Developing Deployable Business Processes Using WebSphere Business Modeler
Before using this information and the product it supports, be sure to read the information general information under "Notices and Trademarks" on page 63.
Contents

Overview ....................................................... v

Chapter 1. Preparing for modeling and testing deployable business processes .... 1
Software prerequisites for testing process deployment ........................................ 2
Setting up the WebSphere Business Modeler environment for deploying processes .......................................................................................... 3
Setting up a WebSphere Business Modeler environment for deploying processes .......................................................................................... 3
Setting up a managed deployment environment on WebSphere Integration Developer ......................................................................................... 4
Configuring security for a managed deployment environment on WebSphere Integration Developer ............................................................. 6
Setting up a managed deployment environment on WebSphere Process Server ......................................................................................... 6
Configuring security for a managed deployment environment on WebSphere Process Server ................................................................. 6
Defining the roles required for human tasks .......................................................... 12
Setting up a server configuration file ..................................................................... 13

Chapter 2. Modeling processes for deployment .................................................. 17
Example: New Employee Setup Process ................................................................ 17
Modeling human tasks for deployment .................................................................. 18
Refining the interfaces for your deployable process ............................................. 19
Adding services to deployable applications .......................................................... 23
Using placeholder tasks for testing processes with unimplemented activities .......... 24
Modeling business measures for deployment ...................................................... 25

Chapter 3. Testing business process deployment ................................................ 27
Deploying business processes on a test server ..................................................... 27
Adding and changing test servers for deploying processes ................................... 29
Testing process flow .............................................................................................. 29
Testing human task assignments .......................................................................... 31
Testing business measures .................................................................................. 32
Deleting deployed processes using WebSphere Business Modeler ...................... 33

Chapter 4. Maintaining the managed deployment environment .......................... 35
Restoring the default layout of a process verification environment ...................... 35
Uninstalling deployed resources using the administrative console ...................... 36
Maintaining the WebSphere Business Monitor server in a managed deployment environment ................................................................. 36

Chapter 5. Troubleshooting business process deployment ................................... 39
Overview of WebSphere Business Modeler problem determination ................. 39
Importing a WebSphere Business Modeler problem determination archive .......... 41
Loading project interchange resources into the workbench ............................... 41
Viewing or rerunning test traces in the integration test client ............................... 43
Viewing business process models ....................................................................... 49
Viewing serviceDeploy build problems ............................................................... 50
Analyzing server console and log records in the Server Logs view ...................... 51
Troubleshooting deployment failures to Windows 2003 and Windows XP servers ......................................................................................... 54
Viewing Eclipse workspace logs .......................................................................... 54
Viewing WebSphere Business Modeler workspace logs .................................... 56

Chapter 6. Deploying business processes to production .................................... 57
Limitations and restrictions for deploying processes from WebSphere Business Modeler ..................................................................................... 59

Notices and Trademarks ...................................................................................... 63
Notices ................................................................................................................. 63
Trademarks and service marks ............................................................................ 65
Overview

Accelerate your time to value by using interactive process design to create 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 deployable, human-centric business process without relying on IT to develop and test the application.

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

<table>
<thead>
<tr>
<th>Work done by IT</th>
<th>Prepare</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work done by business analyst</td>
<td>Model</td>
</tr>
<tr>
<td></td>
<td>Analyze</td>
</tr>
<tr>
<td></td>
<td>Test</td>
</tr>
<tr>
<td></td>
<td>Deploy</td>
</tr>
</tbody>
</table>

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 deployable business processes, the business analyst needs to install only a few products on their own computer. 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.
The type of managed deployment environment that the IT department sets up depends on the number of users that will be using the server for testing, and the type of testing that the users will be doing. For example, if the users expect to test business measures along with their business processes, then a monitoring server component must be included in the managed deployment environment.

For details about the products required, see “Software prerequisites for testing process deployment” on page 2.

Summary

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.
Managed deployment environment for testing

WebSphere Business Modeler Advanced

Managed deployment environment for testing

Business Space powered by WebSphere

WebSphere Process Server

WebSphere Business Monitor

Monitored application events

Process verification environment space

Monitor models Process diagram

Setup: WebSphere test environment on WebSphere Integration Developer OR a stand-alone server with server components

Note: The problem determination archive file is not generated upon deployment, but is instead a separate artifact that is generated on demand by the business analyst who has encountered issues during deployment.

Legend:

- Process verification
- Deployment to production

Exported business process application for deployment to production

Problem determination archive file

Developed deployable business processes using WebSphere Business Modeler
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 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 test along with the business process. If you are testing business measures, the managed deployment environment that is set up needs to include a monitoring server component.

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:
   • Setting up a managed deployment environment on WebSphere Integration Developer
   • Setting up a managed deployment environment on WebSphere Process Server

3. **Business analyst and IT administrator:** Define the roles required for human tasks
   If you are testing human task assignments in 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.

4. **IT administrator:** Setting up the server configuration file

After the IT administrator sends the business analyst the server configuration file, the business analyst has everything necessary to model and test the deployment of business processes. If the business analyst will be testing human task assignments in a deployable human-centric process, they will also need the completed role mapping file, and .mar project of the relevant roles (if this project was created by someone in the IT department).

As testing progresses, business analysts can free up resources on the test server by deleting their own deployed processes using the Process Execution Environment view.
Modeling processes that you plan to deploy is much like other business modeling 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.

Software prerequisites for testing process deployment

To verify the design of business processes on a test server (managed deployment environment), business analysts must ensure that they have the required software installed on their computers. An IT administrator needs to set up a test server (or managed deployment environment) that includes the required server components.

If the business analysts expect to test business measures along with their business processes, then a monitoring server component must be included in the managed deployment environment.

To simplify the testing experience for business users, the IT administrator can point to Lotus Webform Server from the managed deployment environment. The Business Space can then display forms in the process verification environment, so business analysts do not have to install Lotus Forms Viewer on the same computer as WebSphere Business Modeler to verify their process design.

Products required for five or fewer business users

If you plan to have five or fewer users performing the process testing, then the following software prerequisites apply.

<table>
<thead>
<tr>
<th>Computers for business analysts (clients)</th>
<th>Managed deployment environment (server)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required software</td>
<td>Optional software</td>
</tr>
<tr>
<td>• WebSphere Business Modeler Advanced version 7.0</td>
<td>• To customize forms in WebSphere Business Modeler, install the Lotus Forms Designer feature when you install WebSphere Business Modeler</td>
</tr>
<tr>
<td>• Adobe® Flash Player version 10.0.2 or later</td>
<td></td>
</tr>
<tr>
<td>• Lotus Forms Viewer version 3.5.1</td>
<td></td>
</tr>
<tr>
<td>• Mozilla Firefox version 3.0 or Microsoft® Internet Explorer version 7 or Microsoft Internet Explorer version 6.0 SP2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Products required for more than five business users

If you plan to have more than five users performing the process testing, then the following software prerequisites apply.

<table>
<thead>
<tr>
<th>Computers for business analysts (clients)</th>
<th>Managed deployment environment (server)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required software</td>
<td>Optional software</td>
</tr>
<tr>
<td>• WebSphere Business Modeler version 7.0</td>
<td>• To customize forms in WebSphere Business Modeler, install the Lotus Forms Designer feature when you install WebSphere Business Modeler</td>
</tr>
<tr>
<td>• Adobe Flash Player version 10.0.2 or later</td>
<td></td>
</tr>
<tr>
<td>• Lotus Forms Viewer version 3.5.1</td>
<td></td>
</tr>
<tr>
<td>• Mozilla Firefox version 3.0 or Microsoft Internet Explorer version 7 or Microsoft Internet Explorer version 6.0 SP2</td>
<td></td>
</tr>
</tbody>
</table>

Related information

[Installing Lotus Forms]

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.

The following products must be installed on the same computer as WebSphere Business Modeler Advanced:

• One of the following Web browsers: Firefox version 3.0, Microsoft Internet Explorer version 7, or Internet Explorer version 6.
• If a Lotus Webform Server is not included in the managed deployment environment that your IT administrator has set up, you must install Lotus Forms Viewer version 3.5.1.
• Adobe Flash Player version 10.0.2 or later

If you want to customize forms in WebSphere Business Modeler, install the Lotus Forms Designer feature when you do or modify a WebSphere Business Modeler installation using Installation Manager.

If you did not install the local help contents feature when you installed WebSphere Business Modeler, the deployment documentation and the help topics in the Getting Started widgets in the process verification environment are accessed remotely from the IBM documentation library on the Web. If you do not have an Internet connection, you will then get a message telling you that these topics are unavailable. To get the instructions for installing help topics and changing the way that WebSphere Business Modeler accesses these topics, click Help → Help Contents, open the “Troubleshooting and support” node in the help system navigation, and click “Accessing help topics”.

To set up the WebSphere Business Modeler environment:

1. To use a Web browser other than your default system Web browser for testing the deployment of processes:
   a. Click Window → Preferences.
b. Beside the filter text field of the Preferences window, click the Clear icon. All available preferences are displayed.

c. In the navigation tree, click General → Web Browser.

d. On the Web Browser page, click Use external Web browser.

Note: If Use internal Web browser is selected, the process verification environment opens in your default system Web browser.

e. Select the browser that you want to use for testing the deployment of your business processes.

f. Click OK.

2. To see the display of the process diagram in the process testing environment, ensure that Adobe Flash Player is installed for the Web browser that you are using for verifying your process design.

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 12

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 a managed deployment environment on WebSphere Integration Developer

If you want to use the WebSphere test environment as a managed deployment environment, set WebSphere Integration Developer up on a dedicated computer. Your WebSphere test environment can be enabled for testing the deployment of business processes from WebSphere Business Modeler.

The IT administrator should set up the WebSphere 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 WebSphere 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 the WebSphere Process Server test environment and the WebSphere Business Monitor Development Toolkit, 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.

To set up the managed deployment environment:

1. Install WebSphere Integration Developer and WebSphere Process Server as the test environment. The server components for Business Space powered by WebSphere are automatically installed with the test environment.
Note: You must create a WebSphere Process Server profile to complete the installation of WebSphere Process Server as your WebSphere test environment.

2. If your users will be testing business measures along with their business processes, ensure that the Monitor Model editor feature is selected when installing WebSphere Integration Developer ([Deciding which features to install]) and then install WebSphere Business Monitor as your WebSphere test environment ([Installing the WebSphere test environment]).

Note: You must create a WebSphere Business Monitor profile to complete the installation of WebSphere Business Monitor as your WebSphere test environment.

3. An application is provided in the WebSphere Integration Developer installation files to enable the test environment for process deployment and testing. The application is called “TestController70”. This 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. Open the administrative console by right-clicking on your WebSphere Process Server in the Servers view, and selecting Administration > Run administrative console.
   b. In the navigation tree, select Applications → Enterprise Applications.
   c. Look for the “TestController” application in the list of installed applications.
   d. 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.

4. Install the “TestController” application.
   a. In the navigation tree, select Applications > Install New Application.
   b. Click Browse and select TestController70.ear application from the WID_install\wid\util directory.
   c. Accept all default settings during installation.
   d. Save your configuration after installation.

5. Start the “TestController” application from the administrative console.
   Select the “TestController” application from the list and click Start.

6. To allow the testing of services that you call in your deployable processes, you can add a list of WebSphere Service Registry and Repository systems in the WebSphere Integration Developer preferences. You need to ensure that the WebSphere Service Registry and Repository that hosts the called services is set to the default one.

Your test environment is enabled for the deployment of business processes from WebSphere Business Modeler.

After you configure security for the managed deployment environment, you must create a server configuration file, and optionally a role-mapping file, to provide to the WebSphere Business Modeler users who will be testing business processes in the managed deployment environment. For details on how to configure Business Space to use Lotus Webform Server, see this topic in the Business Space powered by WebSphere information center.
Related tasks

“Defining the roles required for human tasks” on page 12
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 a server configuration file” on page 13
To connect to a test server (or managed deployment environment), the business analyst needs a server configuration file that contains the required information about the server instances running in the managed deployment environment. The IT administrator creates a different server configuration file for each managed deployment environment that is set up for testing processes.

Configuring security for a managed deployment environment on WebSphere Integration Developer
To test business processes in a managed deployment environment, users need to be assigned roles with the appropriate privileges. The IT administrator assigns roles using the administrative console.

You need to make sure that the individual users or groups of users who will be deploying processes to the managed deployment environment have WebSphere Application Server deployer or administrative role privileges.

1. Create user definitions.
   a. Open the administrative console by right-clicking on your WebSphere Process Server in the Servers view, and selecting Administration > Run administrative console.
   b. In the navigation tree, select Users and Groups > Manage Users.
   c. 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.

3. To delete deployed processes in WebSphere Business Modeler using the Process Execution Environment view, a user must either be the original owner who deployed the process or have super-user authority in the process testing Business Space. Assign the super-user role to any user ID that you want to be able to delete deployed processes.

Users will now have the required privileges to test 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.

Related tasks

Chapter 4, “Maintaining the managed deployment environment”, on page 35
From time to time, files will need to be removed from server components in the managed deployment environment.

Setting up a managed deployment environment on WebSphere Process Server
If you want to set up a managed deployment environment on WebSphere Process Server, install WebSphere Process Server on a stand-alone server.
The IT administrator needs to prepare a stand-alone server as a 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.

To set up a managed deployment environment:

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 must select the following components:
   - WebSphere Process Server
   - Business Space powered by WebSphere
   You can optionally select the following components:
   - WebSphere Business Monitor

   **Note:** WebSphere Process Server and WebSphere Business Monitor must be setup to run on the same stand-alone server
   - WebSphere Service 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 with the monitor profile template using the WebSphere Application Server Profile Management Tool (PMT). To use Business Space, you must select the **Configure Business Space** option during profile augmentation. For more information see [Creating and augmenting profiles](#)

4. To allow the testing of services that you call in your deployable processes, you can add a list of WebSphere Service Registry and Repository systems in WebSphere Process Server. You need to ensure that the WebSphere Service 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, this 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.


   **Important:** 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.

7. Install the Test Controller application through the WebSphere Process Server administrative console.
   a. Select **Applications > Install New Application**.
   b. Click **Browse** and select the TestController70.ear application from the directory where you unzipped it.
   c. Accept all default settings during installation.
   d. Save your configuration after installation.

8. 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.
For details about configuring the Business Space to use a Lotus Webform Server, see Configuring Lotus Webform Server for Human Task Management widgets in the Business Space.

Related tasks

“Defining the roles required for human tasks” on page 12
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 a server configuration file” on page 13
To connect to a test server (or managed deployment environment), the business analyst needs a server configuration file that contains the required information about the server instances running in the managed deployment environment. The IT administrator creates a different server configuration file for each managed deployment environment that is set up for testing processes.

Configuring security for the managed deployment environment on WebSphere Process Server
To test business processes in 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 WebSphere Process Server component 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.

3. To delete deployed processes in WebSphere Business Modeler using the Process Execution Environment view, a user must either be the original owner who deployed the process or have super-user authority in the process testing Business Space. Assign the super-user role to any user ID that you want to be able to delete deployed processes.

Users will now have the required privileges to test business processes. 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.
Related tasks

Chapter 4, “Maintaining the managed deployment environment”, on page 35

From time to time, files will need to be removed from server components in the managed deployment environment.

**Configuring a managed deployment environment on WebSphere Process Server and WebSphere Business Monitor**

After the IT administrator sets up the managed deployment environment, it might require further configuration and maintenance. For example, the IT administrator can specify the deployment timeout properties for recovery from a system lockup.

The following files are part of the configuration:

- The managed-deployment-environment.properties file is used to configure the WebSphere Process Server component.
- The monitorMSSConfig.properties file is used to configure the WebSphere Business Monitor component.
- The role mapping file provides the mapping of logical roles in WebSphere Business Modeler and the physical user group defined in the target directory.
- The server configuration file lists the server components instances that make up the managed deployment environment. The business analyst uses the server configuration file to establish connectivity to the managed deployment environment.

To configure the managed deployment environment:

   a. To configure the queue management properties, set the values for the `token.queue.timeout.seconds` and `token.queue.capacity` properties in the managed-deployment-environment.properties file. The managed deployment environment supports the concurrent deployment of processes. WebSphere Process Server accepts all the deployment requests and will attempt to process them simultaneously. If it cannot handle a deployment request, WebSphere Process Server stores the request as a token on a queue. Once the token reaches the front of the queue, the corresponding deployment request is then processed. Example of the queue management properties:

   ```
   # The period of time a token remains valid after it has been refreshed. When this period runs out
   # and the token has not been refreshed by its client, the token will be removed from the queue.
   #
   token.queue.timeout.seconds=60

   # The number of tokens that the queue can store before it becomes full. Once the queue is full, it
   # will not accept new tokens to be enqueued until existing tokens have been dequeued or timed out.
   #
   token.queue.capacity=4000
   ```

   b. To configure the timeout properties, set the values for the `sleep.interval` and `intervals.to.timeout` properties in the managed-deployment-environment.properties file. You may need to tune the settings based on the environment and the number of concurrent users. Example of the timeout properties:

   ```
   # 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

2. If the business analyst gets messages about problems with deployment of elements to the monitor
server component, configure the appropriate values in the monitorMSSConfig.properties file to resolve
the problem. The value in this file override the default timeout values in the managed-deployment-
environment.properties file.

a. To activate the monitorMSSConfig.properties file, make a copy of the
monitorMSSConfig.propertiesTemplate file located in the <WAS_home>\scripts.wbm\d2d
directory, paste the copy into the same directory, and rename the copy to
monitorMSSConfig.properties.

b. Uncomment and set the desired properties. Example of a monitorMSSConfig.properties file:

```properties
# Application Installation timeout values
# ************************************************************
# In most cases the default values should not be changed.
# "appmanagement.retry.attempts" is the number of times the monitor server component will retry
# if an error occurs during the installation of a model application.
# The default value is 15.
# "appmanagement.retry.wait.seconds" is the number of seconds to wait before
# retrying the installation of the model application.
# The default value is 5 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.
# "model.deploy.timeout" is the number of seconds to wait before
# the monitor model deployment should time out.
# The default value is 600 seconds (10 minutes).
# To specify the information, uncomment these lines
appmanagement.retry.attempts=15
appmanagement.retry.wait.seconds=5
appmanagement.timeout.minutes=15
model.deploy.timeout=600

# ************************************************************
# 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 timeout. 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.
# To specify the information, uncomment these lines
sleep.interval=60
intervals.to.timeout=15

# ************************************************************
# Deployment targets
# ************************************************************
# The server or cluster where the Moderator and ModelLogic modules
# should be deployed.
# To specify the information, uncomment these lines
#moderator.deployment.target=WebSphere:cell=WBMonSrv_wps_Cell,
#node=WBMonSrv_wps_Node,server=server1
#modelllogic.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"

# "model.cei.config.wait.seconds" is the number of seconds to wait for the asynchronous CEI configuration task to finish. The task is started as part of the installation of the Monitor model, but the task usually finishes several seconds after the model installation has finished.
# The default value is 1 second. The maximum value is 30 seconds.

# CEI configuration information for the Monitor model
# To specify the information, uncomment these lines
#model.cei.location=remote
#model.cei.hostname=myhost.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=myhostCell,myhostNode01,server1
#model.cei.distribution.mode=activeBest
#model.cei.config.wait.seconds=1

# Monitor Data Security values
# "model.data.security.user.role" is the role to which the userid should be assigned. The default value is "KPI-Administrator".
# "model.data.security.user.resource.group" is the resource group to which the userid should be assigned.
# The default value is "root".
# To specify the information, uncomment these lines
#model.data.security.user.role=KPI-Administrator
#model.data.security.user.resource.group=root

# Monitor Stopwatch logging
# The Monitor stopwatch is used for debugging purposes.
# "monitor.mss.stopwatch.output" determines if output from the stopwatch is written to a log file or to the WebSphere Application Server SystemOut.log.
# Possible values are "none" for no output, "file" for output to <java.io.tmpdir>\monStopwatch.txt, or "both" for output to monStopwatch.txt and SystemOut.log.
# The default value is "none".

# To specify the information, uncomment this line
#monitor.mss.stopwatch.output=none

3. Restart WebSphere Process Server and WebSphere Business Monitor, as needed.

Related tasks
Chapter 4, “Maintaining the managed deployment environment”, on page 35

From time to time, files will need to be removed from server components in the 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, 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
   <?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-8da99005236fd4588235e97ead00b7cb7e" 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.

---

: Developing Deployable Business Processes Using WebSphere Business Modeler
a. In the process server administrative console, expand **Users and Groups**, select **Manage Groups**, and click **Search**. A table of defined groups is displayed.

b. Check the values for the uniqueName and groupName attributes for the roles in the role mapping file against the values in the table.

c. Ensure that the value for the uniqueName attribute in the role mapping file matches the full value in the "Unique Name" column.

d. Ensure that the value for the groupName attribute in the role mapping file matches the value for the \texttt{cn=""} argument in the "Unique name" column. If there is no \texttt{cn=""} value in the "Unique name" column, use the full value in the "Unique Name" column.

**Note:** If the operating system and product are running in different languages when the group is created, the characters for the name in the "Group name" column and in the \texttt{cn=""} value might not match. The directory look-up function for deployed processes uses the value in the \texttt{cn=""} argument.

The following example shows a completed entry for a role in the role mapping file:

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

Do not change the value for the \texttt{uid} attribute in the role mapping file.

e. Ensure that the \texttt{peopleDirectory} attribute matches the JNDI name for the people directory that you use for the managed deployment environment. The default value for the \texttt{peopleDirectory} attribute is \texttt{peopleDirectory="bpe/staff/samplevmmconfiguration"}, which targets the default people directory for the WebSphere Integration Developer unit test environment.

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. 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. This relative path is the value of the \texttt{memberMapping} attribute in the configuration file.

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

**Related information**

[Roles](#)

---

### Setting up a server configuration file

To connect to a test server (or managed deployment environment), the business analyst needs a server configuration file that contains the required information about the server instances running in the managed deployment environment. The IT administrator creates a different server configuration file for each managed deployment environment that is set up for testing processes.

The server configuration file is an XML file containing a set of parameters that define the name and location of the server component instances in the managed deployment environment. A server component instance is a logical name for an installation of WebSphere Process Server, Business Space powered by WebSphere, or WebSphere Business Monitor. The business analyst points to the server configuration file on their computer the first time that he or she tests a business process. The business analyst also requires this file to add a new test server in the Process Execution Environment view.

The server configuration file also specifies the location of a role mapping file, if one is required, and the security setting of the managed deployment environment.
Important: If you have a V6.2 server configuration file, you must change the Business Space server component definition in this file. See the code example in step 1 for the new Business Space server component definition.

To set up a server configuration file:

1. Create a server configuration file based on the following sample file. You can copy and paste the following sample code into a text or XML editor.

```xml
<rest:serverConfiguration xmlns:rest="http://rest.dtd.btools.ibm.com"
    name="Test server with monitoring"
    test="true"
    secured="true"
    memberMapping="memberMapping.properties">

    <description>
    Managed deployment environment with WebSphere Process Server and WebSphere Business Monitor
    </description>

    <serverComponent
        name="WebSphere Process Server"
        configuration="https://9.42.77.130:9443/rest/serverComponent/componentConfiguration">
    </serverComponent>

    <serverComponent
        name="WebSphere Business Monitor"
        configuration="https://9.42.77.130:9443/monitorServerComponent/componentConfiguration">
    </serverComponent>

    <serverComponent
        name="WebSphere Business Space"
        configuration="https://9.42.77.130:9443/mum/mycontenthandler?uri=service%3Acollection">
    </serverComponent>

</rest:serverConfiguration>

2. Supply the correct parameter values by modifying the sample code.

   a. name="Testing server with monitoring": Name the managed deployment environment. This name is displayed in the Process Execution Environment view in WebSphere Business Modeler. If you are providing multiple managed deployment environments, provide a meaningful name for each one.

   b. test="true": Set the test parameter to "true". This parameter indicates that the managed deployment environment is intended for testing purposes only and not for production.

   c. secured="true": Set the secured parameter to "true". This parameter indicates that security is enabled on the managed deployment environment. Users will be required to enter an ID and password before testing processes. If this parameter is set to "false", users will still be required to enter an ID. However, they will not have to enter a password.

   d. memberMapping="filenamem.rmf" If roles for human tasks must be mapped to groups in the people directory, add a valid URL for the role mapping file, which can be an absolute or relative path to the file. For more information about creating role mapping files, see “Defining the roles required for human tasks” on page 12.

   e. <serverComponent>...</serverComponent>: For each server component required for the managed deployment environment, provide the name of and configuration (endpoint URL) for the server component instance. The configuration parameter must include the port number. For WebSphere Process Server, the port number is the port that the WebSphere Process Server web container listens to.

   Tip: To find the port number for WebSphere Process Server, open the administrative console and select Servers → Application servers → servername → Port. If security is disabled, use the wc_defaulthost port (for example, 9080). If security is enabled, use the wc_defaulthost_secure port.
3. 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.

Related tasks

“Setting up a managed deployment environment on WebSphere Integration Developer” on page 4
If you want to use the WebSphere test environment as a managed deployment environment, set WebSphere Integration Developer up on a dedicated computer. Your WebSphere test environment can be enabled for testing the deployment of business processes from WebSphere Business Modeler.

“Setting up a managed deployment environment on WebSphere Process Server” on page 6
If you want to set up a managed deployment environment on WebSphere Process Server, install WebSphere Process Server on a stand-alone server.

“Deploying business processes on a test server” on page 27
When you verify the design of your business process, WebSphere Business Modeler deploys the process to the test server for you. If the test server is processing other deployment or deletion requests, your request is put in a queue.
Chapter 2. Modeling processes for deployment

Modeling processes that you plan to deploy is much like other business modeling 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.

The following guidelines can make developing a deployable process more manageable:

- Develop your process in stages, starting with the simple, high-level 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 27](#)

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.

**Related reference**

- [“Limitations and restrictions for deploying processes from WebSphere Business Modeler” on page 59](#)

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

**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 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:
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 them in WebSphere Business Modeler or import them from Lotus Forms Designer. You can modify the layout of forms only in Lotus Forms Designer.
- 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 Verify Process Design 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 Verify Process Design action is only enabled for processes without errors, so any errors in your process must be fixed before you can test your process.

Remember: By default, some messages are filtered from the Errors view until the first Verify Process Design action is invoked. You can choose to have these error messages displayed by default by clicking Window > Preference > Business Modeling > Modes and selecting the option to show errors that prevent deployment.

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

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

Related tasks

"Testing human task assignments" on page 31

As part of verifying your process flow, you can test the human task assignments in your process to ensure that people with the appropriate roles in your organization will be able to complete the tasks assigned to them.

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 that you can make your application easier to use is to create custom forms for your human tasks. If you already have forms created using Lotus Forms Designer, you can import these forms into your workspace and associate them with a human task. You can also customize forms that you create in WebSphere Business Modeler and then update these forms in your process model.

**Important:** When you associate a form with a human task, if the inputs or outputs of the human task do not match the form data, then the inputs and outputs of the human task will be replaced with the form data.

When you test a business process, WebSphere Business Modeler generates default forms for human tasks that do not already have forms associated with them and for any process input or output. The data fields for these generated forms are based on the inputs and outputs of the human tasks or process. However, these data fields are simply laid out in a top-to-bottom fashion according to the order of data elements in the inputs and outputs.

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 human 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:

<table>
<thead>
<tr>
<th>EmployeeRecord</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EmployeeID</td>
<td></td>
</tr>
<tr>
<td>FirstName</td>
<td></td>
</tr>
<tr>
<td>Initial</td>
<td></td>
</tr>
<tr>
<td>LastName</td>
<td></td>
</tr>
<tr>
<td>Street</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td></td>
</tr>
<tr>
<td>StateOrProvince</td>
<td></td>
</tr>
<tr>
<td>Country</td>
<td></td>
</tr>
<tr>
<td>ZipOrPostalCode</td>
<td></td>
</tr>
<tr>
<td>HomePhone</td>
<td></td>
</tr>
<tr>
<td>StartDate</td>
<td></td>
</tr>
<tr>
<td>Department</td>
<td></td>
</tr>
<tr>
<td>PWA</td>
<td></td>
</tr>
<tr>
<td>OfficePhone</td>
<td></td>
</tr>
<tr>
<td>OfficeIP</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td></td>
</tr>
<tr>
<td>ComputerDNSName</td>
<td></td>
</tr>
</tbody>
</table>

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:
Tip: After a form is associated with a human task or process, you or your form designer can move fields to improve the form layout, add headings and graphics, format fonts, and make other visual enhancements to the form. However, it is recommended that you do not add, delete, or re-create form data in the form editor (Lotus Forms Designer).
You can associate forms with human task and 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
- Adding forms for human tasks
- Mapping values
- Creating business rules that are configurable in the 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 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 tasks and global tasks in deployable processes are ‘none’ or “Import - Web Service binding”.
   b. Export the new service for your IT developer to implement.

Chapter 2. Modeling processes for deployment 23
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 as part of your process.

**Note:** You can still include services in your process even if you are not using a WebSphere Service Registry and Repository server. If the WSDL you imported into WebSphere Business Modeler is a single WSDL that includes the PortType as well as the endpoint information, a WebServices Import will be generated that is bound to the mediation component as the default port. This means that if no WebSphere Service Registry and Repository server is configured the WebServices Import will be triggered, using the endpoint that is defined in the original WSDL object.

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 add classifications to services:

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. The classification that you assign to a service or business service in WebSphere Business Modeler must correspond to the classification assigned to the port of the service in WebSphere Service Registry and Repository.

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

**Related tasks**

["Testing process flow" on page 29](#)

After you have deployed one or more processes to a test server, you can test the process flow for them on the Processes tab of the process verification environment. For example, you can check that your activities execute as expected and that the correct data is passed from node to node as your process runs.

**Related information**

- Services
- Business services and business service objects

---

**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 **Verify Process Design** 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 tasks**

- [Testing process flow](#) on page 29

**Related information**

- [Creating human tasks](#)

### Modeling business measures for deployment

To monitor the business performance of the application after it is deployed to a production environment, you can include business measures in your processes. You can test these business measures 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 **Verify Process Design** 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:** The 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 **Verify Process Design** 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
  You can check that the business measures in your process are collecting the data you expect on the **Business Measures** tab of the process verification environment. The default widgets on the **Business Measures** tab are automatically updated with data from each process instance execution on the **Processes** tab.

- “Setting up a server configuration file” on page 13
  To connect to a test server (or managed deployment environment), the business analyst needs a server configuration file that contains the required information about the server instances running in the managed deployment environment. The IT administrator creates a different server configuration file for each managed deployment environment that is set up for testing processes.

**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.

To connect to a test server, you must have the configuration file for it in your file system. Ask the IT administrator who set up your test server to provide you with this file.

The testing function is enabled only when your business process is free of 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.

**Tip:** By default, some messages are filtered from the Errors view until the first Verify Process Design action is invoked. You can choose to have these error messages displayed by default by clicking **Modeling → Preferences → Business Modeling → Modes** and selecting **Show validation errors that prevent deployment**.

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 verify the design of your business process, WebSphere Business Modeler deploys the process to the test server for you. If the test server is processing other deployment or deletion requests, your request is put in a queue.

After you finish preparing for modeling and testing deployable business processes, check that you are ready to start testing your business process:

- You must be in the WebSphere Process Server mode.
- In your file system, you must have the test server configuration file and role mapping file that your IT administrator set up for you. The first time that you test your business process, you are prompted to add a server by pointing to the server configuration file. You might need the account ID and password for the test server to authenticate with it.
- Your process must be free of all errors that prevent deployment. By default, some error messages are filtered from the Errors view until the first **Verify Process Design** action is invoked. To display these error messages by default, click **Window → Preference → Business Modeling → Modes** and select the option to show errors that prevent deployment.
- For consistent testing results, set your browser preferences as part of setting up your modeling environment. The process testing environment runs on the Business Space component of the test server, which opens in an external Web browser.

If business measures are defined for the business process, a monitor model is generated automatically and deployed to the server at the same time as the process model is deployed.

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

Although only one process deployment or deletion can occur at a time on the test server, each deployment or deletion request to the test server is queued. If the test server is busy, you receive a
message indicating that another request is being processed and telling you what position the current request has in the queue. Each request is processed when it reaches the top of the queue. You do not have to reinvoke the **Verify Process Design** action unless you chose to cancel a deployment operation.

To deploy a business process on a test server:

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

   **Tips:**
   - If you have 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 Process Execution Environment view and select **Properties**. If you continue to have connection problems, contact the IT administrator for the test server.
   - If you want to continue working in WebSphere Business Modeler while your process is being deployed to the server, click **Run in Background**. You can still monitor the status of the process deployment in the Process Execution Environment view.
   - If you have problems with the deployment of a business process to a test server, click **Request Help from IT** in the error message window. Send the problem determination archive file generated through this action to an IT developer who can open the file in WebSphere Integration Developer 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 Process Execution Environment 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 Process Execution Environment view.
Related tasks

“Deleting deployed processes using WebSphere Business Modeler” on page 33
After you have deployed multiple processes or multiple versions of the same process, you might want to delete these processes from the test server.

Chapter 4, “Maintaining the managed deployment environment”, on page 35
From time to time, files will need to be removed from server components in the managed deployment environment.

Related reference

“Limitations and restrictions for deploying processes from WebSphere Business Modeler” on page 59
This topic describes the limitations and restrictions for developing deployable business processes using WebSphere Business Modeler.

Adding and changing test servers for deploying processes

Use the Process Execution Environment 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 a new server for testing your processes, open the Process Execution Environment 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 of its icon. 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 Process Execution Environment view, right-click it, and select Make Default Server.

• To test the connection to a server, right-click the server name in the Process Execution Environment view, and select Test Connection.

• To remove a test server, open the Process Execution Environment view, right-click the Test Servers node, and select Remove Server.

Note: When you remove a test server from the Process Execution Environment view, you are given the option to delete processes deployed by the current account ID at the same time. Any processes left on a test server must be removed manually by an IT administrator.

Related tasks

“Deleting deployed processes using WebSphere Business Modeler” on page 33
After you have deployed multiple processes or multiple versions of the same process, you might want to delete these processes from the test server.

Testing process flow

After you have deployed one or more processes to a test server, you can test the process flow for them on the Processes tab of the process verification environment. For example, you can check that your activities execute as expected and that the correct data is passed from node to node as your process runs.

• If your process contains empty subprocesses or services for which you do not yet have a service lookup or implemented business service, you must use a placeholder human task to stand in for the missing subprocess or service. An undefined activity, such as a service task with no implementation, cannot execute in the process verification environment. So your process execution will stop at this activity.

• If you want to test modifying the parameter values in business rules, the if-then rules must be based on rule templates that include these parameters.

Note: If you have not customized the rule presentations for your business rules, they might be difficult to understand in the process verification environment. Remember that the users of your business
process will see the business rule presentations when you deploy your process to production. For information about customizing rule presentations, see “Refining the interfaces for your deployable process” on page 19.

You can test only one process and its dependent subprocesses at a time. The data from your testing sessions remains on the test server until you delete the process in the Process Execution Environment view, redeploy the process to the test server, or deploy another process to the test server. After the process is deployed to the test server, you can continue testing it after you close WebSphere Business Modeler. However, if you close WebSphere Business Modeler, the Request Assistance from IT function becomes unavailable in the process verification environment.

To test your process flow:
1. In the Start Process Instance widget, select the process that you want to start testing and click the Create icon or action. In the Business category column, you can see which elements are processes.

   Note: Tip: You do not need to start receive tasks in a process. Receive tasks start automatically when you reach them in the process flow.

2. To start the process instance, enter input data for it in the Enter Data into Forms widget, press Enter, and click Submit. If you have associated a form with your process input, the form fields are displayed in the Enter Data into Forms widget. If you have not associated a form with the process input, an input form is created for you by default.

3. View the execution path of your process instance highlighted in the Trace Process Execution widget. As each activity is completed, the name of and output data from the activity are displayed under the process diagram. For complex data, click the + (plus sign) to drill down to the data values that make up the output.

4. When the process execution arrives at a human task, accept it in the Claim Available Tasks widget. Also complete the associated input form in the Enter Data into Forms widget and click Submit. The process execution resumes, moving to the next activity in the process flow.

   Tip: If a waiting human task is not displayed in the Claim Available Tasks widget, select the Refresh action from the Claim Available Tasks widget options menu.

5. If your process contains a subprocess that is marked as a BPEL collaboration scope (in the Technical Attributes view), test claiming, skipping, or redoing work on the human tasks in the subprocess using the Step Through Human Workflow widget.

   • When the process execution arrives at a human task in the subprocess, select it in the Claim Available Tasks widget. A diagram containing the human tasks in the process is displayed in the Step Through Human Workflow widget, and the human task that you selected has a forward arrow on it.

   • To claim a human task, accept it in the Claim Available Tasks widget, complete the associated forms in the Enter Data into Forms widget, and click Submit. If you claim a human task, you can redo it later.

   • To skip a human task, click the human task image in the Step Through Human Workflow widget and then click Skip. If you skip a human task, you can redo it later.

   • To redo a human task that you already claimed, click the human task image in the Step Through Human Workflow widget and then click Redo.

   Tip: If the display in the Step Through Human Workflow widget gets out of synch with the sequence of actions that you selected, select the Refresh action in the widget options menu.

6. When the process execution arrives at a receive task, complete the activity:
   a. In the Claim Available Tasks widget, select the Check status of tasks view.

      Note: To test the next human task, you will have to switch back to the Assess and work on tasks view.
b. Click the Edit icon or action for the receive task.

c. In the Enter Data into Forms widget, supply a value for each correlation property.

7. As required, test the modification of the parameter values in your business rules:

   a. In the Change Parameter Values for Business Rules widget, select a business rule for the business rules task. The if-then rules for each business rule are listed on the left under the business rules task. The rule presentation for each if-then rule is displayed on the right, and the configurable parameter values are highlighted in blue.

   Tip: To make it easier to work with the Change Parameter Values for Business Rules widget, you can click the maximize icon to expand it. When you click the minimize icon, the widget returns to its previous size.

   b. Select an if-then rule that has highlighted parameter values, click a parameter value, and supply a new value.

   c. To accept the parameter values, click Save. If you decide to reset the parameter values to their defaults, click Reset. When you start a new instance of this process, you can check that the modified parameter values are displayed in the Change Parameter Values for Business Rules widget during process execution.

8. If you encounter problems with running your process, click Request Assistance from IT on the Trace Process Execution widget. WebSphere Business Modeler packages the log files from your testing session for you to hand off to your IT developer for problem determination. If you have closed WebSphere Business Modeler, this function becomes unavailable.

What to do next

After the process execution is complete, you can return to the Start Process Instance widget and start another process instance for further testing:

- Check or modify the process inputs.
- Rerun the process with different parameter values for configurable business rules, and analyze the effect that these values have on the resulting process output or on the execution path for the process.
- Verify that your human task assignments work as expected by logging in as a different user. See "Testing human task assignments".
- If your process contains business measures, verify that your business measures are working correctly on the Business Measures tab. See "Testing business measures" on page 32.

When you are finished verifying all aspects of your process, either close or log out of the browser window and return to WebSphere Business Modeler to continue modeling. To test any modifications that you make to your process, relaunch the process verification environment by right-clicking the process in the Project Tree view and selecting Verify Process Design.

Related concepts

Chapter 2, “Modeling processes for deployment”, on page 17

Modeling processes that you plan to deploy is much like other business modeling 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.

Testing human task assignments

As part of verifying your process flow, you can test the human task assignments in your process to ensure that people with the appropriate roles in your organization will be able to complete the tasks assigned to them.

Before you can test human task assignments:

- Security on your test server must be enabled.
You must have the account IDs and passwords that your IT administrator has set up for testing role assignments.
You must have the role mapping file on your computer and be using the corresponding roles project to for human task assignments in your process.
You must have the process instance for which you want to test human task assignments running on the Processes tab of the process verification environment. See "Testing process flow" on page 29.

Note: If you have included scheduling information in your human task, the human task will not become available for testing until the scheduled time period.

To test the human task assignments in your process:
1. When the process execution pauses at a human task that you want to claim on the Processes tab of the process verification environment, click Log in as Different User on the Trace Process Execution widget.
2. Log in with the account ID and password for the role that is appropriate for completing the human task. For example, you could log in with an ID associated with an HR role so that you can claim and complete a task for setting up an employee record. A Task Completion Space window opens in a new Web browser window or tab.
3. In this new browser window, accept the human task in the Claim Available Tasks widget. The associated input form opens in the Enter Data into Forms widget.
4. Complete the form and click Submit. The process execution resumes on the Processes page, and the submitted data is passed on to the next activity in the process.

Leave Task Completion Space window open until you complete all of the human tasks assigned to the role that you logged in as.

Related tasks
"Modeling human tasks for deployment" on page 18
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.
"Defining the roles required for human tasks" on page 12
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.

Testing business measures
You can check that the business measures in your process are collecting the data you expect on the Business Measures tab of the process verification environment. The default widgets on the Business Measures tab are automatically updated with data from each process instance execution on the Processes tab.

Before you test business measures:
• The test server that you use for testing business measures must be set up with a monitoring server component.
• You must have the process instance for which you want to test human task assignments running on the Processes tab of the process verification environment. See "Testing process flow" on page 29.
• Any business measures flagged with warnings are not included in the monitor model that is deployed to the process verification environment. If you want to test all of your defined business measures, make sure that you clean up any warnings before invoking the Verify Process Design action.
Note: Because the deployment of the monitor model takes significantly longer than the deployment of process model, you might want to defer testing your business measures. You can defer this testing in one of two ways:

- If your process does not yet have business measures defined for it, wait to add them until you have successfully tested the process flow, human tasks, services, receive tasks, and business rules in your process.
- If your process already has business measures defined for it, either run your tests on a managed deployment environment that does not have a monitoring server component or ask your IT administrator to create a server configuration file that comments out this component.

When you open the Business Measures tab, the business measures data is displayed in dashboards in the appropriate monitoring widgets. For example, if your process model includes KPIs with ranges, the monitoring data for each KPI is displayed in a half-gauge dashboard in the KPIs with Ranges widget.

<table>
<thead>
<tr>
<th>Monitoring widget</th>
<th>Data displayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>KPI widgets</td>
<td>Data accumulated across a testing session for a particular process deployment</td>
</tr>
<tr>
<td>Instance Metrics widget</td>
<td>Information from specific, individual process instance runs</td>
</tr>
<tr>
<td>Dimensions widget</td>
<td>Number of instances run</td>
</tr>
</tbody>
</table>

To test your business measures:

1. To test metrics for an element instance, go to the Business Measures tab when that element is executing. To test KPIs, wait until the process has finished executing before opening the Business Measures tab.

   **Tip:** If the dashboards that you expect to see are not displayed, try refreshing the browser window.

2. To verify that you have set up the business measures correctly for your business process, check the values displayed in the dashboards. You might have to resize a widget to see some dashboards. You can also modify widget display attributes, such as the dashboard format, widget name, or colors.

3. To generate additional monitoring data, return to the Processes tab and run another instance of your process.

If the monitoring widgets are not used for some time, you might receive a message that your session has expired. To continue working, simply refresh your browser window.

**Related tasks**

- "Deploying business processes on a test server" on page 27
- "Modeling business measures for deployment" on page 25

**Related information**

- [Dashboards](#)

**Deleting deployed processes using WebSphere Business Modeler**

After you have deployed multiple processes or multiple versions of the same process, you might want to delete these processes from the test server.
Before you can delete processes deployed on a test server, your IT administrator must set up your account ID with the required authority.

The deletion operation removes from the test server all processes owned by an account ID.

Although only one process deletion or deployment can occur at a time on the test server, each deletion or deployment request to the test server is queued. If the test server is busy, you receive a message indicating that another request is being processed and telling you what position the current request has in the queue. Each request is processed when it reaches the top of the queue. You do not have to reinvoke the **Delete Processes from Server** action unless you chose to cancel a deletion operation.

To delete the deployed processes that belong to one or more account IDs:

1. In the Process Execution Environment view, right-click the test server from which you want to delete processes and select **Delete Processes from Server**. If you are using a deployer account ID, you can delete only those processes that belong to this account ID.
2. If you are using an administrator account ID, select the account IDs for which you want to delete processes. You can delete all the processes that belong to a particular account ID, to selected account IDs, or to all account IDs with processes deployed on the test server.
3. If an error marker is displayed on a process after the deletion operation completes, contact your IT administrator to have these processes removed from the test server. The error marker indicates that a problem occurred during the deletion operation.

If you cancel a deletion operation while it is in progress, the deletion of the current element completes and the deletion of any elements remaining on the test server is cancelled. The cancel operation does not roll back any deletions that have started or completed.

**Tip:** If deletions do not show up immediately in the Process Execution Environment view, click the **Refresh** icon on the view toolbar to refresh the display of the Process Execution Environment view.

**Related tasks**

- "Deploying business processes on a test server" on page 27
  When you verify the design of your business process, WebSphere Business Modeler deploys the process to the test server for you. If the test server is processing other deployment or deletion requests, your request is put in a queue.
- Chapter 4, “Maintaining the managed deployment environment”, on page 35
  From time to time, files will need to be removed from server components in the managed deployment environment.
Chapter 4. Maintaining the managed deployment environment

From time to time, files will need to be removed from server components in the managed deployment environment.

Related tasks
- "Setting up a managed deployment environment on WebSphere Integration Developer" on page 4
- "Setting up a managed deployment environment on WebSphere Process Server" on page 6
- "Deleting deployed processes using WebSphere Business Modeler" on page 33
- Chapter 5, “Troubleshooting business process deployment”, on page 39

Restoring the default layout of a process verification environment

If a business analyst wants to return a process verification environment to its default layout, the business analyst or IT administrator must delete the two testing spaces that belong to the relevant user account in the Business Space.

If you use WebSphere Business Modeler to delete the testing spaces, you must be in WebSphere Process Server mode.

As part of the process deployment operation, WebSphere Business Modeler checks whether testing spaces exist for a user account. If none are found, it deploys two preconfigured testing spaces along with the business process application and any monitor models. Removing these two testing spaces in the Business Space allows the preconfigured testing spaces to be redeployed, which restores the default layout of the process verification environment.

You can delete the testing spaces in one of the following ways:

- In WebSphere Business Modeler, open the Process Execution Environment view and delete the processes that belong to the user account. WebSphere Business Modeler deletes both of the testing spaces along with the processes and any monitor models deployed with that user account.
- Delete one or both of the two testing spaces in the Business Space:
  1. Open the Business Space. One way to open the Business Space login page is to select the last testing session in your browser history.
  2. In the menu bar, click Manage Spaces.
  3. Find the Process Verification Environment space owned by the appropriate account ID, click Actions, and select Delete. You must confirm that you want to delete this business space.
  4. Find the Task Completion Space space owned by the appropriate account ID, click Actions, and select Delete. You must confirm that you want to delete this business space.
  5. Click Done.
The next time that the business analyst invokes the **Verify Process Design** action, the Business Space will open in the default Process Verification Environment space.

**Related tasks**

“Deploying business processes on a test server” on page 27

When you verify the design of your business process, WebSphere Business Modeler deploys the process to the test server for you. If the test server is processing other deployment or deletion requests, your request is put in a queue.

### Uninstalling deployed resources using the administrative console

Business analysts should delete deployed processes from the Process Execution Environment view in WebSphere Business Modeler. However, if this deletion process fails, the IT administrator might need to remove applications, monitor models, or both from the server components of the managed deployment environment.

The account ID used to delete resources in the administrative console must be associated with the Administrator or Configurator role. Otherwise, the service integration bus (SIB) queues will not be removed properly from the server. To change the role of an account ID, go to the administrative console and click **Users and Groups → Administrative User Roles**.

To manually uninstall deployed resources from the managed deployment environment:

1. Open the administrative console.
2. To uninstall WebSphere Business Modeler applications, click **Applications → SCA modules** in the console navigation tree.
   a. Select modules that you want to remove.
   b. Click **Uninstall**.
3. To uninstall a monitor model, click **Applications → Application Types → WebSphere 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_DeploymentIDApp`
   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.
   c. Click **Save** to save your changes.

The modules and associated applications are removed from the managed deployment environment.

**Related tasks**

“Deleting deployed processes using WebSphere Business Modeler” on page 33

After you have deployed multiple processes or multiple versions of the same process, you might want to delete these processes from the test server.

### Maintaining the WebSphere Business Monitor server in a managed deployment environment

If your managed deployment environment includes a WebSphere Business Monitor server component, your server file system might over time accumulate a significant number of unneeded files and folders that you can safely delete from the file system to free up disk space.

You must use your WebSphere Business Monitor server from a WebSphere Business Modeler version 7.0 client.
You can safely delete the files and folders within the mde.wbm directory, but do not delete the mde.wbm directory itself.

If you inadvertently delete the mde.wbm directory, it is automatically regenerated by the managed deployment environment. You can find these files using the filepath, <was_profile_home>\mde.wbm.

To help with problem determination, consider keeping the most recent files and folders and deleting only files that are older than one week.
Chapter 5. 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:

Related tasks

"Deploying business processes on a test server" on page 27
When you verify the design of your business process, WebSphere Business Modeler deploys the process to the test server for you. If the test server is processing other deployment or deletion requests, your request is put in a queue.

"Testing process flow" on page 29
After you have deployed one or more processes to a test server, you can test the process flow for them on the Processes tab of the process verification environment. For example, you can check that your activities execute as expected and that the correct data is passed from node to node as your process runs.

Related reference

"Limitations and restrictions for deploying processes from WebSphere Business Modeler” on page 59
This topic describes the limitations and restrictions for developing deployable business processes using WebSphere Business Modeler.

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:

     ![Business Process Editor]

     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:

     ![Process Model Viewer]
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. Switch to the test client, which resembles the following figure:
g. At the top of the Events area in the test client, click the **Invoke** icon 🔄. A new **Invoke** event is displayed at the bottom of the Events area.

h. Ensure that the new **Invoke** event is selected, then specify some values for the parameters in the value editor located in the lower right corner of the test client. For example, in the above figure, the values **10000** and **Widgets** are specified for the parameters **Amount** and **Commodity**.

i. At the top of the Events area, click the **Continue** icon 🔄. The Deployment Location wizard opens, as shown in the following figure:
j. In the **Deployment Location** list, ensure that the server that is listed is the same server for which you have enabled cross-component tracing.

k. Click **Finish**. The module is deployed to the server and the test client reruns the test trace and begins to return the results.

l. 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."

---

**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:

![Business Process Models](image)

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

**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.

**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.
Troubleshooting deployment failures to Windows 2003 and Windows XP servers

If the managed deployment environment is running on a computer with a Windows 2003 or Windows XP operating system and business analysts experience failures when they deploy processes to it from WebSphere Business Modeler, check the server SystemOut.log file for an OutOfMemory error.

To determine whether an OutOfMemory error is causing the problem, check the SystemOut.log file:

- WebSphere Integration Developer: <install dir>/pf/wps/logs/server1/SystemOut.log
- WebSphere Process Server: <install dir>/profiles/<profileName>/logs/server1/SystemOut.log

To troubleshoot a server OutOfMemory error:
1. On the server, right-click My Computer and select Properties.
2. In the System Properties window, click the Advanced tab.
3. In the Startup and Recovery section, click Settings. The Startup and Recovery window is displayed.
5. In the [Operating Systems] section, add the following switch to the end of the startup line that includes the /fastdetect switch: /3GB
6. Save the changes, and close Notepad.
7. To finish accepting the changes, click OK for each open window.
8. Restart the computer.

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 in the WebSphere Business Monitor or WebSphere Process Server components of the managed deployment environment. 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 file.
2. Extract the file contents, and open the workspace-logs directory.
3. Open the Eclipse workspace .log file in a text editor.
4. Use the error codes in the following tables to help you 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 WebSphere Business Monitor or WebSphere Process Server components of the managed deployment environment. If you encounter one of these errors, you might 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 might need to obtain assistance from IBM Software Support.

Table 1. WebSphere Business Monitor

<table>
<thead>
<tr>
<th>Error</th>
<th>Error Code</th>
<th>Translated Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent deployment (deploy, undeploy, or cleanup)</td>
<td>E0001</td>
<td>Cannot perform a deploy or undeploy on this resource because there is an operation still in progress.¹</td>
</tr>
</tbody>
</table>
Table 1. WebSphere Business Monitor (continued)

<table>
<thead>
<tr>
<th>Error</th>
<th>Error Code</th>
<th>Translated Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>No content (query API)</td>
<td>E0002</td>
<td>There is no content found for this resource.</td>
</tr>
<tr>
<td>Resource not found (all APIs that contain the</td>
<td>E0003</td>
<td>The resource with id &quot;{0}&quot; was not found.</td>
</tr>
<tr>
<td>resource ID)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Illegal source state (deploy, undeploy, or cleanup)</td>
<td>E0004</td>
<td></td>
</tr>
<tr>
<td>Model not found (deploy)</td>
<td>E0005</td>
<td>A valid model was not found in the archive.</td>
</tr>
<tr>
<td>Model differs from last (deploy)</td>
<td>E0006</td>
<td>The model sent in the update archive differs from the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>existing one.</td>
</tr>
<tr>
<td>Multiple models in PI (deploy)</td>
<td>E0007</td>
<td>More than one Monitor model file was found.</td>
</tr>
<tr>
<td>Build failure (deploy)</td>
<td>E0008</td>
<td>Build failure. (mmdeploy could not build the Monitor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EAR file)</td>
</tr>
<tr>
<td>Application already installed (deploy)</td>
<td>E0009</td>
<td>The application &quot;[0]&quot; already exists.</td>
</tr>
<tr>
<td>Application does not exist (deploy or undeploy)</td>
<td>E0010</td>
<td>The application &quot;[0]&quot; does not exist.</td>
</tr>
<tr>
<td>Model name too long (deploy)</td>
<td>E0011</td>
<td>The model name is too long. It must be less than 61</td>
</tr>
<tr>
<td></td>
<td></td>
<td>characters.</td>
</tr>
<tr>
<td>Internal server error</td>
<td>E1000</td>
<td>Internal server error.</td>
</tr>
<tr>
<td>Database unavailable</td>
<td>E1001</td>
<td>For a database on a remote database server, either the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>server is not available or the database has been stopped.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The database must be restarted.</td>
</tr>
<tr>
<td>Timeout</td>
<td>E2000</td>
<td>Operation timed out.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Error</th>
<th>Error Code</th>
<th>Translated Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concurrent deployment (deploy, undeploy, or cleanup)</td>
<td>E0001</td>
<td>Cannot perform a deploy or undeploy on this resource because</td>
</tr>
<tr>
<td></td>
<td></td>
<td>there is an operation still in progress.²</td>
</tr>
<tr>
<td>No content (query API)</td>
<td>E0002</td>
<td>There is no content found for this resource.</td>
</tr>
<tr>
<td>Resource not found (all APIs that contain</td>
<td>E0003</td>
<td>The resource was not found.</td>
</tr>
<tr>
<td>resource id)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Illegal source state (deploy, undeploy, or cleanup)</td>
<td>E0004</td>
<td></td>
</tr>
<tr>
<td>Model not found (deploy)</td>
<td>E0005</td>
<td>A valid model was not found in the archive.</td>
</tr>
<tr>
<td>Model differs from last (deploy)</td>
<td>E0006</td>
<td>The model sent in the update archive differs from the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>existing one.</td>
</tr>
<tr>
<td>Multiple modules in PI (deploy)</td>
<td>E0007</td>
<td>More than one SCA module was found.</td>
</tr>
<tr>
<td>Build failure (deploy)</td>
<td>E0008</td>
<td>Build failure.</td>
</tr>
<tr>
<td>Application already installed (deploy)</td>
<td>E0009</td>
<td>The application already exists.</td>
</tr>
<tr>
<td>Application does not exist (deploy or undeploy)</td>
<td>E0010</td>
<td>The application does not exist.</td>
</tr>
<tr>
<td>Application failed to install</td>
<td>E0017</td>
<td>An error in the process model prevents it from being</td>
</tr>
<tr>
<td></td>
<td></td>
<td>installed. Request assistance from IT to identify the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>problem in the process model.</td>
</tr>
</tbody>
</table>

²This 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 (continued)

<table>
<thead>
<tr>
<th>Error</th>
<th>Error Code</th>
<th>Translated Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal server error</td>
<td>E1000</td>
<td>Internal server error.</td>
</tr>
<tr>
<td>Timeout</td>
<td>E2000</td>
<td>Operation timed out.</td>
</tr>
</tbody>
</table>

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.
Chapter 6. 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.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Versioning</td>
<td>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.</td>
</tr>
<tr>
<td>Late or early binding for process-to-process interaction</td>
<td>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.</td>
</tr>
<tr>
<td>Human task binding (role mapping)</td>
<td>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).</td>
</tr>
</tbody>
</table>
| Business service classification    | If the business service classifications for WebSphere Service 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:  
  • 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. |
<p>| Module structure                  | 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>
<table>
<thead>
<tr>
<th>Configuration</th>
<th>Specification</th>
</tr>
</thead>
</table>
| Monitor model          | 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:  
  • 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  
The IT developer can use the compare and merge function in WebSphere Integration Developer to identify such changes.                                                                                                                                                                                                                                                                   |
| Optimized business objects | Your IT developer might need to change the optimization setting for business objects (business items and business service objects). If your IT developer asks you to enable optimized business objects before you export your process, you can access this setting by clicking Window → Preferences → Modes and selecting the WebSphere Process Server mode.                                      |

Related tasks

Chapter 3, “Testing business process deployment”, on page 27

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.

Related information

[Exporting files to WebSphere Integration Developer](#)
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 deployment to the managed deployment environment. 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.

<table>
<thead>
<tr>
<th>Unsupported construct</th>
<th>Example</th>
<th>Recommended replacement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exclusive gateways with downstream parallel gateways.</td>
<td>An upstream exclusive decision flows into a downstream join.</td>
<td>Pair an exclusive decision with a downstream merge instead of a join.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Inclusive gateways with a downstream exclusive gateway.</td>
<td>A decision with multiple branches that can execute simultaneously flows into a downstream merge.</td>
<td>Pair an inclusive decision or fork with a downstream join instead of a merge.</td>
</tr>
<tr>
<td>Parallel gateways with a backward connection.</td>
<td>A merge following by a fork with a backward connection from one of the parallel paths of the fork to the merge.</td>
<td>Use only sequential paths inside a process fragment with a backward connection.</td>
</tr>
<tr>
<td>Inclusive decisions with a backward connection.</td>
<td>A decision with multiple branches, one of which loops back to flow into the same inclusive decision.</td>
<td>Use only sequential paths inside a process fragment with a backward connection.</td>
</tr>
<tr>
<td>Global process invocations that loop back to their own input.</td>
<td>A global process invocation that produces an output that links back with a backward connection to the global process input.</td>
<td>Use local subprocesses instead of global processes to loop backward from the subprocess output to the subprocess input.</td>
</tr>
<tr>
<td>A parallel gateway that has both branches connecting to a single activity.</td>
<td>A fork which has two branches connecting to the input of the one task.</td>
<td>Use an intermediate element such as a map or task between the fork and the target activity.</td>
</tr>
</tbody>
</table>

For information about resolving errors related to the above restrictions, see [Modeling branching behavior](#).

Input and output restrictions for deployable processes

Only processes with a single input set (input criterion) and single output set are deployable. Similarly, processes that contain global subprocesses with multiple input criteria or output criteria cannot be deployed to the managed deployment environment. Processes defined as one-way operations (on the Request tab of the Technical Attributes view) are also not deployable.
If the input and output of a process have the same data type or name, the same form must be used for both the input and the output form. This is a result of a restriction in the underlying architecture, which requires that if the associated process inputs and outputs are the same then you must use a single form as both the input form and the output form.

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

**Human tasks in deployable applications**

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 task must have an assigned role. The primary owner cannot be based on an individual resource definition.

The only people assignment criteria supported for deployed processes 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 “e-mail”, the notify setting cannot be “none” or “Members by group ID”. In order for a notification e-mail 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 e-mail notification at runtime.

**Using services in deployable 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.

**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.

**Business rules in deployable applications**

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. An E0008 error for the WebSphere Business Monitor server component will be logged in the Eclipse workspace log. If you delete the instance metric, the process will deploy successfully.

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.

**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.

**Client and server clocks must be synchronized**

If the clock on the computer where WebSphere Business Modeler is installed is not synchronized with the clock on the server where the process is being tested, you might receive an error when trying to start the process instance in the process testing environment. If you receive such an error, contact your IT administrator to have them synchronize all of the clocks on the computers involved in the managed deployment environment with clock on the computer where WebSphere Business Modeler is installed.
Related tasks

“Modeling human tasks for deployment” on page 18
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.

“Adding services to deployable applications” on page 23
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.

“Modeling business measures for deployment” on page 25
To monitor the business performance of the application after it is deployed to a production environment, you can include business measures in your processes. You can test these business measures in the same process testing environment where you test your process.

Related information

Creating business rules tasks
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:
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 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 7.0
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:

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Company or Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phone No.</th>
<th>E-mail address</th>
</tr>
</thead>
</table>
Readers’ Comments — We’d Like to Hear from You

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