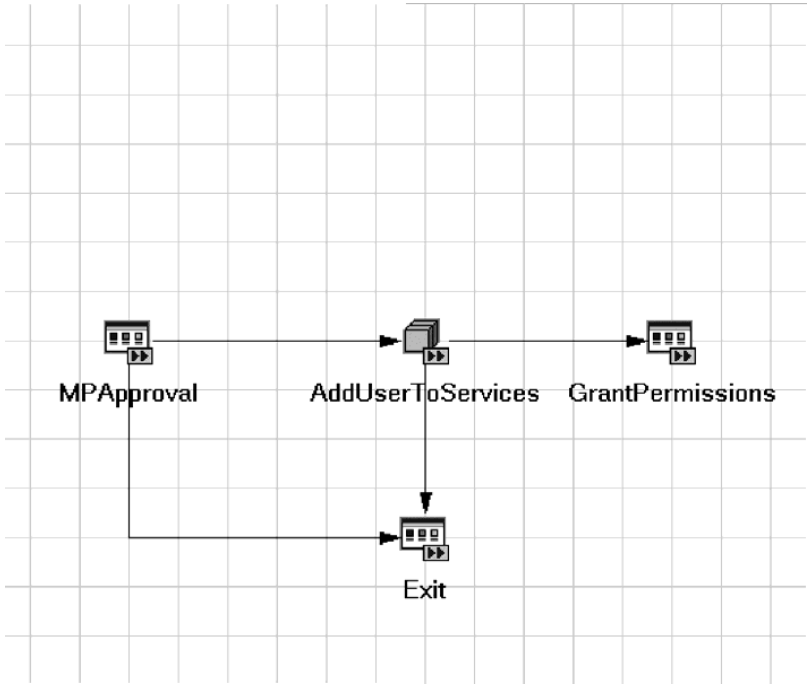

Figure 2. The approval process as modeled in the MQSeries Workflow build-time tool

The subprocess has a single start point and a single endpoint. The end point of the subprocess leads to either of the two endpoints for the process flow. The subprocess as modeled in the MQSeries Workflow build-time tool is illustrated in Figure 3 on page 6.

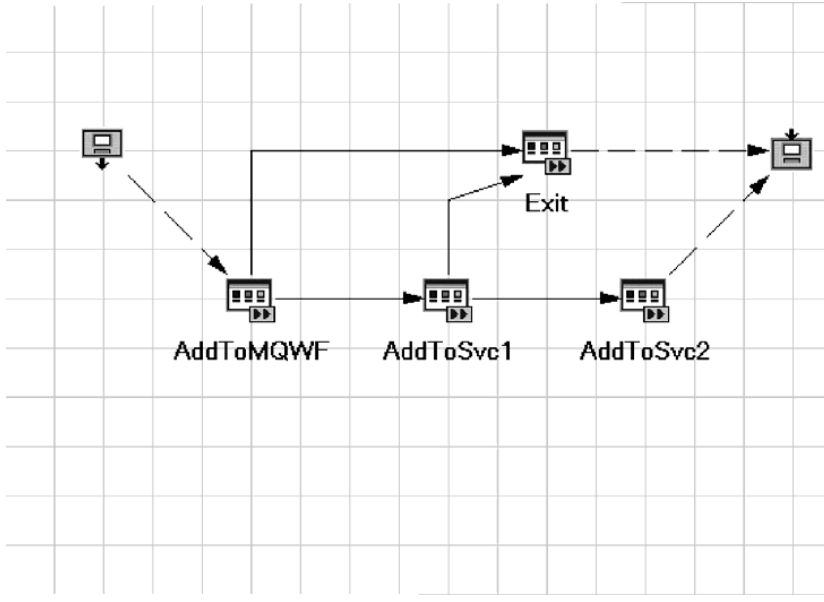

Figure 3. A subprocess as modeled in the MQSeries Workflow build-time tool

Approve registration

The approve-registration process has five tasks associated with it. Most are performed automatically when an administrator gives approval. The tasks are as follows:

1. When the administrator logs on to the e-marketplace, the Logon servlet authenticates the user ID and password though the Business Integrator Trust and Access Manager and checks the authorization for the administrator based on the user's role. The Logon servlet then logs on to the Workflow Services session bean to determine whether there is any work for this user, based on his role; for instance, whether there are new users who need approval. The servlet presents the administrator with a window that lists the following four options:
   - **Approve/Reject new user**
   - **Register new user by proxy**
   - **Revoke registered user**
   - **Modify user data**

   Because a new user has requested to register with the e-marketplace, there is work for this administrator to perform. The servlet presents a window with the **Approve/Reject new user** option displayed as an active link. When the administrator selects this link, a request is sent to the UserReg

servlet. The UserReg servlet requests a list of users with the status of Pending Approval from the RegistrationServices session bean.

Registration Services returns the list of users, and the UserReg servlet provides this list as an HTML form with buttons such as **Approve** and **Reject** to the administrator. The administrator selects the new user and then selects the **Approve** button. This initiates a request to the UserReg servlet. The UserReg servlet sends an Approve User request to RegistrationServices. On the administrator's behalf, RegistrationServices initiates and completes the Workflow activity for approving the new user to the e-marketplace.

2. The user-registration template adds the user to the MQSeries Workflow build time.
3. The user-registration template ensures that the MQSeries Workflow run-time is updated to reflect the new user.
4. The new user is then added to the selected trading-partner services for which the user registered.
5. The new user can log on to the e-marketplace and perform work based on the user role.

Developing and publishing the user-registration template in Solution Studio

This section discusses how to develop and publish the template. Development instructions are provided in "Developing the template"; publishing instructions are provided in "Publishing the template" on page 18.

Developing the template

To develop the user-registration template in Solution Studio, perform the following steps. Note that the Solution Studio wizards include many fields; if a field is not referenced in the following procedure, it is not needed for the successful development of the template, regardless of whether it contains a default value or is blank by default. You can safely disregard any field that is not referenced in the following procedure.

1. Start WebSphere Studio.
2. Create a new project based on the user-registration template, as follows:
   a. If a dialog box appears and asks what you want to do, click **Create new**; otherwise, click **File → New Project**. The New Project dialog box appears.
   b. Type a name for the project in the **Project Name** field; for example, `URTest`.
   c. In the **Project Folder** field, click **Browse** and browse to the shared folder on the ClearCase server machine that holds project files. For information on setting up a ClearCase server, see the *WebSphere Studio Business Integrator Extensions Installation Guide*.

d. In the **Create using** drop-down field, select **UserRegistration**.
e. Click **OK**.

The project view (left-hand pane) in WebSphere Studio now displays the name of your project and the hierarchy of folders contained in the project. These folders contain the artifacts that make up the user-registration template.

3. Configure the Application artifacts for the template as follows:
   a. Select the Application folder in the project view.
   b. Configure two adapters named UserRegRUC and UserRegRUR by performing the following steps for each adapter.
      1) Start the Application Configuration wizard by clicking **Tools → Wizards → Application Configuration Wizard**. The wizard opens to the Application page.
      2) In the **Application Id** field, type one of the following adapter names:
         - `UserRegRUC`
         - `UserRegRUR`
      3) In the **Trace** field, click **On**, and verify that the **Trace Level** field is set to a value of `-1`. Ensure that the **Trace Client** field is left blank.
      4) In the **Logging** field, ensure that **Disabled** is selected.
      5) Ensure that the **Logon Info Class** field is left blank.
      6) Ensure that the **Queue Manager** field is left blank.
      7) In the **Use remote queue manager to send messages?** field, ensure that **No** is selected.
      8) Ensure that the **Host Name**, **Port Number**, and **Channel Name** fields are left blank.
      9) In the **Queue Manager** field under **JMS Queue Connection Factory**, type `DEFAULT`. If the queue manager you are using has a different name, type it into this field.

         Alternatively, click **Define**. The Component Definition for JMS QM Host window opens. Select a valid component and subcomponent from the drop-down boxes in the **Component** and **Subcomponent** fields, respectively. Click **OK** to close the window.
      10) In the **Connection Factory** field, type `DEFAULT`. If the connection factory you are using has a different name, type it into this field.
      11) Under **Message Types**, click **New**. The Adapter Routing wizard starts.
      12) Ensure that the General page on the Adapter Routing wizard is displayed. In the **Message Body Category** field, type `OAG`.

13) In the **Message Body Type** field, type `SyncPersonnel.UserRegistration`.
14) In the **Receive Mode** field, select **JMS**. The wizard now displays the **Message Formatter**, **Receive Timeout**, **Receive Queue**, **Error Queue**, and **Reply Queue** fields.
15) In the **Message Formatter** field, ensure that **com.ibm.epic.adapters.eak.nativeadapter.JMSNMXMLFormatter** is selected.
16) In the **Receive Timeout** field, type `30000`.
17) In the **Receive Queue** field, type one of the following:
    - `JMSUserRegRUC` if you are configuring the UserRegRUC adapter
    - `JMSUserRegRUR` if you are configuring the UserRegRUR adapter
18) In the **Error Queue** field, type `JMSErrorQueue`.
19) Click **Command** or **Next**. The Command page of the Adapter Routing wizard is displayed.
20) In the **Command Type** field, select **Enterprise Bean** from the drop-down list.
21) In the **Command Class Name** field, type one of the following:
    - `ReceiveConfirmationAdapter` if you are configuring the UserRegRUC adapter
    - `UserRegWraparound` if you are configuring the UserRegRUR adapter
22) In the **TDC Mapper Class** field, type one of the following:
    - If you are configuring the UserRegRUC adapter, type `com.ibm.b2bi.templates.userregistration.adapters.userregtarget.UserRegTargetTargetDataInputTDCMapper`
    - If you are configuring the UserRegRUR adapter, type `com.ibm.b2bi.templates.userregistration.adapters.syncpersonneluserregistration.SyncPersonnelUserRegistrationInput_MessageTDCMapper`
23) In the **Method to Invoke** field, type one of the following:
    - If you are configuring the UserRegRUC adapter, type `executeUserRegTarget`
    - If you are configuring the UserRegRUR adapter, type `executeSyncPersonnelUserRegistration`
24) In the **Parameter Type** field, type one of the following:
    - `com.ibm.b2bi.templates.userregistration.adapters.userregtarget.UserRegTargetTargetDataInputTDC` if you are configuring the UserRegRUC adapter
    - `com.ibm.b2bi.templates.userregistration.adapters.syncpersonneluserregistration.`

SyncPersonnelUserRegistrationInput_MessageTDC if you are configuring the UserRegRUR adapter

25) In the **Initial Context Factory** field, ensure that **com.ibm.ejs.ns.jndi.CNInitialContextFactory** is entered.
26) Click **Worker** or **Next**. The Worker page of the Adapter Routing wizard is displayed.
27) In the **Bean Name** field, type one of the following:
    - RUCWorker if you are configuring the UserRegRUC adapter
    - RURWorker if you are configuring the UserRegRUR adapter
28) In the **JNDI Home Name** field, type one of the following:
    - RUCWorker if you are configuring the UserRegRUC adapter
    - RURWorker if you are configuring the UserRegRUR adapter
29) In the **Container Name** field, type UserRegContainer.
30) In the **Maximum Retries** field, type 0.
31) In the **Maximum Sessions** field, type 1.
32) Click **Finish** on the Adapter Routing wizard. The wizard closes and returns you to the Application Configuration wizard.
33) Click **Finish** on the Application Configuration wizard. Solution Studio creates the following files:
    - An adapter configuration file named *adapter_name*.appcfg (UserRegRUC.appcfg or UserRegRUR.appcfg)
    - An XML instructions file named *adapter_name*_inst.xml (UserRegRUC_inst.xml or UserRegRUR_inst.xml)
    - XML configuration files for the workers (RUCWorker.xml and UserRegRUC_workers.xml, or RURWorker.xml and UserRegRUR_workers.xml)
    - A Java Message Service (JMS) configuration file named *adapter_name*.scp (UserRegRUC.scp or UserRegRUR.scp)

    Solution Studio places these files in the Application folder with the other application artifacts, then closes the wizard.

Repeat these steps to define each of the two adapters.

c. Create an adapter named SUC by performing the following steps:
1) Start the Application Configuration wizard by clicking **Tools → Wizards → Application Configuration Wizard**. The wizard opens to the Application page.
2) In the **Application Id** field, type SUC.
3) In the **Trace** field, click **On**, and verify that the **Trace Level** field is set to a value of -1. Ensure that the **Trace Client** field is left blank.
4) In the **Logging** field, ensure that **Disabled** is selected.

5) Ensure that the **Logon Info Class** field is left blank.
6) Ensure that the **Queue Manager** field is left blank.
7) In the **Use remote queue manager to send messages?** field, ensure that **No** is selected.
8) Ensure that the **Host Name**, **Port Number**, and **Channel Name** fields are left blank.
9) In the **Queue Manager** field under **JMS Queue Connection Factory**, type `DEFAULT`. If the queue manager you are using has a different name, type it into this field.

   Alternatively, click **Define**. The Component Definition for JMS QM Host window opens. Select a valid component and subcomponent from the drop-down boxes in the **Component** and **Subcomponent** fields, respectively. Click **OK** to close the window.
10) In the **Connection Factory** field, type `DEFAULT`. If the connection factory you are using has a different name, type it into this field.
11) Under **Message Types**, click **New**. The Adapter Routing wizard starts.
12) Ensure that the General page on the Adapter Routing wizard is displayed. In the **Message Body Category** field, type `OAG`.
13) In the **Message Body Type** field, type `SyncPersonnel.UserRegistration`.
14) In the **Destination Id** field, type `UserRegRUC`, then click **Add To List**. UserRegRUC appears in the **Destination Ids** field.
15) Ensure that the **Receive Mode** field is left blank.
16) Click **Finish** on the Adapter Routing wizard. The wizard closes and returns you to the Application Configuration wizard.
17) Click **Finish** on the Application Configuration Wizard. Solution Studio creates the following files:
    - An adapter configuration file named SUC.appcfg
    - An XML instructions file named SUC_inst.xml
    - A JMS configuration file named SUC.scp

    Solution Studio places these files in the Application folder with the other application artifacts, then closes the wizard.

d. Create an adapter named SUR by performing the following steps:
   1) Start the Application Configuration wizard by clicking **Tools → Wizards → Application Configuration Wizard**. The wizard opens to the Application page.
   2) In the **Application Id** field, type `SUR`.
   3) In the **Trace** field, click **On**, and verify that the **Trace Level** field is set to a value of `-1`. Ensure that the **Trace Client** field is left blank.

4) In the **Logging** field, ensure that **Disabled** is selected.
5) Ensure that the **Logon Info Class** field is left blank.
6) Ensure that the **Queue Manager** field is left blank.
7) In the **Use remote queue manager to send messages?** field, ensure that **No** is selected.
8) Ensure that the **Host Name**, **Port Number**, and **Channel Name** fields are left blank.
9) In the **Queue Manager** field under **JMS Queue Connection Factory**, type `DEFAULT`. If the queue manager you are using has a different name, type it into this field.

   Alternatively, click **Define**. The Component Definition for JMS QM Host window opens. Select a valid component and subcomponent from the drop-down boxes in the **Component** and **Subcomponent** fields, respectively. Click **OK** to close the window.
10) In the **Connection Factory** field, type `DEFAULT`. If the connection factory you are using has a different name, type it into this field.
11) Under **Message Types**, click **New**. The Adapter Routing wizard starts.
12) Ensure that the General page on the Adapter Routing wizard is displayed. In the **Message Body Category** field, type `OAG`.
13) In the **Message Body Type** field, type `SyncPersonnel.UserRegistration`.
14) In the **Destination Id** field, type `UserRegRUR`, then click **Add To List**. UserRegRUR appears in the **Destination Ids** field.
15) In the **Receive Mode** field, select **JMS**. The wizard now displays the **Message Formatter**, **Receive Timeout**, **Receive Queue**, **Error Queue**, and **Reply Queue** fields.
16) In the **Message Formatter** field, ensure that **com.ibm.epic.adapters.eak.nativeadapter.JMSNMXMLFormatter** is selected.
17) In the **Receive Timeout** field, type `30000`.
18) Ensure that the **Receive Queue**, **Error Queue**, and **Reply Queue** fields are left blank.
19) Click **Finish** on the Adapter Routing wizard. The wizard closes and returns you to the Application Configuration wizard.
20) Click **Finish** on the Application Configuration Wizard. Solution Studio creates the following files:
    - An adapter configuration file named SUR.appcfg
    - An XML instructions file named SUR_inst.xml
    - A JMS configuration file named SUR.scp

Solution Studio places these files in the Application folder with the other application artifacts, then closes the wizard.

e. Define three queues named JMSUserRegRUC, JMSUserRegRUR, and JMSErrorQueue by performing the following steps for each queue.

1) Start the MQSeries Queue Definition wizard by clicking **Tools → Wizards → MQSeries Queue Definition Wizard**. The wizard opens to the General page.

2) In the **Queue Manager Name** field, verify the **DEFAULT** entry. If the queue manager you are using has a different name, type it into this field.

Alternatively, click **Define**. The Component Definition for JMS QM Host window opens. Select a valid component and subcomponent from the drop-down boxes in the **Component** and **Subcomponent** fields, respectively. Click **OK** to close the window.

3) In the **Queue Name** field, type one of the following queue names:

- `JMSUserRegRUC`
- `JMSUserRegRUR`
- `JMSErrorQueue`

4) Accept the following defaults listed on the General page:

- **Queue Type:** `Local`
- **Default Priority:** `0`
- **Default Persistence:** `Not Persistent`
- **Put Messages:** `Allowed`
- **Get Messages:** `Allowed`
- **Scope:** `Queue Manager`
- **Usage:** `Normal`

Accept all other defaults listed on the other pages of this wizard.

5) Click **Finish**. Solution Studio creates an XML configuration file named *queue_name*.xml and an MQSeries configuration file named *queue_name*.mqsc. Solution Studio places these files in the Application folder with the other application artifacts, then closes the wizard.

Repeat these steps to define each of the three queues.

4. Configure the Business Process and Roles artifacts for the template as follows:

a. Delete the B2BiTemplatesUserReg.fdl file from the BusinessProcess folder. This item is recreated in a subsequent step.

b. Go to the *x:*\WebSphere\templates\UserRegistration\BusinessProcess directory, where *x* is the drive letter on which WebSphere Studio is installed. Open the B2BiTemplatesUserReg.fdl file in a text editor and

verify that the queue manager name and queue name correspond to the names that were defined for MQSeries Workflow during that product's installation.

c. Import the B2BiTemplatesUserReg.fdl file into the BusinessProcess folder, as follows:

   1) Select the BusinessProcess folder in the project view.
   2) Start the Workflow wizard by clicking **Tools → Wizards → Workflow Wizard**. The wizard opens.
   3) Click **Select file(s) to be imported**. The Windows Open dialog box appears.
   4) Browse to the *x:*\WebSphere\templates\UserRegistration\BusinessProcess directory, where *x* is the drive letter on which WebSphere Studio is installed. Select the B2BiTemplatesUserReg.fdl file, then click **Open**.

   When the file is imported, the Workflow wizard places the FDL file in the BusinessProcess folder in the project view. The wizard also creates a Roles folder if it does not already exist, creates an XML configuration file for the FDL file, and places the XML file in the Roles folder.

d. Define a queue named WFRegServices for MQSeries Workflow as follows:

   1) Select the BusinessProcess folder in the project view.
   2) Start the MQSeries Queue Definition wizard by clicking **Tools → Wizards → MQSeries Queue Definition Wizard**. The wizard opens.
   3) In the **Queue Manager Name** field, click **Define**. The Component Definition for JMS QM Host window opens. Enter the following information:
      a) In the **Component** field, select **BFM** from the drop-down list.
      b) In the **Subcomponent** field, select **WF** from the drop-down list.
      c) Click **OK** to close the Component Definition for JMS QM Host window.
   4) In the **Queue Name** field, type `WFRegServices`.
   5) In the **Queue Type** field, select **Local** from the drop-down list.
   6) If desired, in the **Description** field, type an appropriate description (for example, `Queue for MQSeries Workflow`).
   7) In the **Default Priority** field, type `0` (zero).
   8) In the **Default Persistent** field, select **Not Persistent** from the drop-down list.

9) In the **Put Messages** field, select **Allowed** from the drop-down list.
10) In the **Get Messages** field, select **Allowed** from the drop-down list.
11) In the **Scope** field, select **Queue Manager** from the drop-down list.
12) In the **Usage** field, select **Normal** from the drop-down list.

Accept any other defaults and click **Finish**. Solution Studio creates two configuration files named WFRegServices.mqsc and WFRegServices.xml, places them in the BusinessProcess folder, and closes the wizard.

5. Configure the Client artifacts for the template as follows:
   a. Select the Client folder in the project view, then open the web folder.
   b. Start the WebSphere Artifact Configuration wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.
   c. In the **Configuration File** field, type an appropriate name for the configuration file. The name WebDocConfig is used for this example.
   d. In the **Resource Type** field, select **Document**.
   e. Click **Select All** to select all the artifacts in the folder.
   f. In the **Web Application** field, type `UR`.
   g. Click **Finish**. Solution Studio creates an XML configuration file (in this case, WebDocConfig.xml), places it in the web folder with the other client artifacts, and closes the wizard.
6. Configure the data artifacts for the template as follows:
   a. Select the Data folder in the project view.
   b. Start the WebSphere Artifact Configuration wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.
   c. In the **Configuration File** field, type an appropriate name for the configuration file. The name Config is used for this example.
   d. In the **Resource Type** field, select **Enterprise Bean**.
   e. Select the **UserRegwas.jar** file in the list that appears.
   f. In the **Bean Type** field, click **Entity Bean**.
   g. In the **Container Name** field, type `UserRegContainer`.
   h. In the **Data Source** field, type `B2BITemplatesDS`.
   i. In the **Persistence** field, click **Bean Managed**.
   j. In the **Create Data Source?** field, click **Yes**.
   k. In the **Database Name** field, type `B2BiTMPL`.

l. Click **Finish**. Solution Studio creates an XML configuration file (in this case, UserRegEJBConfig.xml), places it in the Data folder with the other data artifacts, and closes the wizard.

7. Configure the roles artifacts for the template as follows:
   a. Select the Roles folder in the project view.
   b. Start the Solution Deployment wizard by clicking **Tools → Wizards → Solution Deployment Wizard**. The wizard opens to the Solution Deployment page.
   c. In the **Solution Name** field, type `B2BiTemplates`.
   d. In the **Solution Description** field, type an appropriate description—for instance, `User-registration update in LDAP`.
   e. In the **Solution Entry Point** field, type `/ePortal/jsp/B2BiDesktop.jsp`.
   f. In the **List of Roles** field, select all roles, including the following:
      - **MP Studio Developer**
      - **MP Solution Developer**
      - **MP Solution Publisher**
      - **MP Administrator**
      - **MP EDI Specialist**
   g. Click **Menu Items** to display the Menu Items page.
   h. Create four menu items as follows:
      1) Create a menu item named UserAdministration by entering the following information in the fields under **Menu Item Data**:
         - **Menu Item Id:** `UserAdministration`
         - **Menu Item Title:** `UserAdministration`
         - **Action:** Select **UR/UserManag.jsp** from the drop-down list, or type this value into the field.
         - **Image:** Select **/ePortal/Images/UserAdministration.gif** from the drop-down list, or type this value into the field.
         - **Roles:** Check **MPAdministrator**.
         - **Parent Menu Item Id:** Leave this field blank.
         - **Target Frame:** `Content`

         Click **Add** to add this menu item to the list in the left-hand pane of the wizard.
      2) Create a menu item named ReviewApplication by entering the following information in the fields under **Menu Item Data**:
         - **Menu Item Id:** `ReviewApplication`
         - **Menu Item Title:** `ReviewApplication`
         - **Action:** Select **UR/Admin?Action=10** from the drop-down list, or type this value into the field.

- **Image:** Select **/ePortal/Images/ReviewApplication.gif** from the drop-down list, or type this value into the field.
- **Roles:** Check **MPAdministrator**.
- **Parent Menu Item Id:** Select **UserAdministration** from the drop-down list, or type this value into the field.
- **Target Frame:** `Content`

Click **Add** to add this menu item to the list in the left-hand pane of the wizard.

3) Create a menu item named UserManagement by entering the following information in the fields under **Menu Item Data**:
   - **Menu Item Id:** `UserManagement`
   - **Menu Item Title:** `UserManagement`
   - **Action:** Select **UR/UserManag.jsp** from the drop-down list, or type this value into the field.
   - **Image:** Select **/ePortal/Images/UserManagement.gif** from the drop-down list, or type this value into the field.
   - **Roles:** Check **MPAdministrator**.
   - **Parent Menu Item Id:** Select **UserAdministration** from the drop-down list, or type this value into the field.
   - **Target Frame:** `Content`

   Click **Add** to add this menu item to the list in the left-hand pane of the wizard.
4) Create a menu item named RegistrationStatusGuest by entering the following information in the fields under **Menu Item Data**:
   - **Menu Item Id:** `RegistrationStatusGuest`
   - **Menu Item Title:** `RegistrationStatus`
   - **Action:** Select **UR/Guest.html** from the drop-down list, or type this value into the field.
   - **Image:** Select **/ePortal/Images/RegistrationStatus.gif** from the drop-down list, or type this value into the field.
   - **Roles:** Check **Guest**.
   - **Parent Menu Item Id:** Leave this field blank.
   - **Target Frame:** `Content`

   Click **Add** to add this menu item to the list in the left-hand pane of the wizard.

i. Click **Finish**. Solution Studio creates configuration files named B2BiTemplates.xml, B2BiTemplatesCMD.pdsc, and B2BiTemplatesPD.xml, places them in the Roles folder with the other roles artifacts, and closes the wizard.

8. Configure the servlet artifacts for the template as follows:
   a. Select the servlet folder in the project view.
   b. Start the WebSphere Artifact Configuration wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.
   c. In the **Configuration File** field, type an appropriate name for the configuration file. The name ServletConfig is used for this example.
   d. In the **Resource Type** field, select **Servlet**.
   e. Perform the following tasks for each servlet listed by the wizard:
      1) Select the servlet from the list of class files in the wizard. Each servlet is specified by its fully qualified name (for instance, \com\ibm\b2bi\templates\userregistration\servlets\Admin.class for the Admin servlet).
      2) In the **Web Application** field, type `UR`.
      3) In the **Alias** field, type the short name of the servlet (for example, `Admin` for the Admin.class servlet).
      4) Click **Save Configuration**.
   f. When you have entered information for all servlets, click **Finish**. Solution Studio creates an XML configuration file (in this case, ServletConfig.xml), places it in the servlet folder with the other data artifacts, and closes the wizard.

Publishing the template

To publish the user-registration template, perform the following steps:

1. Create four servers named BFM-WAS, BFM-WF, IM, and TAM-LDAP. To create a server, perform the following steps:
   a. In the publishing view (right-hand pane) of WebSphere Studio, right-click **Test**, then select **Insert → Server**. The Insert Server dialog box is displayed.
   b. Type the name of the server (`BFM-WAS`, `BFM-WF`, `IM`, or `TAM-LDAP`) in the **Server Name** field, then click **OK**. WebSphere Studio creates the server and displays it in the publishing view under the Test stage.
2. Copy (drag and drop) the Solution Studio artifacts from the project view to the appropriate server in the publishing view, as follows:
   - Copy all artifacts from the BFMAdapters subfolder of the Application folder and all artifacts from the Data folder to the BFM-WAS server.
   - Copy all artifacts from the WFAdapters subfolder of the Application folder and all artifacts from the BusinessProcess folder to the BFM-WF server.
   - Copy all artifacts from the web subfolder of the Client folder and from the servlet folder to the IM server. Do not copy the folders themselves, just the artifacts that the folders contain.

- Copy all artifacts from the Roles folder to the TAM-LDAP server.

3. Right-click **Test** in the publishing view, then select **Publish whole Project**. The Publishing Options dialog box is displayed.
4. Verify and change the options according to your preference, then click **OK**.
5. The Publishing dialog box appears and displays the progress of the publishing process. The following events occur during the publishing process:
   - As needed, WebSphere Studio prompts you to confirm the publication of certain files. Confirm them and click **OK** to continue the publishing process.
   - For each server to which artifacts are being published, Solution Studio launches the Deployment wizard and asks you to indicate the type of Business Integrator manager that is associated with the server. Select the following Business Integrator managers:
     – For the BFM-WAS server, select **BFM-WAS** from the list of managers.
     – For the BFM-WF server, select **BFM-WF** from the list of managers.
     – For the IM server, select **IM-IM** from the list of managers.
     – For the TAM-LDAP server, select **TAM-LDAP** from the list of managers.

     If you wish to enter manual instructions for a particular server, verify the file name in the **Instruction Manual(s)** field, then click **Browse** to browse to a file that already contains instructions, or click **Open** to launch the Windows Notepad and type in instructions. You must create the following manual instructions for the user-registration template's BFM-WAS server:
     – Create a note indicating that the *x*:*bi*\solution\SQL\registration.ddl file must be manually run *after* the solution is deployed to the BFM-WAS manager. In the path, *x* is the drive letter on which Business Integrator is installed and *bi* is the root directory of the Business Integrator installation.
     – Create a note indicating that the driver for the B2BiTemplatesDS database must be manually set as JTA enabled from the WebSphere Application Server Admin Console.

     When you have finished selecting the manager and adding any manual instructions, click **Finish**.

Note: The publishing process can take a considerable length of time to complete, depending on the number, size, and complexity of the artifacts being published and on the resources available on the Solution Studio client machine.

6. When WebSphere Studio finishes creating the package, it displays an information box that lists a success message and the location of the package file (for example, C:\URTest\URTest.zip). Click **OK** to acknowledge the success message and to close the information box.
7. WebSphere Studio creates a publishing report in HTML format and displays it in your system's default browser, launching the browser if necessary. The HTML file name is in the format *month-day-year-hour-minutes-seconds*-{AM|PM}.html, to indicate the date and time of file creation. If desired, review the publishing report for errors and warnings. Close the browser when you are finished reviewing the report.

   Note: This report does not reflect whether the artifacts were successfully configured in Solution Studio; it reflects only whether the artifacts were successfully published by WebSphere Studio.
8. Move the package file to a machine on which the Business Integrator run time is installed. Suggested ways of moving the package file include the following:
   - Diskette, CD-ROM, or other portable media
   - Mapped network drive
   - FTP

Deploying the user-registration template

To deploy the user-registration template after it is developed and published, perform the following steps:

1. Ensure that the required Business Integrator managers are running on the requisite number of machines. For the user-registration template, a minimum of four managers are required, as follows:
   - Business Flow Manager
   - Interaction Manager
   - Trust and Access Manager
   - WebProxy and WebSeal
2. On the Business Integrator run-time machine, start the Platform Console, then click **File → Deploy Solution** to start the Deploy Solution Package wizard. For additional information about the run time, the Platform Console, and the Deploy Solution Package wizard, see the *WebSphere Business Integrator Run Time* book.
3. Enter the fully qualified file name of the solution package (for example, `C:\WebSphere Business Integrator\solution\URTest.zip`), or click **Browse** to browse to the location of the solution package file. Click **Next**. The wizard displays details of the solution package.

4. Click **Start** to start deploying the solution package to the run time. A progress bar indicates the overall status of the deployment. Log messages indicate the solution element currently being deployed and the current stage of the deployment.
5. As needed, the wizard prompts you to carry out the manual instructions that you entered in Step 5 on page 19. Perform the instructions, then click **Completed**.
6. When the deployment is finished, a message is displayed and the Topology Repository status of the solution changes to Activated. Click **Finish** to exit from the Deploy Solution Package wizard.
7. Configure the WebSeal Web proxy to use a custom-challenge page by performing the following steps. These steps assume that a custom-challenge page has been created as described in the *WebSphere Business Integrator Run Time* book.
   a. On the machine that is running the Integration Manager, change to the *x*:\WebSphere\AppServer\hosts\default_host\web\webseal directory, where *x* is the drive letter on which WebSphere Application Server is installed, and locate the RegistrationForm.html file.
   b. Use FTP or another file-transfer facility to copy the RegistrationForm.html file to the machine on which WebSeal is running. Copy the file into the *x*:*WebSeal_install*\www\docs directory, where *x* and *WebSeal_install* are the drive letter and root directory, respectively, on which WebSeal is installed.
   c. Edit the RegistrationForm.html file that you copied to the WebSeal machine. Edit the following line:

      ```
      <form name="SIGNUP" method="post"
      action="/junction_name/UR/UserRegistration"
      onSubmit="return checkForm()">
      ```

      Change *junction_name* to the WebSeal junction name that was created during WebSeal's installation and configuration.

Running the user-registration template

To run the user-registration template after it is deployed, start a Web browser and enter the following URL:

```
https://webseal_host
```

where *webseal_host* is the name of the machine running WebSeal. The Web browser displays the custom-challenge page, as shown in Figure 4 on page 22.

Figure 4. The WebSphere Business Integrator custom-challenge page

You can perform the following tasks from the custom-challenge page:

- If you do not already have a user account, submit a new registration request. See "Submitting a new registration request" for details.
- If you already have a user account, log on to the solution and perform the tasks for which you are authorized. See "Logging onto the user-registration solution" on page 24 for details.

Submitting a new registration request

To submit a new registration request, click the **New User Registration** link in the left-hand pane of the custom-challenge pane. An HTML form, as shown in Figure 5 on page 23, is displayed and requests information such as your name, requested user ID and password, and the role or roles in which you want to participate in the solution. Enter the requested information and click **Submit Registration**. The browser then displays a confirmation page, as shown in Figure 6 on page 24.

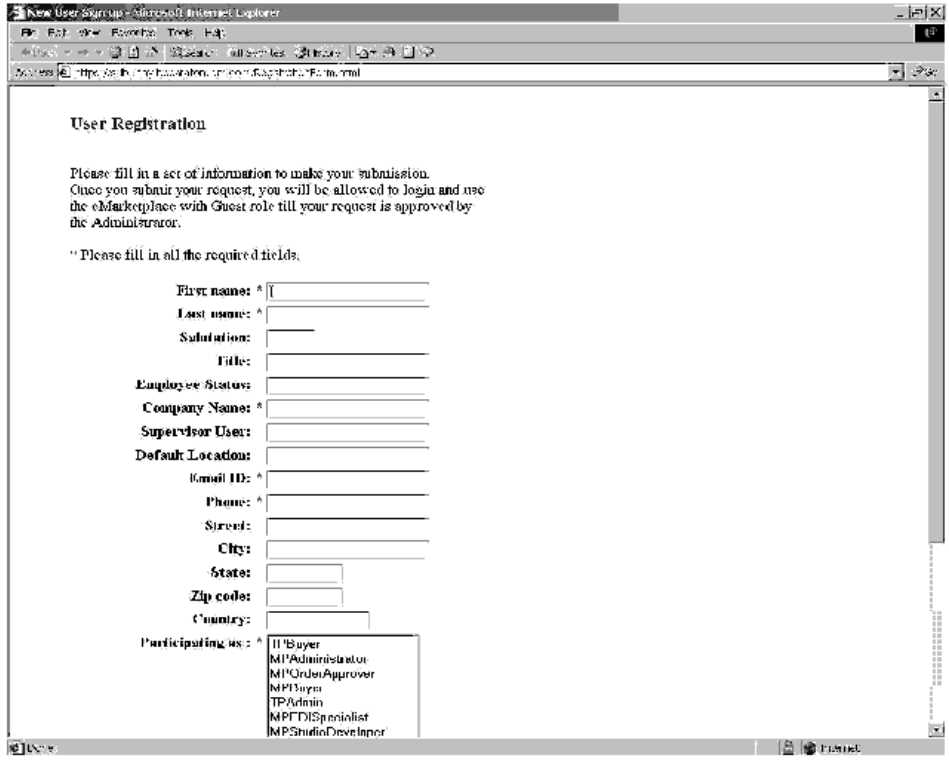

Figure 5. The new user registration page

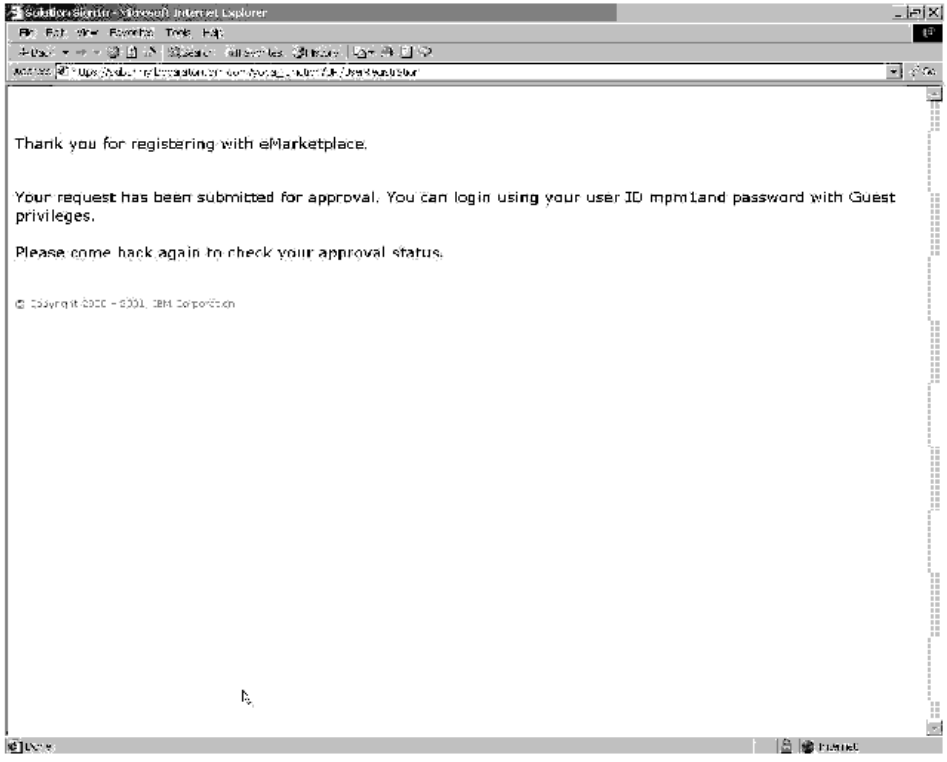

Figure 6. The registration confirmation page

The user-registration solution performs the following tasks when you submit a registration request:

- It places the submitted registration information into an entity bean called RegistrationBDO.
- It starts a workflow process to complete the approval. The workflow process is identified by the requested user ID.
- It enters you into LDAP and Policy Director with guest permissions. An administrator must approve your registration and perform additional steps to grant you additional permissions; see "Approving new user-registration requests" on page 25 for details.

Logging onto the user-registration solution

To log onto the user-registration solution, enter your user ID and password in the **Registered Users** panel on the custom-challenge form, then click **Login**. After you log in, Business Integrator displays links to the tasks that you can perform and an inbox that lists any new requests for your action, as shown in Figure 7 on page 25.

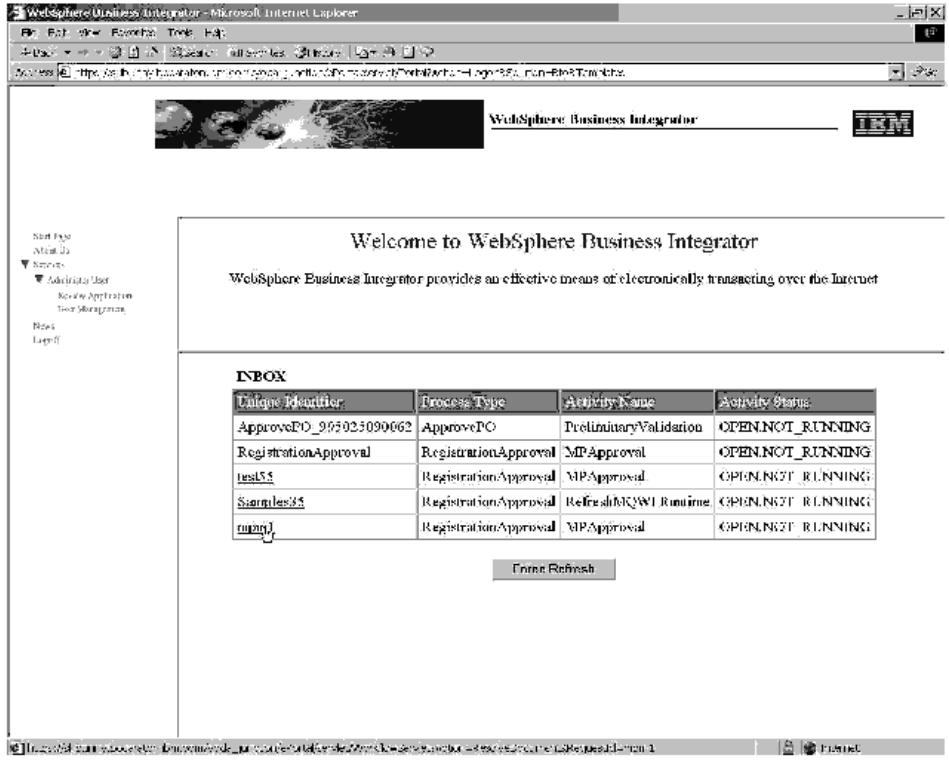

Figure 7. The WebSphere Business Integrator inbox page

Approving new user-registration requests

New user-registration requests must be approved by someone acting in the role of an administrator (MPAdministrator) for the solution. When the administrator logs onto the solution, his or her initial page shows an inbox with pending registration requests. To enable a new user, the administrator must review and approve the new user's request, as shown in Figure 8 on page 26. The administrator must refresh the inbox by clicking **Force Refresh**, and then approve the new user being added to the MQSeries Workflow build time and run time, as shown in Figure 9 on page 27, Figure 10 on page 28, and Figure 11 on page 29, respectively. When the work is approved, the PO-management template performs the work automatically.

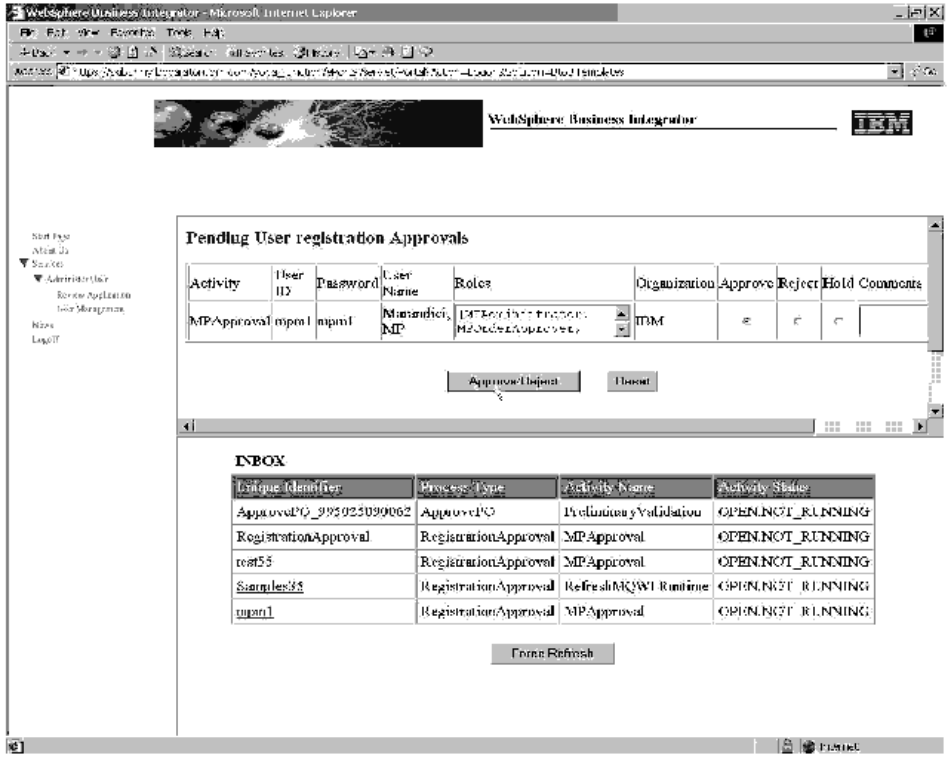

Figure 8. Approving a new user

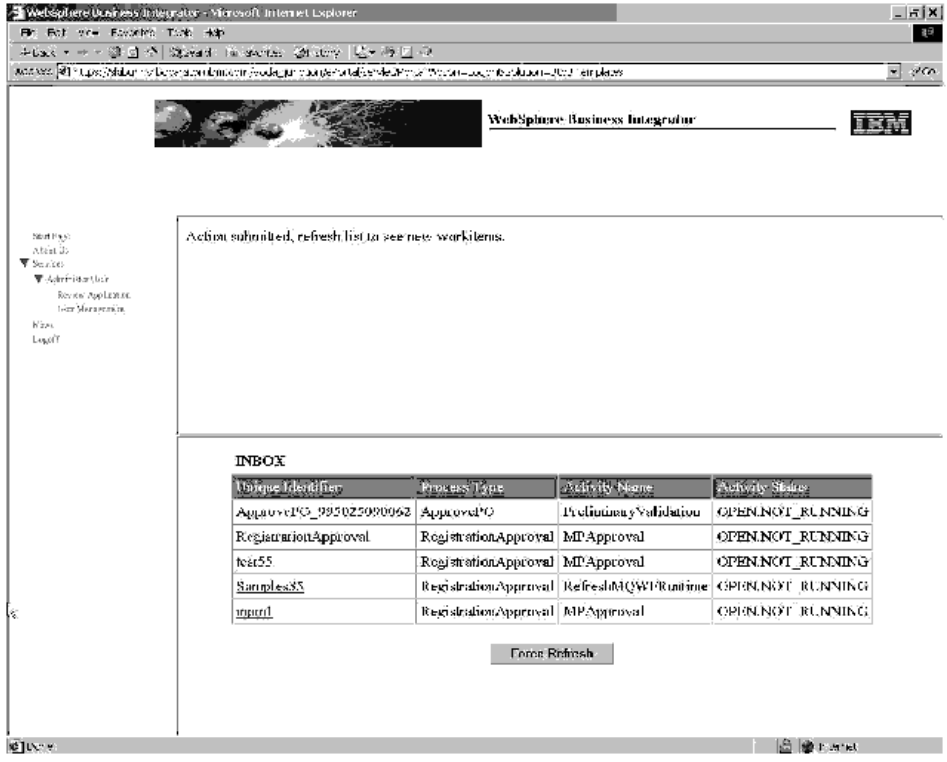

Figure 9. Refreshing the inbox

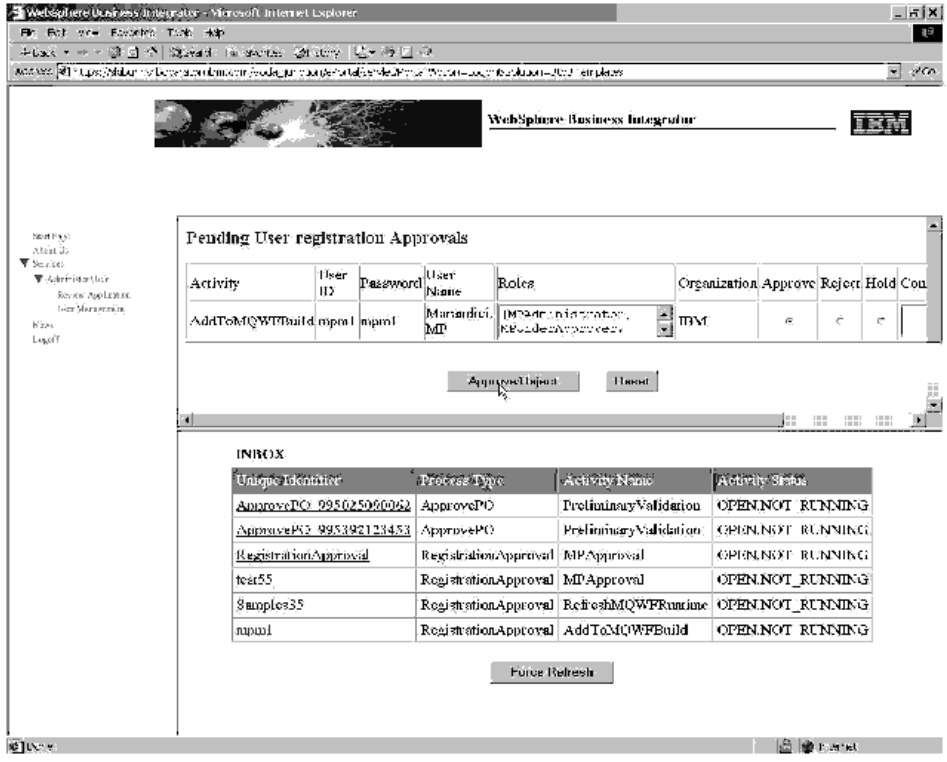

Figure 10. Approving the addition of a new user to the MQSeries Workflow build time

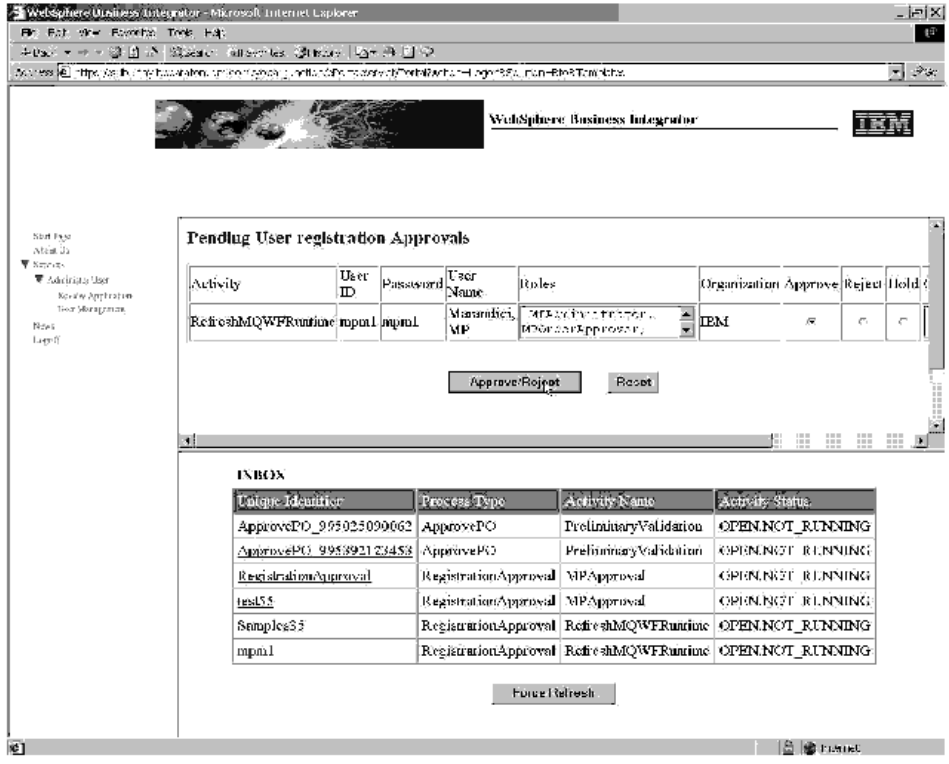

Figure 11. Approving the addition of a new user to the MQSeries Workflow run time

Note: If an administrator rejects a new user's registration request, the user is removed from LDAP and Policy Director, thus revoking the user's initial guest privileges.

Chapter 3. The purchase-order management template

This chapter discusses the purchase-order (PO) management template. It includes the following sections:

- "About the PO-management template"
- "Developing and publishing the PO-management template in Solution Studio" on page 41
- "Running the PO-management template" on page 51

About the PO-management template

The PO-management template is included with WebSphere Business Integrator to show how Business Integrator components can be used to design and implement a well defined solution. The need to manage purchase orders for an enterprise in the e-marketplace is used as an scenario example. "The programming model of the PO-management template" on page 35 discusses the programming model of the PO-management template, including state management and the Business Integrator run-time components that are used to run the template.

PO-management overview

This section provides a functional overview of the PO-management template.

The end-to-end process for managing POs is shown in Figure 12.

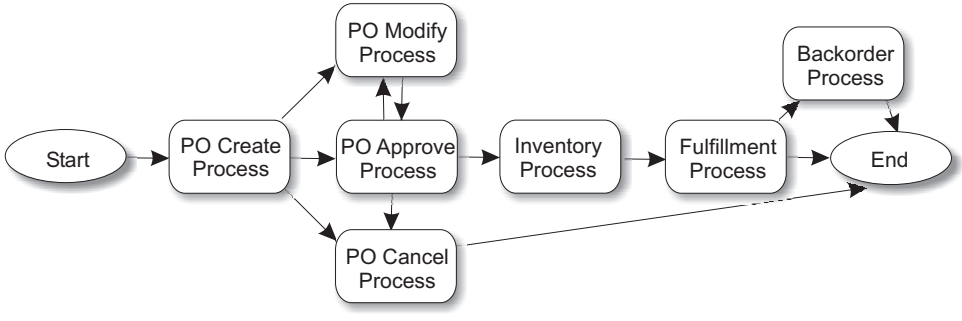

Figure 12. Process for managing purchase orders

The PO-management process has many subprocesses. As shown in Figure 12, the process begins when a user of the e-marketplace logs on in a role that is authorized to participate in the PO-management process. For the solution, these roles can include the following:

- Buyer
- Administrator
- Financial approver

Business Integrator's Interaction Manager renders the logon screen. The Logon servlet services the user logon request by authenticating the user ID and password through the Trust and Access Manager, checking the user's authorization based on the role with which the user is associated. When the user is successfully authenticated, the Interaction Manager renders a preconfigured screen on the user's browser, including a role-related navigation pane on the left side of the browser.

Based on the user's role, the rendered screen displays one or more of the following options:

- **Approve PO**
- **Create PO**
- **Modify PO**
- **Cancel PO**

For example, a user with the role of buyer is presented with the **Create PO** item in the navigation pane.

Simultaneously, the user's inbox, which is displayed in the bottom portion of the browser, is populated with work items that the user's role has authority to access. These items are generated by the POInboxServlet servlet, which sends a request to the Workflow Services session bean to determine whether any work is pending for the user's role.

Note: The buyer role does not include approving, modifying, or canceling POs. These actions are reserved for administrators.

Create a new PO

If the buyer chooses to create a new PO, the POHandlerServlet servlet presents the buyer with the Create Purchase Order JavaServer Page™. The buyer fills in required trading-partner information, creates line items, and submits the PO by clicking **Submit PO** on the form page, which sends a create request to the POHandlerServlet servlet.

The POHandlerServlet servlet sends a create request to the POServices session bean, which handles all requests for interaction with back-end components, including persistent entity beans (for example, PO and POLineItem). POServices creates an instance of the PO entity bean to create the PO as a persistent business data object (BDO). Because the request includes creation of PO line items, the PO entity bean sends a request to the POLineItem entity bean to create the line items as persistent BDOs.

When the needed BDOs are successfully created, an MQSeries Workflow process instance is created. This process instance is defined by the PO Approval process definition and is the first of three MQSeries Workflow process instanced used by the PO-management template. At this point, the purchase order is created and awaits approval. Approval can be performed only by a user with administrator authority, not by a user with buyer authority.

Modify a PO

If a user in the role of administrator chooses to modify an existing PO, he or she clicks **Review PO** in the navigation pane. This produces a screen that displays a list of POs that are in a modifiable state. When a PO is selected from this list, the POHandlerServlet servlet sends a request to the POServices bean to retrieve from the persistent BDO the PO data that can be modified. The data is then presented to the user. When the data is displayed, the user can make the appropriate changes (for instance, changing the quantity of an item) and click **Submit PO** to submit the modifications.

The POHandlerServlet servlet sends a change-PO request to the POServices bean to update the BDO data. The MQSeries Workflow process instance associated with the PO is terminated, and a new process instance is created to reflect the new parameters of the purchase order. When the new process instance is created, the POHandlerServlet servlet updates the list of purchase orders displayed on the screen.

The administrator's view shows the modified PO. The administrator can then choose to modify other POs that are pending action or return to perform other activities that he or she is authorized to perform.

Cancel a PO

If an administrator chooses to cancel a PO, he or she selects the PO from the list of POs he or she has access to. This list is presented by the POHandlerServlet servlet as described in "Modify a PO". The buyer clicks **Cancel PO** on the HTML form, which sends a request to cancel the PO to the POHandlerServlet servlet.

The POHandlerServlet servlet sends this request to POServices, which in turn initially delegates the cancel request to the PO BDO. The PO BDO tells the POLineItem BDO to delete the line items from the database. The PO BDO returns a successful or unsuccessful indication to POServices, then removes the PO. These results are returned to the POHandlerServlet servlet, which renders a revised HTML page.

Approve a PO

The PO-approval process is the process discussed for the PO-management solution. See "PO-approval task model" on page 34 for details. The role that

can perform this action is the administrator role. The buyer role cannot approve a purchase order in the PO-management solution.

A user with administrator privileges must approve a purchase order before it can be processed. The administrator selects a PO for review by clicking **Review PO** in the navigation pane, then by selecting the PO to be reviewed from a list of POs awaiting approval. The POHandlerServlet servlet processes this request by issuing a request to the POServices bean to lock (or *claim*) that PO to prevent other authorized users for working on it. The servlet then sends a change-PO request to the POServices bean to continue the approval process. The approval process from this point forward is automatic unless the PO requires financial approval; this is discussed in Step 16 on page 35. When the MQSeries Workflow process instance is completed for this purchase order, the POServices bean automatically starts the next MQSeries Workflow process in the PO-management template; this process is named CheckInventory.

The CheckInventory process is a simple MQSeries Workflow process that can be enhanced by a developer with the appropriate skills in using the MQSeries Workflow build-time tool. In this scenario, the CheckInventory process is completed quickly, and the POServices bean then creates and starts the PO-management template's final MQSeries Workflow process, which is named Fulfillment. This process is also a simple MQSeries Workflow process that can be enhanced in conjunction with an associated POServices state-machine command to perform simple tasks (for example, sending an email notifying a user of a shipment).

PO-approval task model

The PO-approval process is composed of several steps, including several activities that are identified in the ApprovePO workflow process definition. These steps, from logon to completion of the approval, can include the following:

1. An administrator enters the URL for the e-marketplace's welcome screen.
2. The Logon servlet renders the logon screen.
3. The administrator enters the appropriate user ID and password and is authenticated and authorized.
4. The Interaction Manager renders the initial screen as determined by the user's role.
5. The administrator selects **Purchase Order Management**, then **Review PO** from the menu.
6. The request is sent to the POHandlerServlet servlet.
7. The servlet issues a request to the Workflow Services session bean to retrieve any work items for this user's role, such as POs that require approval. If outstanding work exists, a list of POs is returned to the servlet.

8. The servlet renders a JavaServer Page that lists any POs that are awaiting action from an administrator.
9. The administrator selects the PO to approve from the list.
10. A request is sent to the POHandlerServlet servlet for the particular PO (identified by a PO identifier in the BDO), which is forwarded to POServices. This also initiates the processing of activities for the Approve PO workflow process instance by locking (or claiming) the Preliminary Validation, the first activity of the process instance
11. POServices forwards the PO data to the POHandlerServlet servlet, which renders a JavaServer Page of the PO with the options of approving or rejecting it and the option of canceling any action on the form and returning to the list of POs.
12. The administrator approves the PO, which sends a request to POServices to complete the Preliminary Validation activity.
13. The next activity, Get PO Amount, is programmed to occur automatically.
14. An MQSeries User Process Execution Services (UPES) message is sent to the JMS Listener's worker to call a method on the POServices bean.
15. This method retrieves the cost amount of the PO and passes the amount to the TestPOAmount activity.
16. Logic in the Test PO Amount activity determines the path the workflow takes, based on whether the amount is greater or less than $10,000. If the amount of the PO is less than $10,000, the PO Approve Success activity is initiated. If the amount is greater than $10,000, the FA Approval activity is initiated; this places the PO in a state for the financial approver to review and manually act on by using the FA Approve form.
17. If the amount is less than $10,000, the PO Approve Success activity sends a UPES message that results in a call to a method on POServices to set the state of the PO to approved, and the PO-approval process completes.

The programming model of the PO-management template

This section discusses the programming model of the PO-management template, including a detailed analysis of the template's PO-creation process.

The PO-management template uses the following Business Integrator and related facilities:

- Business Flow Manager
- Interaction Manager
- DataInterchange gateway, in conjunction with MQSeries and MQSeries Adapter Kernel
- MQSeries Workflow

For a complete discussion of these facilities, see the *WebSphere Business Integrator Installation Guide*. This section discusses some general considerations of the facilities, particularly the Business Flow Manager, as they pertain to the PO-management template.

The Business Flow Manager runs in WebSphere Application Server Advanced Edition. It consists of the following elements:

- WebSphere Messaging services, the Java Message Service (JMS) Listener, and associated worker beans. These elements use the services of MQSeries and MQSeries Adapter Kernel to provide asynchronous communication from endpoints and gateways.

  The JMS Listener provides message routing based on message type, sending each message object it receives to the particular worker bean that is associated with the message object's type. For instance, if a message object has a body type of create, the JMS Listener sends the message body to the worker bean that handles create requests. The worker bean in turn calls the appropriate method on the POServices session bean to execute the request. For more information on the JMS Listener, see the MQSeries Adapter Kernel *Quick Beginnings* book.

  MQSeries provides guaranteed delivery of messages among the solution's elements. At least one set of MQSeries queues exists for each of the following communications paths:

  – To and from the DataInterchange (DI) gateway

  – To and from MQSeries Workflow. The messages exchanged with MQSeries Workflow are User Process Execution Services (UPES) messages, which are used to trigger processes in the receiving application.
- WebSphere Workflow services, a Joint Flow-based application programming interface (API) that provides access to Workflow Management Coalition (WfMC)-compliant workflow engines such as MQSeries Workflow.
- Solution Management services, including audit, exception handling, logging, and monitoring of business processes. These services are not specific to the Business Flow Manager; they are provided with all components of the Business Integrator run time.

Along with these elements, the PO-management template uses a *state machine*—that is, software that monitors the transition of data from one state to another, providing guards, entry and exit actions, and transactional guarantees (that is, commitment of successfully completed units of work and rollback of unsuccessfully completed units of work). The data on which a state machine acts is contained in what is known as an *adaptive document*. The interaction between the state machine and the adaptive document is dynamic; that is, the current state of the adaptive document can determine the action that the state machine takes upon it.

The PO-management template utilizes the part of the Business Integrator programming model that enables the state machine to be implemented as a session bean and the adaptive document to be implemented as an entity bean. Specific components involved in the model are as follows:

- A *business data object*, an entity bean that encapsulates the persistent data that comprises the business object—for example, a purchase order.
- The *adaptive business service*, a session bean that wraps a set of operations collectively called a *business service*. Each operation (or *business-service operation*) can be implemented by using command beans. This development pattern enables well defined functionality to be developed and implemented in a variety of ways (for instance, a graphical tool that produces a generated command that is realized as subclasses of a CommandBean Java class). The PO-management template currently uses the CommandBean model in both the POServices session bean and the actions of the state machine. As an example, an adaptive business service named purchase order can wrap the following business service operations:
  - Create purchase order
  - Update purchase order
  - Delete purchase order

  To create a purchase order, you call the create purchase order method on the adaptive business service; this in turn runs the specific command bean to create a purchase order.

  The adaptive business service is initially invoked by an *operations signature message*, which takes the form of either a message delivered by MQSeries and MQSeries Adapter Kernel or a remote method invocation (RMI) from the Interaction Manager. Operations signature messages can be issued by business-service operations or by other parts of the solution.
- The state machine is an important component of adaptive documents and of the programming model in general. State-based behavior is a powerful and necessary aspect of real-time business integration. The ability to modify an already created purchase order (that is, to cause an *event*) depends largely on the location of the purchase order in its continuing processing (that is, its *state*). For example, a purchase order that is already in a state of approved cannot be changed; however, it can be canceled. The combination of a state-machine implementation with the versatility of an adaptive document enables these complex scenarios to exist.

  Command beans are readily utilized by the state machine to initiate actions based on the state of the adaptive document. When the state machine is invoked based on a specific event, it obtains the state of the adaptive document, then uses the combination of the event and the state to execute the configured action.

The business service provided with the PO-management template is called Purchase Order Management Services. The purchase-order create operation of this business service is described in "Model of creating a purchase order".

Model of creating a purchase order

Figure 13 on page 39 shows a diagram of the process to create a purchase order, only one of the business operations possible with Purchase Order Management Services. Other processes include modifying a purchase order, canceling a purchase order, and checking inventory.

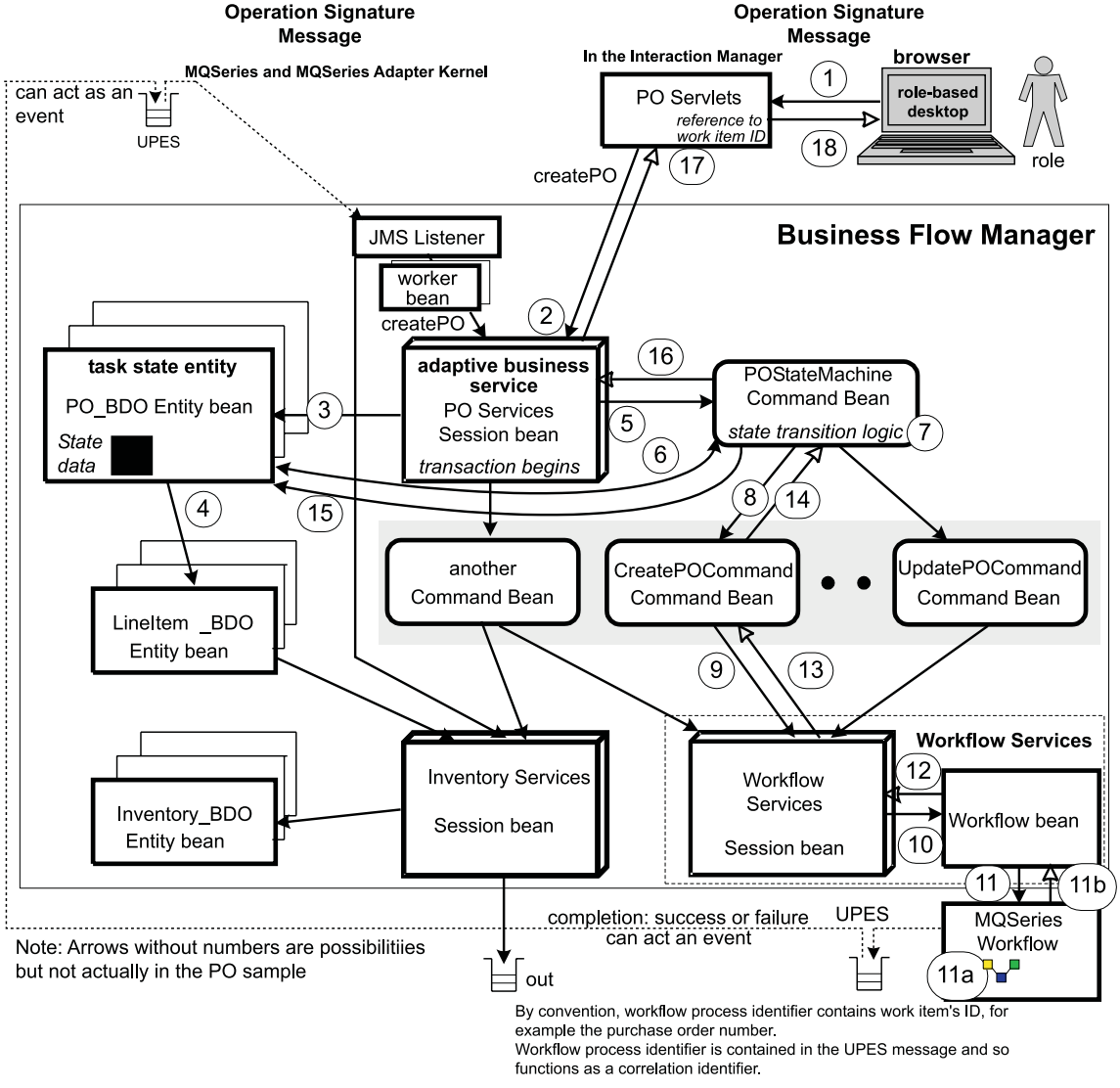

Figure 13. Annotated diagram of the create-PO process in Purchase Order Management Services

The create-PO business operation works as follows. Note that the steps listed here correspond to the numbers in Figure 13.

1. A purchase order is created as follows: A user at a browser-based session makes a request to create the purchase order. The user has previously been authenticated and authorized to Business Integrator through the user-registration template. The form of the request is an RMI. The request in the browser triggers the PO servlet, which runs as part of the

Interaction Manager in WebSphere Application Server. The RMI from the browser functions as the operation signature message.

2. The PO servlet invokes a session bean named POServices that serves as the adaptive business service. All work from this step to step 11 is under transactional control provided by WebSphere Application Server's EJB container. For more information on WebSphere Application Server and transactional controls, see the *Introduction to WebSphere Application Server* and *Writing Enterprise Beans in WebSphere* books, which are provided with that product's Info Center.
3. The POServices session bean invokes an entity bean named PO to hold persistent data about the purchase order. The PO bean serves as the primary business data object, holding state data about the purchase order.
4. The PO bean creates an instance of an entity bean named LineItem_BDO to hold line-item data for the purchase order.
5. The POServices session bean creates an instance of the POStateMachine command bean, which serves as the state machine, holding state-transition logic for the purchase order.
6. The PO and POStateMachine beans communicate to ensure that all the data is in a consistent state before additional work begins. Note that the PO and POStateMachine beans remain in constant contact so that data held by the PO bean is updated according to the state-transition logic held by the POStateMachine bean.
7. The POStateMachine bean initializes its state-transition logic.
8. Based on the transition rules that you defined, the POStateMachine bean invokes a command bean named CreatePOCommand to create the purchase order.
9. The CreatePOCommand bean invokes a session bean in WebSphere Workflow Services, a subservice hosted by the Business Flow Manager inside WebSphere Application Server.
10. The WebSphere Workflow Services session bean invokes a workflow bean, a specialized type of bean that works with MQSeries Workflow.
11. The workflow bean passes data to MQSeries Workflow, which performs one or more of the following steps, as appropriate for the data. Note that the data at this point passes out of WebSphere Workflow Services, the Business Flow Manager, and WebSphere Application Server.
    a. If the specified activity is configured with the UPES property enabled, a UPES message is routed and used to trigger additional events outside of the Business Flow Manager. At this point, the data does not return to the Business Flow Manager, although the UPES message can be used to trigger events that directly or indirectly invoke additional work in the Business Flow Manager. This step can be taken if, for example, the create-PO request failed for any reason; the UPES

message can then be used as a trigger for additional events (for instance, restarting the process or informing the user of failure).

b. It performs the requested work and returns the data to the workflow bean in WebSphere Workflow Services. The remaining steps in this discussion assume that the data is returned from MQSeries Workflow to the WebSphere Workflow Services workflow bean.

12. The workflow bean returns any updated data to the WebSphere Workflow Services session bean.
13. The WebSphere Workflow services session bean returns the updated data to the CreatePOCommand bean.
14. The CreatePOCommand bean returns the updated data to the POStateMachine bean, which evaluates the returned data according to the state logic with which it has been programmed.
15. The POStateMachine bean performs the appropriate state transition on the PO bean. Recall that the data held by the PO bean is persistent and kept under transactional control by WebSphere Application Server.
16. The POStateMachine returns control to the POServices bean.
17. Assuming that the initial request came from the PO servlet running in the Interaction Manager, the POServices bean relays the updated data to the PO servlet. At this point the data passes out of the Business Flow Manager and back to the Interaction Manager.
18. The PO servlet updates the user's browser, informing the user of the success or failure of the request along with any other data that was created or updated by the request.

Developing and publishing the PO-management template in Solution Studio

This section discusses how to develop and publish the template. Development instructions are provided in "Developing the template"; publishing instructions are provided in "Publishing the template" on page 49.

Note: The PO-management template depends upon the user-registration template for user authentication and authorization. The user-registration template must therefore be deployed before the PO-management template can be run successfully. See "Chapter 2. The user-registration template" on page 3 for details on developing, deploying, and running the user-registration template.

Developing the template

To develop the PO-management template, perform the following steps. Note that the Solution Studio wizards include many fields; if a field is not referenced in the following procedure, it is not needed for the successful development of the template, regardless of whether it contains a default value or is blank by default. You can safely disregard any field that is not referenced

in the following procedure. Also note that the artifacts for the PO-management template do not require the extensive configuration in Solution Studio that is required by the artifacts for the user-registration template.

1. Start WebSphere Studio.
2. Create a new project based on the PO-management template, as follows:
   a. If a dialog box appears and asks what you want to do, click **Create new**; otherwise, click **File → New Project**. The New Project dialog box appears.
   b. Enter a name for the project in the **Project Name** field; for example, `POTest`.
   c. For the **Project Folder** field, click **Browse** and browse to the shared folder on the ClearCase server machine that holds project files. For information on setting up a ClearCase server, see the *WebSphere Studio Business Integrator Extensions Installation Guide*.
   d. In the **Create using** drop-down field, select **Purchase Order**.
   e. Click **OK**.

   The project view (left-hand pane) in WebSphere Studio now displays the name of your project and the hierarchy of folders contained in the project. These folders contain the artifacts that make up the PO-management template.
3. Create a JMS Listener adapter for the template as follows:
   a. Select the Application folder in the project view.
   b. Start the Application Configuration wizard by clicking **Tools → Wizards → Application Configuration Wizard**. The wizard opens to the Application page.
   c. In the **Application Id** field, type `UPESPOAPP`.
   d. In the **Trace** field, click **On**, and verify that the **Trace Level** field is set to a value of `-1`. Ensure that the **Trace Client** field is left blank.
   e. In the **Logging** field, ensure that **Disabled** is selected.
   f. Ensure that the **Logon Info Class** field is left blank.
   g. Ensure that the **Queue Manager** field is left blank.
   h. In the **Use remote queue manager to send messages?** field, ensure that **No** is selected.
   i. Ensure that the **Host Name**, **Port Number**, and **Channel Name** fields are left blank.
   j. In the **Queue Manager** field under **JMS Queue Connection Factory**, type `DEFAULT.BPIP`. If the queue manager you are using has a different name, type it into this field.

      Alternatively, click **Define**. The Component Definition for JMS QM Host window opens. Select a valid component and subcomponent from

the drop-down boxes in the **Component** and **Subcomponent** fields, respectively. Click **OK** to close the window.

k. In the **Connection Factory** field, type `DEFAULT.BPIP`. If the connection factory you are using has a different name, type it into this field.

l. Under **Message Types**, click **New**. The Adapter Routing wizard starts.

m. Ensure that the General page on the Adapter Routing wizard is displayed. In the **Message Body Category** field, type `DEFAULT`.

n. In the **Message Body Type** field, type `DEFAULT`.

o. In the **Receive Mode** field, select **JMS**. The wizard now displays the **Message Formatter**, **Receive Timeout**, **Receive Queue**, **Error Queue**, and **Reply Queue** fields.

p. In the **Message Formatter** field, ensure that **com.ibm.epic.adapters.eak.support.JMSNMXMLFormatter** is selected.

q. In the **Receive Timeout** field, type `30000`.

r. In the **Receive Queue** field, type `jmsPOMgmt`.

s. In the **Error Queue** field, type `JMSErrorQueue`.

t. Click **Command** or **Next**. The Command page of the Adapter Routing wizard is displayed.

u. In the **Command Type** field, select **Enterprise Bean** from the drop-down list.

v. In the **Command Class Name** field, type `com/ibm/b2bi/templates/po/POServices`.

w. In the **TDC Mapper Class** field, type `com.ibm.mqao.mqak.ejbclient.TDCGenericMapper`.

x. In the **Method to Invoke** field, type `dispatchJMSMessage`.

y. In the **Parameter Type** field, type `com.ibm.mqao.mqak.ejbclient.TerminalDataContainer`.

z. In the **URL** field, ensure that **IIOP:///** is selected.

aa. In the **Initial Context Factory** field, ensure that **com.ibm.ejs.ns.jndi.CNInitialContextFactory** is selected.

ab. Click **Worker** or **Next**. The Worker dialog box of the Adapter Routing wizard opens.

ac. In the **Bean Name** field, type `POMgmtWorker`.

ad. In the **JNDI Home Name** field, type `POMgmtWorker`.

ae. In the **Container Name** field, type `POMgmtJMSContainer`.

af. In the **Maximum Retries** field, type `0`.

ag. In the **Maximum Sessions** field, type `1`.

ah. Click **Finish** on the Adapter Routing wizard. The wizard closes and returns you to the Application Configuration wizard.

ai. Click **Finish** on the Application Configuration wizard. Solution Studio creates the following files:
   - An adapter configuration file named UPESPOAPP.appcfg
   - An XML instructions file named UPESPOAPP_inst.xml
   - XML configuration files for the workers (POMgmtWorker.xml and UPESPOAPP_workers.xml)
   - A JMS configuration file named UPESPOAPP.scp

   Solution Studio places these files in the Application folder, then closes the wizard.

4. Configure an MQSeries queue for the Application folder's BFMAdapters subfolder as follows:
   a. Open the Application folder in the project view, then select the BFMAdapters folder.
   b. Start the MQSeries Queue Definition wizard by clicking **Tools → Wizards → MQSeries Queue Definition Wizard**. The wizard opens to the General page.
   c. In the **Queue Manager Name** field, click **Define**. The Component Definition for QM Host window opens. For the **Component** field, select **BFM** from the drop-down list; for the **Subcomponent** field, select **WAS** from the drop-down list. Click **OK** to close the Component Definition for QM Host window.
   d. In the **Queue Name** field, type `POMgmt`.
   e. Accept the following defaults listed on the General page:
      - **Queue Type:** `Local`
      - **Default Priority:** `0`
      - **Default Persistence:** `Not Persistent`
      - **Put Messages:** `Allowed`
      - **Get Messages:** `Allowed`
      - **Scope:** `Queue Manager`
      - **Usage:** `Normal`

      Accept all other defaults listed on the other pages of this wizard.
   f. Click **Finish**. Solution Studio creates an XML configuration file named POMgmt.xml and an MQSeries configuration file named POMgmt.mqsc. Solution Studio places these files in the BFMAdapters subfolder of the Application folder with the other artifacts, then closes the wizard.
5. Manually create two files for the Application folder's BFMAdapters subfolder as follows:

a. Use a text editor such as Notepad to create two files named WFPOChannel.mqsc and WFPOChannel.xml. Type the following into the WFPOChannel.mqsc file:

```
DEFINE CHANNEL('WFPOChannel')+
CHLTYPE(RCVR) +
TRPTYPE(TCP)
```

Type the following into the WFPOChannel.xml file:

```
<?XML version="1.0"?>
<instructions>
<instruction>
<type>MQ</type>
<method>runMQSCScript</method>
<parameter type="String" value="@{top:comp=BFM,subcomp=WAS,
type=ComputerSystem,name=hostname,case=upper}.@{top:type=TopologyType,
name=mqClusterName}"></parameter>
</instruction>
<manual_instructions>
<manual_instruction>Queue manager @{top:comp=BFM,subcomp=WAS,
type=ComputerSystem,name=hostname,case=upper}.@{top:type=TopologyType,
name=mqClusterName} needs to exist or be created.
For example, the following command line could be executed:
 crtmqm @{top:comp=BFM,subcomp=WAS,type=ComputerSystem,
 name=hostname,case=upper}.@{top:type=TopologyType,name=mqClusterName}
For information on creating a queue manager, refer to the MQSeries
System Administration manual.
</manual_instruction>
</manual_instructions>
</instructions>
```

Save the files in a convenient location.

b. Open the Application folder in the project view and select the BFMAdapters subfolder.

c. For each of the two files you created, do the following:

1) Click **Insert → File**. The Insert File window opens to the Create New page.

2) Click **Use Existing**, then click **Browse**. Browse to the location in which you saved the file, then select the file. Click **OK**. WebSphere Studio imports the file into the BFMAdapters folder and closes the Insert File window.

6. Configure the business-process artifacts for the template as follows:

a. Select the BusinessProcess folder in the project view.

b. Start the MQSeries Queue Definition wizard by clicking **Tools → Wizards → MQSeries Queue Definition Wizard**. The wizard opens to the General page.

c. In the **Queue Manager Name** field, click **Define**. The Component Definition for QM Host window opens. For the **Component** field,

select **BFM** from the drop-down list; for the **Subcomponent** field, select **WAS** from the drop-down list. Click **OK** to close the Component Definition for QM Host window.

d. In the **Queue Name** field, type `WFPOServices`.

e. Accept the following defaults listed on the General page:

- **Queue Type:** `Local`
- **Default Priority:** `0`
- **Default Persistence:** `Not Persistent`
- **Put Messages:** `Allowed`
- **Get Messages:** `Allowed`
- **Scope:** `Queue Manager`
- **Usage:** `Normal`

Accept all other defaults listed on the other pages of this wizard.

f. Click **Finish**. Solution Studio creates an XML configuration file named WFPOServices.xml and an MQSeries configuration file named WFPOServices.mqsc. Solution Studio places these files in the BusinessProcess folder with the other artifacts, then closes the wizard.

g. Use a text editor such as Notepad to create two files named WFPOChannel.mqsc and WFPOChannel.xml. Type the following into the WFPOChannel.mqsc file:

```
DEFINE QLOCAL('WFPOTransQ') REPLACE +
DEFPRTY(0) +
DEFPSIST(NO) +
DESCR(' ') +
PUT(ENABLED) +
BOQNAME(' ') +
BOTHRESH(0) +
DEFBIND(OPEN) +
DEFSOPT(SHARED) +
DISTL(NO) +
GET(ENABLED) +
INITQ(' ') +
MAXDEPTH(5000) +
MAXMSGL(4194304) +
MSGDLVSQ(PRIORITY) +
NOHARDENBO +
SHARE +
NOTRIGGER +
PROCESS(' ') +
QDEPTHHI(80) +
QDEPTHLO(20) +
QDPHIEV(DISABLED) +
QDPLOEV(DISABLED) +
QDPMAXEV(ENABLED) +
QSVCIEV(NONE) +
QSVCINT(999999999) +
RETINTVL(999999999) +
```

```
SCOPE(QMGR) +
TRIGDATA(' ') +
TRIGDPTH(1) +
TRIGMPRI(0) +
TRIGTYPE(FIRST) +
USAGE(XMITQ)

DEFINE CHANNEL('WFPOChannel') +
CHLTYPE(SDR) +
CONNAME('@{computername(BFM,WAS)}(1414)') +
TRPTYPE(TCP) +
XMITQ('WFPOTransQ')
```

Type the following into the WFPOChannel.xml file:

```
<?XML version="1.0"?>
<instructions>
<instruction>
<type>MQ</type>
<method>runMQSCScript</method>
<parameter type="String" value="FMCQMSV"></parameter>
<parameter type="String" value="WFPOChannel.mqsc"></parameter>
</instruction>
<manual_instructions>
<manual_instruction>Queue manager FMCQMSV needs to exist or be created.
For example, the following command line could be executed:
 crtmqm FMCQMSV
For information on creating a queue manager, refer to the MQSeries
System Administration manual.
</manual_instruction>
</manual_instructions>
</instructions>
```

Ensure that the value listed as FMCQMSV in this file matches the name of the queue manager that was defined for MQSeries Workflow during that product's installation.

Save the files in a convenient location.

h. For each of the two files you created, do the following:
   1) Click **Insert → File**. The Insert File window opens to the Create New page.
   2) Click **Use Existing**, then click **Browse**. Browse to the location in which you saved the file, then select the file. Click **OK**. WebSphere Studio imports the file into the BusinessProcess folder and closes the Insert File window.

7. Configure the Client artifacts for the template as follows:
   a. Select the Client folder in the project view, then open the web folder.
   b. Start the WebSphere Artifact Configuration wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.

c. In the **Configuration File** field, type an appropriate name for the configuration file. The name WebDocConfig is used for this example.
d. In the **Resource Type** field, select **Document**.
e. Click **Select All** to select all the artifacts in the folder.
f. In the **Web Application** field, type `PO`.
g. Click **Finish**. Solution Studio creates an XML configuration file (in this case, WebDocConfig.xml), places it in the web folder with the other client artifacts, and closes the wizard.

8. Configure the data artifacts for the template as follows:
   a. Select the Data folder in the project view.
   b. Start the WebSphere Artifact Configuration Wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.
   c. In the **Configuration File** field, type an appropriate name for the configuration file. The name EJBConfig is used for this example.
   d. In the **Resource Type** field, select **Enterprise Bean**.
   e. Select the **POManagementEJBSource.jar** file in the list that appears.
   f. In the **Bean Type** field, click **Entity Bean**.
   g. In the **Container Name** field, type `PO`.
   h. In the **Data Source** field, type `PODataSource`.
   i. In the **Persistence** field, click **Bean Managed**.
   j. In the **Create Data Source?** field, click **Yes**.
   k. In the **Database Name** field, type `B2BiTMPL`.
   l. Click **Finish**. Solution Studio creates an XML configuration file (in this case, EJBConfig.xml), places it in the Data folder with the other data artifacts, and closes the wizard.
9. Configure the servlet artifacts for the template as follows:
   a. Select the servlet folder in the project view.
   b. Start the WebSphere Artifact Configuration wizard by clicking **Tools → Wizards → WebSphere Artifact Configuration Wizard**. The wizard opens.
   c. In the **Configuration File** field, type an appropriate name for the configuration file. The name POServletConfig is used for this example.
   d. In the **Resource Type** field, select **Servlet**.
   e. Perform the following tasks for each servlet listed by the wizard:
      1) Select the servlet from the list of class files in the wizard. Each servlet is specified by its fully qualified name (for instance, \com\ibm\b2bi\templates\po\servlets\POHandlerServlet.class for the POHandlerServlet servlet).
      2) In the **Web Application** field, type `PO`.

3) In the **Alias** field, type the short name of the servlet (for example, `POHandlerServlet` for the POHandlerServlet.class servlet).
4) Click **Save Configuration**.

f. When you have entered information for all servlets, click **Finish**. Solution Studio creates an XML configuration file (in this case, POServletConfig.xml), places it in the servlet folder with the other data artifacts, and closes the wizard.

Publishing the template

To publish the PO-management template, perform the following steps:

1. Create four servers named BFM-WAS, BFM-WF, IM, and TAM-LDAP. To create a server, perform the following steps:
   a. In the publishing view (right-hand pane) of WebSphere Studio, right-click **Test**, then select **Insert → Server**. The Insert Server dialog box is displayed.
   b. Type the name of the server (`BFM-WAS`, `BFM-WF`, `IM`, or `TAM-LDAP`) in the **Server Name** field, then click **OK**. WebSphere Studio creates the server and displays it in the publishing view under the Test stage.
2. Copy (drag and drop) the Solution Studio artifacts from the project view to the appropriate server in the publishing view, as follows:
   - Copy all artifacts that are contained in the BFMAdapters subfolder of the Application folder and all artifacts that are contained in the Data folder to the BFM-WAS server.
   - Copy all artifacts that are contained in the BusinessProcess folder to the BFM-WF server.
   - Copy all artifacts that are contained in the Client and servlet folders to the IM server.
   - Copy all artifacts that are contained in the Roles folder to the TAM-LDAP server.
3. Right-click **Test** in the publishing view, then select **Publish whole Project**. The Publishing Options dialog box is displayed.
4. Verify and change the options according to your preference, then click **OK**.
5. The Publishing dialog box appears and displays the progress of the publishing process. The following events occur during the publishing process:
   - As needed, WebSphere Studio prompts you to confirm the publication of certain files. Confirm them and click **OK** to continue the publishing process.
   - For each server to which artifacts are being published, Solution Studio launches the Deployment wizard and asks you to indicate the type of Business Integrator manager that is associated with the server. Select the following Business Integrator managers:

- For the BFM-WAS server, select **BFM-WAS** from the list of managers.
- For the BFM-WF server, select **BFM-WF** from the list of managers.
- For the IM server, select **IM-IM** from the list of managers.
- For the TAM-LDAP server, select **TAM-LDAP** from the list of managers.

If you wish to enter manual instructions for a particular server, verify the file name in the **Instruction Manual(s)** field, then click **Browse** to browse to a file that already contains instructions, or click **Open** to launch the Windows Notepad and type in instructions.

Create a manual instruction for the PO-management template's BFM-WAS server instructing the user to manually run the *x*:*bi*\solution\SQL\B2BiTemplatesPOMgmt.ddl file *after* the solution is deployed to the BFM-WAS manager. In the path, *x* is the drive letter on which Business Integrator is installed and *bi* is the root directory of the Business Integrator installation. When you have finished selecting the manager and adding manual instructions, click **Finish**.

Note: The publishing process can take a considerable length of time to complete, depending on the number, size, and complexity of the artifacts being published and on the resources available on the Solution Studio client machine.

6. When WebSphere Studio finishes creating the package, it displays an information box that lists a success message and the location of the package file (for example, C:\POTest\POTest.zip). Click **OK** to acknowledge the success message and to close the information box.
7. WebSphere Studio creates a publishing report in HTML format and displays it in your system's default browser, launching the browser if necessary. The HTML file name is in the format *month-day-year-hour-minutes-seconds*-{AM|PM}.html, to indicate the date and time of file creation. If you wish, review the publishing report for errors and warnings. Close the browser when you are finished reviewing the report.
8. Move the package file to a machine on which the Business Integrator run time is installed. Suggested ways of moving the package file include the following:
   - Diskette, CD-ROM, or other portable media
   - Mapped network drive
   - FTP

Deploying the PO-management template

To deploy the PO-management template after it is developed and published, perform the following steps:

1. Ensure that the required Business Integrator managers are running on the requisite number of machines. For the PO-management template, a minimum of three managers are required, as follows:
   - Business Flow Manager
   - Interaction Manager
   - Trust and Access Manager

   Additionally, the user-registration template must be deployed. The user-registration template requires a machine running WebProxy and WebSeal; see "Deploying the user-registration template" on page 20 for details.

   Note: For illustrative purposes, the PO-management template includes DataInterchange connections; however, the template itself does not require messages to be exchanged with DataInterchange. If you have a functioning DataInterchange machine, you can use it with the PO-management template, but a DataInterchange machine is not required to run the template.
2. On the Business Integrator run-time machine, start the Platform Console, then click **File → Deploy Solution** to start the Deploy Solution Package wizard. For additional information about the run time, the Platform Console, and the Deploy Solution Package wizard, see the *WebSphere Business Integrator Run Time* book.
3. Enter the fully qualified file name of the solution package (for example, `C:\WebSphere Business Integrator\solution\POTest.zip`), or click **Browse** to browse to the location of the solution package file). Click **Next**. The wizard displays details of the solution package.
4. Click **Start** to start deploying the solution package to the run time. A progress bar indicates the overall status of the deployment. Log messages indicate the solution element currently being deployed and the current stage of the deployment.
5. As needed, the wizard prompts you to carry out any manual instructions that you entered during the publication phase. Perform the instructions, then click **Completed**.
6. When the deployment is finished, a message is displayed and the Topology Repository status of the solution changes to Activated. Click **Finish** to close the Deploy Solution Package wizard.

Running the PO-management template

After the PO-management template is deployed, you can go directly to a number of screens to perform various tasks, including the following:

- Creating a purchase order; see "Creating a purchase order" on page 52.
- Approving a purchase order; see "Approving a purchase order" on page 52.

- Obtaining financial approval for a purchase order with a total cost greater than $10,000; see "Obtaining financial approval for a high-cost purchase order" on page 53.

This section provides instructions, including predefined user IDs and passwords, for working directly with different functions of the template. Alternatively, you can log on through the user-registration template and perform the tasks that an administrator has authorized you to perform.

Creating a purchase order

To create a purchase order, perform the following steps:

1. On the machine running WebProxy, start a Web browser and enter the following URL:

   ```
   http://localhost/ePortal/servlet/Portal?Action=Logon&Solution=BtoBiTemplates&
   UserName=Girish&Password=girish
   ```

   For the purposes of this demonstration, the user ID `Girish` and password `girish` are predefined for a user who has authority to create a purchase order.
2. Enter template information in the **Trading Partner 1** and **Trading Partner 2** fields—for instance, `ABC Corporation` and `XYZ and Associates`.
3. Click **Add Line Items**. The browser displays items available from a template manufacturer's catalog.
4. Select one or more line items by changing the value in the item's **quantity** field to a positive, nonzero integer.
5. Click the browser's **Back** button to return to the Create PO screen.
6. Verify the selected items, then click **Submit PO**. When the PO is successfully submitted, the browser displays a success message and the purchase-order identifier assigned to your request.
7. Close the browser to avoid problems with cached data in subsequent steps.

Approving a purchase order

To approve a purchase order, perform the following steps:

1. On the machine running WebProxy, start a Web browser and enter the following URL:

   ```
   http://localhost/ePortal/servlet/Portal?Action=Logon&Solution=BtoBiTemplates&
   UserName=Yoda&Password=yoda
   ```

   For the purposes of this demonstration, the user ID `Yoda` and password `yoda` are predefined for a user who has authority to approve a purchase order.

   The browser displays a list of purchase-order identifiers. Each identifier is a hyperlink.

2. Click one of the identifiers. The browser displays the Approve PO screen.
3. Click **Approve** to begin the PO-approval process.
4. Close the browser to avoid problems with cached data in subsequent steps.

Obtaining financial approval for a high-cost purchase order

If the total cost of a purchase order is greater than $10,000, a Business Integrator user in the role of a financial approver must approve the PO before it can be fully processed. To approve a high-cost purchase order, perform the following steps:

1. On the machine running WebProxy, start a Web browser and enter the following URL:

   ```
   http://localhost/ePortal/servlet/Portal?Action=Logon&Solution=BtoBiTemplates&
   UserName=GaryP&Password=garyp
   ```

   For the purposes of this demonstration, the user ID `GaryP` and password `garyp` are predefined for a user who has financial-approver authority.

   The browser displays a list of purchase-order identifiers. Each identifier is a hyperlink.
2. Click the identifier of the high-cost purchase order. The browser displays the Approve PO screen.
3. Click **Approve** to approve the purchase order.
4. Close the browser to avoid problems with cached data in subsequent work with the template.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make

improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM United Kingdom Limited,
Intellectual Property Department
Hursley Park
Winchester SO21 2JN
United Kingdom

Such information may be available, subject to appropriate terms and conditions, including, in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained here in was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measures may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available system. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the application data of their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claim related to non-IBM products. Questions on capabilities of non-IBM products should be addressed to the suppliers of those products.

Any performance data contained here in was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measures may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available system. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the application data of their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claim related to non-IBM products. Questions on capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains template application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these template programs in any form without payment to IBM, for the purpose of developing, using, marketing or distributing application programs conforming to IBM's application programming interfaces for the operating platform for which the template programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States or other countries, or both:

- DB2
- IBM
- MQSeries
- Process Choreographer
- SecureWay
- VisualAge

- WebSphere

Java and all Java-related trademarks are trademarks of Sun Microsystems, Inc. in the United States, or other countries, or both.

Microsoft, Windows, and Windows NT are registered trademarks of Microsoft Corporation in the United States and/or other countries.

Rational and ClearCase are registered trademarks of Rational Software Corporation.

Other company product, and service names may be trademarks or services marks of others.

Bibliography

This bibliography lists the books in the IBM WebSphere Business Integrator and associated libraries.

IBM WebSphere Business Integrator library

The Business Integrator library consists of the following books:

- *WebSphere Business Integrator Concepts and Planning*, GC34-5960

  This book introduces the Business Integrator system, providing a high-level system overview, defining the system capabilities, and describing its value to e-businesses. This book also provides the information that you need to plan the installation of Business Integrator.
- *WebSphere Business Integrator Installation Guide*, GC34-5961

  This book is a guide to installing and configuring Business Integrator. It contains information about:
  - Selecting your required topology
  - Installing and configuring the base products and software components of Business Integrator on each machine in the topology
  - Installing and configuring firewalls and proxies
- *WebSphere Studio Business Integrator Extensions Installation Guide*, SC34-5962

  This book is a guide to installing and configuring Solution Studio. It also contains information about setting up clients and servers, and creating projects.
- *WebSphere Business Integrator Run Time*

  This book is a comprehensive guide to the Business Integrator runtime system, providing the following information:
  - Detailed conceptual information about the run-time components of Business Integrator
  - Deployment of solutions to the run-time system
  - System administration, such as starting and stopping software components and base products, defining users, and using the Exception Console
  - General problem determination information, including how to trace and debug, and information on obtaining help from technical support

- *WebSphere Business Integrator Messages*

  This book lists the error messages that are produced by Business Integrator and provides references to the documentation for the messages of base products.
- *WebSphere Studio Business Integrator Extensions Developer's Guide*

  This book describes how to create a Business Integrator solution, beginning with the solution design phase, to the solution implementation phase, and finally the solution deployment phase by using a sample business problem. This book also provides procedures for assembling a Business Integrator solution in the run-time environment and a description of how to use Solution Studio for solution design and implementation.
- *WebSphere Business Integrator DataInterchange for Windows NT User's Guide*, SC34-5963

  This book is a guide to installing and using DataInterchange in the Business Integrator environment.

You can find the latest versions of the books at the following Web site:

`http://www-4.ibm.com/software/webservers/btobintegrator/`

This site contains links to the Web sites of the underlying products of IBM WebSphere Business Integrator.

Related documentation

The utilities subdirectory on the Documentation CD contains documentation about utilities that can prove useful in building and running solutions. This documentation is not available on the IBM WebSphere Business Integrator Web site.

WebSphere Business Integrator also provides a number of external application programming interfaces (APIs). HTML documentation that is generated by using the Javadoc tool is provided for these APIs. For a list of the APIs, refer to the *WebSphere Business Integrator Run Time* book.

WebSphere Partner Agreement Manager library

The Partner Agreement Manager Version 2 Release 1 library consists of:

- *Partner Agreement Manager Installation Guide*, GC34-5964
- *Partner Agreement Manager Administrator's Guide*
- *Partner Agreement Manager User's Guide*
- *Partner Agreement Manager Adapter Developer's Guide*
- *Partner Agreement Manager Script Developer's Guide*
- *Partner Agreement Manager API Guide*
- *Partner Agreement Manager Adapters for MQSeries User's Guide*
- *Partner Agreement View User's Guide*, GC34-5965
- *WebSphere Partner Agreement Manager Business Process Integration Adapter Guide*

DataInterchange library

The DataInterchange Version 3 Release 1 library consists of:

- *DataInterchange Client User's Guide,* SB34-2010
- *DataInterchange Administrator's Guide,* SB34-2002
- *DataInterchange Installation Guide,* GB09-8070
- *DataInterchange Messages and Codes,* SB34-2000
- *DataInterchange Programmer's Reference*, SB34-2001

Other libraries

You can find important information in the libraries of the following products:

- DB2® UDB
  - *IBM DB2 Universal Database Quick Beginnings Version 6.1*, S10J-8149
- MQSeries®
  - *MQSeries for Windows NT Quick Beginnings*, GC34-5389
  - *MQSeries System Administration*, SC33-1873
  - *MQSeries Using Java*, SC34-5456
  - *MQSeries MQSC Command Reference*, SC33-1369
  - *MQSeries Queue Manager Clusters*, SC34-5349
  - *MQSeries Integrator Introduction and Planning*, GC24-5599
  - *MQSeries Workflow Getting Started with Buildtime*, SH12-6286
  - *MQSeries Workflow Getting Started with Runtime*, SH12-6287
  - *MQSeries Adapter Kernel for Multiplatforms: Quick Beginnings*, GC34-5855
  - *MQSeries Adapter Kernel: Problem Determination Guide*, GC34-5897

  - *MQSeries Adapter Builder for Windows NT: Using the Control Center*, GC34-5882
- SecureWay®
  - *SecureWay Policy Director Up and Running*, SCT6-3KNA
  - *SecureWay Policy Director Base Administration Guide*
  - *SecureWay Firewall User's Guide*, CG31-8658
- VisualAge®
  - *VisualAge Java, Enterprise Edition Getting Started*
  - *VisualAge C++ Professional for Windows NT Getting Started*
- WebSphere® Application Server
  - *Introduction to WebSphere Application Server*, SC09-4430

Index

File Number: BIZAAK00

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.