



Developing deployable business processes using WebSphere Business Modeler

Note

Before using this information and the product it supports, be sure to read the information general information under "Notices and Trademarks" on page 59.

Contents

Overview: Developing deployable business processes using WebSphere Business Modeler	v
--	----------

Chapter 1. Preparing for modeling and testing deployable business processes

Setting up the WebSphere Business Modeler environment for deploying processes	2
Enabling the unit test environment for direct deployment.	3
Implementing the managed deployment environment using WebSphere Process Server.	4
Defining the roles required for human tasks	5
Defining security for the managed deployment environment	6
Configuring the managed deployment environment	7
Uninstalling deployed resources	11

Chapter 2. Modeling processes for deployment.

Example: New Employee Setup Process	13
Modeling human tasks for deployment	14
Refining the interfaces for your deployable process	15
Adding services to deployable applications.	21
Modeling business measures for deployment	22

Chapter 3. Testing business process deployment.

Deploying business processes on a test server	25
Adding and changing test servers for deploying processes	26
Testing process flow	26
Testing human task assignments	29

Using placeholder tasks for testing processes with unimplemented activities	30
Testing configurable business rules	31
Testing business measures	32

Chapter 4. Troubleshooting business process deployment

Overview of WebSphere Business Modeler problem determination	35
Importing a WebSphere Business Modeler problem determination archive	37
Loading project interchange resources into the workbench	37
Viewing or rerunning test traces in the integration test client	39
Analyzing server console and log records in the Server Logs view	45
Viewing serviceDeploy build problems	47
Viewing Eclipse workspace logs	48
Viewing WebSphere Business Modeler workspace logs	50
Viewing business process models	50

Chapter 5. Deploying business processes to production

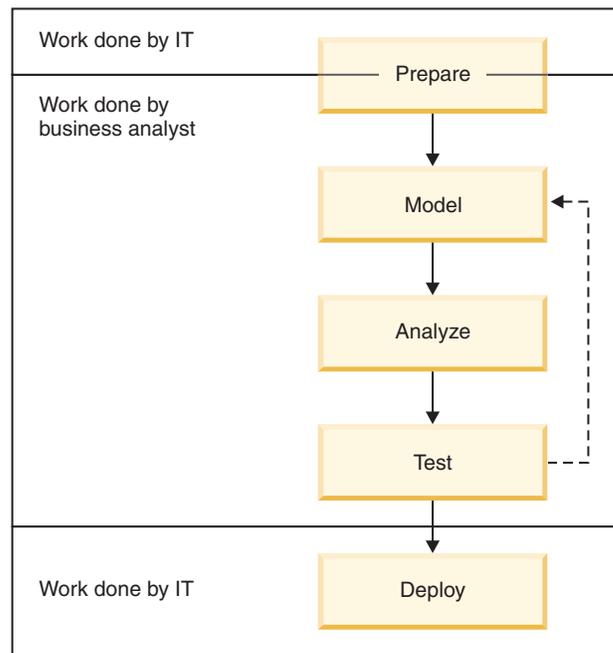
Limitations and restrictions for deploying processes from WebSphere Business Modeler	55
---	-----------

Notices and Trademarks	59
Notices	59
Trademarks and service marks	61

Overview: Developing deployable business processes using WebSphere Business Modeler

Accelerate your time to value by creating business process applications on your timeline to exactly match your line of business (LOB) requirements. You can use WebSphere Business Modeler to define exactly what you want in a business process application for a human-centric process without relying on IT to develop and test the application.

The process of developing a deployable business process using WebSphere Business Modeler involves the following high-level stages.



If you need to add services to your business process, you might require IT to implement services that do not already exist. To meet governance requirements, IT should deploy the completed business process application to production.

Roles required

The person who models a deployable process must know how to use WebSphere Business Modeler and be comfortable with providing the more technical specifications required to fully define a business process.

- Business analyst
- IT administrator (to set up a managed deployment environment for process testing)
- IT developer (to provide service implementations, if required, and to troubleshoot deployment problems)

Products required

To model and then test the deployment of business processes, the business analyst needs to install only a few products:

- WebSphere Business Modeler version 6.2
- Lotus Forms Viewer version 3.5 (included with WebSphere Business Modeler and the WebSphere Integration Developer unit test environment)
- Lotus Forms Designer version 3.5 (included with WebSphere Business Modeler)

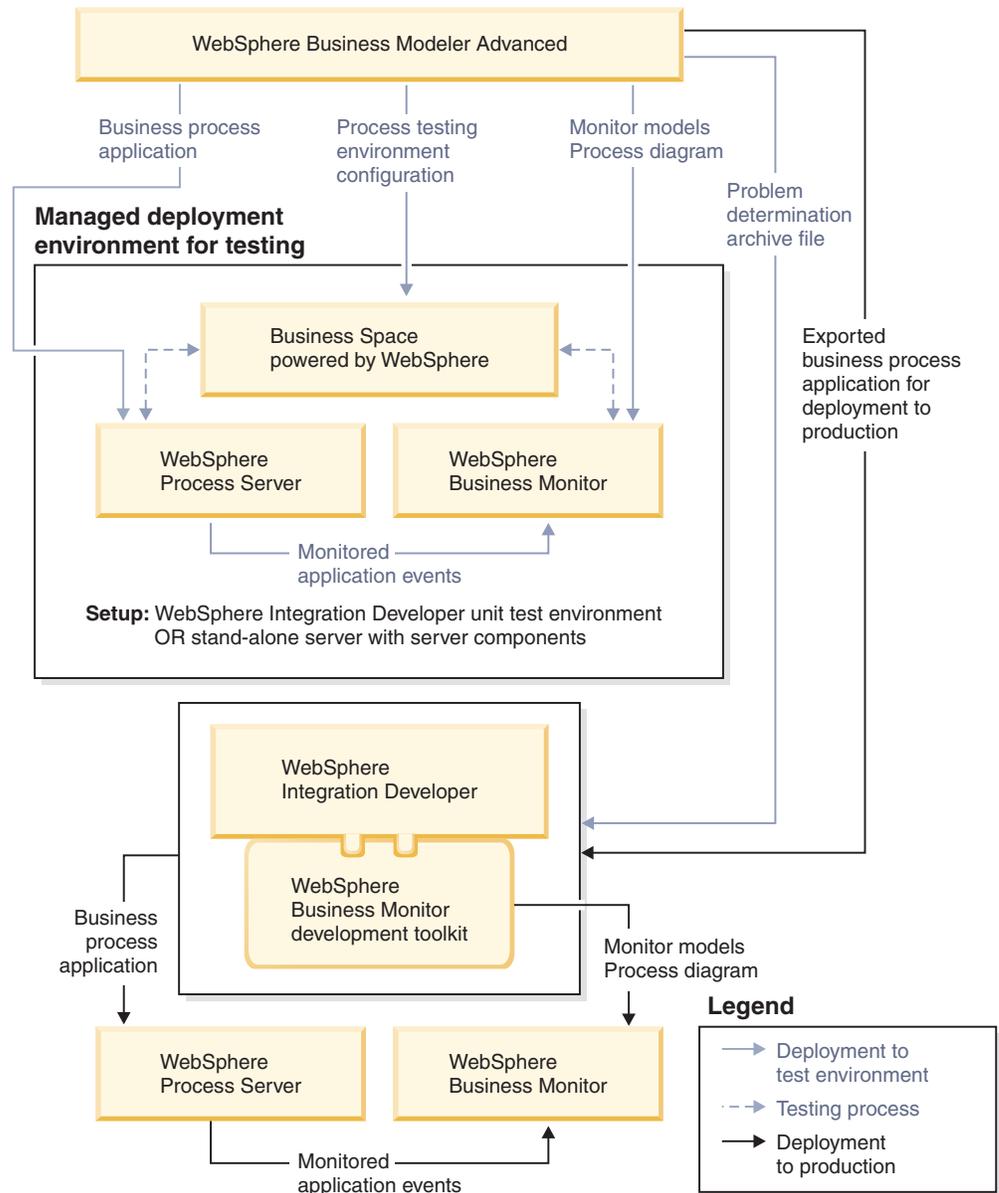
Note that Lotus Forms Viewer is also required by end users of the business process application so that they can view and complete forms.

To support the iterative testing of the business process application being developed, the IT department needs to set up a test server (or managed deployment environment) that includes the required server components. You can either use the enhanced unit test environment in WebSphere Integration Developer version 6.2 or configure a stand-alone server with WebSphere Process Server, the Business Space, and WebSphere Business Monitor (if required) configured as a managed deployment environment. If you use the WebSphere Integration Developer unit test environment, you should set this up on a dedicated computer. If you do not plan to monitor a business process, you do not require WebSphere Business Monitor or the WebSphere Business Monitor development toolkit.

- WebSphere Integration Developer version 6.2
- Business Space powered by WebSphere (a component included with WebSphere Process Server and the WebSphere Integration Developer unit test environment)
- WebSphere Business Monitor development toolkit version 6.2 (for monitoring a process in the WebSphere Integration Developer unit test environment)
- WebSphere Process Server version 6.2 (for deployment to production)
- WebSphere Business Monitor version 6.2 (for monitoring business processes in production)
- WebSphere Services Registry and Repository version 6.2 (optional)

Troubleshooting of any deployment problems must be done using WebSphere Integration Developer version 6.2, but should not be done on the test server.

The following diagram summarizes the flow of artifacts between the products required to use WebSphere Business Modeler to develop a deployable process and the products required for the final deployment of that process to production.



Chapter 1. Preparing for modeling and testing deployable business processes

Before you start modeling processes, you need to complete a number of setup tasks. Ask your IT administrator to set up one or more servers for testing deployed business processes. Also ensure that the roles you use for assigning people to human tasks map to groups in your organization's people directory.

The IT administrator has two choices for setting up a testing server (or managed deployment environment):

- If five or fewer business users will be testing process deployment, the IT administrator can install WebSphere Integration Developer version 6.2 on a dedicated computer and enable the unit test environment for testing from WebSphere Business Modeler. To test the dashboards for monitoring business measures, WebSphere Integration Developer must be installed with the WebSphere Business Monitor development toolkit.
- If a larger number of business users will be testing process deployment, the IT administrator might choose to set up a managed deployment environment using WebSphere Process Server and WebSphere Business Monitor (if required).

Important: Ensure that IT knows whether the business process application will include business measures that you plan to monitor at run time.

To model a deployable human-centric process, you need the relevant roles defined in a modeling project and mapped to actual groups in the people directory using a role mapping file.

To prepare for modeling and testing:

1. **Business analyst:** Set up the WebSphere Business Modeler environment.
2. **IT administrator:** To set up a managed deployment environment for testing deployed business processes, choose one of the following options:
 - Enable the unit test environment in WebSphere Integration Developer.
 - Implement the managed deployment environment using WebSphere Process Server.
3. **Business analyst and IT administrator:** Define the roles required for human tasks.
4. **IT administrator:** Enable security for users and roles.
5. **IT administrator:** Configure the managed deployment environment.

After the IT administrator sends the business analyst the server configuration file, completed role mapping file, and .mar project of the relevant roles (if this project was created by someone in the IT department), the business analyst has everything necessary to model and test the deployment of business processes.

As testing progresses, the IT administrator might find it necessary to free up resources on the test server.

Related tasks

“Uninstalling deployed resources” on page 11

WebSphere Business Modeler can directly deploy applications to the managed deployment environment. If the need arises to free up resources on WebSphere

Process Server, the IT administrator can do this by uninstalling these applications using the administrative console.

Setting up the WebSphere Business Modeler environment for deploying processes

To test the deployment of your business processes, you need to prepare your WebSphere Business Modeler environment.

Some or all of the following products must be installed on the same computer as WebSphere Business Modeler:

- Lotus Forms Viewer version 3.5 (required)
- Lotus Forms Designer version 3.5 (if the business analyst wants to customize forms)
- Microsoft Internet Explorer version 7 is recommended, but you can also choose to use Microsoft Internet Explorer version 6 or Firefox version 3.x.

Important: To test human task assignments in deployed business processes by logging in as different users to complete tasks, you must use Internet Explorer as your Web browser. If you intend to log in to the process testing environment under one user ID at a time, you can use Firefox as your Web browser.

- Adobe Flash Player version 9 or 10

To set up the WebSphere Business Modeler environment:

1. Install Lotus Forms Viewer.
2. To customize forms in WebSphere Business Modeler, install Lotus Forms Designer in the WBModeler62 program files directory.
3. To ensure consistent testing results for human task assignments, set your Web browser preferences instead of using the system default settings. To access these preferences, click **Window** → **Preferences** and then select **General** → **Web Browser**.

Setup example: If you want to set Internet Explorer as your Web browser for testing deployed processes, click **Use external Web browser** and select **Internet Explorer**.

Note: If you leave the default **Use internal Web browser** and **Default system Web browser** settings, the process testing environment still opens in an external Web browser but might not work as expected.

4. To see the display of the process diagram in the process testing environment, ensure that Adobe Flash Player is installed on your system.

To test role assignments for human tasks, you must also define the roles required for human tasks.

Related tasks

“Defining the roles required for human tasks” on page 5

For human tasks to be assigned to the correct people at runtime, the roles used to model a business process for deployment must be mapped to the groups in your organization’s people directory. This mapping requires coordination between the modeling team and IT.

Related information

 Installing Lotus Forms products

Enabling the unit test environment for direct deployment

If you want to use the unit test environment as a test server, set WebSphere Integration Developer up on a dedicated computer. Your unit test environment can be enabled for direct deployment.

The IT administrator should set up the unit test environment as a test server when five or fewer business users will be testing process deployment. (Note that the license agreement refers to this setup of the unit test environment as the managed test environment.) If a larger number of business users need to test deployment, it is advisable to set up a managed deployment environment using WebSphere Process Server and, optionally, WebSphere Business Monitor.

It is recommended that you use one of

- Windows Server 2008
- SUSE Linux
- Red Hat Linux

as the operating system of your test server and that you have 3 GB RAM on this server. Other operating systems are supported, but as the number of testers increases the memory demands on the server increase and the listed operating systems are the preferred choice. WebSphere Business Monitor is not supported on Linux, if you intend to use both WebSphere Process Server and WebSphere Business Monitor it is recommended that you use Windows Server 2008.

Recommendation: To prevent users from getting a browser error that the security certificate is not trusted, use a certificate for the test server signed by a trusted certifying authority.

An application is provided to enable the unit test environment for direct deployment. The application is called “TestController62”. Take the following steps to install the application and enable your test environment for direct deployment.

1. Verify that TestController62.ear is installed.

The application is installed automatically the first time that you use the integration test client. Perform the following steps to verify that the application is installed on your test server.

- a. On your test environment server, open the administrative console and browse to **Applications** → **Enterprise Applications**.
- b. Look for the “TestController” application in the list of installed applications.
- c. If the application is installed then you do not need to do anything more.

If the “TestController” application is not installed, complete the next step to install the application.

2. Install the “TestController” application.
 - a. Browse to **Applications** > **Install New Application**.
 - b. Click **Browse** and select **TestController62.ear** application from the *WID_install\wid\util* directory.
 - c. Accept all default settings during installation.
 - d. Save your configuration after installation.
3. Start the “TestController” application from the administrative console. Select the “TestController” application from the list and click **Start**.

Your test environment is enabled for direct deployment.

Implementing the managed deployment environment using WebSphere Process Server

If you want to set up the managed deployment environment on WebSphere Process Server, install WebSphere Process Server on a stand-alone server.

The IT administrator will need to prepare a stand-alone server as the managed deployment environment. It is recommended that you have 3 GB of RAM on this server.

Recommendation: To prevent users from getting a browser error that the security certificate is not trusted, use a certificate for the test server signed by a trusted certifying authority.

1. Install WebSphere Process Server software on a stand-alone server.
See *Installing the Software* for more details on how to perform a stand-alone server installation.
2. Determine the components that you will use. You can select the following:
 - WebSphere Process Server
 - Business Space powered by WebSphere
 - WebSphere Business Monitor - in stand-alone server configuration
 - WebSphere Services Registry and Repository - likely separate shared installation
3. If you are monitoring business measures, when installing WebSphere Business Monitor, use the Advanced installation option. Then you must augment the existing WebSphere Process Server profile within the monitor profile template. To use Business Space, you must select the **Configure Business Space** option during profile augmentation. For more information see *Configuring WebSphere Business Monitor*.
4. In WebSphere Process Server, you can add a list of WebSphere Services Registry and Repository systems. You need to ensure that the WebSphere Services Registry and Repository that hosts the services is set to the default one.
5. The stand-alone server has to run with development mode enabled. This will ensure, that applications containing business processes can be republished or uninstalled. To enable development mode:
 - a. Open the administrative console and select **Servers > Application servers**.
 - b. Click your server link (if you followed default installation will be called "server1").
 - c. Select the **Run in development mode** check box.
 - d. After the **Run in development mode** option is set in the administrative console, you will need to restart the server.
6. Download and install the integration test client update from the WebSphere Business Modeler support site: <http://www-01.ibm.com/support/docview.wss?rs=2308&uid=swg24021508>. This update is installed using the IBM Update Installer. Follow the installation instructions provided on the support page to install the update.

Note: If you download the update using Microsoft Internet Explorer, the file is downloaded as a .zip file instead of a .pak file. Ensure that the file is renamed to .pak before using the IBM Update Installer to install the update.

7. Download the Test Controller application from the same support page:
<http://www-01.ibm.com/support/docview.wss?rs=2308&uid=swg24021508>.

Note: If you download the update using Microsoft Internet Explorer, the file is downloaded as a .zip file instead of a .ear file. Ensure that the downloaded .zip file is renamed to a .ear file before installing it through the WebSphere Process Server administrative console.

8. Install the Test Controller application through the WebSphere Process Server administrative console.
 - a. Select **Applications > Install New Application**.
 - b. Click **Browse** and select the TestController62.ear application from the directory where you unzipped it.
 - c. Accept all default settings during installation.
 - d. Save your configuration after installation.
9. Start the Test Controller application from the administrative console by selecting it from the list, and clicking **Start**.

After installation is complete, you can configure your managed deployment environment.

Defining the roles required for human tasks

For human tasks to be assigned to the correct people at runtime, the roles used to model a business process for deployment must be mapped to the groups in your organization's people directory. This mapping requires coordination between the modeling team and IT.

To create deployable business processes using WebSphere Business Modeler, the primary owner for a human task must be specified using a role. Before the business analyst assigns roles to human tasks, you need to ensure that the roles correspond to groups in your people directory. When you deploy the business process application, WebSphere Business Modeler requires a role mapping file to map roles to these groups in the people directory.

Recommendation: Establish a common set of roles for human-centric business processes that everyone who works on them can use. Because these roles are defined in a modeling project, they are easy to share and incorporate into process models.

In the following procedure, the first three steps might be done by someone in the line of business or in the IT department.

To define the roles required for human tasks:

1. Identify the list of groups in your people directory for the line of business for which the business process application is being developed.
2. If your modeling team does not already have one, create a project in WebSphere Business Modeler that contains only the role definitions that match these groups. Although for mapping purposes you need only specify the name for each role, you can add further specifications to the roles. You can use WebSphere Business Modeler Basic to create this roles project.
3. In WebSphere Business Modeler Advanced version 6.2, generate the template for the role mapping file:

- a. Ensure that the roles project is in your workspace and that you are in the WebSphere Process Server mode.
 - b. Right-click the roles project, select export, and select the **WebSphere people directory data** export type. A *ProjectName.rmf* file is generated that IT can use to map the role names to the unique IDs of the group names required by the business process application at run time. For a roles project that contains a role called “Manager”, a role mapping file with the following contents would be generated:


```
<?xml version="1.0" encoding="ASCII"?>
<logicalMapping:LogicalEntityRoot
xmlns:logicalMapping="http://www.ibm.com/logicalMapping"
peopleDirectory="bpe/staff/samplevmmconfiguration">
  <role name="Resources/Manager" uniqueName="" uid="BLM-
8da9005236fd458235e97ead00b7cb7e" description="" groupName="Manager"/>
</logicalMapping:LogicalEntityRoot>
```
 - c. If a business analyst generates the role mapping file, the business analyst needs to give it to an IT administrator for completion.
4. **IT administrator:** In the role mapping file, complete the mapping between the roles and the groups in the people directory.
 - a. In the WebSphere Process Server administrative console, expand **Users and Groups**, select **Manage Groups**, and click **Search**. A table of defined groups is displayed.
 - b. Check the group names in the “Group name” column against the **groupName** attributes in the .rmf file, and update the **groupName** value if necessary.
 - c. Copy the value in the “Unique name” column into the **uniqueName** attribute. The following example shows a completed entry for a role in the role mapping file:


```
<role name="Resources/Manager"
uniqueName="cn=WPSManager,o=defaultWIMFileBasedRealm"
uid="BLM-8da9005236fd458235e97ead00b7cb7e"
description="" groupName="WPSManager"/>
```

In the configuration file for the managed deployment environment on which the business users tests applications, a relative path must be added to the completed role mapping file.

After the IT administrator has created the configuration file for the test server, he or she will send the business analyst both the completed role mapping file and the test server configuration file in a .zip file.

Important: If you need to make changes to your roles project, ensure that the role mapping file is also updated.

Related information

 Creating roles

Defining security for the managed deployment environment

To deploy business processes to the managed deployment environment, users need to be assigned roles with the appropriate privileges. The IT administrator assigns roles using the administrative console.

WebSphere Business Monitor data security will be set up automatically by the code when a model is deployed. For more information see *Securing your environment*.

After WebSphere Process Server is installed, you need to make sure you have already enabled security. Refer to *Securing a deployment environment of WebSphere Process Server*.

Clients communicate with the managed deployment environment in WebSphere Process Server through the REST (Representational State Transfer) Services Gateway.

You need to make sure the individual users or groups of users that invoke the deployment APIs have been given WebSphere Application Server *deployer* or *administrative* role privileges.

1. Create user definitions.
 - a. Open the administrative console and select **Users and Groups > Manage Users**.
 - b. Click **Create** and complete the user definition.
2. Make sure that the individual users or group of users that need to invoke the deployment APIs are assigned the required administrative security roles.
 - a. In the administrative console, select **Users and Groups**.
 - b. Select either **Administrative User Roles** or **Administrative Group Roles** as required.
 - c. Add individual users or groups of users to either the *administrator* or *deployer* roles. Typically, the business analyst is assigned the deployer role.

Users will now have the required privileges to deploy business processes to the managed deployment environment. WebSphere Business Modeler passes these credentials to the REST Services Gateway as part of command invocations. The REST Services Gateway will verify that the user has the appropriate privileges to deploy business processes.

Configuring the managed deployment environment

To establish communication between WebSphere Business Modeler and the managed deployment environment, the IT administrator needs to ensure that the server configuration file is properly configured. The IT administrator can also specify the deployment timeout properties for recovery from a system lock up.

The managed deployment environment is the set of servers configured for testing. You need to configure the managed deployment environment to use this functionality.

The following files are part of the configuration:

- The server configuration file refers to a list of one or more server components instances. A server component instance is a logical name for an installation of Business Space powered by WebSphere, WebSphere Business Monitor or WebSphere Process Server. The server configuration file is imported into WebSphere Business Modeler and used by Modeler to establish connectivity to the managed deployment environment.
- The logical role mapping file provides the mapping of logical roles in WebSphere Business Modeler and the physical user group defined in the target directory, such as LDAP.
- The managed-deployment-environment.properties file is used to configure the managed deployment environment in WebSphere Process Server.

- The `monitorMSSConfig.properties` file is used to configure the managed deployment environment in WebSphere Business Monitor.

To configure the managed deployment environment:

1. When WebSphere Process Server is installed, send the IT administrator the host name or IP address where the managed deployment environment is running.
2. Configure the following parameters in the server configuration file. The IT administrator needs to configure and provide the server configuration file to the WebSphere Business Modeler user. The Modeler user should import the server configuration file during the initial deployment or when explicitly adding a new managed deployment environment in the Servers View.
 - a. Specify the name of the test server and make sure the server is in test mode (`test="true"`) and whether security is switched on (`secured="true"`). The test server name should be properly set because this will be the name that appears in the Server View in Modeler.
 - b. Specify the name of the logical role mapping file. This is needed if your business process uses human tasks. See *Defining the roles required for human tasks* for more information.
 - c. Specify the endpoint URLs for each of the server components that you will use in your runtime environment. The server components include Business Space powered by WebSphere, WebSphere Process Server and WebSphere Business Monitor. WebSphere Process Server and WebSphere Business Monitor must be setup to run on the same server.

For each server component you can provide information for the:

- **name** - name of the server component
 - **configuration** - the URL for the server component, including port
- d. There is no specific configuration for the managed deployment environment. However, the IT administrator may need to specify a port number that WebSphere Process Server web container listens to. In the administrative console, select **Servers > Application servers > *servername* > Port**. If security is disabled, use the `wc_defaulthost` port (for example, 9080). If security is enabled, use `wc_defaulthost_secure`. After this the client, for example WebSphere Business Modeler, queries the managed deployment environment using REST (Representational State Transfer) application programming interfaces to establish communication with the runtime.
3. Optional: Configure the deployment timeout properties. Normally you should be able to use the default deployment timeout properties. You may need to tune the settings based on the environment and the number of concurrent users. You can set the values for `sleep.interval` and `intervals.to.timeout` properties in the `install_root/properties/managed-deployment-environment.properties` file. The managed deployment environment allows only one deployment in process at any given time. If a deployment has not finished within the specified timeout interval, the deployment will abort and the managed deployment environment will be ready to accept a new deployment.

```
#
# The interval time in seconds in which the environment checks for
# hanging deployments
#
sleep.interval=60

#
# The timeout in intervals after which the managed deployment environment
# will timeout a hung deployment.
#
```

```

# For example if your sleep interval is 60 seconds and the timeout is set
# to 30 intervals, then a hung deployment will timeout after 30 minutes.
#
intervals.to.timeout=30

```

4. Configure the monitorMSSConfig.properties file as needed. For more information see *Configuring WebSphere Business Monitor*.
5. Restart WebSphere Process Server and WebSphere Business Monitor, as needed.
6. Send the business analyst a .zip file that contains the config.xml file and the completed role mapping .rmf file. To make it easier for the business analyst to identify these files, you could put them in a folder with the same name as the test server.

The IT administrator now knows not only where the managed deployment environment is running from but also which port the REST API for the managed deployment environment listens to. This provides enough information to configure the client (WebSphere Business Modeler) to talk to the managed deployment environment.

Here is an example of a server configuration file that has configuration information for three server components:

```

<?xml version="1.0" encoding="UTF-8"?>
<rest:serverConfiguration xmlns:rest="http://rest.dtd.btools.ibm.com"
name="Test Server with Monitor" test="true" secured="true"
memberMapping="Organization.rmf">
<description>
Test MDE with WPS with Monitor
</description>
<serverComponent
name="WebSphere Process Server"
configuration="https://xxx.yyy.zzz:9446/rest/serverComponent/componentConfiguration">
</serverComponent>
<serverComponent
name="WebSphere Monitor Server"
configuration="https://xxx.yyy.zzz:9446/monitorServerComponent/componentConfiguration">
</serverComponent>
<serverComponent
name="WebSphere Business Space"
configuration="https://xxx.yyy.zzz:9446/BusinessSpace/services/request/deployConfig">
</serverComponent>
</rest:serverConfiguration>

```

Here is an example of a logical role mapping file:

```

<?xml version="1.0" encoding="ASCII"?>
<logicalMapping:LogicalEntityRoot
xmlns:logicalMapping="http://www.ibm.com/logicalMapping" peopleDirectory="bpe/staff">
<role name="Employee" uniqueName="" uid=BLM-fd0c51" description=""/>
<role name="Manager" uniqueName="" uid=BLM-eob771" description=""/>
</logicalMapping:LogicalEntityRoot>

```

Here is an example of a monitorMSSConfig.properties file:

```

# Application Installation timeout values
# *****
# In most cases the default values should not be changed.
# "appmanagement.retry.attempts" is the number of times Monitor MSS will retry
# if an error occurs during the installation of a model application.
# The default value is 3.
# "appmanagement.retry.wait.seconds" is the number of seconds to wait before
# retrying the installation of the model application.
# The default value is 20 seconds.
# "appmanagement.timeout.minutes" is the number of minutes to wait for the
# installation of a model application to finish.

```

```

# The default value is 15 minutes.
#appmanagement.retry.attempts=3
#appmanagement.retry.wait.seconds=20
#appmanagement.timeout.minutes=15

# *****
# Global lock timeout values
# *****
# "sleep.interval" is the number of seconds to sleep before
# checking the global lock again. The default value is 60 seconds.
# "intervals.to.timeout" is the number of times to sleep before
# releasing the global lock due to time out. The default value is 15.
# The total time to wait for a time out is sleep.interval * intervals.to.timeout,
# so the default time out value is 15 minutes.
#sleep.interval=60
#intervals.to.timeout=15

# *****
# Deployment targets
# *****
# The server or cluster where the Moderator and ModelLogic modules
# should be deployed.
#moderator.deployment.target=WebSphere:cell=WBMonSrv_wps_Cell,
#node=WBMonSrv_wps_Node,server=server1
#modellogic.deployment.target=WebSphere:cell=WBMonSrv_wps_Cell,
#node=WBMonSrv_wps_Node,server=server1
# *****
# CEI configuration information
# *****
# Valid values for model.cei.location are "local" and "remote".
#
# If model.cei.location=remote, set model.cei.hostname
# to the hostname of the remote CEI server, and set model.cei.rmiport
# to the bootstrap port for the remote CEI server.
#
# Valid values for model.cei.security are "enabled" and "disabled".
# If model.cei.security=enabled, set model.cei.userid to the
# userid on the remote CEI server, and set model.cei.password to
# the password for that userid.
#
# Valid value for model.cei.profile.list is "Event groups list".
#
# Valid values for model.cei.scope.type are
# "cell", "node", "server" and "cluster".
# The value of model.cei.scope.value depends on the scope.type.
# For "cell", set model.cei.scope.value=<cellName>.
# For "node", set model.cei.scope.value=<cellName>,<nodeName>.
# For "server", set model.cei.scope.value=<cellName>,<nodeName>,<serverName>.
# For "cluster", set model.cei.scope.value=<cellName>,<clusterName>.
#
# Valid values for model.cei.distribution.mode are "activeBest",
# "activeBypass", "active" and "inactive"

# CEI configuration information for the Monitor model
# To specify the information, uncomment these lines
#model.cei.location=remote
#model.cei.hostname=lkcnh03.raleigh.ibm.com
#model.cei.rmiport=2814
#model.cei.security=enabled
#model.cei.userid=admin
#model.cei.password=admin
#model.cei.profile.list=Event groups list
#model.cei.scope.type=server
#model.cei.scope.value=lkcnh03Node03Cell,lkcnh03Node02,server1
#model.cei.distribution.mode=activeBest

```

Uninstalling deployed resources

WebSphere Business Modeler can directly deploy applications to the managed deployment environment. If the need arises to free up resources on WebSphere Process Server, the IT administrator can do this by uninstalling these applications using the administrative console.

A user can uninstall models before all the deployed resources are removed. In this case, the installed application will remain on the server.

The IT administrator can manually uninstall applications using these steps.

1. Open the administrative console.
2. To uninstall a Websphere Business Modeler model, click **Applications > SCA modules** in the console navigation tree.
 - a. Select modules that you want to remove.
 - b. Click **Uninstall**.
3. To uninstall a Websphere Business Monitor model, click **Applications > Enterprise applications**.
 - a. Select the applications that you want to remove and click **Stop**.
 - b. Select again the applications that you want to remove and click **Uninstall**.

The format of the application name associated with the generated monitor model is as follows:

ProcessName_Mon_DeploymentIDAppl

Due to length limitations on the generated application name, the *ProcessName* string might be truncated so that it does not exactly match the name of the deployed process.

Note: Uninstalling a Websphere Business Monitor model does not remove information from the related direct to deploy tables in the Websphere Business Monitor database.

- c. Click **Save** to save your changes.
4. Periodically you may want to cleanup the direct to deploy tables in the Websphere Business Monitor database. To remove an item from the tables, follow these general steps. The specific steps depend on whether the database is DB2 or Derby.
 - a. To remove items from the tables, you need to know the *deployment id*. Use this URL to get the *deployment id* of undeployed and failed items:
http://<server>:<port>/monitorServerComponent/deployedResources You can get the values of *<server>* and *<port>* from the server configuration file.
 - b. Connect to the WebSphere Business Monitor database.
 - c. Run this SQL statement: **Delete From Monitor.D2D_Item where ID='<deployment id>'**

Here is an example of the information returned by the URL:

```
<?xml version="1.0" encoding="UTF-8" ?>
<rest:deployedResources xmlns:rest="http://rest.dtd.btools.ibm.com">
  <rest:deployedItem appName="Vacation Process Monitor_1226687623343_1Application"
  id="1226687623343_1"
  moduleName="Vacation Process Monitor_1226687623343_1" state="deployed"
  uri="/monitorServerComponent/deployedResources/1226687623343_1" version="1.0">
    <rest:metadata>
      <rest:property name="applicationType" value="process" />
      <rest:property name="blmid" value="BLM-572803bfbdfc33cb18a68f00882ea9f0" />
    </rest:metadata>
  </rest:deployedItem>
</rest:deployedResources>
```

```
<rest:property name="deployedArtifactDescription" value="[Vacation Process]" />
<rest:property name="deployedTimestamp" value="1226687631343" />
<rest:property name="deployedVersion" value="1.0.0" />
<rest:property name="name" value="Vacation Process" />
<rest:property name="userId" value="admin" />
<rest:property name="wpsDeploymentIdentifier" value="1226687623468_1" />
</rest:metadata>
</rest:deployedItem>
</rest:deployedResources>
```

The modules and associated applications are removed from the managed deployment environment, thereby freeing up resources on WebSphere Process Server.

Chapter 2. Modeling processes for deployment

Modeling processes that you plan to deploy is much like modeling other business processes using WebSphere Business Modeler: you create model elements like activities, business items, and resources, and associate them in the same way that you do for other modeling projects. However, there are a few guidelines to follow that can make developing a deployable process more manageable.

- Develop your process in stages, starting with the simple, high-level process flow, and gradually adding more complex elements like business services, business rules tasks, and business measures.
- Resolve validation errors after each significant update to the process. The error validation can be turned off temporarily by working in Basic mode to perform your high-level modeling tasks. However, it is important to validate the process frequently in WebSphere Process Server mode to identify any errors before you move onto the next stage of development.
- Test your process after each development stage to make sure that problems have not been introduced.

Related tasks

Chapter 3, “Testing business process deployment”, on page 25

After you have modeled your business process for deployment, you can test it iteratively on a server set up for you by IT (as a managed deployment environment). If you include business measures in your process, monitoring dashboards are automatically generated that you can also test. The testing function is enabled when your business process is free of errors.

Example: New Employee Setup Process

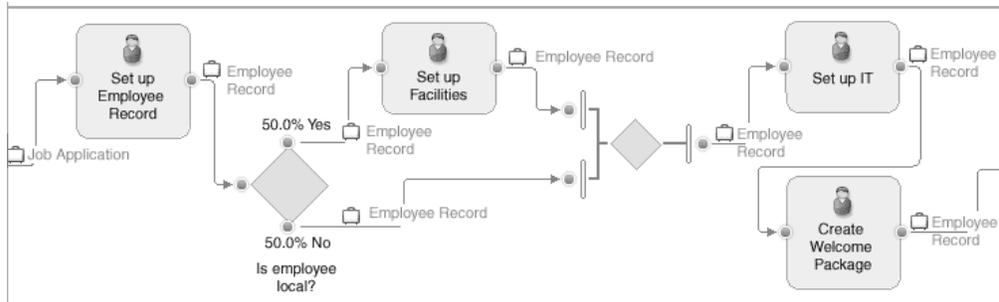
Human-centric business processes are one type of process that can be modeled in WebSphere Business Modeler, and then deployed as an application. Human-centric processes involve work done by people using data that is passed from person to person, typically through electronic forms. Throughout the documentation in this scenario, we will use a fictional business process as a reference point to illustrate concepts in the modeling and testing of deployable processes.

The example business process is a basic new employee setup process. In this process, the Human Resources department receives the data about a newly-hired employee and creates an employee record for her.

Once the employee record is created, it is passed on to the Facilities department so that her office and computer can be set up if the employee requires an on-site office (that is, if she is not remote), and to the IT department to setup her system IDs and passwords.

After these items are completed, the employee record is returned to the Human Resources person again so that he can put together a welcome package for the new employee with the all of the information that the employee needs to find her office, log on to the network, and so on.

Here is the initial process model representing the New Employee Setup process:



Related information

 Modeling human-centric processes

Modeling human tasks for deployment

You can create deployable processes that include tasks that are assigned to, and performed by a human. Human tasks are typically associated with a particular role and are completed using a form. For example, in the Set up New Employee process, the Set up Employee Record task is performed by someone in the Human Resources role, and it is associated with the Employee Record electronic form.

- Before you assign roles to your human tasks, ensure that you have an up-to-date copy of the roles project in your workspace.
- Lotus Forms Viewer must be installed to view forms for human tasks.
- If you want to associate electronic forms with your human tasks, you can create or modify them in Lotus Designer before modeling your human tasks, or you can create them later. To create or modify forms, you must have Lotus Forms Designer installed.
- You can begin modeling human tasks for deployment in Basic mode. However, to specify a primary owner for the human task, and to have the entire process validated for deployment, you must be in WebSphere Process Server mode.

While you are modeling your human tasks for deployment, you can test your process incrementally by invoking the **Test On Server** action from the Project tree view after you have made and saved updates to your process. You can test processes that have warnings, for example, about incomplete specifications of certain model elements, and clean up the warnings at a later point in development. However, the **Test On Server** action is only enabled for processes without errors, so any errors in your process must be fixed before you can test your process.

1. Create the human task.

Note: You can create local or global human tasks in deployable processes. The global human tasks behave slightly differently when you are testing your process, so you might want to begin with a local human task initially until you are satisfied with your process flow and have all of your data elements set up correctly. Once you are satisfied with the overall process, you can convert your local human tasks into global human tasks that can be reused in other processes.

For example, the Set Up Facilities human task in the Set Up New Employee process could be modelled initially as a local task, but could later be converted to a global task and reused in another business process that deals with moving existing employees from one office to another.

2. Create the inputs and outputs for the human task. Every human task must be associated with an input form to be deployed. If no form is specified, a default form is generated based on the data structure of the task input. You can leave the default form for your initial process development and testing, but you will probably want to improve the form layout and appearance before finalizing your process for deployment.

3. Assign a primary owner to your human tasks.

For processes that are to be deployed, primary owners must be identified as a role, not as an individual resource definition. Furthermore, the runtime people assignment criterion must be "Members by role name".

For example, in the Set Up New Employee process, the primary owner of the Set up Employee Record human task is the HR role.

Note: During the initial stages of modeling and testing, you can leave the primary owner unspecified, but you will receive a warning. When you test the process without a primary owner specified, it will behave as though any person can claim the task and complete it.

After you have completed modeling your human tasks, you can test your process again to make sure that the data flows as expected between your human tasks and other activities in your process, and you can make sure your primary owner specifications work as expected.

If you have not done so already, you might want to customize the forms used in your human tasks using Lotus Forms Designer.

Related tasks

"Modeling human tasks for deployment" on page 14

You can create deployable processes that include tasks that are assigned to, and performed by a human. Human tasks are typically associated with a particular role and are completed using a form. For example, in the Set up New Employee process, the Set up Employee Record task is performed by someone in the Human Resources role, and it is associated with the Employee Record electronic form.

"Testing human task assignments" on page 29

After you have tested the flow of your process and determined that the process execution paths work correctly, you can test the human task role assignments in your process to ensure that the appropriate people in your organization are able to complete the human activities included in your deployed process.

"Using placeholder tasks for testing processes with unimplemented activities" on page 30

As you develop and test your deployable process, there might be global elements that you need in your process that are not yet ready to be tested. For example, you might plan to include invocations of global services that are not yet implemented, or global processes that you have not finished modeling.

Related information

 [Creating human tasks](#)

Refining the interfaces for your deployable process

After you have modeled the basic flow of your process and verified that the human task assignments work as expected, you might want to refine your process to make it easier for the people who will use the application to perform their assigned tasks.

Improve the interfaces for your tasks

One of the ways you can make your application easier to use is to create custom forms for your human tasks. If you already have forms created, you can import these forms into your workspace and associate them with a human task.

Important: If you associate an existing form with a human task, and if the inputs or outputs of the human task do not match the data required by the form, then the inputs and outputs of the human task will be updated to match the data required by the form.

Although WebSphere Business Modeler generates default forms for any human task that does not already have forms associated with it, as well as a human form for any process input or output, these generated forms are a simple series of editable fields laid out in a top-to-bottom fashion according to the order of elements in the associated business item or data element.

However, a custom form designed in Lotus Forms Designer can provide a more user-friendly and attractive interface for reviewing and entering the data associated with a task. With a custom form, you can group related fields and provide section headings, field label formatting, and so on. For information about how to create custom forms, see the documentation for Lotus Forms Designer.

In the New Employee Setup process, if we use the default form generated for the Employee Record business item used to pass data between the human tasks, the user will enter data in a form that looks like the following form:

EmployeeRecord	
EmployeeID	<input type="text"/>
FirstName	<input type="text"/>
Initial	<input type="text"/>
LastName	<input type="text"/>
Street	<input type="text"/>
City	<input type="text"/>
StateOrProvince	<input type="text"/>
Country	<input type="text"/>
ZipOrPostalCode	<input type="text"/>
HomePhone	<input type="text"/>
StartDate	<input type="text"/> 
Department	<input type="text"/>
PWA	<input type="text"/>
OfficePhone	<input type="text"/>
OfficeIP	<input type="text"/>
Email	<input type="text"/>
ComputerDNSName	<input type="text"/>

If we customize a form based on the Employee Record business item, then we can present users with a better form layout and format, allowing users at each step of the process to more easily find the fields in the form that are relevant to the current task:

Employee Record

Employee ID:

Section 1: Employee Information

First Name:

Initial:

Last Name:

Street:

City:

State or Povince:

Country:

Zip or Postal Code:

Home Phone:

Section 2: HR

Start Date:



Department:

Section 3: Facilities

Office Number:

Office Phone:

Tip: If you are planning to design custom forms for your process, there are a few things to keep in mind:

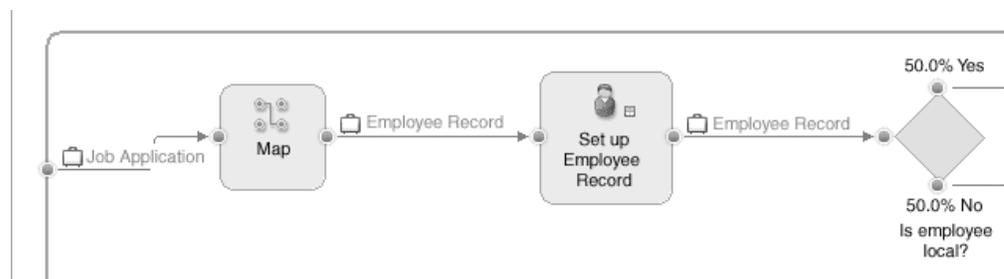
- The easiest forms to customize are forms based on business items that have a flat structure and do not contain other complex data types.
- It is not recommended that you try to reorder fields in your forms, or that you delete fields and recreate them.

You can also associate a form with the input that begins a process instance and the output that is generated from a process instance. As with a human task within a process, if you do not specify a form for a human task that requires input data to begin, or that generates output data when it completes, a default form is generated based on the data structure of the input and output. However, if you want to provide a custom form for entering input data or reviewing output data, associate the form with the process inputs and outputs on the **Forms** tab in the Attributes view.

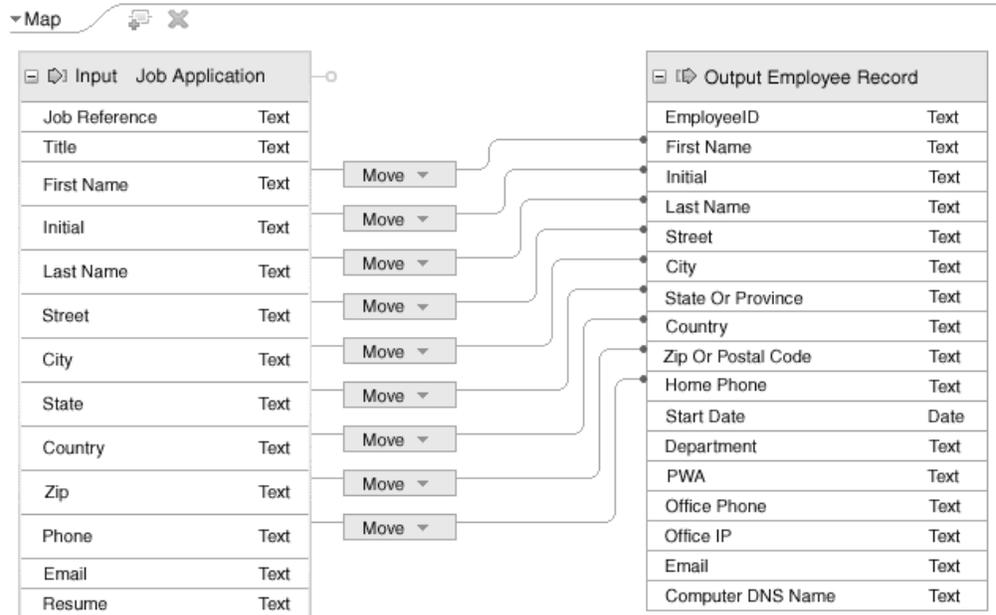
Save users time in data entry

Another way that you can make your business process application more usable is by mapping different data types that are generated by one activity in the process, such as data returned by a service, and data types expected as input to another activity in the process. Adding a data map allows you to map individual data elements from one data structure to data elements in another data structure.

For example, the input to the New Employee Setup process is a Job Application business item. This business item contains information about the new employee, such as name and contact information. The first task in the process is to create a new employee record, which contains much of the same information that is provided in the Job Application business item. Rather than have the HR person re-enter the information from the job application, we can map the data fields from the Job Application business item to the Employee Record application, and these mapped values will appear automatically in the Employee Record form that is presented to the HR person completing the Setup Employee Record task.



You define the map between the Job Application attributes and the Employee Record attributes in the Map editor, formalizing the link between, for example, the First Name attribute in the Job Application business item and the First Name attribute in the Employee Record business item, so that the relevant data can be propagated automatically to the Employee Record business item instance when a Job Application instance comes into the New Employee Setup process.



Write meaningful rule presentations for your business rules tasks

If you have business rules in your process that are based on rule templates with parameters, the parameter values are configurable in your process at runtime.

While you are testing your application, you can view the associated business rules and update rule parameter values to test the effect of different values on the execution path. This rule is also viewable and configurable by Business Space users once the application is deployed.

The rule presentation is the text representation for your business rule. When you create a business rule, a default rule presentation is generated for you. Since the rule presentation that you set up in WebSphere Business Modeler is the same string that is presented to users for configuration in the Business Space interface, it is particularly important to customize the text into a clear and readable format so that the Business Space user can understand the rule logic and identify which parameters to modify.

To customize the text for presentation of the rule in Business Space, open the rule in the Business task editor, click **Customize the text for the rule template presentation**, and type your text in the text area. In order for Business Space users to be able to configure parameter values, you must include the configurable parameters in the rule presentation. To add parameters to a customized rule template presentation, right-click in the text box and select the parameter you want from the pop-up menu.

Related information

-  Creating human tasks
-  Mapping values
-  Creating business rules that are configurable in Business Space

Adding services to deployable applications

As you are modeling your deployable business process, you might identify certain activities that can be automated. These activities can be modeled in your process as services or business services, depending on whether or not they have already been implemented as a service and exist on your WebSphere Service Registry and Repository server.

1. If the business service already exists on your WebSphere Service Registry and Repository server, you can import the business service and related business service objects for inclusion within your process model. For example, if there is already an implemented service that performs all of the steps required to set up all of the account IDs and passwords required for a new employee, you could replace the Set Up IT human task in your process model with the Set Up IT service:
 - a. Import the Set Up IT business service and related business service objects from your WebSphere Service Registry and Repository server. The imported business service and related business service objects appear in the selected catalog.
 - b. Add the business service to your business process from the Project tree view, just as you would add any other global element.
 - c. Use the map editor to perform any mappings between data required for or provided by the business service, and the connecting tasks in your process model. For example, if the service requires an Activation Date as an input, you can map the Employee Start Date attribute from the Employee Record business item to the Activation Date value required by the business service.
2. If your service has not yet been implemented, you can model the type of service that you would like to have implemented, including the types of input and output data and a description of the operations performed by the service, and pass the service specification to your IT developer to implement the service.
 - a. Create a service to add to your process model.

Note: The supported implementation types for services and business services in deployable processes are 'none' or "Import - Web Service binding".

- b. Export the new service for your IT developer to implement.
 - c. After the developer has implemented the service and published the endpoint to the WebSphere Service Registry and Repository server, you should be able to test your service within your deployable process.
3. Optional: You can improve the efficiency of how your service is located at runtime by using a classification system that is defined for services in your WebSphere Service Registry and Repository server. Such a classification system might include, for example, values that distinguish services used by one geographic region from another. For example, there may be one Set Up IT service that is specific to setting up Canadian-based employees and one for US-based employees. When modeling the Set Up New Employee process, you might want to ensure that the service for US-based employees is invoked and therefore tag the service invocation in your process model with the appropriate classifier.

To use classifications in developing your deployable processes:

- a. Import the classifications from your WebSphere Service Registry and Repository server. The classifications that you import are added as classifier values in the specified classifier catalog.

- b. Assign classifications to your services or business services. You can assign your classifications to global services or business services by adding them to the specification of the global element, or you can assign a classification to the service invocation in your process diagram.

Note: The classification added to a service invocation overrides the classification specified for the global element.

Related tasks

“Using placeholder tasks for testing processes with unimplemented activities” on page 30

As you develop and test your deployable process, there might be global elements that you need in your process that are not yet ready to be tested. For example, you might plan to include invocations of global services that are not yet implemented, or global processes that you have not finished modeling.

“Testing process flow” on page 26

The first step in testing your process should be to test the overall flow of your process, to make sure that your activities execute in the correct order, and that your gateways and connections work as expected. You should also make sure that the correct data is passed from node to node as your process executes.

Related information

 Services

 Business services and business service objects

Modeling business measures for deployment

To monitor the business performance of the application after it is deployed, you can include business measures in your deployable processes. You can also test business measures that you define in the same process testing environment where you test your process.

Before you define your business measures, you should have a process model that is in the late stages of development, having already been thoroughly tested for any problems with data flow, human tasks assignments, and business rules tasks. After you are satisfied with your model, you can add business measures.

1. Add business measures to your process model. You might want to develop your business measures in stages, first adding a set of business measures to your process, and then verifying they work as expected in the process testing environment. For example, because instance metrics form the basis of other metrics and KPIs, you could start by creating instance metrics in your process and test the generated monitor model to make sure that each instance metric is collecting the data as expected.

For example, in the New Employee Setup process, we could collect data about the city that our applicants live in, because many of the new employees are remote. The first step would be to create an instance metric that records the City value from the Employee Record business item output of the Setup Employee Record task.

After creating this instance metric, we could invoke the **Test on Server** action from the Project tree view and verify that the instance metric is behaving as expected. For more information about testing your business measures in the process testing environment, see “Testing business measures” on page 32.

Note: Deployable business measures must be error free, and either based on a predefined business measure template or on a fully-specified calculation

expression. Any business measures that contain warnings, either because they are incomplete or because they are not valid, are not included in the monitor model that is deployed to the process testing environment. Also, any business measures that are based on business measures with warnings are not included in the monitor model that is deployed to the process testing environment.

2. After you have verified that the first set of instance metrics are set up correctly, you can create aggregate metrics, KPIs or other instance metrics that are based on expressions that include the tested instance metrics. For example, we could create an aggregate metric that uses the City instance metric as a dimension for analysis. In the monitoring dashboards of the process testing environment, we should be able to see the number of employees from each city after a sufficient number of process instances have been run.

Note: During the deployment of the process an artifact called a monitor model is generated and deployed. Because the deployment of the monitor model takes significantly longer than the deployment of process model, it is recommended that you defer your business measures testing until late in the development of your business model. You can defer the monitor model testing in two ways:

- Delay adding business measures to your process until you have tested the process flow, human tasks, services, and business rules in your process, and confirmed that there are no errors. Testing each of these components might require many iterations and many invocations of the **Test on Server** action. If you have even a single business measure defined prior to running these tests, a monitor model is deployed each time. As a result, your test session will take significantly more time to set up.
- If you have already added business measures to your process, for example, if you are modifying a process that is already in production, you do not need to remove the business measures from your model to simplify your process testing. Instead, you can either run your tests on a managed server environment that does not have a WebSphere Business Monitor component, or you can ask your IT developer to create a server configuration file that mimics a managed server environment that does not have WebSphere Business Monitor component. (To do this, your IT developer has to remove the configuration information for the WebSphere Business Monitor component from the **configuration.xml** file that he gives you.)

Related tasks

“Testing business measures” on page 32

After you have added business measures to your process model, you can test your business measures in the process testing environment to ensure that the metrics that you have defined are collecting the data that you expect.

“Configuring the managed deployment environment” on page 7

To establish communication between WebSphere Business Modeler and the managed deployment environment, the IT administrator needs to ensure that the server configuration file is properly configured. The IT administrator can also specify the deployment timeout properties for recovery from a system lock up.

Related information

 Creating business measures

Chapter 3. Testing business process deployment

After you have modeled your business process for deployment, you can test it iteratively on a server set up for you by IT (as a managed deployment environment). If you include business measures in your process, monitoring dashboards are automatically generated that you can also test. The testing function is enabled when your business process is free of errors.

First, ensure that your business process shows no errors in the WebSphere Process Server mode. You can test a business process if warnings are still displayed in the Errors view. However, any business measures with warnings that they require further editing to be monitored will not be deployed to the test server.

To connect to a test server, you must have the configuration file for it in your file system.

Note: Although you can use human tasks as placeholders for services during the testing process, remember to replace any placeholder human tasks with the actual service implementations after you are finished testing.

Deploying business processes on a test server

When you test your business process on a server, WebSphere Business Modeler deploys the process to the test server for you.

Before you start testing a business process, ensure that you are in the WebSphere Process Server mode and that the configuration file and role mapping file for the test server is in your file system. You might need the account ID and password for the test server to authenticate with it.

The process testing environment runs on the Business Space component (on the test server), which opens in an external Web browser. For consistent testing results, ensure that you have set your browser preferences as part of setting up your modeling environment.

The first time that you test your business process, you are prompted to add a server. After you add this server, information about the server and about the processes deployed on it is displayed in the Servers view. This view is located in the bottom pane of the two- or four-pane layout.

Note: If you have a business process that includes other global processes or business measures, it can take some time to deploy on the test server.

To deploy a business process on a test server:

1. Right-click the process in the Project Tree view, and select **Test on Server**. If this is the first time that you are deploying a process, point to the server configuration file in the Add Server window. You might have to provide an account ID and password for the server.

Tips:

- If you have any problems connecting to a test server, first make sure that your account ID and password are correct. To update the authentication information for a server that you have already added, right-click the server

- name in the Servers view and select **Properties**. If you continue to have connection problems, contact the IT administrator for the test server.
- If you have problems with the deployment of a business process to a server, you can click the **Request Help from IT** button in the error message. You should send the problem determination archive file generated through this action to an IT developer who can open the file in WebSphere Integration Developer version 6.2 to identify the cause of the problem.
2. After your process deploys, log in to the Business Space to access the process testing environment using the same account ID and password that you used to log on to the test server. You can test your process in the preconfigured space that opens after you log in.

Important: If the browser opens with an error page about a problem with the security certificate or that the secure connection failed, check with your IT administrator before continuing to open the Web site (in Internet Explorer) or adding an exception (in Firefox).

3. To check which processes are deployed on test servers, expand the **Test Servers** and server nodes in the Servers view. By clicking the **Show My Processes** and **Show All Processes** icons, you can view either the processes that you deployed on test servers or all the processes deployed on test servers.

When you redeploy a business process, the version of the process that you deploy overwrites the version of the process on the test server. If you need to test processes on a different server, you can add or change test servers in the Servers view.

Related information

Troubleshooting process deployment using WebSphere Integration Developer

Adding and changing test servers for deploying processes

Use the Servers view to add new servers, change the server on which you want to test processes, test server connections, change authentication information, and see which processes are deployed on a server.

To add or change test servers:

- To add another server for testing your processes, open the Servers view, right-click the **Test Servers** node, and select **Add Server**. This server becomes the default test server , which has a triangle overlay on the bottom right corner. Processes that you test are deployed on the default test server.
- To change test servers, select the test server that you want to use in the Servers view, right-click it, and select **Make Default Server**.
- To test the connection to a server, right-click the server name in the Servers view, and select **Test Connection**.

Testing process flow

The first step in testing your process should be to test the overall flow of your process, to make sure that your activities execute in the correct order, and that your gateways and connections work as expected. You should also make sure that the correct data is passed from node to node as your process executes.

- You must be working in WebSphere Process Server mode.
- Your process must be free of validation errors.

- You must have the configuration file for your managed server environment in your file system.
- You must have set up your WebSphere Business Modeler environment for deploying processes, which includes installing Lotus Forms Viewer.
 1. In the Project tree view, right-click on the process you want to test and select **Test on Server**. If your process contains subprocesses or services that you have not created yet, you can use a placeholder task to stand in for the missing subprocess or service.
 2. If you have not already configured your workspace to communicate with the test managed server environment, the Configure Test Server window opens. Browse to the **configuration.xml** file provided by your IT developer. Enter your ID and your password, if security is enabled. The ID that you provide is the ID that is used to authenticate you to the managed server environment, and also to identify you as the owner of processes that you upload to the server. To change the server used for testing, or to update any of the authentication information, go to the Servers view. The process testing environment opens in an external browser.
 3. Log in to the Business Space to begin your test session. In most cases, the login ID and password should be the same ones that you provided when you set up your test server.
 4. Start the process instance by selecting the process name in the Start Process Instance widget and clicking **Create**.
 5. If you have defined inputs for your process, enter or review input values in the Enter Data into Forms widget. If you have associated your process input with a form, the form will appear in the Enter Data into Forms widget. If you have not yet associated a form with the process input, one is created for you by default.

Note: The form that displays for entering data is labeled the "Output form"

6. After you have finished reviewing or editing the form, click **Submit**. The process instance is started, and an entry appears in the View Process Instance History widget.
7. To view the execution path of your process, use the Process Execution and Process Execution Trace and Data Values widgets.

The Process Execution widget displays the process diagram that you are testing. After you create a process instance for testing, the execution path through the process is highlighted in the diagram.

As each activity is completed, it is added to the Process Execution Trace and Data Values widget. The name of the completed activity and the output data from the activity is displayed in this widget.
8. You can view the data associated with different points in the execution path by selecting an activity from the list of completed activities in the Process Execution Trace and Data Values widget. When you select an activity from the list, the output data for the selected activity is displayed.

For complex data, you can click the + to expand and drill down to the data values that make up the output for the activity.
9. When a human task is encountered, the task is highlighted in the process execution diagram, and it is added to the Claim Available Tasks widget. The task must be claimed and the associated input form completed before the process flow will continue.

- a. Complete the waiting human task by selecting it in the Claim Available



Tasks widget and clicking the  button to claim and work on the task. If a waiting human task is not showing up in the Claim Available Tasks widget as expected, you might need to select the **Refresh** menu action from the menu in the Claim Available Task widget.

- b. Enter or review the required data in the Enter Data into Forms widget and click **Submit**. The process execution resumes, moving to the next activity in the process flow.

Note: Because the account ID that launched the process testing environment is, by default, the owner of the Business Space, you are able to claim and complete any task listed in the Claim Available Tasks widget. You can also choose to log in as a different user to make sure your human task assignments work as expected.

10. After the process has completed its execution, you can:
 - See the path of the process execution in the Process Execution area. The execution path is highlighted in the diagram.
 - See the list of activities that were completed during the process execution in the Process Execution Trace widget.
 - View the data associated with each executed activity in the execution path by selecting an activity in the Process Execution Trace and Data Values widget, and examining the associated output data.
 - Return to the Start Process Instance widget and start another process instance for testing, potentially modifying the process inputs or the values of configurable business rules to verify that the process runs as expected.
11. When you have completed your testing, or if you would like to make modifications to your process model and continue testing, close the Web browser and return to WebSphere Business Modeler to continue modeling. To test any modifications that you make to your process, relaunch the process testing environment by right-clicking the process in the Project tree view, and selecting **Test On Server** again.

If you encounter errors in the process testing environment at any point during your testing, you can click the **Request Help From IT** button in the Process Execution Trace and Data Values widget. When this button is clicked, the log files from your testing session are packaged up for you to pass off to your IT developer for problem determination.

For information about importing, analyzing, and troubleshooting a WebSphere Business Modeler problem determination archive file in WebSphere Integration Developer, see Troubleshooting processes for deployment.

Note:

- You can only test one process including any dependent subprocesses at a time in the process testing environment.
- When you close the Web browser containing the process testing environment, you terminate the current testing session, and you can no longer review previous test runs of the process. When you invoke the **Test On Server** action from the Project tree view, the previous version of your process are replaced by the updated version.
- If you close WebSphere Business Modeler or switch workspaces while the Web browser containing the process testing environment is still open, the Process

Execution and Process Execution Trace and Data Values widgets will not function properly. You must keep the WebSphere Business Modeler workspace containing the deployed business process open while you are performing your process testing.

After you have tested the overall flow of your process, you can test your human task assignments, configurable business rules, or monitor models.

Related concepts

Chapter 2, “Modeling processes for deployment”, on page 13

Modeling processes that you plan to deploy is much like modeling other business processes using WebSphere Business Modeler: you create model elements like activities, business items, and resources, and associate them in the same way that you do for other modeling projects. However, there are a few guidelines to follow that can make developing a deployable process more manageable.

Related tasks

“Configuring the managed deployment environment” on page 7

To establish communication between WebSphere Business Modeler and the managed deployment environment, the IT administrator needs to ensure that the server configuration file is properly configured. The IT administrator can also specify the deployment timeout properties for recovery from a system lock up.

“Testing human task assignments”

After you have tested the flow of your process and determined that the process execution paths work correctly, you can test the human task role assignments in your process to ensure that the appropriate people in your organization are able to complete the human activities included in your deployed process.

“Testing configurable business rules” on page 31

If your process includes business rules based on a rule template with configurable parameters, you can validate these business rules in the process testing environment by executing process instances with different rule parameter values.

“Testing business measures” on page 32

After you have added business measures to your process model, you can test your business measures in the process testing environment to ensure that the metrics that you have defined are collecting the data that you expect.

Testing human task assignments

After you have tested the flow of your process and determined that the process execution paths work correctly, you can test the human task role assignments in your process to ensure that the appropriate people in your organization are able to complete the human activities included in your deployed process.

Before you can test human task assignments:

- Security on your test server must be enabled.
- You must have the account IDs and passwords that have been set up for testing by your IT developer.
- The role mapping must already be done on your test server, and you must have imported the roles project that corresponds to the role mapping file for the people directory used by your test server.
- You must be in the process testing environment, and your test process instance must be started from the Start Process Instance widget.

- If you have included scheduling information in your human task, be aware that the human task is not be testable in the process until the scheduled time period.

To test the human task role assignments in your process, you can login as a different user ID that is associated with one of the roles used in your process, and accept and complete the task as that user.

1. When the process execution pauses at a human task, click the **Log in as Different User** button Process Execution Trace and Data Values widget. A new external browser window opens.
2. Log in with the ID that is associated with the role that is appropriate for completing the task. For example, if we wanted to test that the roles have been set up correctly in the New Employee Setup process, we could log in with an ID associated with the HR role so that we can claim and complete the Set Up Employee Record task.
3. Select a task in the Claim Available Tasks widget, and click **Accept**. Tasks you have claimed are listed in the Work on Tasks widget.
4. Select a task in the Work on Tasks widget and click **Edit**. The form associated with the task opens in the Enter Data into Forms widget.
5. Complete the form and click **Submit**. The process execution resumes, and the submitted data is passed onto the next activity in the process.

Related tasks

“Modeling human tasks for deployment” on page 14

You can create deployable processes that include tasks that are assigned to, and performed by a human. Human tasks are typically associated with a particular role and are completed using a form. For example, in the Set up New Employee process, the Set up Employee Record task is performed by someone in the Human Resources role, and it is associated with the Employee Record electronic form.

“Configuring the managed deployment environment” on page 7

To establish communication between WebSphere Business Modeler and the managed deployment environment, the IT administrator needs to ensure that the server configuration file is properly configured. The IT administrator can also specify the deployment timeout properties for recovery from a system lock up.

“Defining the roles required for human tasks” on page 5

For human tasks to be assigned to the correct people at runtime, the roles used to model a business process for deployment must be mapped to the groups in your organization’s people directory. This mapping requires coordination between the modeling team and IT.

“Setting up the WebSphere Business Modeler environment for deploying processes” on page 2

To test the deployment of your business processes, you need to prepare your WebSphere Business Modeler environment.

Using placeholder tasks for testing processes with unimplemented activities

As you develop and test your deployable process, there might be global elements that you need in your process that are not yet ready to be tested. For example, you might plan to include invocations of global services that are not yet implemented, or global processes that you have not finished modeling.

Even if there are existing services or subprocesses available when you first begin modeling, it's a good practice to test your process initially without such dependencies so that you can verify the overall process flow before verifying the dependencies. When you test your processes using placeholders for your dependencies, you can emulate the execution of the missing dependencies by providing the expected output data that would be generated by the service or global process, and continuing with the execution of the parent process.

1. To use a place holder for an unimplemented task, service, or global process in a process that you are developing for deployment, use a global human task in your model that has inputs and outputs that correspond to the unimplemented element.

Alternatively, if you plan to invoke a global subprocess that you have not implemented, you could create the process and add a single human task within the process. Connect all the inputs to the process to the inputs of the human task, and the outputs of the human task to the outputs of the process.

2. When you invoke the **Test On Server** action, and start a process instance in the process testing environment, you can emulate the execution of the missing element by providing the output that would be generated in the form that is associated with the placeholder human task. The data that you provide in the form is passed on to the next activity in the process as though the missing element has completed.
3. After you have verified the overall flow of your parent process, and you have the dependent elements available for testing, you can replace the placeholder task with the actual global process or service that you intend to include in your process application.

Related concepts

Chapter 2, "Modeling processes for deployment", on page 13

Modeling processes that you plan to deploy is much like modeling other business processes using WebSphere Business Modeler: you create model elements like activities, business items, and resources, and associate them in the same way that you do for other modeling projects. However, there are a few guidelines to follow that can make developing a deployable process more manageable.

Related tasks

"Adding services to deployable applications" on page 21

As you are modeling your deployable business process, you might identify certain activities that can be automated. These activities can be modeled in your process as services or business services, depending on whether or not they have already been implemented as a service and exist on your WebSphere Service Registry and Repository server.

"Testing process flow" on page 26

The first step in testing your process should be to test the overall flow of your process, to make sure that your activities execute in the correct order, and that your gateways and connections work as expected. You should also make sure that the correct data is passed from node to node as your process executes.

Related information

 [Creating human tasks](#)

Testing configurable business rules

If your process includes business rules based on a rule template with configurable parameters, you can validate these business rules in the process testing environment by executing process instances with different rule parameter values.

- Your process must contain business rules based on rule templates that include parameters.
- You must be in an active testing session in the process testing environment.

In the process testing environment:

1. The Change Parameter Values for Business Rules widget lists the business rules tasks included in the current process and dependent subprocesses. To change the parameter values for a rule, select from the list the business rules task that contains the rule to modify. The business rules associated with this business rules task are listed and the configurable parameter values are highlighted.

Note: If you have not customized the rule presentations for your business rules, they might be difficult to understand in the process testing Business Space. It is important to remember that this is the same way that the users of your business process will see the business rule when you deploy your process. For information about customizing rule presentations, see “Refining the interfaces for your deployable process” on page 15.

2. Edit the values for selected rule by clicking on a value and typing in the active field.
3. Click **Save** to save the values that you specified. To reset the values to their defaults, click **Reset**.
4. Start a new process instance using the Start Process Instance widget. During the process instance execution, the relevant business rules tasks will run using the specified parameter values.

After you have tested your process with one set of values, you can re-run the process with different values and analyze the effect that these values have on resulting process output or on the execution path for the process.

Related information

 [Creating business rules tasks](#)

 [Creating business rules that are configurable in Business Space](#)

Testing business measures

After you have added business measures to your process model, you can test your business measures in the process testing environment to ensure that the metrics that you have defined are collecting the data that you expect.

When you invoke the **Test on Server** action for a process model that has business measures defined, a WebSphere Business Monitor monitor model is generated automatically. If the target managed deployment environment is configured with a WebSphere Business Monitor, then the monitor model is deployed to the server at the same time as the process model is deployed.

Note: Because the deployment of the monitor model takes significantly longer than the deployment of process model, it is recommended that you defer your business measures testing until late in the development of your business model. You can defer the monitor model testing in two ways:

- Delay the addition of business measures to your process until you have successfully tested the process flow, human tasks, services, and business rules in your process. Testing each of these components might require many iterations and many invocations of the **Test on Server** action. If you have even a single

business measure defined prior to running these tests, a monitor model is deployed each time, and your test session will take significantly more time to set up.

- If you have already added business measures to your process, for example, if you are making modifications to a process that is already in production, you do not need to remove the business measures from your model to simplify your process testing. Instead, you can either run your tests on a managed server environment that does not have a WebSphere Business Monitor component, or you can have your IT administrator create a server configuration file that mimics a managed server environment that does not have WebSphere Business Monitor component. (To do this, your IT administrator has to remove the configuration information for the WebSphere Business Monitor component from the **configuration.xml** file that he provides you.)
1. When you are ready to test your business measures, ensure that the server that you have selected for your test environment is set up with a WebSphere Business Monitor server component. To change your currently selected managed server environment to a different one, open the Servers view by going to **Window** → **Show view** → **Servers**. The list of available servers is shown.
 2. To add a new server for testing business measures, right-click **Test Servers** in the tree, and select **Add Server**. This brings up the **Add Server** window where you can browse to a configuration file with the appropriate server components defined.
 3. Right-click on the process that contains the business measures that you want to test, and select **Test on Server**. The process testing environment opens with a Business Space login page.
 4. Follow the same procedure as you did when testing the flow of your process, starting a process instance in the **Start Process Instance** widget, and providing any required data in the **Enter Data into Forms** widget.
 5. The monitoring dashboard widgets appear at the bottom of the page. The type of monitoring widgets that appear depend on the types of business measures that you include in your model. For example, if you have included instance metrics in your model, the **Instances** widget appears on the page.

Note: Any business measures flagged with warnings are not included in the monitor model that is deployed to the test environment and therefore these business measures will not show in the monitoring dashboard widgets. If you want to test all of your defined business measures, make sure that you clean up any warnings before invoking the **Test on Server** action.

6. To verify that you have set up the business measures correctly for your business process, you can re-run the process multiple times in the same testing session and verify, for example, that your aggregate metrics and KPIs are returning the type of data that you expect. As each process instance completes, the monitoring dashboards should reflect the new values.

Note: If you close the Business Space browser, you lose the monitoring data that has been collected during your testing session. If you intend to run multiple tests on the same version of your process by running several processes instances, do not close your browser window until you have completed your testing session.

Any customizations that you make to the Monitor Dashboard widgets are overwritten the next time you invoke Test on Server. Customizations to the Dimensions view, for example, or the addition of other monitoring widgets, do not persist between test sessions.

Related tasks

“Configuring the managed deployment environment” on page 7

To establish communication between WebSphere Business Modeler and the managed deployment environment, the IT administrator needs to ensure that the server configuration file is properly configured. The IT administrator can also specify the deployment timeout properties for recovery from a system lock up.

Related information

 [Creating business measures](#)

 [Dashboards](#)

Chapter 4. Troubleshooting business process deployment

In WebSphere Integration Developer, an IT developer can use specialized tools to troubleshoot WebSphere Business Modeler deployment problems based on information provided in a problem determination archive file imported from WebSphere Business Modeler.

The following topics describe the key problem determination concepts and explain how to use the problem determination tools in WebSphere Integration Developer:

Overview of WebSphere Business Modeler problem determination

In WebSphere Business Modeler, business users can generate BPEL business processes and modules from business process models and then transmit them for deployment in a managed deployment environment and test them without IT support. If problems are encountered when generating, transmitting, deploying, or running the business processes, the business users can generate a problem determination archive file and send it to an IT developer for analysis and resolution in WebSphere Integration Developer.

There are several stages required to ultimately run a business process in a managed deployment environment. The stages are:

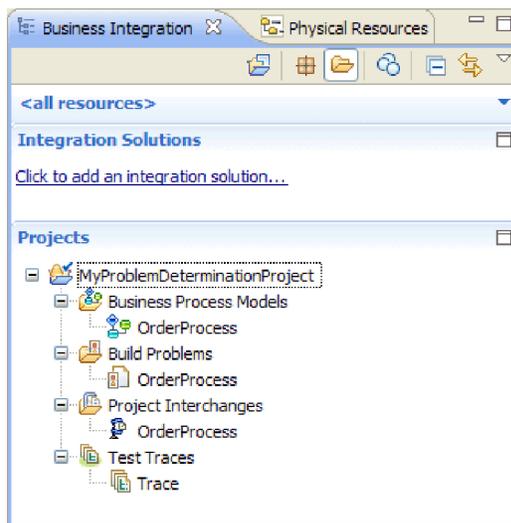
1. In WebSphere Business Modeler, business process models are transformed to BPEL business processes and modules and the transformation results are logged in WebSphere Business Modeler workspace log files.
2. The generated BPEL business processes and modules are transmitted to a managed deployment environment that is running on a server and the transmission results are logged in an Eclipse workspace log file.
3. In the managed deployment environment, a serviceDeploy build generates J2EE applications from the business processes and modules and the build results are logged in serviceDeploy build log files.
4. The deployed J2EE application is run as an EAR file in the managed deployment environment.

If problems occur at any stage and the business user generates a problem determination archive in WebSphere Business Modeler, the archive zip file will contain some or all of the following problem determination resources:

- One or more business process models.
- One or more WebSphere Business Modeler workspace log files. These log files can contain transformation errors and other errors that were logged when the business process models were transformed to business processes and modules in WebSphere Business Modeler.
- One project interchange file for each business process model. The project interchange files contain the business processes and modules that were generated from the business process models in WebSphere Business Modeler. If a problem occurred and no business processes or modules were generated, there may not be any project interchange files in the problem determination archive (depending on the specific nature of the problem).

- An Eclipse workspace log file. This log file can contain transmission and communication errors that occurred when the generated business processes and modules were transmitted from WebSphere Business Modeler to the managed deployment environment.
- One serviceDeploy build log file for each business process. This log file can contain errors and warnings that occurred when the build took place to generate J2EE applications from the business processes and modules.
- One test trace for the J2EE application. If a problem occurred in building the J2EE application from the business processes and modules, there may not be any test trace (depending on the specific nature of the problem).

The following figure shows the typical contents of a WebSphere Business Modeler problem determination archive in WebSphere Integration Developer:



In WebSphere Integration Developer, the IT developer has a number of problem determination tools that are optimized for working with the resources contained in the problem determination archive, such as:

- A specialized import wizard for importing WebSphere Business Modeler problem determination archives into WebSphere Integration Developer.
- A process model viewer for viewing (but not editing) visualization files for business process models.
- A business process editor for viewing business processes generated from the business process models.
- A build problems viewer for viewing error and warning data in serviceDeploy build logs.
- An integration test client for viewing and rerunning the test trace.
- A Server Logs view for analyzing server console and log records (and especially invocation records).

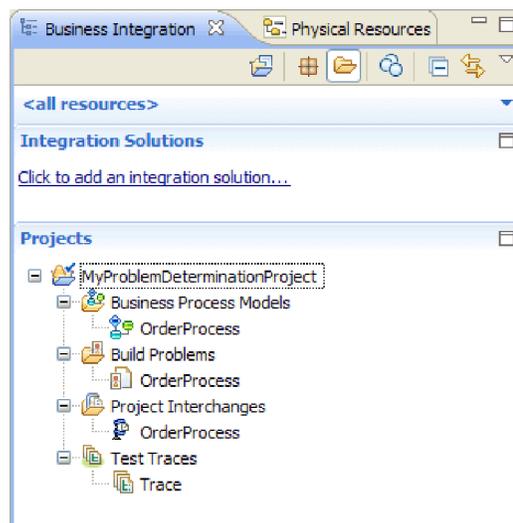
After an IT developer has determined the nature of any problems, the developer typically advises the WebSphere Business Modeler business user about the changes that need to be made to the business process models to resolve the problems.

Importing a WebSphere Business Modeler problem determination archive

In WebSphere Integration Developer, you can import a problem determination archive that was generated in WebSphere Business Modeler. This enables you to use the specialized problem determination tools of WebSphere Integration Developer to investigate and resolve the problems that are found in the archive.

To import a WebSphere Business Modeler problem determination archive:

1. In WebSphere Integration Developer, switch to the **Business Integration** view.
2. From the **File** menu, select **Import**. The Import wizard opens.
3. In the Import wizard, expand **Business Integration** and select **WebSphere Business Modeler problem determination archive**.
4. Click **Next**. The Import Problem Determination Archive wizard opens.
5. Beside the **From file** field, click **Browse**. The Open dialog box is displayed.
6. In the dialog box, locate and select the problem determination archive zip file and then click **Open**. The **From file** field now displays the name of the problem determination archive file.
7. Click **Next**.
8. In the **Project name** field, specify a name for the new project that will contain the imported problem determination archive.
9. Click **Finish**. In the Business Integration view, the problem determination archive is imported into the new project, as shown in the following figure:

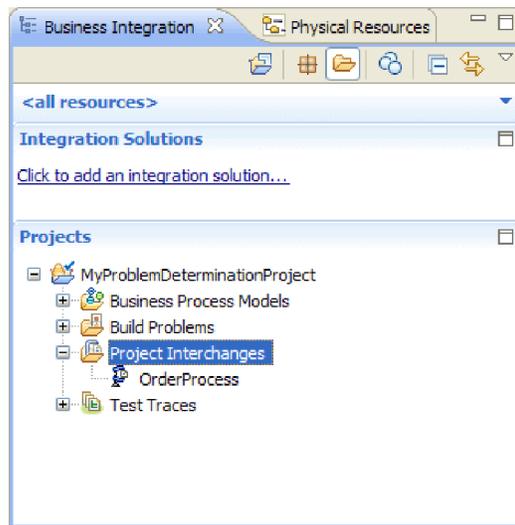


Loading project interchange resources into the workbench

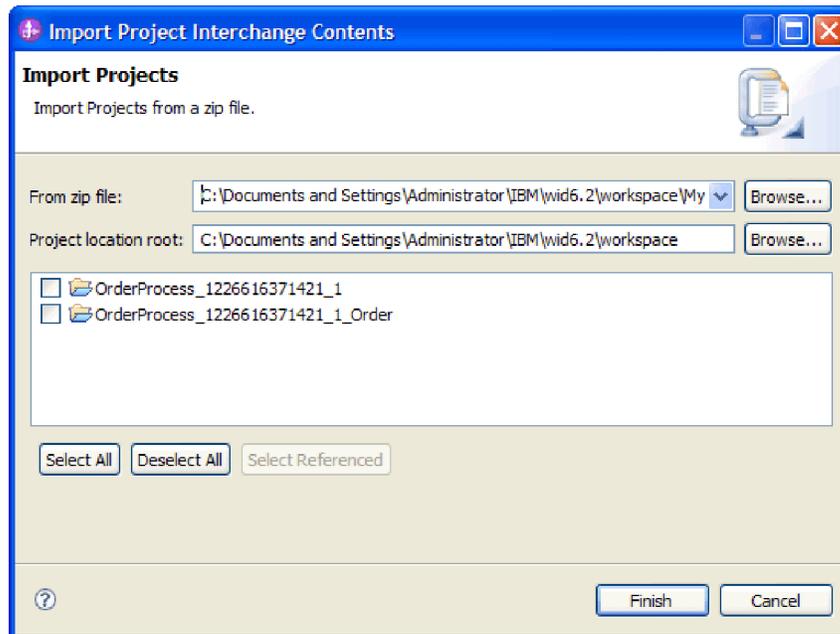
If business processes and modules are successfully generated from business process models in WebSphere Business Modeler but problems are later encountered and a problem determination archive is generated, the archive will contain one or more project interchange files that hold the generated business processes and modules. If you want to accurately view or rerun a test trace in the integration test client, or if the WebSphere Business Modeler workspace log files indicate problems in the business processes and modules and you want to examine them, you first need to load these project interchange resources into the workbench.

To load project interchange resources into the workbench:

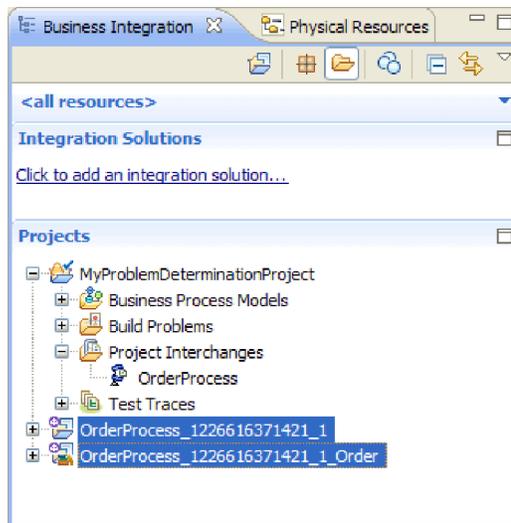
1. In the Business Integration view, expand your problem determination project and then expand the **Project Interchanges** folder. The Project Interchanges folder contains one or more project interchange files, as shown in the following figure:



2. In the Project Interchanges folder, right-click a project interchange file and select **Import to Workspace**. The Import Project Interchange Contents wizard opens, as shown in the following figure:



3. Click **Select All** to select all of the resources in the list.
4. Click **Finish**. The project interchange resources are loaded into the Business Integration view, as shown at the bottom of the following figure:



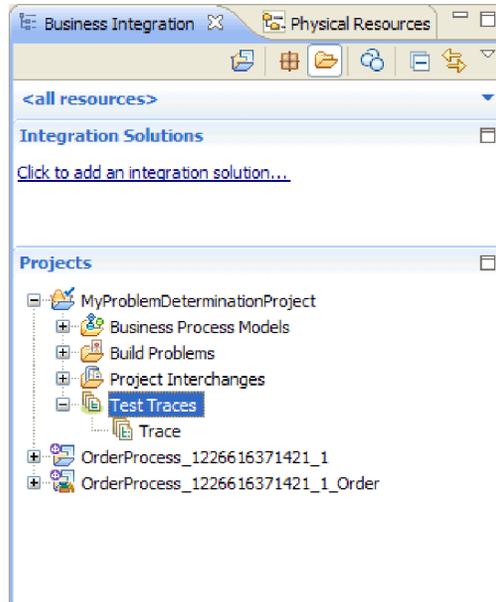
Viewing or rerunning test traces in the integration test client

When business processes and modules are generated from business process models and are transmitted to a managed deployment environment, a build automatically occurs that generates a J2EE application from the business processes and modules. If the build is successful but problems are encountered when the application is run and a problem determination archive is subsequently generated, the archive will typically contain a test trace that you can view or rerun in the integration test client.

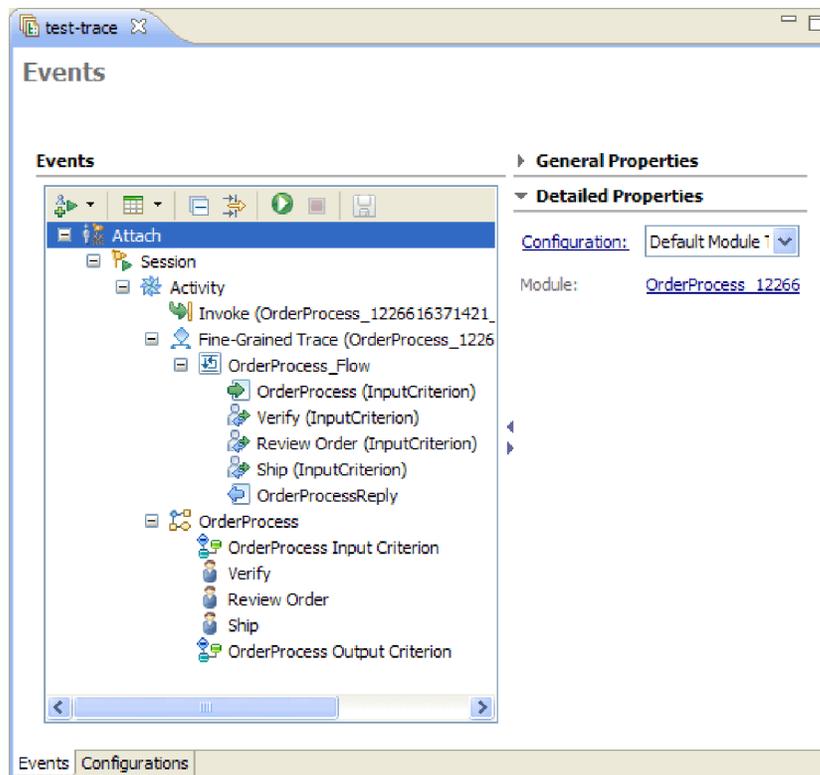
If you want to accurately view a test trace in the integration test client or rerun a test trace in the test client, you must first load the project interchange resources into the workbench, as described in the topic "Loading project interchange resources."

To view or rerun test traces:

1. Complete the following steps to ensure that when you select a fine-grained trace event in the integration test client, the process model viewer or the process editor will automatically open in split-editor mode:
 - a. From the **Window** menu, select **Preferences**. The Preferences window opens.
 - b. In the Preferences window, expand both **Business Integration** and **Integration Test Client** and then select **Fine-Grained Trace**. The Fine-Grained Trace preferences page opens.
 - c. In the Fine-Grained Trace page, select the check box **When clicking an event, open the component editor in split-editor mode with the test client**.
 - d. Click **OK**.
2. In the Business Integration view, expand your problem determination project and then expand the **Test Traces** folder, as shown in the following figure:



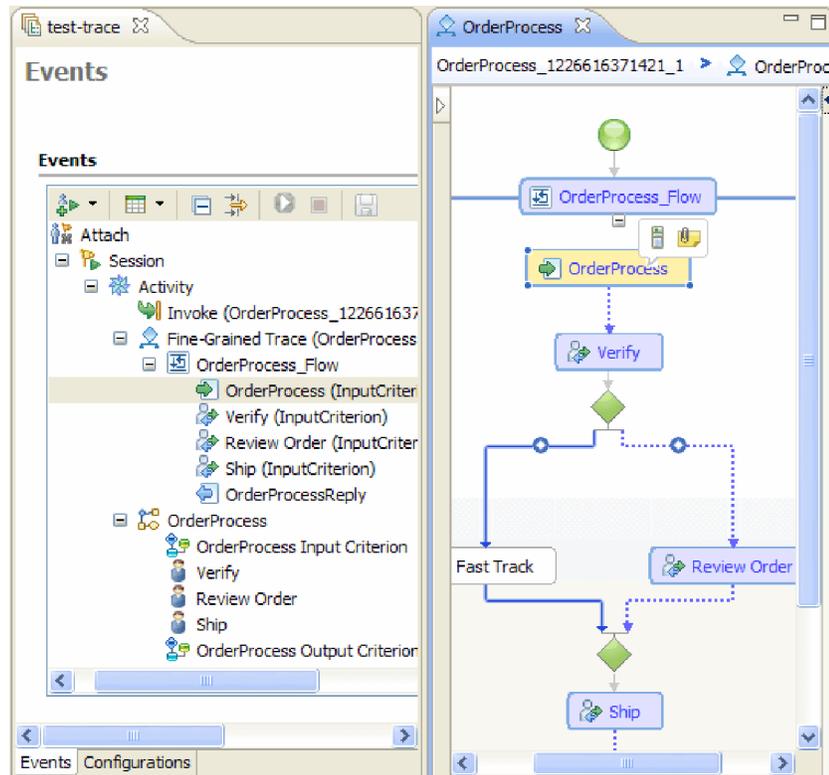
3. Right-click the **Trace** file and select **Open**. The test trace opens as an attachment in the test client, as shown in the following figure:



In the **Events** area, fine-grained trace events are displayed for both the business process models and the generated business processes. These events correspond to the request and response actions that occurred in the business process when it was run. The events for a business process model are nested under an event that is identified by the business process model symbol . The events for a generated business process are nested under an event that is identified by the

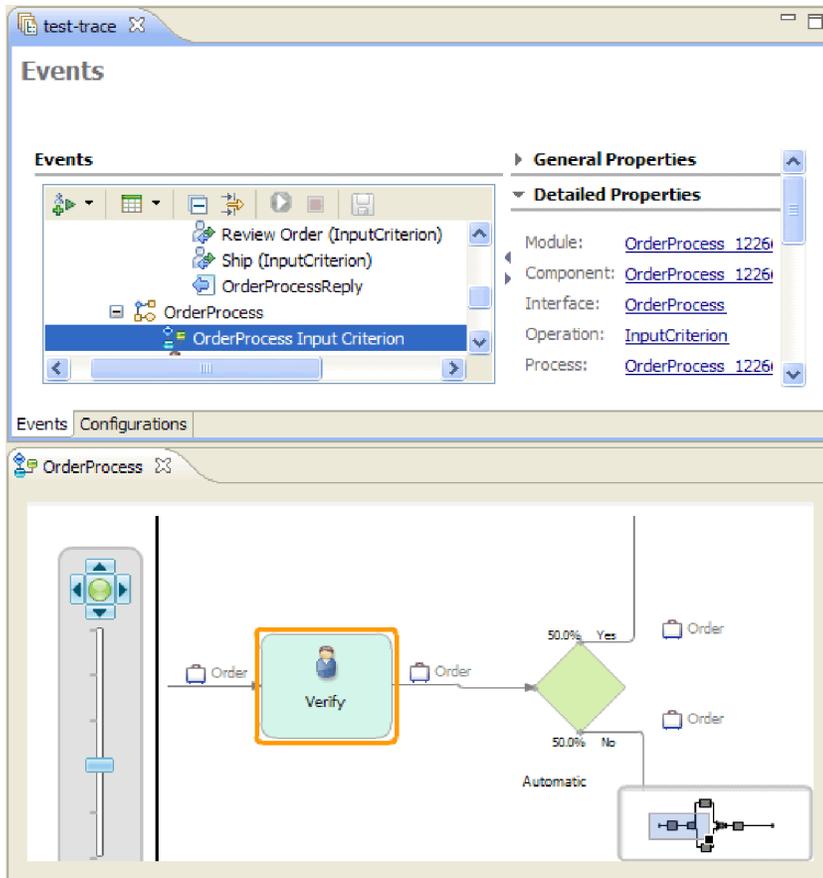
business process symbol . In the Events area, you can select one event after another to see the data that is associated with each event in the **Detailed Properties** area.

4. If you simply want to *view* the fine-grained trace events in the test client without rerunning the test trace, complete the following steps:
 - a. In the Events area of the test client, select a fine-grained trace event for a generated business process. When a Generated File Warning dialog box opens, click **Yes**. The Detailed Properties area of the test client displays the data that is associated with the selected event and the business process editor opens in split-screen mode to display the business process. In the business process editor, the business process activity that is associated with the selected event is highlighted in yellow, as shown in the following figure:



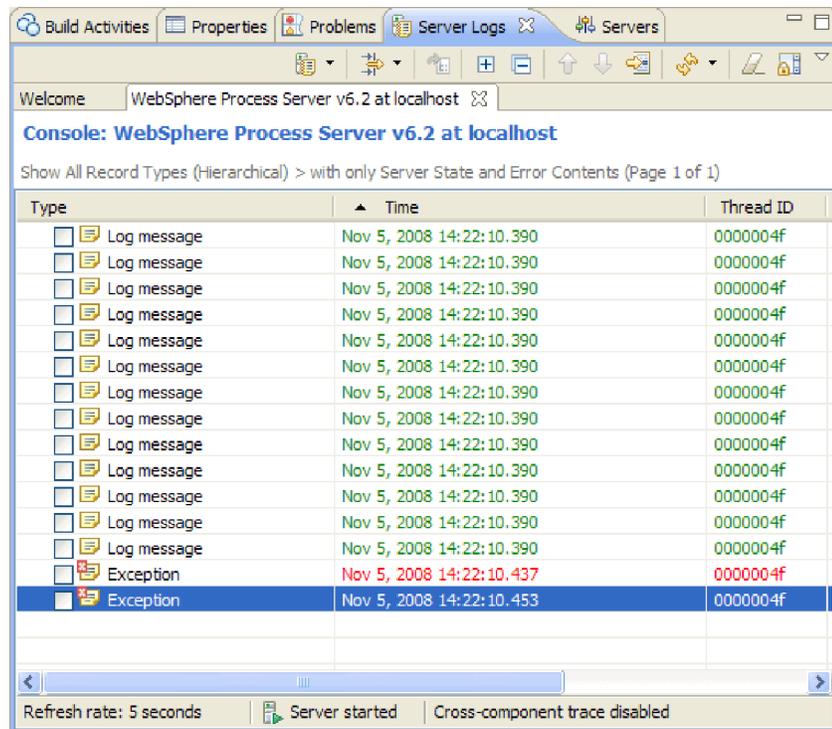
Each time you select a different event for the business process in the Events area of the test client, the associated activity is automatically selected and highlighted in the business process editor.

- b. Close the business process editor.
- c. In the Events area of the test client, select a fine-grained trace event for a business process model. The Detailed Properties area of the test client displays the data that is associated with the selected event and the process model viewer opens in split-screen mode to display the associated business process model. In the process model viewer, the task associated with the selected event is highlighted with a yellow border, as shown in the following figure:



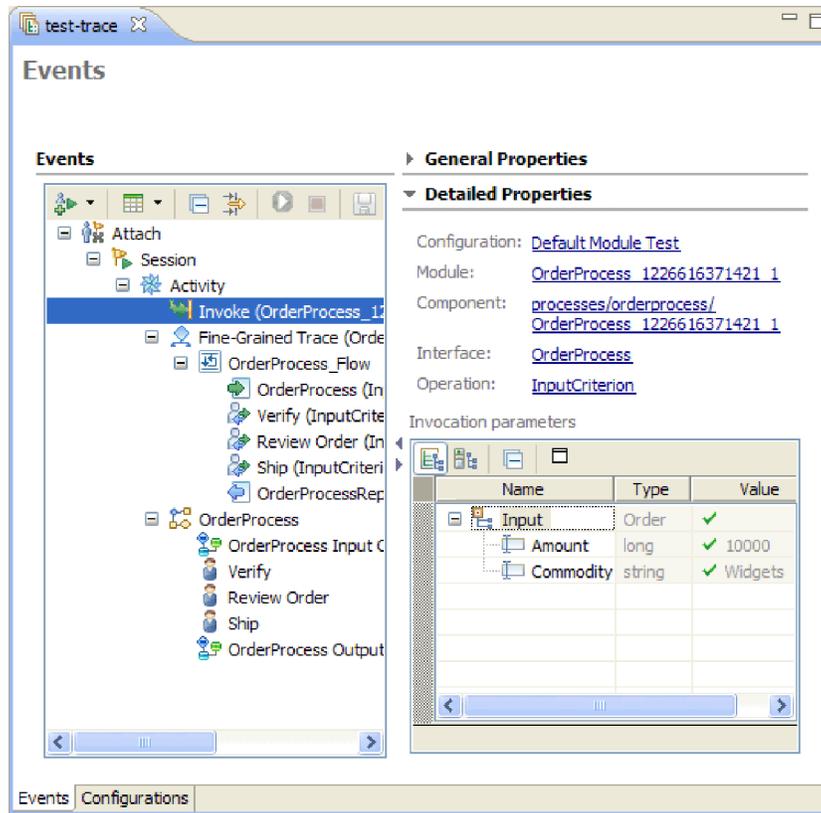
Each time you select a different event for the business process model in the Events area of the test client, the associated task is automatically selected and highlighted in the process model viewer.

- d. Close the process model viewer.
5. If you want to *rerun* the test trace in the test client, complete the following steps:
 - a. In the workbench, click the **Server Logs** tab to open the Server Logs view in preparation for enabling cross-component tracing on the server that you will use to rerun the test trace. (When you enable cross-component tracing and rerun the test trace, invocation records and invocation input and output data are generated and displayed as part of the server console records in the Server Logs view. These invocation records and data can help you analyze and resolve problems.)
 - b. In the Server Logs view, ensure that the server console tab is open and selected for the *running* server where you want to enable or disable cross-component tracing, as shown in the following figure:

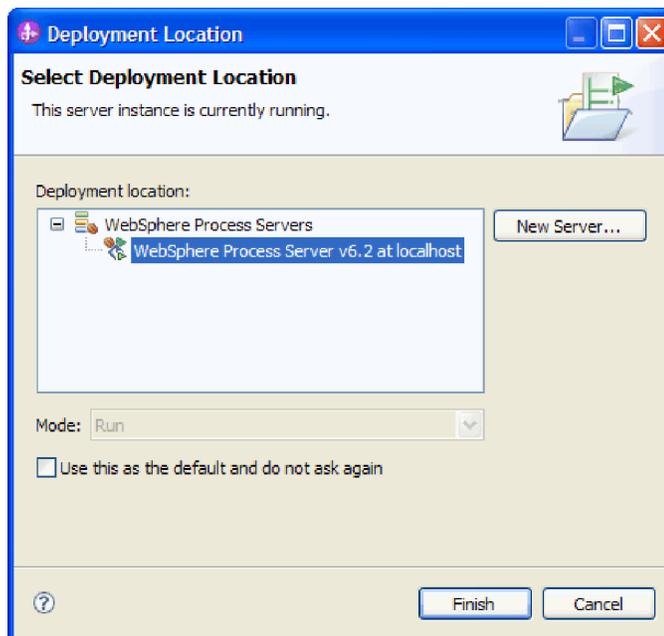


By default, the contents of a server console are automatically loaded into a new tab in the Server Logs view whenever a server is started.

- c. If the server console tab is not visible in the Server Logs view, click the down arrow beside the Load Server Console or Log icon , then select **Load from Server Console > server_name** (where **server_name** is the name of the server for the console that you want to load). The contents of the console for the selected server are loaded into a tab in the Server Logs view.
- d. In the Server Logs view, click the **View Menu** icon . A menu opens.
- e. From the **Cross-Component Trace State** menu, select **Enabled with Data Snapshot** to generate invocation records with invocation data.
- f. In the Events area of the test client, select the **Invoke** event as shown in the following figure:



- g. Right-click the selected **Invoke** event and select **Rerun**. The Deployment Location wizard opens, as shown in the following figure:



- h. In the **Deployment Location** list, ensure that the server that is listed is the same server for which you have enabled cross-component tracing.
- i. Click **Finish**. The module is deployed to the server and the test client reruns the test trace and begins to return the results.

- j. If your business process contains a human task, you need to claim and complete the task before the test client will return the results of your rerun test trace. You can use either Business Process Choreographer Explorer or Business Space to claim and complete the task and you can open them by right-clicking your server in the **Servers** view and selecting either **Launch > Business Process Choreographer Explorer** or **Launch > Business Space**.

After you have finished rerunning your test trace and you have examined the results in the test client, you can analyze the server console and log records in the Server Logs view and gain additional insight into any problems, as described in the topic "Analyzing server console and log records."

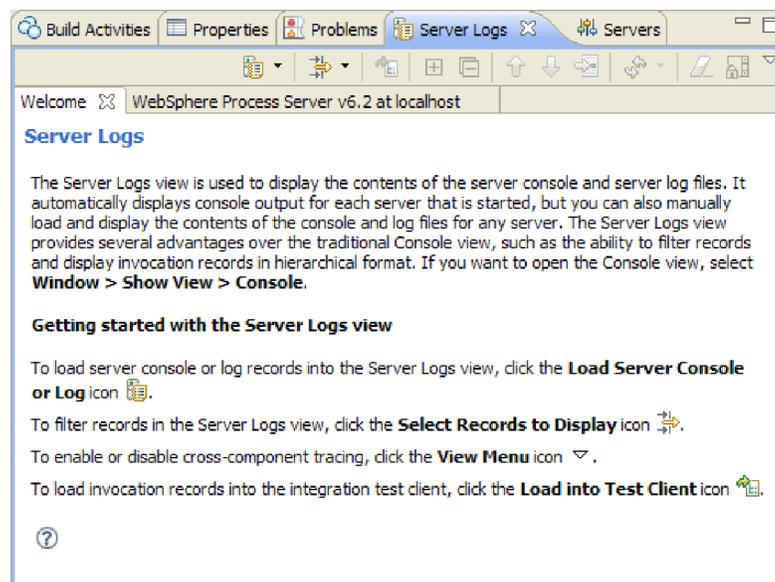
Analyzing server console and log records in the Server Logs view

In WebSphere® Integration Developer, the Server Logs view is used to display the contents of server consoles and log files. It automatically displays console output for each server that is started, but you can also manually load and display the contents of the server console and log files for any server. If you have enabled cross-component tracing, the Server Logs view will also display invocation records that can contain the invocation data that passed between components.

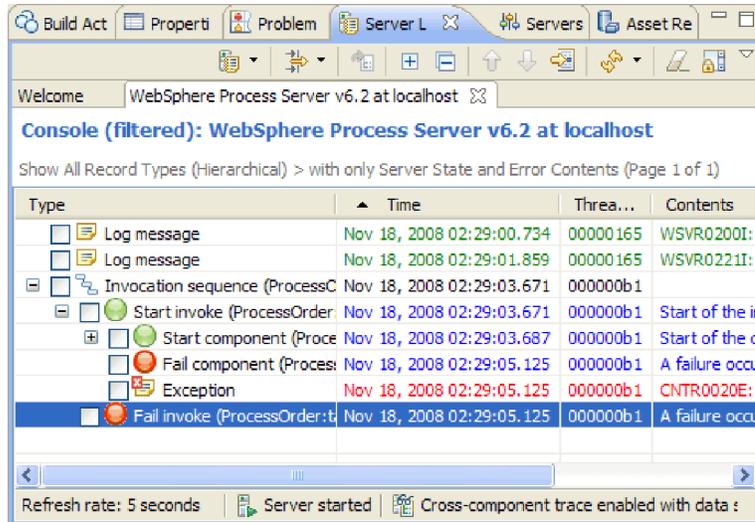
Before you work with the Server Logs view, you should enable cross-component tracing with data snapshot, as described in the topic "Viewing and rerunning test traces."

To analyze server console and log records:

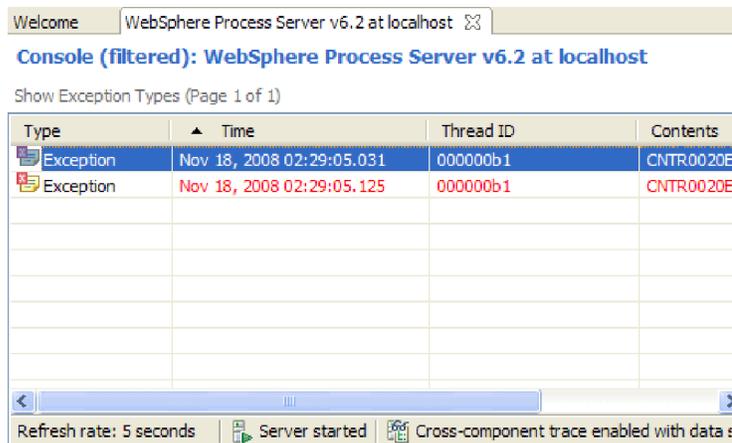
1. In the workbench, click the **Server Logs** tab to open the Server Logs view, as shown in the following figure:



2. In the Server Logs view, click the tab for your server console. The tab displays some invocation records, as shown in the following figure:

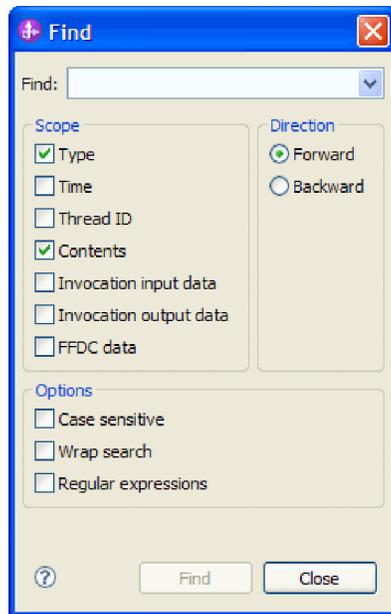


3. If there are multiple pages in the server console tab, you can use the **Page Up** and **Page Down** icons to move up or down one page at a time.
4. Right-click an invocation record and select **Properties**. The Properties dialog box opens and displays the contents of the record. Depending on the type of record that you selected, the Properties dialog box may also contain the invocation data that was passed when the test trace was rerun.
5. Click the **Select Records to Display** icon . A menu opens.
6. From the menu, select **Exception Types**. The console tab is filtered to only display exception records, as shown in the following figure:



Note that the menu also enables you to filter the server console tab in numerous other ways, such as by specific invocation types and FFDC records.

7. Click the **View Menu** icon . A menu opens.
8. From the menu, select **Find**. The Find dialog box opens, as shown in the following figure:



9. Use the Find Dialog box to search the server console records for a specific search string.
10. When you have finished analyzing the server console records, click the **View Menu** icon . A menu opens.
11. From the **Cross-Component Trace State** menu, select **Disabled** to disable cross-component tracing (which improves server performance).

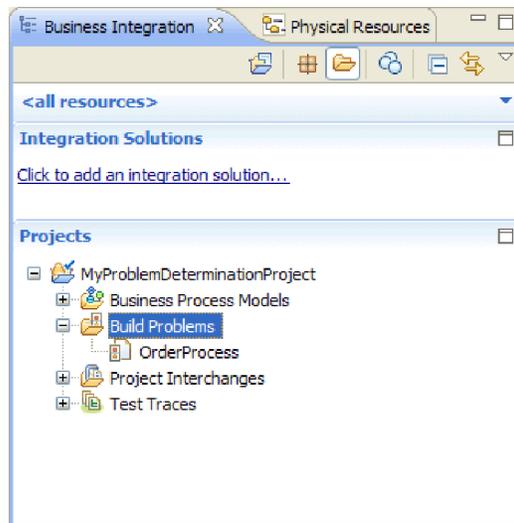
Detailed information about working with server console and log records in the Server Logs view is found in the WebSphere Integration Developer topic "Using the Server Logs view for problem determination" and its subtopics.

Viewing serviceDeploy build problems

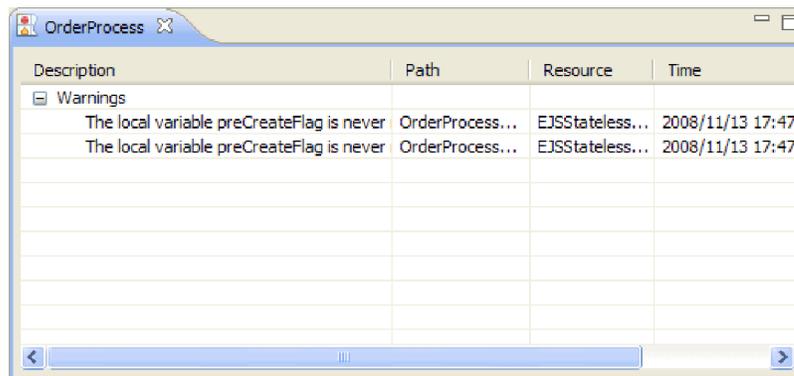
When business processes and modules are generated from business process models and are successfully transmitted from WebSphere Business Modeler to a managed deployment environment, a serviceDeploy build automatically occurs that generates a J2EE application from the business processes and modules. If problems are encountered during the build and a problem determination archive is generated, the archive will typically contain one serviceDeploy build log for each business process. You can open the build logs in the build problems viewer and view any error or warning data that was generated during the build.

To view serviceDeploy build problems:

1. In the Business Integration view, expand your problem determination project and then expand the **Build Problems** folder. The Build Problems folder contains one or more build logs, as shown in the following figure:



2. In the Build Problems folder, right-click a build log and select **Open**. The build problems viewer opens, as shown in the following figure:



3. Use the build problems viewer to analyze the error and warning data in the build log.

Viewing Eclipse workspace logs

In WebSphere Business Modeler, an Eclipse workspace log is created when generated business processes and modules are transmitted to a managed deployment environment. The Eclipse workspace log contains error codes and the communication messages that occurred between WebSphere Business Modeler and the managed deployment environments of WebSphere Business Monitor or WebSphere Process Server. If project interchange files are present in the WebSphere Business Modeler problem determination archive, the Eclipse workspace log is of little use for problem determination purposes.

To view Eclipse workspace logs:

1. In your file system, navigate to the location of the zip file for the WebSphere Business Modeler problem determination archive.
2. Unzip the file and change to the **workspace-logs** directory.
3. Open the Eclipse workspace **.log** file in a text editor.
4. Use the error codes in the tables below to help resolve any problems that you find in the Eclipse workspace log.

The error codes in the following tables are primarily related to internal server errors that occur in the managed deployment environments of WebSphere Business Monitor or WebSphere Process Server. If you encounter one of these errors, you may be able to resolve it by retrying the operation that resulted in the error, fixing any content problems indicated in the error message, or having your system administrator correct or reset the server state. However, if the errors persist, you may need to obtain assistance from IBM Software Support.

Table 1. WebSphere Business Monitor

Error	Error Code	Translated Error Message
Concurrent deployment (deploy/undeploy/cleanup)	E0001	Cannot perform a deploy or undeploy on this resource because there is an operation still in progress. ¹
No content (query API)	E0002	There is no content found for this resource.
Resource not found (all APIs that contain resource id)	E0003	The resource with id "{0}" was not found.
Illegal source state (deploy/undeploy/cleanup)	E0004	
model not found (deploy)	E0005	A valid model was not found in the archive.
model differs from last (deploy)	E0006	The model sent in the update archive differs from the existing one.
multiple models in PI (deploy)	E0007	More than one Monitor model file was found.
build failure (deploy)	E0008	Build failure. (mmdeploy could not build the Monitor EAR file)
application already installed (deploy)	E0009	The application "{0}" already exists.
application does not exist (deploy, undeploy)	E0010	The application "{0}" does not exist.
model name too long (deploy)	E0011	The model name is too long. It must be less than 61 characters.
Internal server error	E1000	Internal server error.
time out	E2000	Operation timed out.

¹ This error code indicates that another user is currently deploying or undeploying applications to the server. If the server consistently returns this error although no deployments or undeployments are active, you should attempt to restart the server. If the problem persists, contact IBM Software Support.

Table 2. WebSphere Process Server

Error	Error Code	Translated Error Message
Concurrent deployment (deploy/undeploy/cleanup)	E0001	Cannot perform a deploy or undeploy on this resource because there is an operation still in progress. ²
No content (query API)	E0002	There is no content found for this resource.

Table 2. WebSphere Process Server (continued)

Error	Error Code	Translated Error Message
Resource not found (all APIs that contain resource id)	E0003	The resource was not found.
Illegal source state (deploy/undeploy/cleanup)	E0004	
model not found (deploy)	E0005	A valid model was not found in the archive.
model differs from last (deploy)	E0006	The model sent in the update archive differs from the existing one.
multiple modules in PI (deploy)	E0007	More than one SCA module was found.
build failure (deploy)	E0008	Build failure.
application already installed (deploy)	E0009	The application already exists.
application does not exist (deploy, undeploy)	E0010	The application does not exist.
Internal server error	E1000	Internal server error.
time out	E2000	Operation timed out.

² This error code indicates that another user is currently deploying or undeploying applications to the server. If the server consistently returns this error although no deployments or undeployments are active, you should attempt to restart the server. If the problem persists, contact IBM Software Support.

Viewing WebSphere Business Modeler workspace logs

In WebSphere Business Modeler, one or more WebSphere Business Modeler workspace logs are created when business processes and modules are generated from business process models. These workspace logs can contain transformation errors and other errors. If project interchange files are present in the WebSphere Business Modeler problem determination archive, these workspace logs are of little use for problem determination purposes.

To view WebSphere Business Modeler workspace logs:

1. In your file system, navigate to the location of the zip file for the WebSphere Business Modeler problem determination archive.
2. Unzip the file and change to the **workspace-logs** directory.
3. Open one or more WebSphere Business Modeler workspace log files in a text editor. These files have the format **WBModeler*.log**.
4. Use the transformation errors and other errors to help resolve the problems in the problem determination archive.

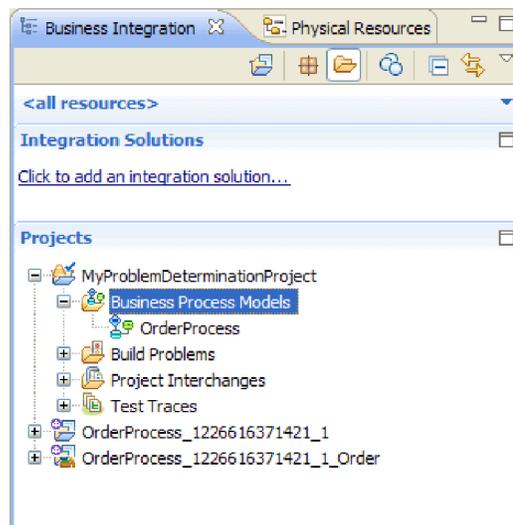
Viewing business process models

If problems are encountered at any stage in the end-to-end deployment of business processes to a managed deployment environment and a problem determination archive is subsequently generated, the archive will typically contain visualization files for one or more business process models. Although you can use the process

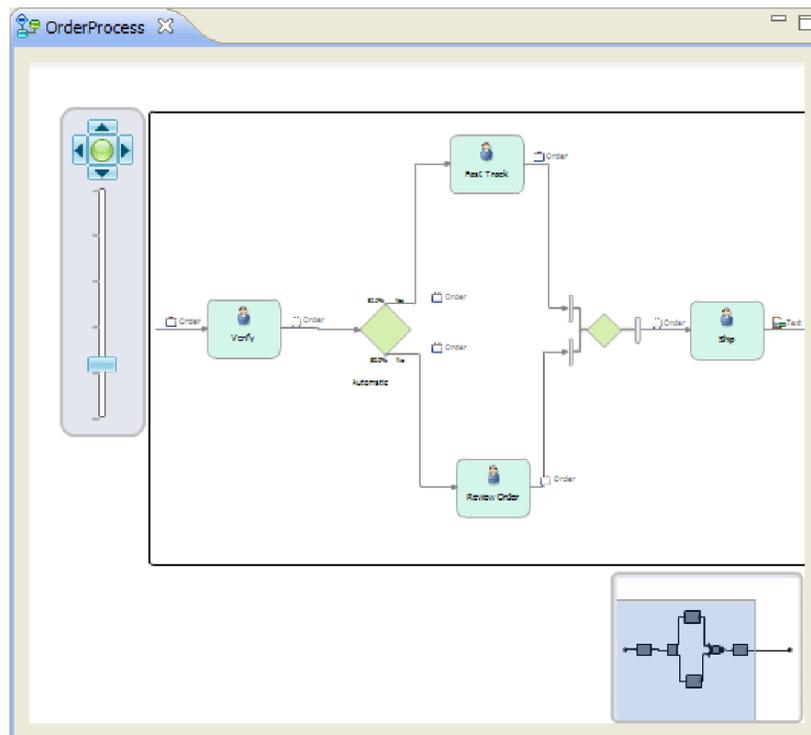
model viewer to open the visualization files and view their content and structure, the process model viewer is most useful when it is used in conjunction with the integration test client.

To view business process models:

1. In the Business Integration view, expand your problem determination project and then expand the **Business Process Models** folder. The Business Process Models folder contains one or more business process models, as shown in the following figure:



2. In the Business Process Models folder, right-click a business process model and select **Open**. The process model viewer opens, as shown in the following figure:



In the lower right corner, an overview image shows the process model in its entirety. In the upper left corner, you can use the zoom tool to zoom the process model in and out or to shift the process model in any direction.

3. In the process model viewer, scroll or zoom the process model to view its content and structure.

Chapter 5. Deploying business processes to production

After you finish testing your business process application and resolve any deployment problems, you should export it to WebSphere Integration Developer for deployment to production. Some configurations, such as the module version, might require modification by an IT developer to conform with IT governance in the production environment.

Before you export your business process, ensure that you have replaced any placeholder human tasks with service implementations or specifications.

If you have not associated forms with a human task in your business process, the export process will generate default forms for this task based on the task inputs and outputs.

To send a tested business process application to IT for deployment:

1. Ensure that you are in WebSphere Process Server mode.
2. Export the project for your business process to WebSphere Integration Developer using the Module + Library export option.

Depending on the production environment and governance process required, one or more of the following configurations might need to be set by an IT developer before the business process application can be deployed to production.

Configuration	Specification
Versioning	The default module version number is 1.0.0. This version number might require updating based on the set of deployed applications in the target deployment environment. The IT developer can use the compare and merge function in WebSphere Integration Developer to set the correct version information.
Late or early binding for process-to-process interaction	Late binding is used by default (that is, one process invokes the latest deployed version of the other process). If this behavior is not appropriate in the target deployment environment, the IT developer can update the template name that refers to the targeted process name.
Human task binding (role mapping)	If the role mapping defined for the test server is not valid for a production environment, the IT developer can update the role assignment for each human task in WebSphere Integration Developer to use the unique name of the group specified in the people directory (which might be an LDAP directory).
Business service classification	If the business service classifications for WebSphere Services Registry and Repository used for testing are different than those used in production, the IT developer can reconfigure these classifications in one of two ways: <ul style="list-style-type: none">• In WebSphere Integration Developer, update the dynamic lookup primitive setting for each mediation component to refer to the new set of classifications.• In WebSphere Business Modeler, ensure that the business process is in the workspace, import the new business service classifications, update the call to the business services that use the new classifications, and then export the business process to WebSphere Integration Developer.

Configuration	Specification
Module structure	<p>If the module structure has to be modified to match the project structure required in the target deployment environment, the processes might need to be packaged into the same or different modules. The IT developer can do this repackaging using WebSphere Integration Developer. Or IT developer can ask the business analyst to change the project structure (in the Project Tree view) to map to the target deployment module structure and then reexport the business process to WebSphere Integration Developer.</p>
Monitor model	<p>If changes are made to finalize the business process for deployment to production, the corresponding monitor model also needs to be updated using WebSphere Integration Developer installed with the WebSphere Business Monitor development toolkit. For example, any of the following changes would require updates to the monitor model:</p> <ul style="list-style-type: none"> • Module version was modified • Process template name was updated • Hierarchy of process content was changed (for example, a task was moved into a local subprocess) • Business measure type was changed • Process inputs or outputs were changed <p>The IT developer can use the compare and merge function in WebSphere Integration Developer to identify such changes.</p>

Related information

-  [Exporting files to WebSphere Integration Developer](#)
-  [Creating services](#)

Limitations and restrictions for deploying processes from WebSphere Business Modeler

This topic describes the limitations and restrictions for developing deployable business processes using WebSphere Business Modeler.

Control flow restrictions

The following types of modeling constructs are not supported for direct deployment. In most cases, the semantic validation returns errors when these unsupported constructs are detected. However, in some cases, the constructs cannot be identified with certainty, so a warning is returned instead. When you receive a warning about an unsupported construct, you should examine your model and revise it accordingly.

Unsupported construct	Example	Recommended replacement
Exclusive gateways with downstream parallel gateways.	An upstream exclusive decision flows into a downstream join.	Pair an exclusive decision with a downstream merge instead of a join. When using disjoint output criterion to model alternate paths, make sure that downstream input criterion are not set up to receive inputs that can never be produced in a single execution.
Inclusive gateways with a downstream exclusive gateway.	A decision with multiple branches that can execute simultaneously flows into a downstream merge.	Pair an inclusive decision or fork with a downstream join instead of a merge.
Parallel gateways with a backward connection.	A merge following by a fork with a backward connection from one of the parallel paths of the fork to the merge.	Use only sequential paths inside a process fragment with a backward connection.
Inclusive decisions with a backward connection.	A decision with multiple branches, one of which loops back to flow into the same inclusive decision.	Use only sequential paths inside a process fragment with a backward connection.
Global process invocations that loop back to their own input.	A global process invocation that produces an output that links back with a backward connection to the global process input.	Use local subprocesses instead of global processes to loop backward from the subprocess output to the subprocess input.

For information about resolving errors related to the above restrictions, see Modeling branching behavior.

Using services in directly deployed applications

The only supported implementation type for tasks or services is Import - Web Service binding or None. Processes containing tasks or services with other implementation types cannot be deployed.

If you have multiple WebSphere Service Registry and Repository servers, ensure that the default WebSphere Service Registry and Repository server that is configured for your test environment is the same server where the service endpoints for any services in your processes are published.

If you used a classification system to assign classifications to tasks or services in your process, the classification system must be the same classification system as the default WebSphere Service Registry and Repository server that is configured for your test environment.

Human tasks in a directly deployed application

Human tasks in deployable processes should have a primary owner specified. If no primary owner is specified for a human task, the task will run with the Everyone query, which means that any user can claim and work on the human task.

The primary owner for a human tasks must have an assigned role. The primary owner cannot be based on a individual resource definition.

The only people assignment criteria supported for deployed process is **Members by role name**. You must specify both a role for the human task and the **Members by role name** people assignment criteria for a human task to be deployable.

All human tasks in deployable processes must have input and output forms associated with them. If no forms are specified, default forms are generated based on the data structure of the input and output.

When setting up escalations on human tasks, if the notification type is "email", the notify setting cannot be "none" or "Members by group ID". In order for a notification email to be sent, a target for notification must be specified, therefore the setting cannot be "none". The "Members by group ID" setting results in too many targets for email notification at runtime.

Business rules in a directly deployed application

Processes cannot be deployed unless all business rules tasks are fully specified, including the following specifications:

- At least one fully-specified business rule must be specified for each business rules task.
- Rule conditions and rule actions must be specified for if-then rules.
- Rule presentations must be defined for each business rule template.
- A default business rule must be selected and scheduled for each business rules task.

Business measures

Instance metrics must be based on predefined business measure templates or fully-specified calculation expressions.

Certain business measure templates are not tied to an implementation in WebSphere Process Server, and therefore are not deployable. These include the Is Delayed and the Calling Process Name templates.

Certain business measure templates are also not tied to an implementation in WebSphere Process Server when associated with loops or subprocesses, and therefore are not deployable. These include the Working Duration, Business Item Input, and Business Item Output templates.

Aggregate metrics must be based on instance metrics that are fully-specified or that are based on a predefined business measure template.

KPIs must be based on fully-specified expressions that make reference to instance metrics that are supported for deployment.

Some types of monitored values are not supported for deployment when associated with loops or subprocesses. These include the values for Processing Cost, Start up Cost, and Revenue.

When defining a business measure based on an expression, the current-date, current-datetime, and current-time functions cannot be used to fully specify a business measure for deployment.

Business measures that are not fully specified are flagged with warnings. These warnings do not prevent the deployment of the business process, however, if the business process is deployed, the business measures with warnings are omitted from the deployed monitor model.

Business measures that are flagged with errors prevent the business process from being deployed. These errors must be resolved before the process can be tested in the process testing environment.

Input and output restrictions for deployable processes

If the input and output of a process are the same in data type or name, the same form must be used for both the input and the output form.

If a process does not have a 1-to-1 mapping between input criteria and output criteria, then the process output will not be viewable in a form.

Only processes with a single input set and single output set will be deployable.

Processes that contain global subprocesses with multiple input criteria or output criteria cannot be directly deployed.

Namespace length limitation

There is a 220 byte limitation on namespaces for deployed elements. In some cases, due to character expansion in translation, WebSphere Business Modeler might generate a namespace for an element that exceeds this limit, and the process cannot be deployed. If your process encounters this limitation, it is identified in the error log file. You can remedy this problem by shortening the element name or path, or by specifying a value for the element's Target namespace in the Technical Attributes view.

Using maps to swap attribute values in a business item

You can use a map element in your process to transform the data produced by one activity so that it conforms to the data expected by another activity. In some situations, you might also want to use a map to swap around the values of different attributes in the same business item. For example, if you need to transform your employee data so that the values for primary and secondary beneficiaries are swapped.

You cannot use a map element where both the source (the map input) and the target (the map output) are the same business item, and the values that are being mapped overwrite each other. For example, you can have a map that takes the value of field A and maps it to field B in the same business item, but the same map can't also map the value of field B to the value of field A, swapping the values of fields A and B.

If you want to use a map element to swap the values in a business item, you need to create a dummy business item that has the same values as the real business item and use this dummy business item as the intermediate target or source for the map. Once the data has been swapped in the dummy business item, use another map element to pass the data values to the real business item.

Restricted concurrency in deploying to a shared test server

Only one process deployment can occur at a time on the test server. If multiple users are using the same managed server environment to test their business processes, a user might receive a message indicating that the server is busy if another user is currently deploying a process. The second user must wait until the first user's process is deployed, then they can re-invoke the Test on Server action.

No validation of imported XML schema definition files

WebSphere Business Modeler does not validate XML schema definition (XSD) files that you import into your workspace and use in your process models. Because no validation is performed, WebSphere Business Modeler does not report any validation errors even if the XSD files are not valid. WebSphere Business Modeler does not prevent you from testing processes that contain files that are not valid, nor does it prevent you from exporting such processes to WebSphere Integration Developer for subsequent deployment.

Related information

-  [Modeling branching behavior](#)
-  [Mapping values](#)
-  [Creating business measures](#)
-  [Services](#)
-  [Business services and business service objects](#)

Notices and Trademarks

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 documentation does not grant 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 character set (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 106-0032, 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 the 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:

*Intellectual Property Dept. for WebSphere Software
IBM Corporation
3600 Steeles Ave. East
Markham, Ontario
Canada L3R 9Z7*

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 document 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 herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for 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 claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the 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 sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample 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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

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

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Readers' Comments — We'd Like to Hear from You

Business Process Management products

Version 6 Release 2

Developing deployable business processes using WebSphere Business Modeler

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

E-mail address

Readers' Comments — We'd Like to Hear from You



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Canada Ltd. Laboratory
Information Development for WebSphere Business Modeler
8200 Warden Avenue
Markham, Ontario
Canada L6G 1C7

Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line