WebSphere WebSphere Business Modeler FileNet Integration

IBM

SupportPac-BA77 Documentation

Contents

IBM WebSphere Business Modeler FileNet Integration

IBM® WebSphere® Business Modeler FileNet Integration SupportPac-BA77 enables you to export processes as XPDL files that you can open with the FileNet P8 Process Designer. The SupportPac™ also includes the FileNet Business Process Manager mode, which supports creating FileNet-friendly models.

About this document

This document contains the following information:

- Instructions for installing the IBM WebSphere Business Modeler FileNet Integration.
- Known issues and limitations of the SupportPac.
- Documentation on modeling in the FileNet Business Process Manager mode. This documentation includes examples of two recommended modeling patterns.
- Documentation on exporting using the FileNet Business Process Manager export.
- A reference of XPL and ZXP error messages.
- A tutorial on how to model the Loan Application sample that accompanies the SupportPac.

New in this release

SupportPac-BA77 has been enhanced with additional functionality and fixes to known issues.

Business items supported

Previously, FileNet Business Process Manager mode only supported basic types such as String and Integer. The mode now supports business items. When you export a process that contains business items using the FileNet Business Process Manager export, the export transform the business items into XML data fields.

Fixes

The following known issues in the previous release of the SupportPac have been addressed:

- Invalid queue type and participant combination error when transferring a workflow
- Complex types export with no error or warning
- Unsupported expressions export with no error or warning

Known issues and limitations

This section contains the known issues and limitations for the IBM WebSphere Business Modeler FileNet Integration SupportPac.

Validation not performed on existing projects

The WebSphere Business Modeler FileNet Integration SupportPac provides the FileNet Business Process Manager mode, which validates projects to support

exporting the projects using the FileNet Business Process Manager export. However, WebSphere Business Modeler does not automatically trigger the validation for existing projects. It only triggers the validation for new models. To trigger validation on an existing project, export the project using the WebSphere Business Modeler export (creating a MAR file) and then import the MAR file into a new project.

WebSphere Business Modeler does not require symmetrical processes

In FileNet P8, workflows must be symmetrical, which means that every AND-split must have a corresponding AND-join (collector) step. The paths between the split and the join must either terminate or reach the collector step. A path may only enter the split/join area through the split and a path may only leave the split/join area through the collector step. For more information on symmetrical processes, refer to the "Create a valid workflow map" topic in the FileNet P8 documentation.

WebSphere Business Modeler does not require symmetrical flows. If the flow you create in WebSphere Business Modeler is not symmetrical, the exported process will not be symmetrical and it will result in a validation error in the Process Designer. When you are creating processes in WebSphere Business Modeler, make sure that their flows are symmetrical.

Due to the complexity of splits and joins, it is also possible to create a symmetrical process in WebSphere Business Modeler, but the resulting exported process is not symmetrical. In this case, fix the paths in Process Designer by changing the routing of a step to collector or by adding a step as a collector step.

Global repositories do not have validation errors

The FileNet Business Process Manager mode does not allow you to add global repositories. However, if you model in another mode, drop a global repository into the process, and then switch to the FileNet Business Process Manager mode, the global repository will not have a validation error. Other unsupported elements will have the error.

Calls to global processes need further development in the Process Designer

While the FileNet Business Process Manager mode and the FileNet Business Process Manager export support global processes, to make the call to the global process fully functional, you need to perform several steps:

1. Open the called global process (a workflow process in the Process Designer) by selecting **View** → **Edit Workflow** → *global process name*.
2. Select **View** → **Workflow Properties**. In the Workflow Properties window, select the **Advanced** tab.
3. In the **Condition Identifier** field, specify a unique identifier expression. A unique identifier expression is usually one or more data fields.
4. Click **Close**.
5. Return to the calling process (a workflow process in the Process Designer) by selecting **View** → **Edit Workflow** → *process name*.
6. Select the global process System step. In the General Properties, double-click the **WaitForCondition** function. The WaitForCondition window opens.

7. In the table, double-click the Condition cell for the first condition. The Condition Parameters window opens.

8. Check that the Workflow Name field contains the called global process name and the Condition Identifier field contains the condition identifier you specified in step 3.

   Important: If the global process name is not listed in the Workflow Name field, the global process was not included in the export and therefore not defined in the XPDL file. In WebSphere Business Modeler, re-export the calling process and ensure that you also include the global process in the export. After completing the export, verify that the definition for the global process is in the XPDL file.

9. Specify the operator and the expression that the runtime engine will be compared to the condition identifier in the work item that the flow is waiting for. The expression can reference one or more data fields in the work item but it cannot reference an operation parameter.

   Note: The data types of the expression and the condition identifier must be the same.

10. Click **Close** to close the Condition Parameters window and then click **Close** to close the WaitForCondition window.

11. Validate the workflow.

Installing the SupportPac

To create processes that you can open with FileNet P8 Process Designer, you need to install the WebSphere Business Modeler FileNet Integration SupportPac.

The IBM WebSphere Business Modeler FileNet Integration SupportPac requires Interim Fix 7 for WebSphere Business Modeler. You can obtain the interim fix from www.ibm.com/software/integration/wbimodeler/advanced/support/. Install the interim fix using the instructions that accompany the interim fix.

The SupportPac also requires FileNet P8 Service Pack 2 if you use business items when you model processes in WebSphere Business Modeler.

The IBM WebSphere Business Modeler FileNet Integration SupportPac is a ZIP file that has the following contents:

- BA77.pdf (this file)
- License folder, which contains the license agreement files
- Samples folder, which contains the Loan Application sample.mar file and the Loan Application sample.xpdl file. You can use these samples in conjunction with the tutorial in this document to help understand the FileNet Business Process Manager mode and the FileNet Business Process Manager export.
- Plugins.zip, which contains the code for the SupportPac.
- Readme.htm

To install the SupportPac, complete the following steps:

1. Unzip the SupportPac ZIP file to a temporary folder.
2. Extract the plugins.zip folder to the *WebSphere Business Modeler install folder*\eclipse folder.

   Note: If you have already installed a version of the SupportPac, a window will open asking whether to overwrite existing files. If this occurs, click **Yes**.

   For example, if you installed WebSphere Business Modeler into the C:\Program Files\IBM\WebSphere\Modeler6 folder, extract the zip file to C:\Program Files\IBM\WebSphere\Modeler6\eclipse folder. The *WebSphere Business Modeler install folder*\eclipse\plugins folder now contains the following folders:
   - com.ibm.btools.mode.xpdl_1.0.0
   - com.ibm.btools.te.bomxpdl_1.0.0

You will be able to use the FileNet Business Process Manager mode and the FileNet Business Process Manager export the next time you start WebSphere Business Modeler.

FileNet Business Process Manager mode

If you plan to use the FileNet P8 Platform as the runtime platform for models developed in WebSphere Business Modeler, you should begin modeling in FileNet Business Process Manager mode. If you have already created a model in another mode, you can switch to FileNet Business Process Manager mode and then fix the errors that the Errors view shows you before you try to export the model.

The FileNet Business Process Manager mode provides limitations and additional validation to the Advance Business Modeling mode that support creating business process models that you can open with FileNet P8 Process Designer. The mode helps you bridge the differences between the modeling paradigm of WebSphere Business Modeler and the modeling paradigm of FileNet P8.

- In WebSphere Business Modeler, there are numerous elements for which an equivalent in FileNet P8 does not exist or the equivalent is too semantically different to use. These elements are unsupported in the FileNet Business Process Manager mode and are not available in the palette. These unsupported elements are:
  - Observers
  - Timers
  - Notification broadcasters
  - Notification receivers
  - Global tasks
  - Global services
  - Global repositories
  - Business services
  - Business service objects

  If you have already created these elements in the model, WebSphere Business Modeler marks them with an error indicator and provides a message in the Errors view. The messages have a ZXP code. If you export a process containing unsupported elements, the FileNet Business Process Manager export transforms the unsupported elements into XPDL activities. When you open the process in FileNet P8 Process Designer, these activities display as connected general steps.

- In WebSphere Business Modeler, there are a number of basic data types for which an equivalent in FileNet P8 does not exist. These basic data types are not supported. The supported basic data types are:
  - Boolean
  - Date
  - DateTime
  - Float
  - Integer
  - String
  - Time

  Important: The unsupported data types are available for you to select; however, when you save your process model, the validation performed by the FileNet Business Process Manager mode will mark them with a warning. If you export an unsupported data type, the FileNet

Business Process Manager transforms it into the appropriate supported data types in the XPDL file:

Unsupported basic data type	Data type after being exported
Byte	String
Double	Float
Duration	String
Long	Integer
Short	Integer

- In WebSphere Business Modeler, you model data flow within business processes with one process element passing data to another through a connection. However, in FileNet P8 you model process-level variables that the steps reference to get data. To bridge the two paradigms, in WebSphere Business Modeler you use repositories to contain the data and connect the repository to each process element that needs that data. To support this, map elements perform a special role in the FileNet Business Process Manager mode. Each process can only have one supported map element and if the process does, the map must be the first element in the process flow. The purpose of the map is to initialize repositories with data. If a map appears anywhere else in the process, it is an unsupported element. Flow control elements (decisions, forks, joins and merges) within a WebSphere Business Modeler process either merge into an adjacent step as routing information or export as general steps in the FileNet P8 process.
- In WebSphere Business Modeler, business items have a greater variety of types than FileNet P8 supports. For this reason WebSphere Business Modeler exports business items as XML data fields that reference type definitions in XSD files. While this use of XML data fields may not by typical for many FileNet P8 users, it has the advantage of preserving the rich amount of details that can be modeled within WebSphere Business Modeler

    Important: To import business items successfully, FileNet P8 must have Service Pack 2 installed.
- When you are modeling in FileNet Business Process Manager mode, you should use unique names for your processes. While WebSphere Business Modeler allows you to create many processes or other elements with the same name provided that they are in different folders, if you try to export two or more processes with the same name, you will get an error and the Process Designer of FileNet P8 will not be able to open the XPDL file.

See the reference pages for more details of constraints and the error messages page for specific validation errors.

Modeling in FileNet Business Process Manager mode

The FileNet Business Process Manager mode provides additional validation to the Advanced Business Modeling mode to support creating a model that can be successfully imported into the FileNet P8 platform.

When modeling in FileNet Business Process Manager mode, follow these practices for the best results when importing the process into FileNet Business Process Manager:
- If you are using a particular type for any data element more than once in a process (for example, you are using String as the type for Customer and for

Employee), place a local repository with a capacity of 1 in the process to represent the data element. If a process element needs the data represented by the repository, create a connection from the repository to the process element. Similarly, if the process element outputs data, create a connection from the process element to the appropriate repository. By modeling in this pattern, the repository becomes a process-level variable when you import it into FileNet P8.

Note: In FileNet Business Process Manager mode, the default capacity of a repository is 1. This differs from the default capacity of repositories created in the Basic Business Modeling mode, the Intermediate Business Modeling, or the Advanced Business Modeling, which have a default capacity of unlimited.

Tip: If you have many process elements accessing data from the same repository, consider removing repository connections from the diagram to make it tidier. Creating processes that import cleanly into FileNet P8 often requires the use of many repository connections. To prevent the repository connections from displaying in the diagram, use the preference.

- Do not use merge elements to merge data flow in WebSphere Business Modeler. A data merge does not have a corresponding semantic equivalent in FileNet P8.
- In WebSphere Business Modeler, when you have a repository and tasks on concurrent paths that access the data in the repository, the tasks are accessing the same data. When you export this to FileNet P8, the tasks will access their own copy of the data.
- If you want to have a process that has an input, place a map as the first element in the path to accept data entering the process. Use the map to initialize the repositories with the data received from the input.
- Use roles or the individual resources based on the Staff or Person predefined individual resources when setting the resources for a task. This practice ensures that the export creates the corresponding participants and workflow groups.
- Use unique names for processes even if the processes are in different catalogs. While WebSphere Business Modeler allows you to reuse names within separate process catalogs, exporting two or more processes with the same name causes an error.

Using a map to initialize repositories

When you are modeling a process in FileNet Business Process Manager mode, use a map to initialize repositories with data.

Before you begin, make sure that you are modeling in FileNet Business Process Manager mode.

Process flow in FileNet P8 does not include data flow. This means that you cannot pass data into the process like you would when you model in other modes. Instead, the FileNet Business Process Manager mode uses a map to initialize repositories with data entering the process. When you export the process using the FileNet Business Process Manager export, WebSphere Business Modeler transforms the map into the proper XPDL code.

To use a map to initialize repositories, complete the following steps:

1. Unless you want concurrent flows to start the process, delete the start node element ⬤. The problem with concurrent flows at the start is that there is no

guarantee that the flow that initializes the repositories will have done so before the flow originating from the start node tries to access the repositories for a value.

2. Right click where you want to place the map and select **New → Map**.
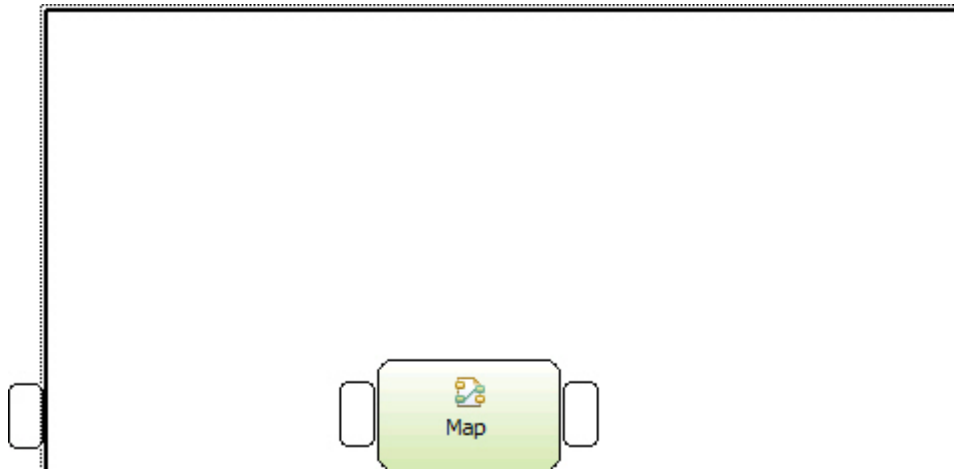
3. To create the data inputs for the process, click any blank area in the diagram. The Attributes view displays the attributes of the process.

4. Select the **Inputs** tab.

5. Click **Add** for each piece of data you are passing into the process. For example, say you are passing a String and an Integer into the process. You would click the **Add** button twice. This action creates two inputs.

6. For each piece of data that is not a String, change its data type by completing the following steps:

   a. Click **String** in the associated data cell. The cell displays a button.

 | Name | Associated data | Minimum | Maximum | |
   |---|---|---|---|---|
   | Input | String | 1 | 1 | |
   | Input:2 | String | 1 | 1 | |
   | | | | | |
   | | | | | |

   b. Click the button. The Type Selection window opens. The Type Selection window enables you to specify what type of data is entering the process.

   > **Important:** Even though the **Basic type** drop down list contains many different types, not all of them are supported in FileNet Business Process Manager mode. The supported types are: Boolean, Date, DateTime, Float, Integer, String, and Time.

   c. From the **Basic type** drop down list, select **Integer**. Click **OK**.

 | Name | Associated data | Minimum | Maximum | |
   |---|---|---|---|---|
   | Input | Integer | 1 | 1 | |
   | Input:2 | String | 1 | 1 | |
   | | | | | |
   | | | | | |

7. In the palette, click the Connection tool. Click a process input ⟫ and then click the input criterion of the map. This action creates a data flow from outside of the process to the map inside the process. To continue the example, a connection with Integer data joins the process input to a map input.
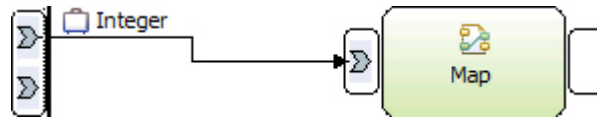
When you also connect the input with a String data type, the diagram looks like this:
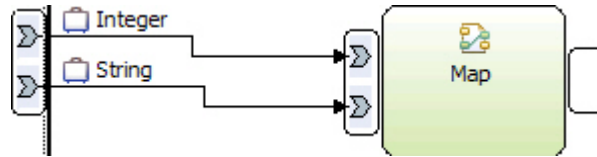

8. Create a local repository for each piece of data you want to pass into the process. For example, if you are passing a String and an Integer into the process, create two local repositories.
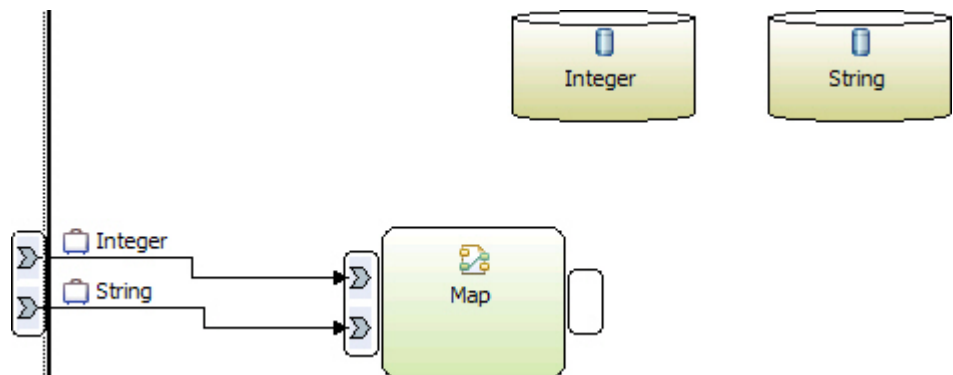
9. If a repository is going to store non-String data, right-click the repository and select **Associate data**. In the window that opens, select the data type from the **Basic type** drop down list and then click **OK**. For example, right-click the repository that will store an Integer and select **Associate data**. For the basic type, select **Integer**.

10. Select the map. In the Attributes view, click the **Outputs** tab.

11. Add outputs to the map using the same technique you used to add inputs to the process. Add an output for each input that the map has. In the example, the map has a String input and an Integer input. You need to create an output that has String associated data and an output that has Integer associated data.

12. For non-String data, set the repository for the data to store the correct data type:

    a. Right-click the Loan Amount repository and select **Associate data**. The Type Selection window opens.

    b. From the **Basic type** drop down list, select **Integer**. Click **OK**.

    Note: You do not have to change the data type for repositories that are storing a String because that is the default type for repositories.

13. From the palette, click the connection tool. Click an output in the map and then click the appropriate repository. For example, connect the output with the String to the String repository and connect the output with the Integer data to the Integer repository.
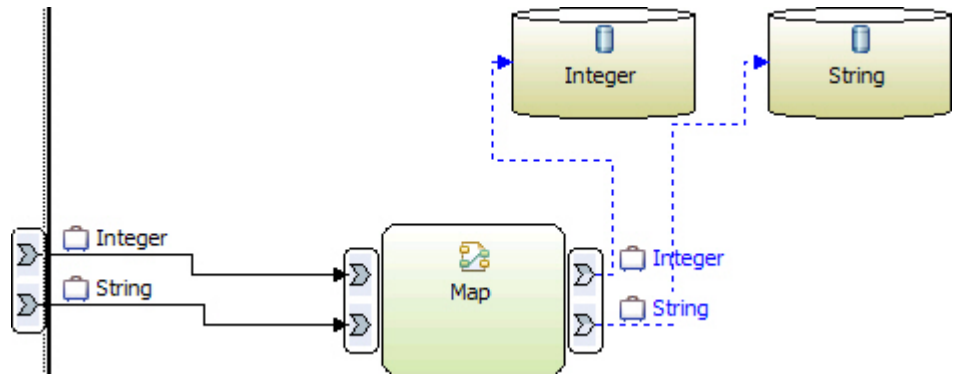
The repositories now contain the data that was passed to the process. You can make connections to these repositories to access the data.

Using repositories to pass data

When you are modeling in FileNet Business Process Manager mode, use repositories to pass data between elements in the process.

If you are going to use business items in your process, ensure that FileNet P8 has Support Pack 2 applied.

Process flow in FileNet P8 does not include data flow. This means that you should avoid including data flow in your WebSphere Business Modeler process. Instead, elements should store data in repositories and other elements access the repositories to get the data.

To use repositories to pass data, complete the following steps:

1. Create a local repository to store the data by right-clicking where you want to place a repository and select **New → Local Repository**. For example, say that you have a customer ID, which is an Integer data type. Create a repository called `Customer ID`.

2. If a repository is going to store non-String data:

   a. Right-click the repository and select **Associate data**.

   b. In the window that opens, select **Basic type** or **Complex type**.

   c. If you selected basic type, select the specific type from drop down list. If you selected complex type, select the business item from the tree.

      Important: Even though the **Basic type** drop down list contains many different types, not all of them are supported in FileNet Business Process Manager mode. The supported types are: Boolean, Date, DateTime, Float, Integer, String, and Time.

   For example, right-click the Customer ID repository and select **Associate data**. For the basic type, select **Integer**.
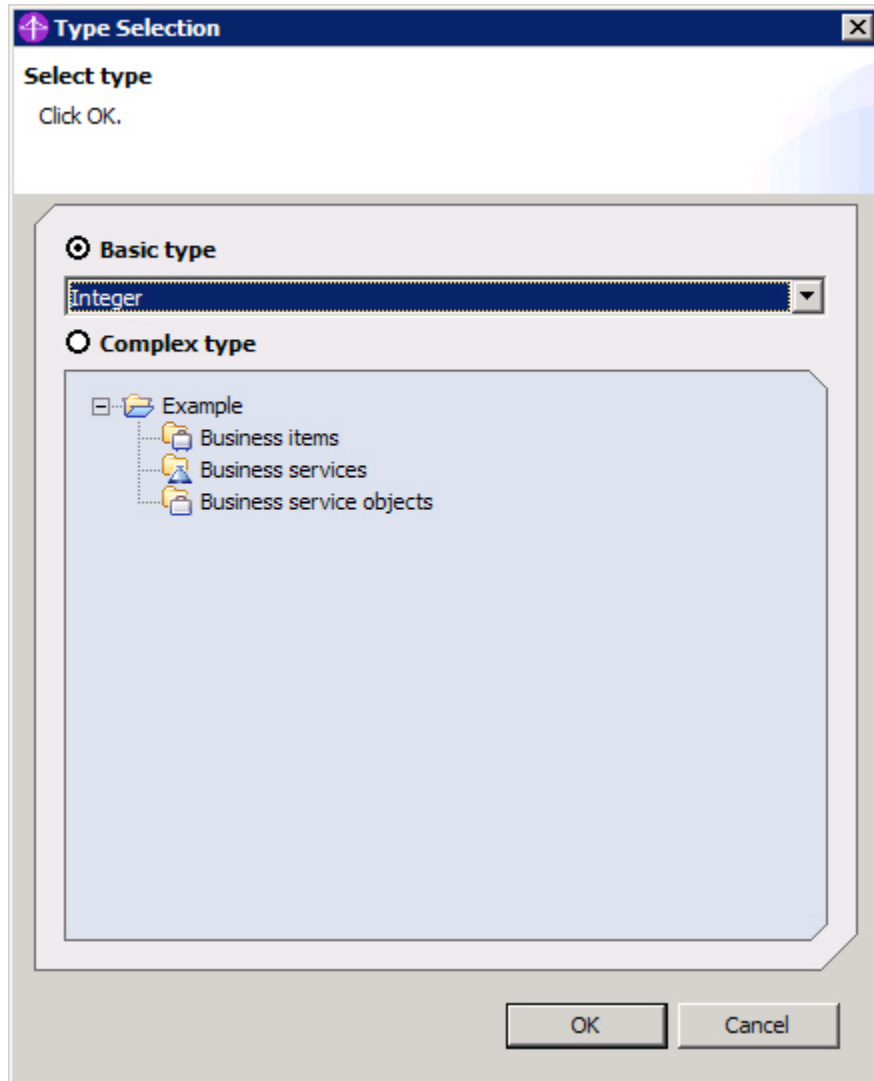
d. Click **OK**.

3. In the process element that is outputting the data, create an output:

   a. Click the process element. The Attributes view displays the attributes of the element.

   b. In the Attributes view, click the **Outputs** tab.

   c. Click **Add**.

   d. Change the associated data for the output named **Output** from String to the same data type associated with the repository.

   For example, the Create Customer task creates a customer ID for a new customer. To reflect this in the diagram, create an output for the Create Customer task and set the data type associated with the output to Integer.

4. From the palette, click the connection tool. Click the repository and then click the input for the process element inputting the data.

   Note: The connection creates the input and sets the data type automatically. For example, connect the Customer ID repository to the Get Customer task.
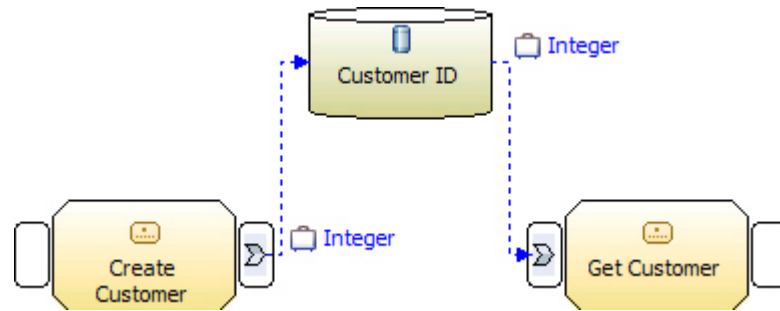
The Create Customer task is now able to pass the Customer ID to the Get Customer task.

Process elements can now pass data to other process elements. Note that any number of elements may input into a repository and any number may output from the repository.

Tip: If your diagram looks untidy and confusing, you can make it tidier by hiding the repository connections in the diagram. Right-click the diagram and select **Show** → **Repository Connections**.

Exporting to FileNet P8

You can export one or more processes (and process catalogs) so that you can then open them in FileNet P8. The FileNet Business Process Manager export also exports any processes or other model elements referenced by the processes you are exporting.

The following steps describe how to export the processes as an XPDL file and then open the file with the Process Designer of FileNet P8 to create implementations for the FileNet P8 runtime.

Exporting FileNet P8 files

You can export a process as an XPDL file for importing into the IBM FileNet P8 Platform.

Before exporting a process destined for FileNet P8, switch to FileNet Business Process Manager mode and check that your process does not have errors or warnings. If there are errors or warnings, you can still export the process but the resulting XPDL file may contain extra elements and data that you will need to handle in FileNet P8 when you import the process. For example, if you have an error about an unsupported element, the element will still be exported in the XPDL file and displays as a connected general step in the Process Designer of FileNet P8.

Important: If you are exporting processes from more than one folder or project, make sure that all of the processes you are exporting have unique names. If you export multiple processes with the same name, you will get an error. Additionally, you will not be able to open the XPDL file with FileNet P8.

If your process includes business items, the FileNet Business Process Manager export will also create XSD files, which contain schemas that define an equivalent set of data fields in FileNet P8.

Important: To import business items successfully, FileNet P8 must have Service Pack 2 installed.

To export a model for FileNet P8, complete the following steps:
1. In the Project Tree, select the project or process that you want to export.
2. Right-click and select **Export**. The Export wizard appears.
3. Select **FileNet Business Process Manager (.xpdl)** and click **Next**.
4. Click **Browse** to select the target directory where you want to put the files resulting from the export.
5. Select the project to export from the drop-down list.
6. Click **Export entire project** to export all of the processes in the project and the resources they reference, or **Export specific elements** to export only the selected processes and the resources they reference. If you chose **Export specific elements**, use the navigation tree to select the processes. You can select more than one element by holding down the Ctrl key.

Note: the export will include elements (including processes) that the selected processes reference, whether you select them here or not

7. Optional: If you want to the export to place all of the files with in a ZIP file, select the **Generate ZIP** check box.

8. Optional: If you want existing files in the directory to be replaced by the ones you are exporting, select the **Overwrite files** check box. Otherwise, you will be warned when a file is about to be overwritten.

9. When you have finished specifying your export options, click **Finish**. A window opens when the export process is complete.

10. If there were any errors or warnings during the export process, click **Details** to read them. Otherwise, click **OK**.

The export process places the exported files in the target folder. You can now import the files into FileNet P8.

Importing into FileNet P8 Process Designer

You can import the processes exported using the FileNet Business Process Manager export into the IBM FileNet P8 Platform.

1. Open the Process Designer in the FileNet P8 workplace.

2. If the export created XSD files to represent business items, copy the XSD files to the defined location for IBM FileNet P8 Platform (*P8_install_directory*/ FNSW_LOC/sd/xsd.) Make sure that you preserve the folder structure of the XSD files.

3. Click **File → Open**

4. Browse to the XPDL file and select it.

Once you have imported the model elements, you can create an implementation for the FileNet P8 Process Engine. For more information on FileNet P8 and how to use its tool, refer to the FileNet P8 documentation.

Mapping elements to FileNet P8 constructs

Many WebSphere Business Modeler elements map to equivalent FileNet P8 constructs.

The following table shows the mapping between WebSphere Business Modeler elements and their equivalent FileNet P8 object or property. If the table does not list the element, it is unsupported in FileNet P8 mode. Unsupported elements still export when you export the process using the FileNet Business Process Management export. When you import the process, the unsupported elements display as connected general steps.

Note: You can only choose the export processes. The export includes any resources that the process reference.

WebSphere Business Modeler element	FileNet P8 construct	Comments
Business item	Data field of XML type	
Connection with associated data	Route	
Connection with no associated data	Route	

WebSphere Business Modeler element	FileNet P8 construct	Comments
Decision (simple and exclusive multiple choice decisions)	Routing information or a general step	The export transforms the decision into an OR split in the routing information of the preceding step. If the export cannot do that, it transforms the decision into a general step with routing information.
Decision (inclusive multiple choice decisions)	Routing information or a general step	The export transforms the decision into an AND split in the routing information in the preceding step. If the export cannot do that, it transforms the decision into a general step with routing information.
Do-while loop	Two general steps	One step represents the condition check and the other represents the behavior contained within the loop.
End node	None	
Expression	Routing condition	
For loop	Two general steps	One step represents the condition check and the other represents the behavior contained within the loop. The condition check has three data fields to track and check the number of time the loop has been performed.
Fork	Routing information or a general step	The export transforms the fork into routing information in the preceding step. If the export cannot do that, it transforms the fork into a general step with routing information.
Input	Parameter	
Input criterion	None	Only one input criterion is supported.
Join	Routing information or a general step	The export transforms the join into routing information in the following step. If the export cannot do that, it transforms the join into a general step with routing information.
Map	None or a connected general step	If the process starts with a map, the export creates data fields based on the output of the map. This map does not appear in the imported workflow. The export creates unconnected general steps for any other maps in the process.
Merge	Routing information or a general step	The export transforms the merge into routing information in the following step. If the export cannot do that, it transforms the merge into a general step with routing information.
Output	Parameter	
Output criterion	None	Only one output criterion is supported.
Top-level global process	Workflow process	

WebSphere Business Modeler element	FileNet P8 construct	Comments
Local subprocess (local process within another process)	Submap	
Global process within another process	System step	The export transforms the global process into a system step associated with the create function and, if the call synchronously attribute is enabled, the waitForCondition function.
Local repository	Data field	
Resource definition, resource definition template	None	
Individual resource	None	
Individual resource requirement	Participant	Maps to a participant for the step if the individual resource is based on the predefined staff or a person resource definition. Otherwise, it does not export.
Role	Workflow group	At the workflow level, a role requirement maps to a workflow group.
Role requirement	Participant	The role in the role requirement maps to a participant for the step
Start node	None. The export creates a single Launch step activity for the process	
Stop node	System step or none	If the stop node is in the top-level process, the export transforms the stop node into a system step that has the terminateProcess function. Otherwise, the export does not transform the stop node.
Local task	General step	
While loop	Two general steps	One step represents the condition check and the other represents the behavior contained within the loop.

Tutorial: Modeling for FileNet Business Process Manager

This tutorial introduces you to the FileNet Business Process Manager mode and shows you how to create a model in WebSphere Business Modeler and then implement it on the FileNet Business Process Manager platform.

The tutorial modules use the **FileNet Business Process Manager** business modeling mode. By completing the full tutorial, you will have an understanding of how you can take rich and useful models in WebSphere Business Modeler and implement them on the IBM FileNet P8 Platform.

The Modeling for FileNet Business Process Manager tutorial creates a simple loan application process. The process receives a loan application and evaluates it to approve or reject the application.

Important: The sample used in this tutorial is simplified to improve the ease of accomplishing the learning objectives. As a result, a number of the tasks and data that might typically be in a real-life loan application process may be absent.

In Lesson 1, you model the process in WebSphere Business Modeler using the FileNet Business Process Manager mode. The tutorial demonstrates the modeling pattern that optimizes the data exported when you export the process. In Lesson 2, you export the process using the FileNet Business Process Manager export and then you import the resulting XPDL file into the Process Designer of FileNet Business Process Manager. In Lesson 3, you set up the workflow environment and associated the process steps with the queues, operations, and data fields in the workflow environment.

Learning objectives

After completing the entire tutorial, you will know how to do the following:
- How to model a process to optimize its import into FileNet Business Process Manager
- How to export the process from WebSphere Business Modeler and then import it into the Process Designer for FileNet P8
- How to complete the process in FileNet P8

Time required

This tutorial should take approximately 90 minutes to complete. If you explore other concepts related to this tutorial, it could take longer to finish.

If you would rather just read the exercises, or want to jump ahead to Lesson 2 without completing Lesson 1, you can import the Loan Application sample project. This sample project is a companion to the tutorial, providing the process diagram and other modeling elements that you would create if you completed each exercise in the tutorial.

If you want to do Lesson 3 without doing or completing Lesson 2, you can use the Loan Application sample.xpdl file that accompanies the sample project. This file is an export of the Loan Application sample.mar project.

System requirements

To do Lesson 1, your computer must have the following software installed:
- WebSphere Business Modeler Advanced Version 6.0.2.1 with Interim Fix 7 or higher
- WebSphere Business Modeler FileNet integration

To do Lesson 2, you must have the required software for Lesson 1 and Author access to the IBM FileNet P8 Platform

To do Lesson 3, you must have Admin access to the IBM FileNet P8 Platform

Prerequisites

While it is not necessary to complete this tutorial, to better understand the concepts and modeling practices of WebSphere Business Modeler, you should complete the Quickstart tutorial, which you can find in the WebSphere Business Modeler documentation. WebSphere Business Modeler also comes with a number of sample projects that you can load and explore.

Lesson 1: Creating the process model

In this lesson, you model the loan application process in FileNet Business Process Manager mode and then export it.

The FileNet Business Process Manager mode provides additional validation to the Advanced Business Modeling mode. This validation helps you create models that will successfully import into FileNet P8 while minimizing the amount of configuring you need to do in the Process Designer to create an implementation of the process. When you are modeling in FileNet Business Process Manager mode, WebSphere Business Modeler marks unsupported elements and attributes with error or warning indicators and creates entries in the Errors view. For more information, see "FileNet Business Process Manager mode" on page 7.

Note: The lesson models a loan application process. You can see the completed process in the loan application sample. You can use the sample to help you understand what is happening in the tutorial and use it as an additional guide in completing the lesson.

Lesson 1 consists of the following topics:
- "Switching to FileNet P8 mode"
- "Creating the project and process" on page 21
- "Passing data into the process" on page 22
- "Storing the data" on page 25
- "Adding the process elements" on page 27
- "Connecting the process elements" on page 28
- "Lesson checkpoint" on page 38

Switching to FileNet P8 mode

The best way to model the process to minimize the amount of work you need to do in FileNet Business Process Manager is to work in the FileNet Business Process

Manager mode . This mode restricts the palette to supported elements and provides validation to help you optimize your modeling.

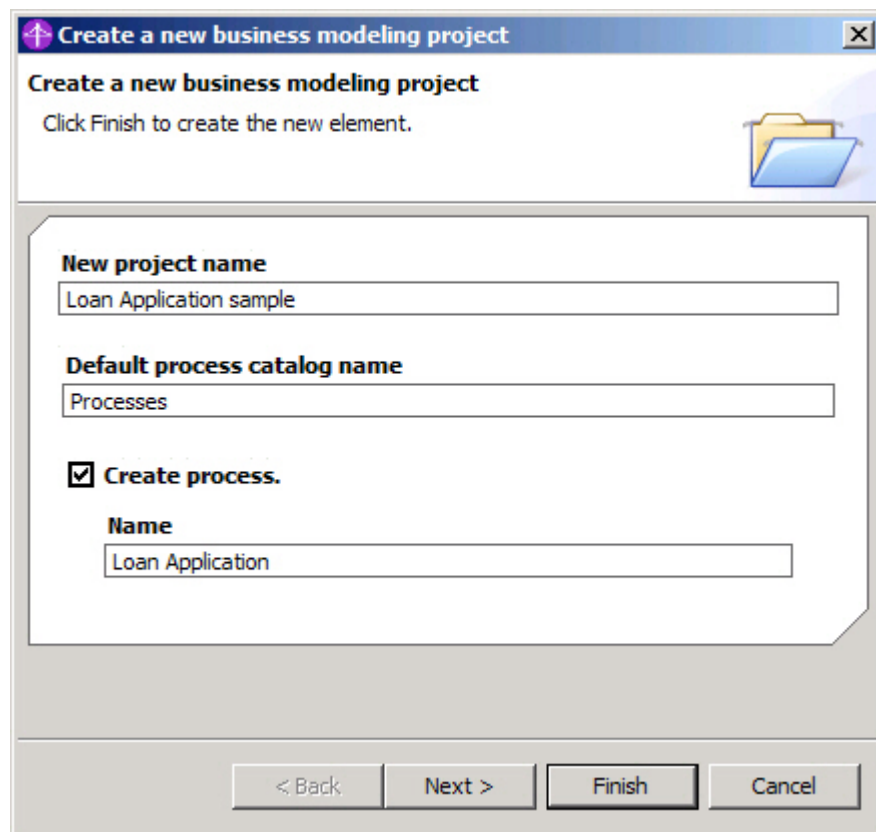To change to FileNet Business Process Manager mode:

1. From the Window menu, select **Window → Preferences**
2. In the Preferences window, expand the **Business Modeling** node and then select **Modes**.
3. Select the **FileNet Business Process Manager** mode and then click **OK**.

Creating the project and process

Before you can begin creating your process model, you must have a project to contain the process.

To create the project and the process within it:

1. From the File menu, select **File → New Business Modeling Project**. The Create a new business modeling project wizard opens.

2. For the name of the project, type: `Loan Application sample`
3. For the name of the process, type: `Loan Application`
4. Click **Finish**.

Note:

If you switch to the four-pane layout , the project will look like this:

```
☐ 🗁 Loan Application sample
      🗅 Business items
  ☐ 🗗 Processes
        🗗 Loan Application
      🗀 Resources
      🗀 Organizations
      🗀 Classifiers
      🗀 Reports
  ☐ 🗀 Queries
      🗀 Business services
      🗀 Business service objects
```
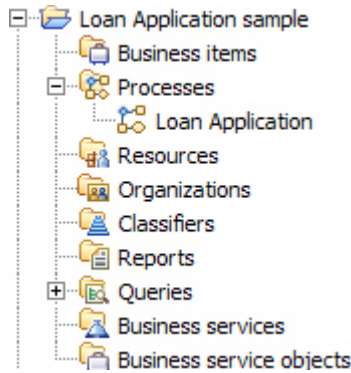
Passing data into the process

The Submit Application process has input data. This data comes from a loan application form filled out by or on behalf of a customer. Within the form are fields for the amount of the loan and the type of loan such as a home equity loan or an automobile loan. The Submit Application process requires this data. The recommended way to model the import of the data in FileNet Business Process Manager mode is to create a map that receives the data and then places the data into repositories.

Elements in a WebSphere Business Modeler diagram input data and control flow from the left and output data and control flow to the right. Many of the elements use an input criterion as their entry point and an output criterion as their output. The following table displays input and output criteria and what the symbols inside them mean:

| Icon	Description	Comments
( )	Input or output criterion	If the criterion is on the left of the element, it is an input criterion. If it is on the right, it is an output criterion
▷	Input or output with control flow only	
▷	Input or output with control flow and data flow	FileNet P8 does not have a data flow paradigm so this type of input or output should only occur between the process input and the map that initializes repositories
▷	Input or output with data flow only	

To model the loan application process in FileNet Business Process Manager mode, do the following:

1. The Submit Application process will not use the start and stop nodes that are on the diagram by default. Remove them by doing the following

   a. In the upper left corner of the diagram, click the start node element ⬤ and then press the Delete key.

   b. Scroll to the lower right corner of the diagram.

   c. Click the stop node element ⬤ and then press the Delete key.

2. Scroll back to the left side of the diagram. Right-click where you want to place the map and select **New** → **Map**. Rename the map to `Submit Application`.

> **Note:** If you save the process now, the map displays an error indicator ▣ to indicate that the map must have at least one output that produces data. You will fix the problem when you make some connections later in the tutorial.

3. Right click where you want to place a repository and select **New** → **Local Repository**. Rename the repository to `Loan Amount`. Create another repository and rename it to `Loan Type`.
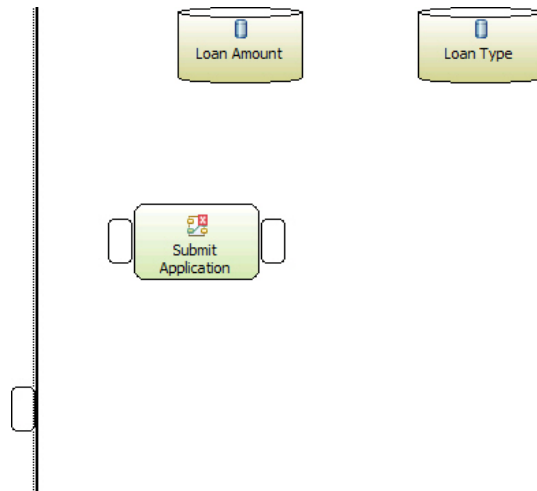
> **Tip:** If you are unhappy with where you placed an element, you can drag and drop it to another location.

4. Now you are going to create the data inputs for the process. Click any blank area in the diagram. The Attributes view displays the attributes of the Submit Application process.

5. Select the **Inputs** tab.

6. Click **Add** twice. This action creates two inputs.
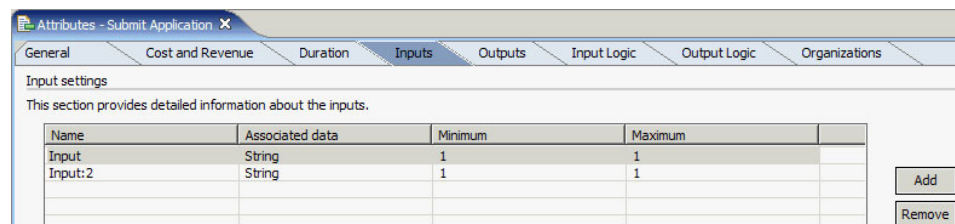
7. Use the first input to represent the amount of the loan. Change the data type for the input named **Input** by clicking **String** in the associated data cell. The cell displays a button.

8. Click the button. The Type Selection window opens.

Type Selection ✕

Select type

Click OK.

⊙ **Basic type**

```
String                                              ▼
```

○ **Complex type**

```
⊟ 📂 Loan Application Sample
     📒 Business items
     📇 Business services
     📒 Business service objects
```

OK   Cancel
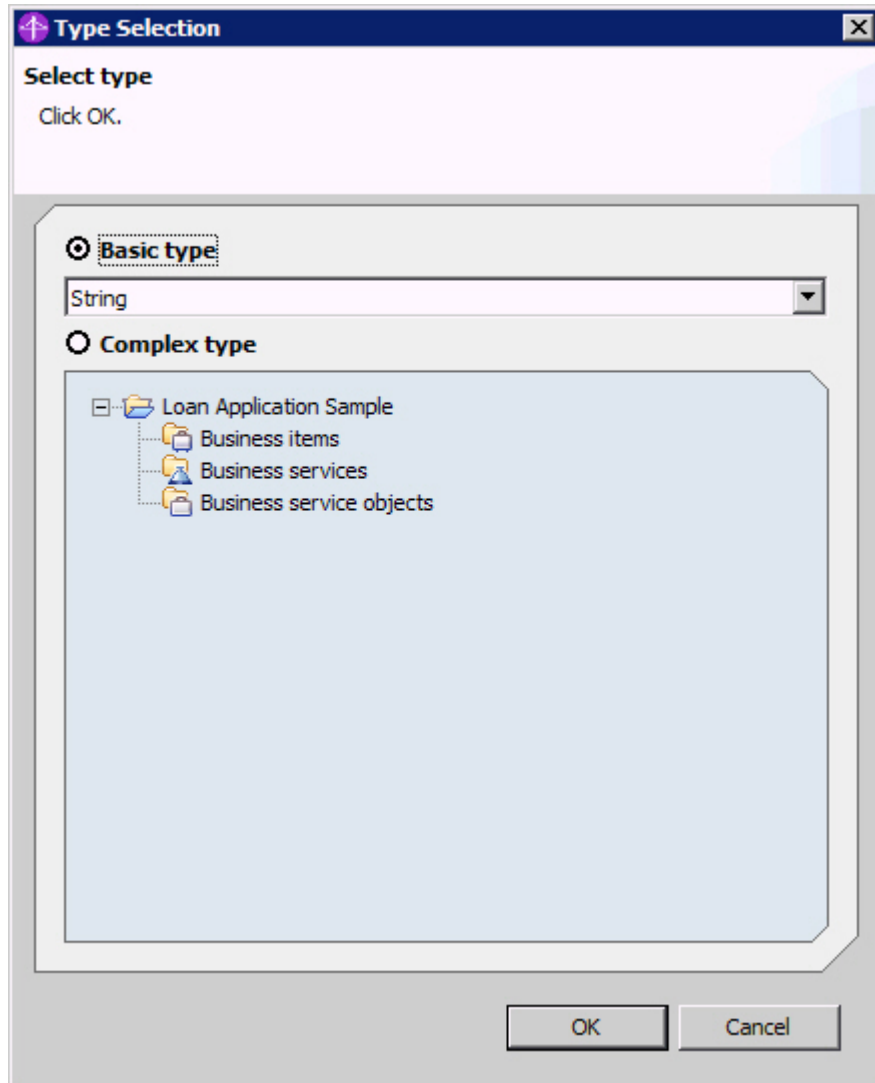
The Type Selection window enables you to specify what type of data is entering the process. The data type becomes important later on when you implement the process.

Important: Even though the **Basic type** drop down list contains many different types, not all of them are supported in FileNet Business Process Manager mode. The supported types are Boolean, Date, DateTime, Float, Integer, String, and Time.

9. From the **Basic type** drop down list, select **Integer**. Click **OK**.

Name	Associated data	Minimum	Maximum	
Input	Integer	1	1	
Input:2	String	1	1	

There is no need to change the associated data for Input:2 because it is representing the loan type, which is a String data type.

10. In the palette, click the Connection tool. Click the upper process input ≫ and then click the input criterion of the Submit Application map. This action creates a data flow from outside of the process to the map inside the process. A connection joins the process input to a map input. The connection has

Integer data.
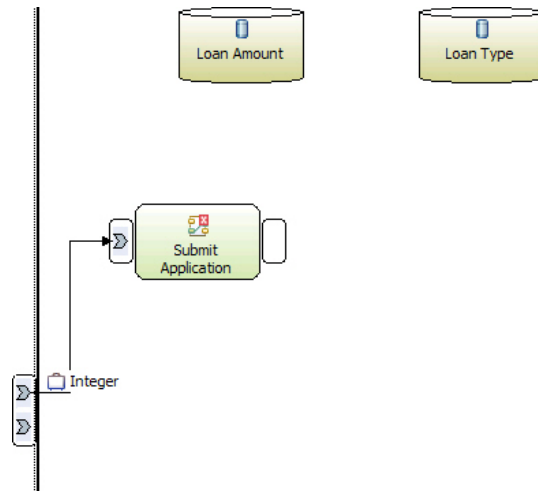
11. Connect the lower process input to the input criterion of the Submit Application map.
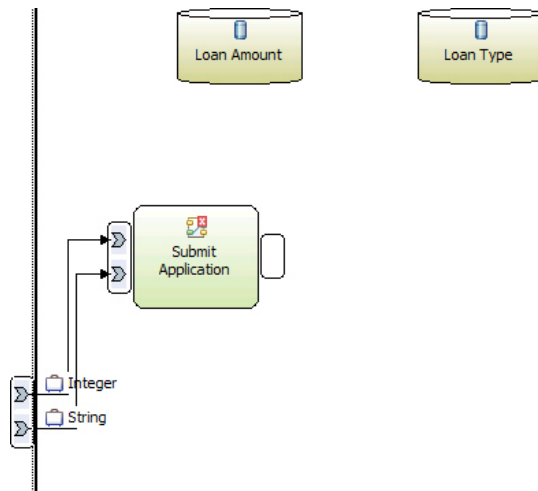
The data has now entered the process. Now you need to store it so that it is available to other elements in the process.

Storing the data

Unlike WebSphere Business Modeler, FileNet P8 does not have the concept of data flow. In FileNet Business Process Manager mode, you use repositories to store data, and the diagram elements access these repositories to retrieve or modify the data.

To store the data:

1. Click the **Submit Application** map. The Attributes view displays the attributes of the Submit Application map.
2. In the Attributes view, click the **Outputs** tab.
3. Add two outputs. Change the associated data for the output named **Output** from String to Integer. The steps for adding the outputs is similar to the same steps you used to add inputs to the Submit Application process. See "Passing data into the process" on page 22.

4. Because the loan amount is an Integer data type, you need to set the Loan Amount repository to store Integers. Right-click the Loan Amount repository and select **Associate data**. The Type Selection window opens.

5. From the **Basic type** drop down list, select **Integer**.

   Important: Even though the **Basic type** drop down list contains many different types, not all of them are supported in FileNet Business Process Manager mode. The supported types are: Boolean, Date, DateTime, Float, Integer, String, and Time.

   Note: You do not have to change the Loan Type repository because repositories store Strings by default. The loan type data is a String.

6. Click **OK**.

7. From the palette, click the connection tool. Click the upper output (named Output) in the Submit Application map and then click the Loan Amount repository. A data connection links the output to the repository.
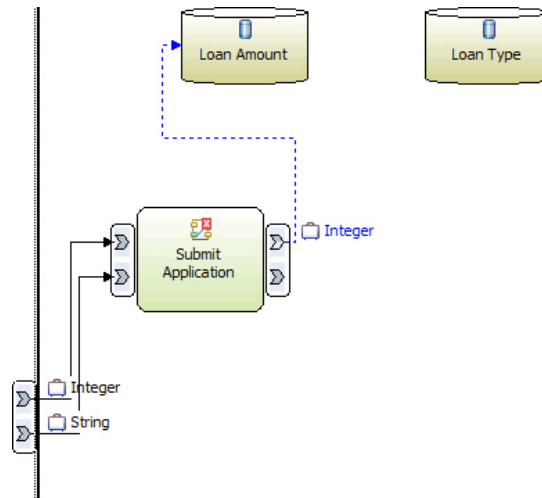
8. Click the lower output (Output:2) on the Submit Application map and then click the Loan Type repository. The diagram should look something like this:
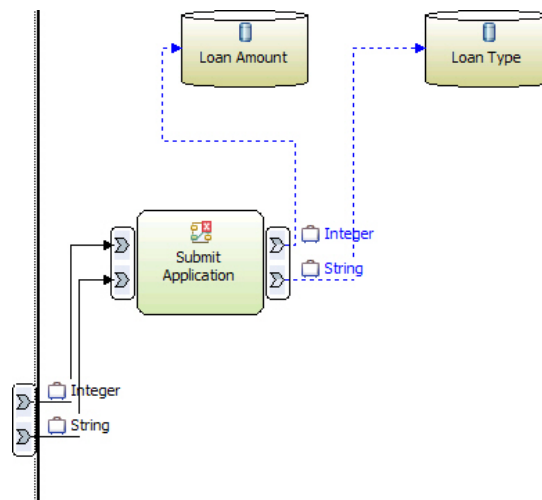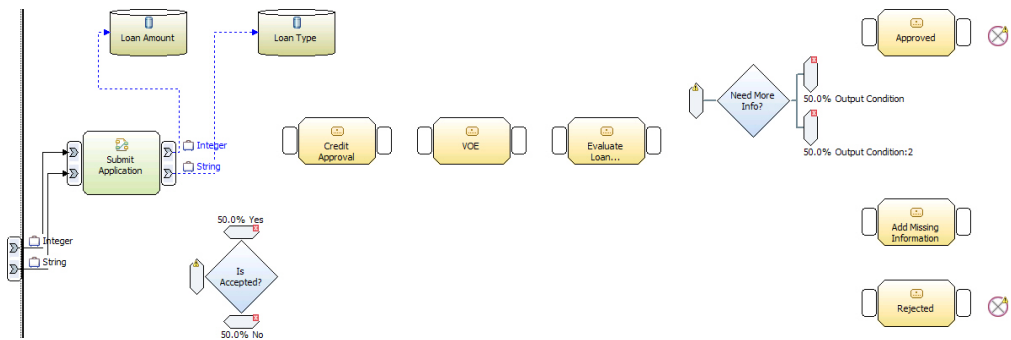
Adding the process elements

Now that you have dealt with the data entering the process, it is time to add the process elements. To add an element, right-click where you want place the element and select **New** → *Element type*.

The following table lists the elements in the order they appear in the flow from left to right:

| Element type	Rename it to...	Description
Simple decision	Is Accepted?	This decision contains business logic that checks whether the loan amount is equal to a specified value. If the loan amount is equal to or above the value, the loan is rejected outright.
Task	Credit Approval	This task prepares the application to enter the VOE system.
Task	VOE	This task represents the processing done by the VOE system. The VOE system calculates the cost of the loan.
Task	Evaluate Loan Application	This task represents the evaluation of the loan application done by the underwriter.
Multiple choice decision	Need More Info?	This decision represents whether the underwriter accepts the loan application, requires additional information, or rejects the loan application.
Task	Approved	This task processes the approved loan application.
Task	Add Missing information	This task gathers additional or missing information so that the underwriter can reevaluate the loan application.
Task	Rejected	This task processes the rejected loan application.
End		This represents the control flow reaching an end point.
End		This represents the control flow reaching an end point.

The process diagram should resemble this example:

You will fix the errors and warnings shown in the diagram when you connect the process elements.

## Connecting the process elements

With the process elements in place, you can now connect them to create a flow.

To connect the elements, complete the following steps:

1. Click the connection tool.
2. Move the pointer close to the top or bottom of outputs of the Submit Application map. The outputs are surrounded by a dotted line that represents the expanded output criterion. If the dotted line does not appear try moving the pointer a bit closer.
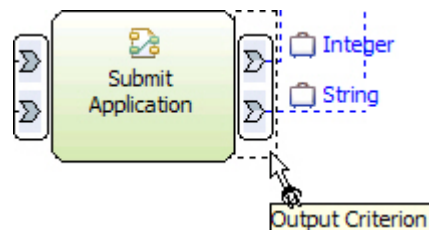
3. Click the expanded output criterion and then click the input branch of the Is Accepted decision. The Is Accepted decision applies business logic to evaluate the application. For example, it can check the amount of the loan application and check whether it falls within the range acceptable for the type of loan. In WebSphere Business Modeler, the business logic can be modeled as an expression or it can be left as a written description that is implemented in FileNet P8.
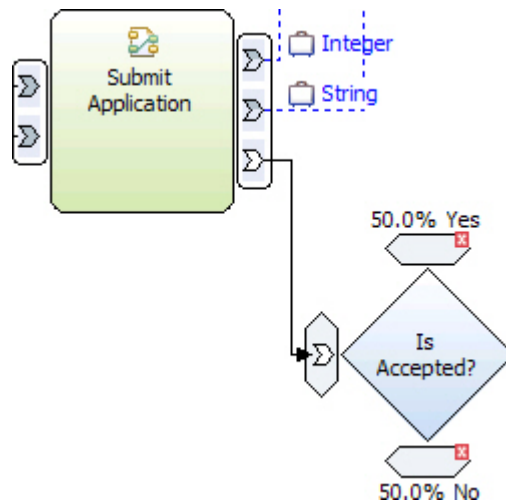
4. The output branches of a decision must have an expression:

   a. Click the Is Accepted decision. The Attributes view displays the attributes of the decision.

   b. In the Attributes view, click the **Output branches** tab.

   c. Select the Yes branch. The details for the branch display:

   d. Click **Edit**. The Expression Builder opens.

   e. In the Expression Composer area, confirm that the **First term** drop down list is set to **Modeling artifact**. Expand the tree in the First term details and select the **Loan Amount** repository. The Operator section displays a list of operators.
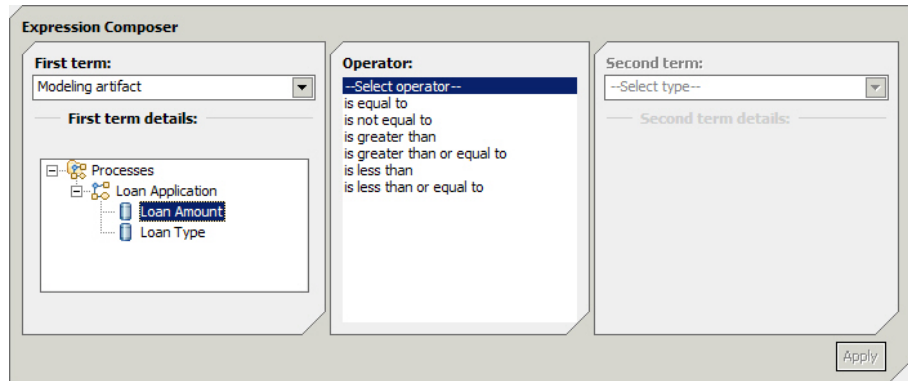
f. In the operator section, select **is less than**. The second term section enables.
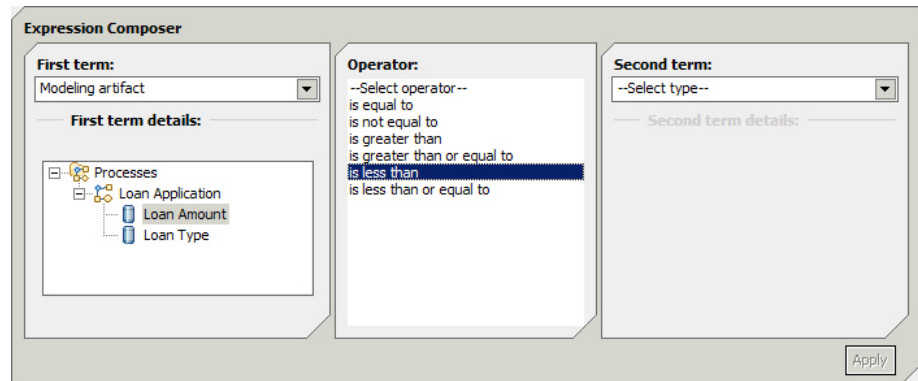
g. In the **Second term** drop down list, select **Number**. In the Number value field, type: 100000
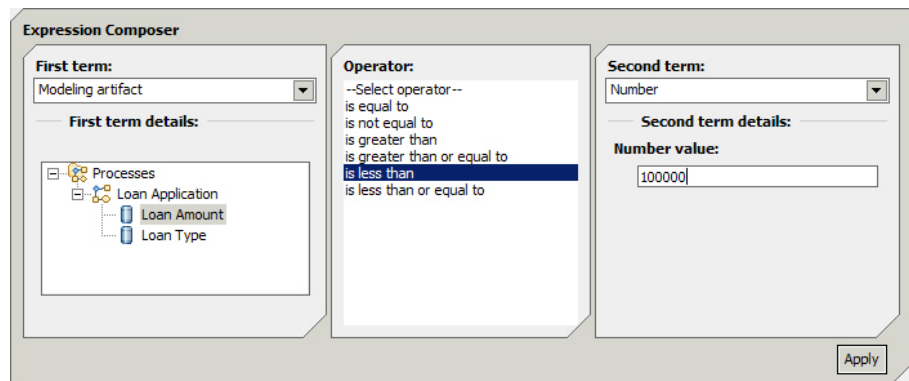
h. Click **Apply**. The Expression Builder generates the expression and populates the other fields with information based on the expression. The key field is the **Expression String** field because it contains the actual expression.

i. Click **OK** to close the Expression Builder. The **Expression** field now contains the expression you created in the Expression Builder.
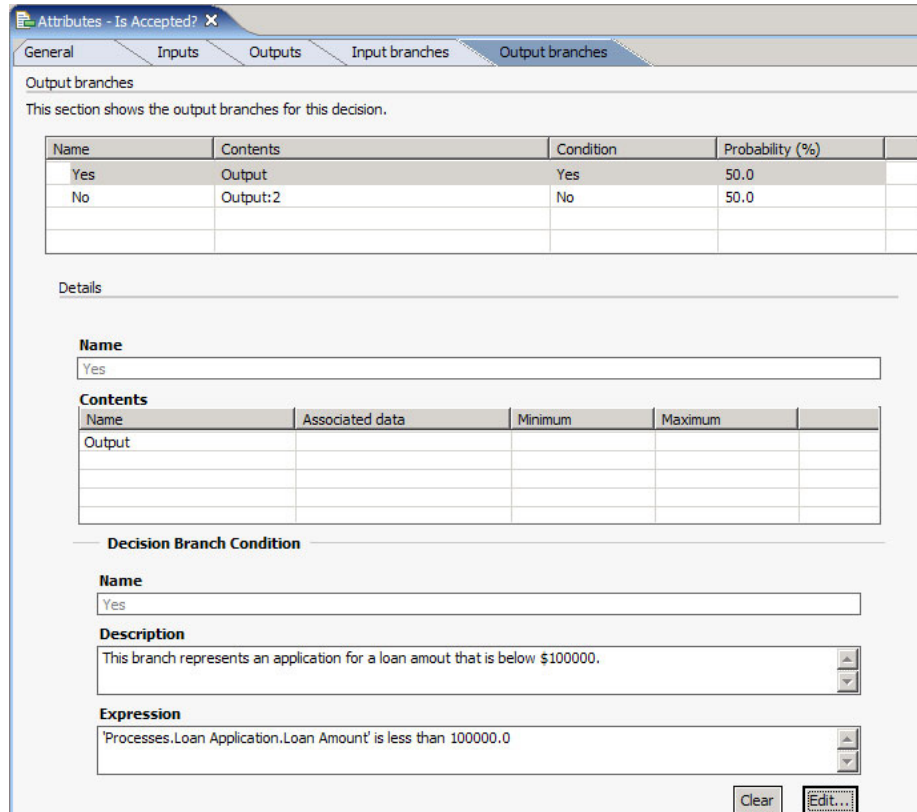
WebSphere Business Modeler automatically generates the expression for the No branch because the element is a simple decision and the flow can only go down one branch or the other. Select the No branch and take a look at its expression.

For more information on the Expression Builder, refer to the WebSphere Business Modeler documentation.

5. The Is Accepted decision has two branches that reflect the outcome of the decision. Connect the Yes branch to the Credit Approval task and then connect the Credit approval task to the VOE task.
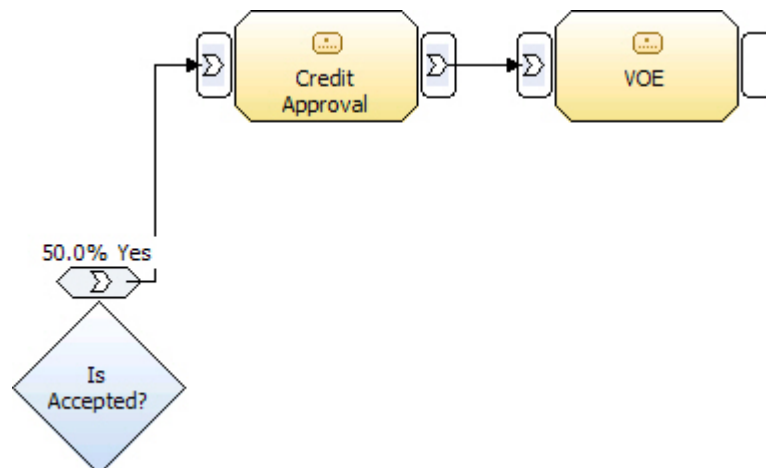
6. The VOE system calculates the cost of the loan. To determine the cost, it retrieves the default cost from the cost repository and then updates that cost. To reflect this in the model:

   a. Create a local repository named Cost.

> **Tip:** Refer to"Passing data into the process" on page 22 for information on how to create the repository.

  b. Change the type of data stored in the repository from String to Float.

> **Tip:** Refer to "Storing the data" on page 25 for information on how to associate data with the repository.

 c. Connect the repository to the input criterion of the VOE task.

The inputs for the VOE task should look something like this:
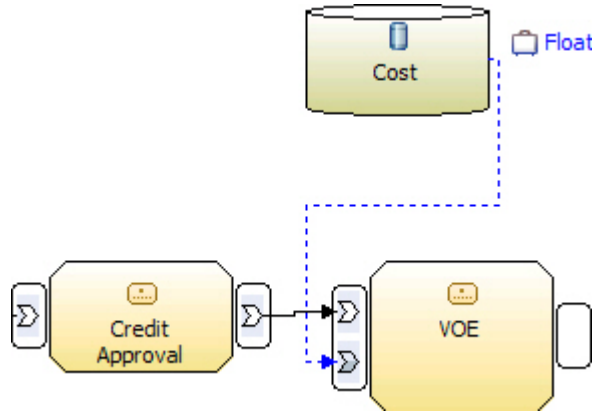
7. The VOE system updates the cost and it sets the Reject repository, which contains Boolean data (true or false). The VOE system also passes the flow to the next task, Evaluate Loan Application.

  a. Connect the VOE task to the Cost repository.

  b. Create the Reject repository and set its data type to Boolean.

  c. Connect the VOE task to the Reject repository.

 d. Connect the VOE task to the Evaluate Loan Application.

The outputs to the VOE task should look something like this:
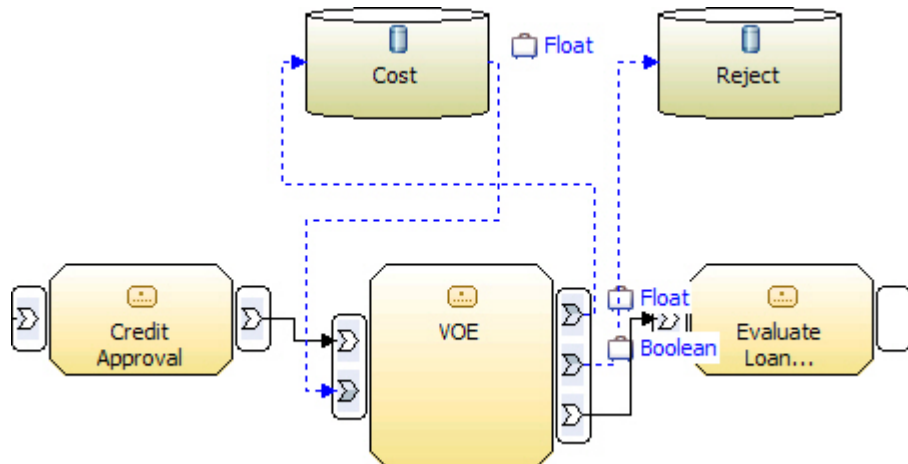
8. Add connections to the Evaluate Loan Application task.

  a. Add a connection from the Loan Amount repository to the Evaluate Loan Application task.

  b. Add a connection from the Cost repository to the Evaluate Loan Application task.

  c. Add a connection from the Evaluate Loan Application task to the Loan Amount repository.

d. Add a connection from the Evaluate Loan Application task to the Cost repository.

e. Connect the Evaluate Loan Application to the Need More Info decision.

These steps represent the underwriter using the loan amount and cost to make a decision on the loan. The result of the evaluation could include a change to the loan amount and the cost, which is what the latter two connections represent.
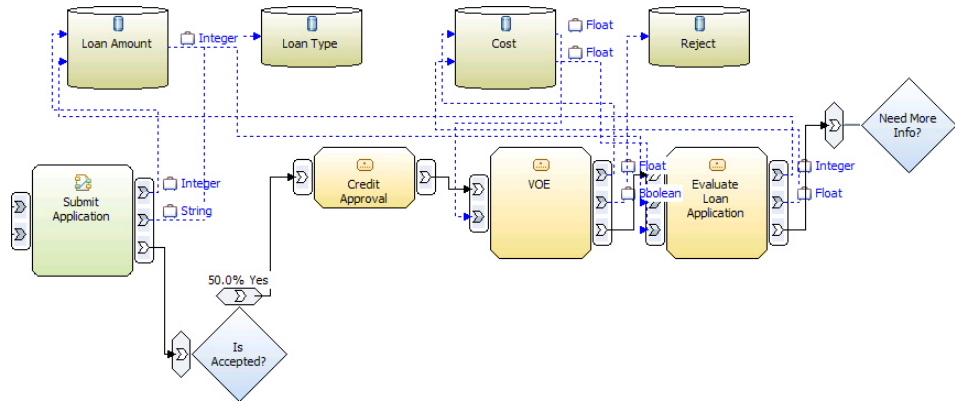
Tip: At this point, the diagram looks untidy and confusing. You can make it tidier by hiding the repository connections in the diagram. Right-click the diagram and select **Show** → **Repository Connections**.
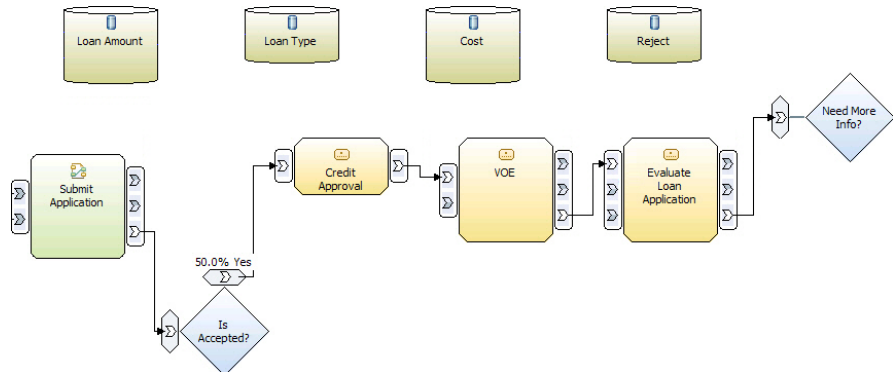

You can also use auto-layout to arrange the elements for you. To conserve space in the screen captures, the tutorial is not going to use the auto-layout option.
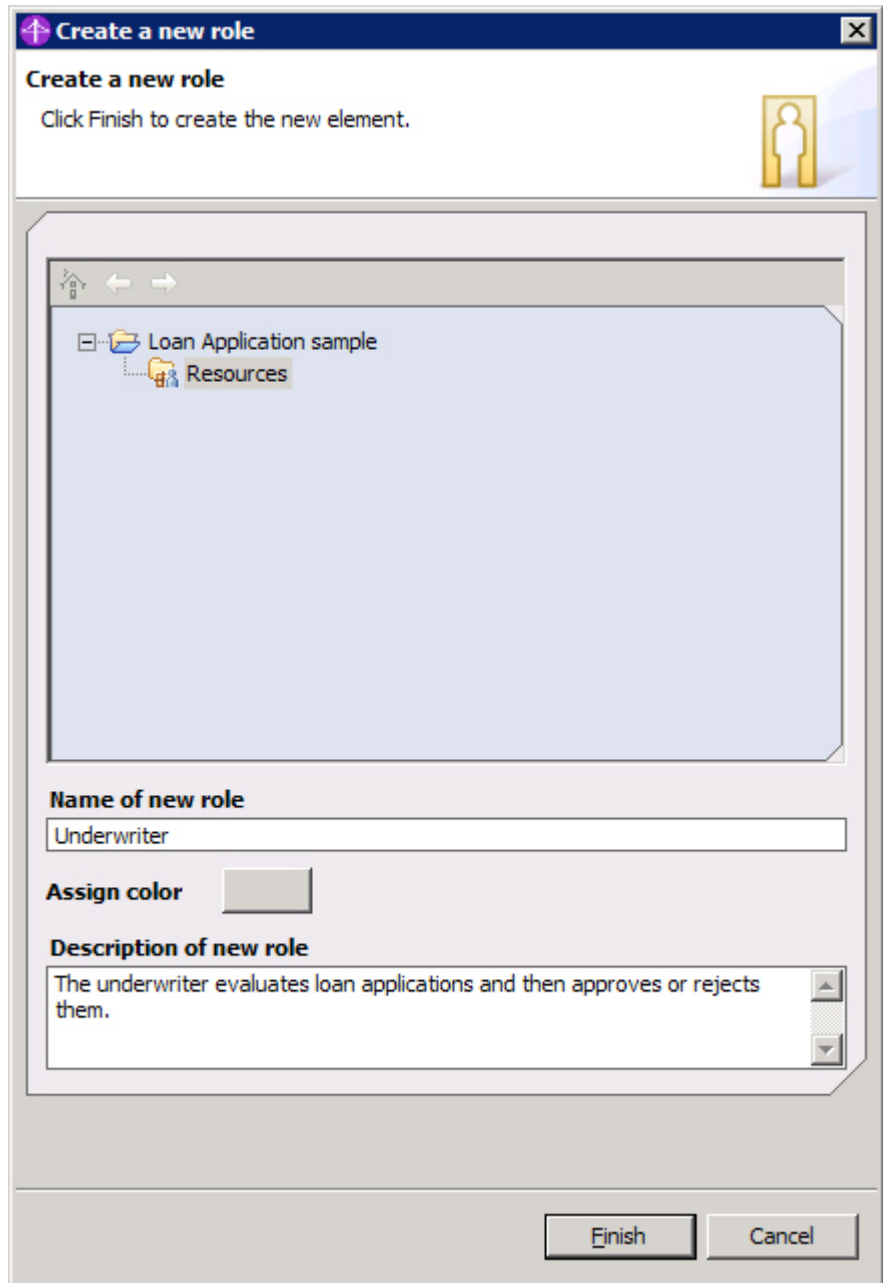
9. An underwriter performs the Evaluate Loan Application task. To add this information to the model:

a. Select the Evaluate Loan Application task.

b. In the Attributes view, click the **Resources** tab.

c. The person doing this task is performing the role of underwriter. Expand the **Role requirements** section and then click **Add**.

d. Click the Role cell. The cell displays a button.

	Name	Role	Time required	Quantity	Unit of mea...	Resource d...	
	Role requir...		0 seconds	1	units		

e. Click the button. The Select Role Requirement window opens.
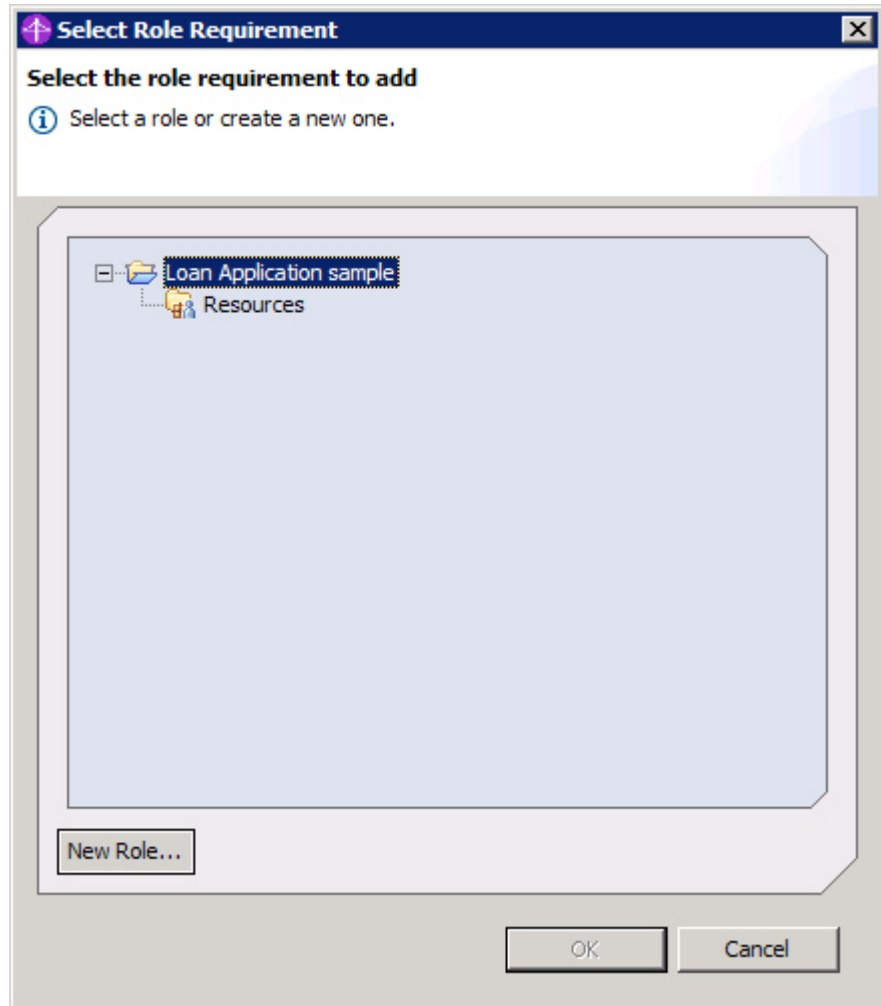
f. Click **New Role**. The Create a new role window opens.

g. In the **Name of new role** field, type: Underwriter

h. Click **Finish**. The Underwriter role displays in the resource tree of the Select Role Requirement window.

i. Click **OK**. In the attributes of the Evaluate Loan Application task, the role requirements table displays the Underwriter role.

|	Name	Role	Time required	Quantity	Unit of mea...	Resource d...	
	Role requir...	Underwriter	0 seconds	1		units	

10. The Need More Info decision has three possible results with varying degrees of probability: Yes, No, and Reject. To modify the decision and then connect its branches:

a. Right-click the Need More Info decision and select **Add branch**.

b. In the Attributes view, click the **Output branches** tab.

c. In the Output branches table, rename the branches to No, Yes, and Reject=true.

d. Change the probability of No to 60%, Yes to 10%, and Reject=true to 30%. When you created the Reject repository, you set its data type to Boolean. The VOE task sets the value held in the repository to true or false. This decision checks the repository value when processing the flow.

e. Add an expression to each branch. For example, for the first term, select the Reject repository, for the operator, select "is equal to," and for the second term, select Boolean and set the value to "false". Click **Apply**.

f. Connect the No branch to the Approved task.

g. Connect the Yes branch to the Add Missing Information task.

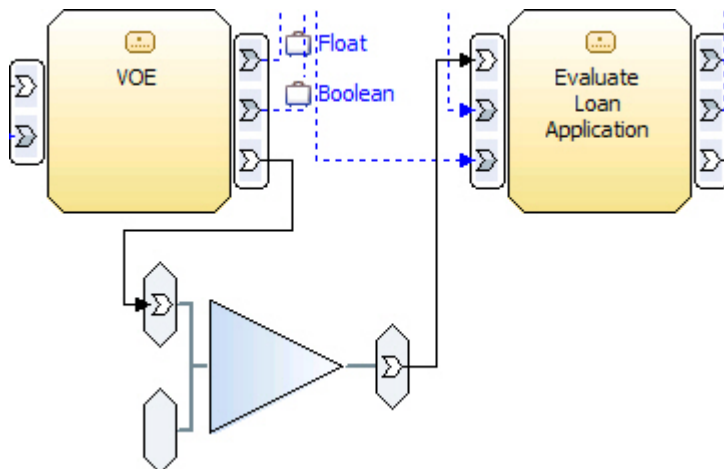h. Connect the Reject=true branch to the Rejected task.
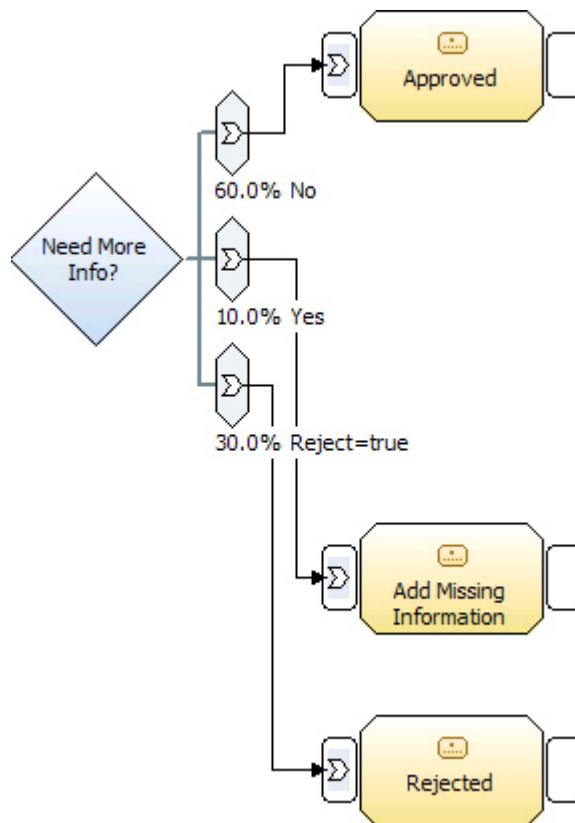
11. The output of the Add Missing Information task needs to feed back into the Evaluate Loan Application. You need to combine the output of the Add Missing Information task with the output of the VOE so that the Evaluate Loan Application task will happen. If you were to directly connect the Add Missing Information task to the Evaluate Loan Application, the Evaluate Loan Application waits for the output of the VOE task and the output of the Add Missing Information task. The latter output will never occur and the process will be stuck. The solution for combining the two outputs is a merge element.

a. Right click where you want to place the merge and select **New → merge**.

Tip: You may want to move some elements to make room for the merge.
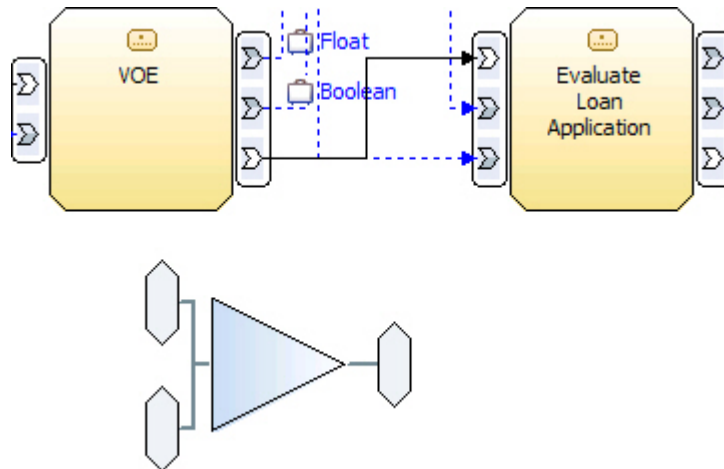

b. Select the connection between the VOE task and the Evaluate Loan
   Application task and press Delete.

c. Connect the empty output of the VOE task to the top branch of the merge
   and then connect the output of the merge to the empty input of the
   Evaluate Loan Application task.

12. Connect the output of the Add Missing Information task to the lower branch
    of the merge.
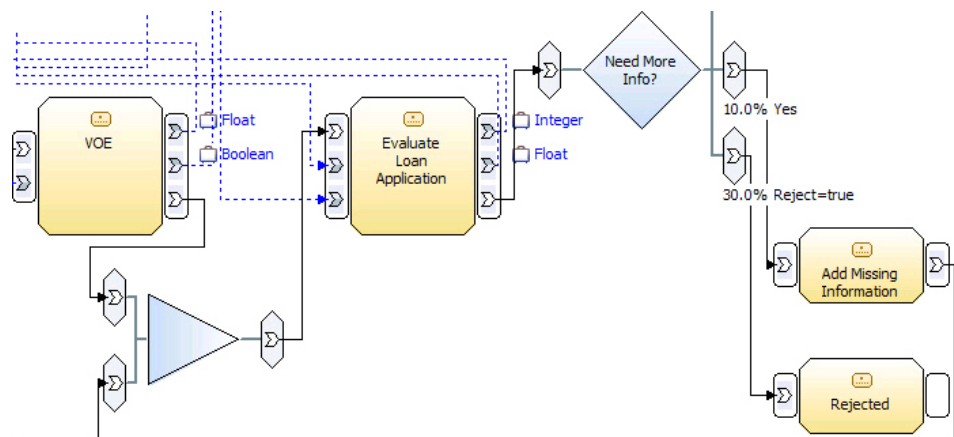
13. The No branch of the Is Accepted decision needs to connect to the Rejected task. The Reject=true branch of the Need More Info decision is already connected to the rejected task. In the same way that you added the merge for the input to the Evaluate Loan Application, add a merge and connect the No branch to it. Delete the Reject=true branch connection to the Rejected task and connect the Reject=true to the new merge. Finally, connect the output of the new merge to the Rejected task.
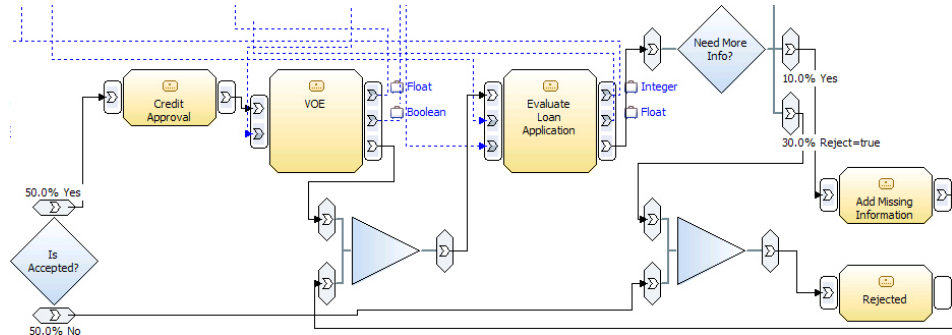
14. The Approved task and the Rejected task represent to possible paths and each one is also the end of the process. To model this, connect the Accepted task to the closest end node. Connect the Rejected task to the other end node The completed process should look something like this:
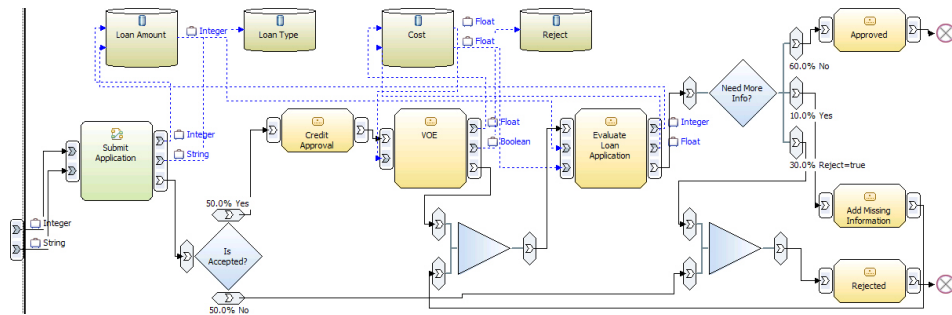
Lesson checkpoint

You have created a process that is ready for importing into FileNet P8.

In this lesson, you have learned how to model a process in WebSphere Business Modeler in a way that optimizes the amount of information passed to FileNet P8. The lesson demonstrated:

- How to import data into a process in FileNet Business Process Manager mode
- How to add and connect process elements
- How to modify element attributes

Lesson 2: Opening the process in FileNet P8 Process Designer

In this lesson, you export the loan application process from WebSphere Business Modeler and then open it FileNet P8 Process Designer.

Lesson 1 taught you how to create a simplified loan application process. If you did Lesson 1, you can use the loan application process you created in that lesson as the base for what you do in this lesson. If you did not do Lesson 1 or you did not complete it, you can import the loan application sample and use it instead.

To import the loan application sample, complete the following steps:

1. From the File menu, select **File → Import**. The Import wizard opens.
2. Select **WebSphere Business Modeler Import** and then click **Next**.
3. Select **WebSphere Business Modeler project** and then click **Next**.
4. Browse to the location of the Loan Application.mar file.
5. Beside the Target project field, click **New**. A wizard opens.
6. Rename the project to a name like `Loan Application sample` and then click **Finish**. The wizard creates the project and then closes. The Target project field in the Import wizard now contains the name of the project you created.
7. Click **Finish**. The Import wizard imports the sample loan application process into the project you specified.

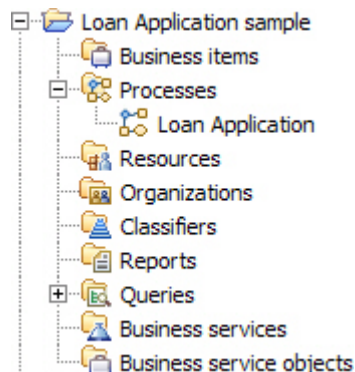Lesson 2 consists of the following topics:

- "Exporting the loan application process"
- "Opening the process in FileNet P8 Process Designer" on page 40

Exporting the loan application process

Before you begin this lesson, ensure that you have a version of the loan application process ready to export. You can use the process created in Lesson 1 or you can use the Loan Application sample. You must also be able to access theFileNet P8 Process Designer.

To export the loan application process, complete the following steps:

1. If you cannot see the Project Tree view, switch to the four-pane layout ⊞ . The Project Tree view opens and displays the Loan Application sample project. Within the Loan Application sample is the Loan Application process.

2. Expand **Loan Application sample → Processes**.
3. Right-click the process and select **Export**. The Export wizard opens.
4. Select **FileNet Business Process Manager** and then click **Next**. The "Destination and source" page opens.
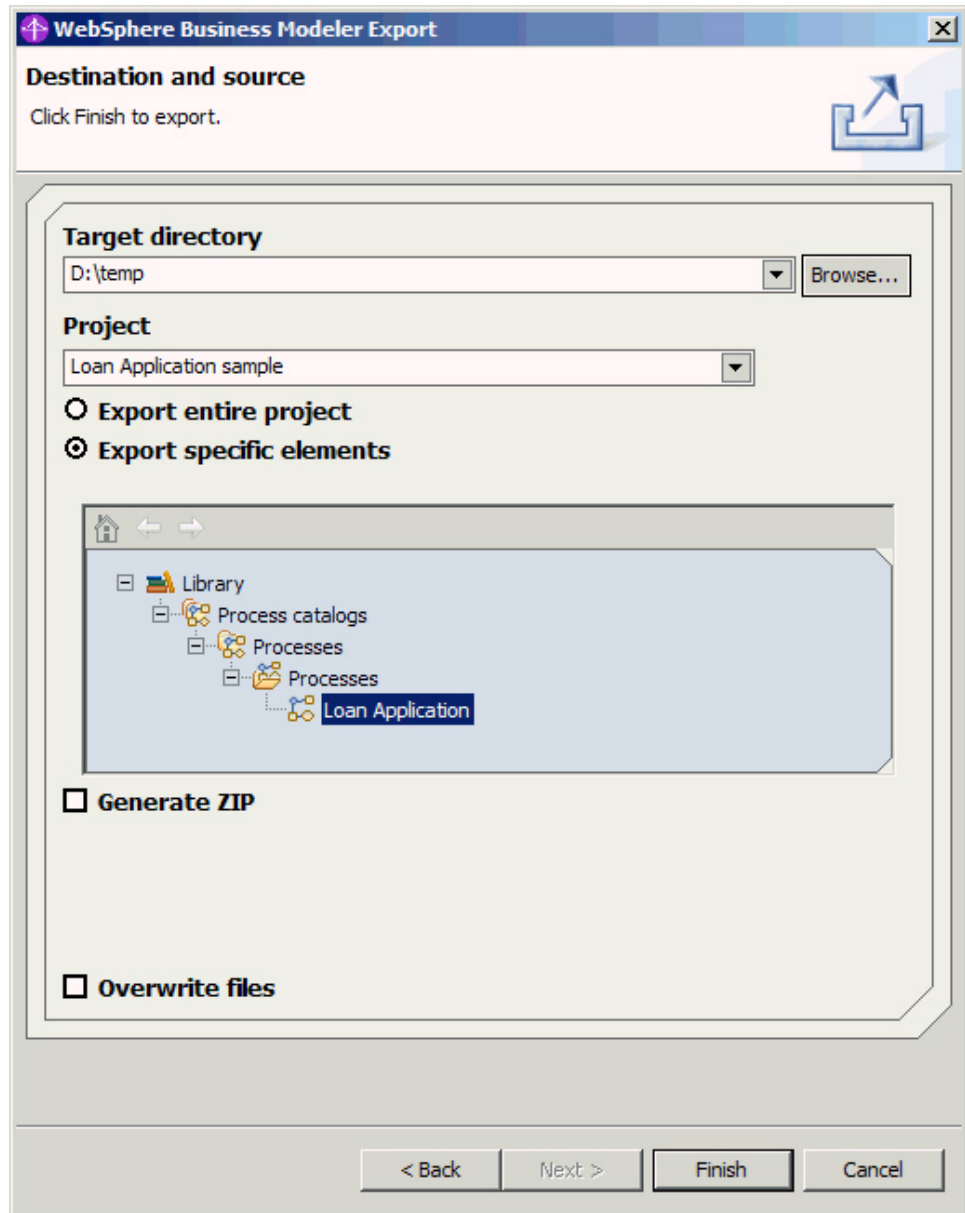
5. For the Target directory field, browse to where you want to place the exported file. In the page, the project and the process are already identified.

6. Click **Finish**. The wizard exports the process into a file named after the project.

Opening the process in FileNet P8 Process Designer

When you exported the Loan Application process in "Exporting the loan application process" on page 39, you created the Loan Application sample.xpdl file, which you can open using FileNet P8 Process Designer.

To open the XPDL file in FileNet P8 Process Designer, complete the following steps:

1. In the FileNet P8 Workplace™, expand the **Author** node and select **Advanced Tools**.

2. Click **Process Designer**.

3. From the File menu in the Process Designer, select **File → Open**
4. In the window that opens, browse to the Loan Application sample.xpdl and click **Open**.

The process will look something like this:
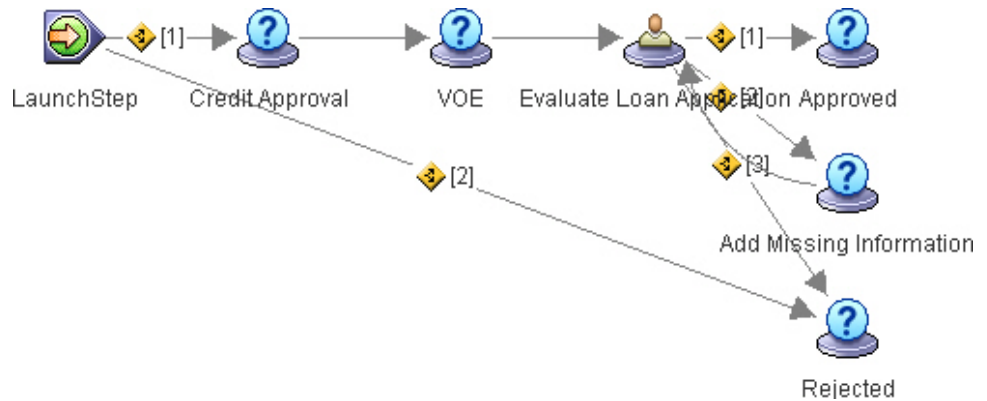
Tip: Rearrange the elements in the diagram to make them easier to distinguish and read.
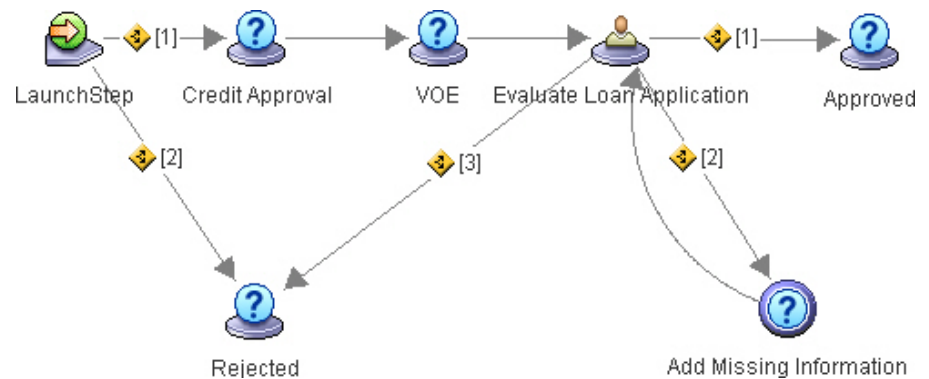
If you select the Evaluate Loan Application step, you can see in the **General** properties of the step that the description for the model element has been transformed into the instruction for the step. The Underwriter role in WebSphere Business Modeler has been transformed into the workflow group called Underwriter, which is acting as the participant who processes this task. The Evaluate Loan Application step is the only step in this process with an assigned participant.

On the **Parameters** tab, you can see that the repositories have been transformed into the Cost and LoanAmount data fields for this step:
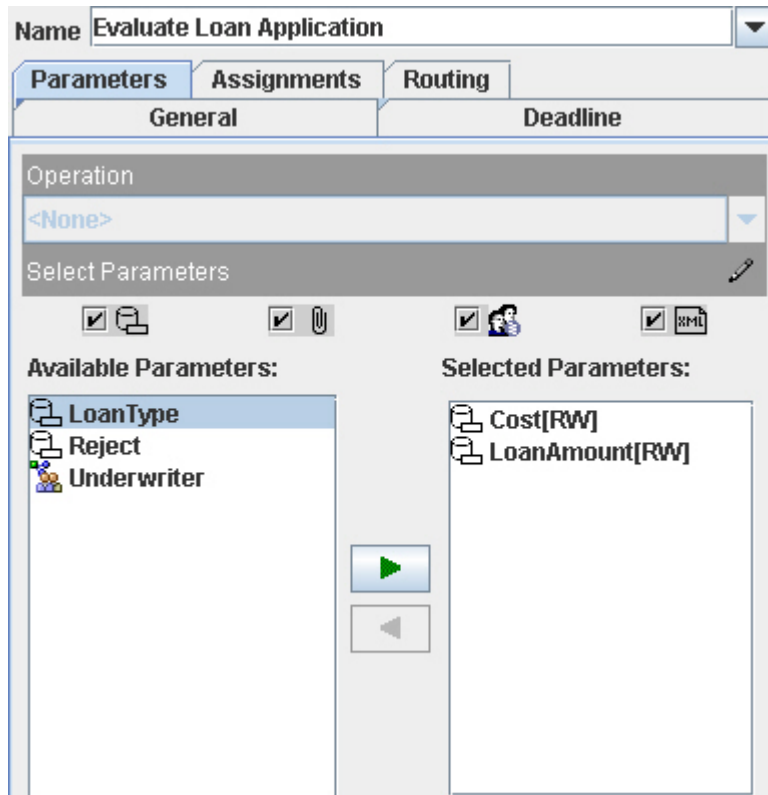
If you select a route with a number, such as the route [1] from the launch step, you can see that the expression ('Processes.Loan Application.Loan Amount' is less than 100000.0) on the Yes branch of the Is Accepted decision has been transformed into the routing condition for the route (LoanAmount < 100000):
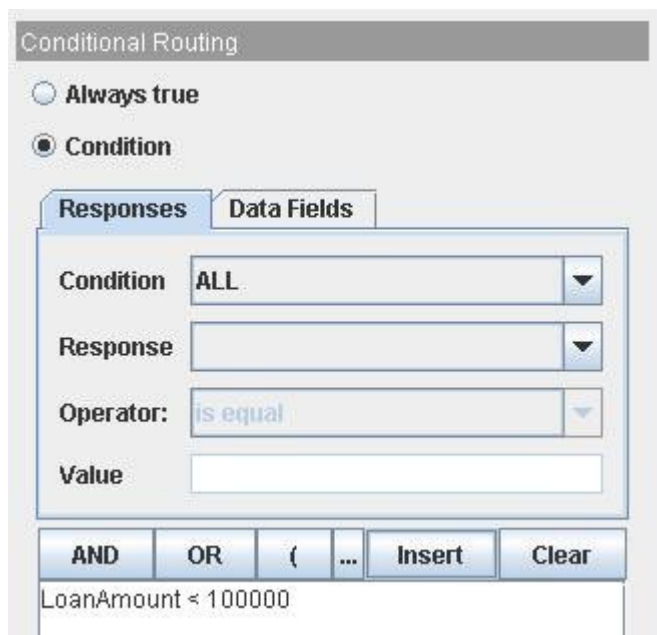
At this point, the process will validate successfully, but you can use the Process Designer to further enhance the process by assigning participants or queues to other steps and by adding other implementation details. You will also need to

verify that Underwriter is a valid user in the current environment. For additional information, refer to the FileNet P8 documentation for Process Designer.

Lesson checkpoint

You have exported a WebSphere Business Modeler process and opened it inFileNet P8 Process Designer.

In this lesson, you have learned:
* How to export a process as an XPDL file from WebSphere Business Modeler using the FileNet Business Process Manager export.
* How to open the XPDL file in FileNet P8 Process Designer.
* How elements and attributes in WebSphere Business Modeler transform into steps, routes, and properties inFileNet P8 Process Designer.

Lesson 3: Implementing the process in FileNet P8

In this lesson, you create an implementation of the process that will run in the FileNet P8 Process Engine.

Lesson 2 taught you how to export the simplified loan application process from WebSphere Business Modeler and open the resulting file in FileNet P8 Process Designer. If you did not do Lesson 2 or you did not complete it, you can use the Loan Application sample.xpdl file that accompanies the sample MAR file when you reach the steps in the tutorial that require the XPDL file. If you completed Lesson 2, save your process and close Process Designer.

Lesson 3 consists of the following topics:
* "Configuring the workflow environment"
* "Integrating the process with the workflow environment" on page 48
* "Validating and saving the process definition" on page 52

Configuring the workflow environment

You are now ready to set up the workflow environment for the loan application process from the sample. For this tutorial, you will be defining the workflow queues used by the process. To demonstrate the steps in this lesson, the tutorial uses a VW host server called p8platform and an isolated region called P8PlatformCP. If you following the tutorial using your own host and region, substitute your names for these names.

To configure the workflow environment, complete the following steps:
1. In the Workplace, click **Admin**. In the Admin page, click **Process Configuration Console**. The Process Configuration Console opens.
2. Connect to the isolated region that will host the loan application process. For example, double-click the isolated region node called P8PlatformCP. The console connects to the region and displays its contents.
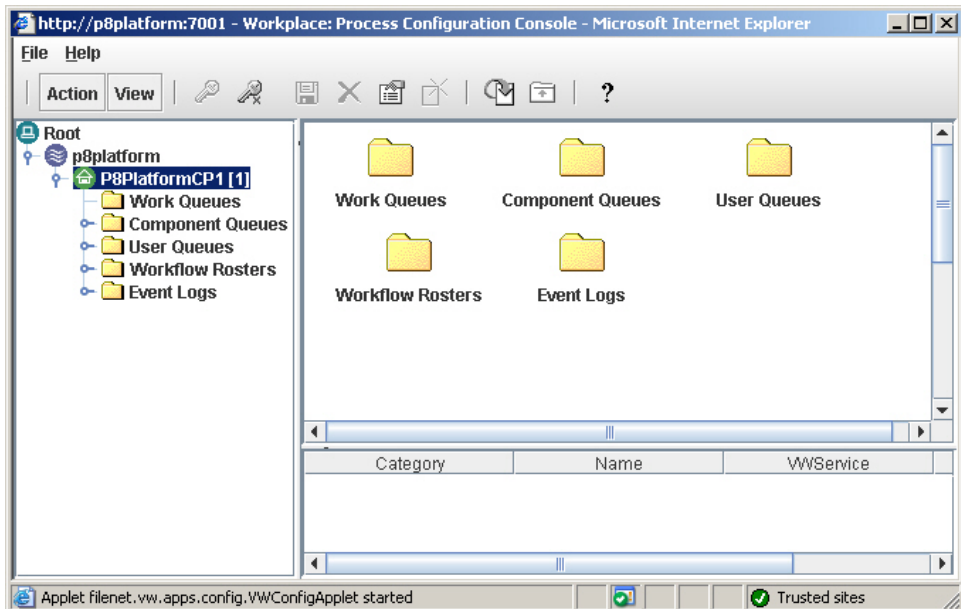
3. Now you are ready to define the work queues for the loan application process.

   a. Right-click the **Work Queues** node and select **New**. The New Queue window opens.

   b. In the **Queue Name** field, type: LoanProcessor
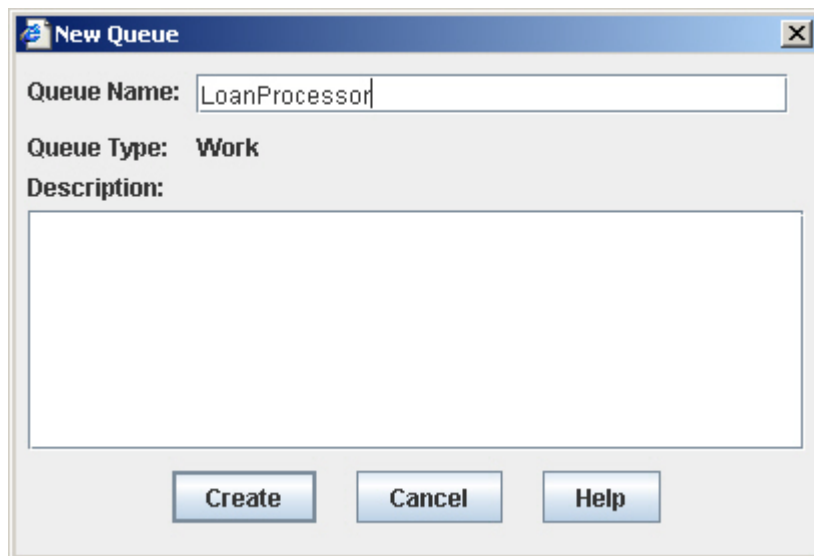
   c. Click **Create**.

   d. Repeat steps a-c to create a work queue called CreditApproval.
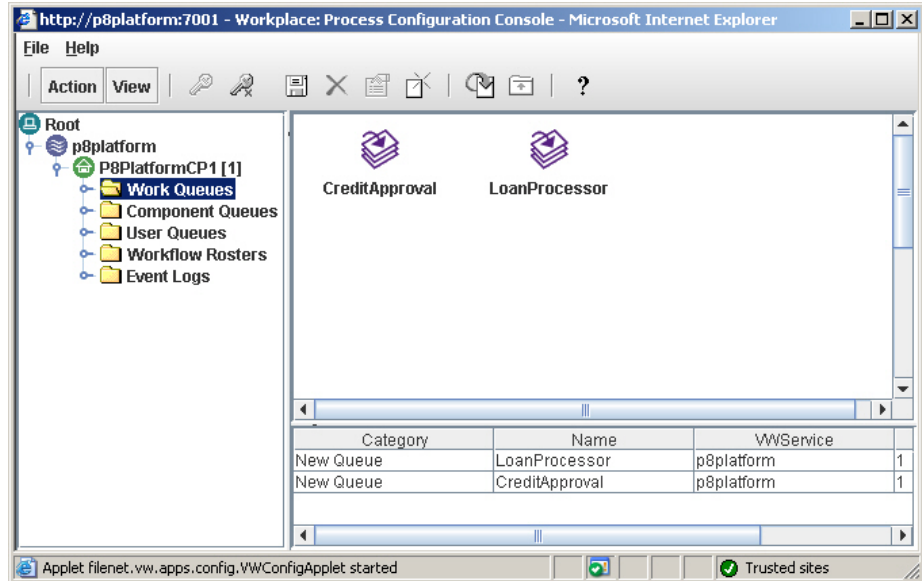
4. With the work queues defined, you can define the operations associated with the work queues. Operations contain the parameters that define the interface for components that are processing work in the work queue. For the loan application process, the LoanProcessor needs to have an operation to handle approved loans and an operation to handle rejected loans.

   a. Right-click the LoanProcessor queue and select **Properties**. The Queue Properties window opens.
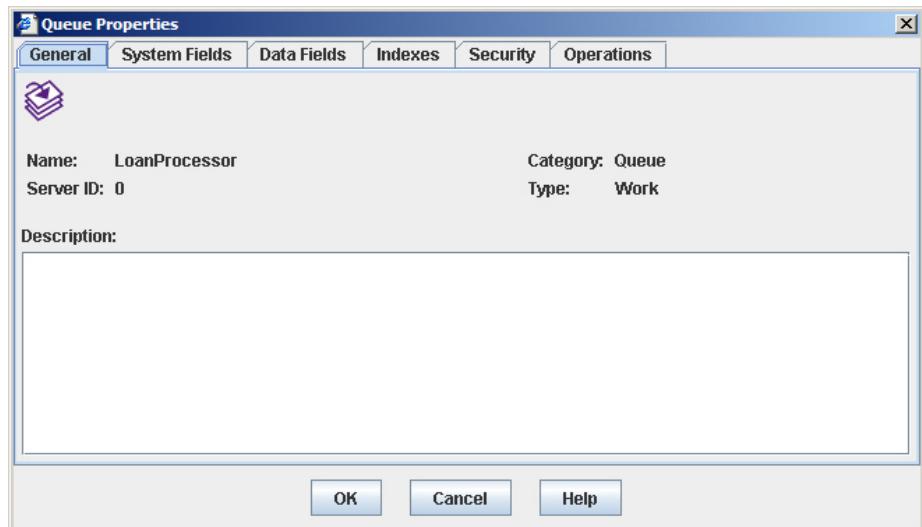
   b. Click the **Operations** tab.

   c. Create an operation called ApproveLoan by selecting the first available line in the Name box, typing the name, and then pressing the Enter key.
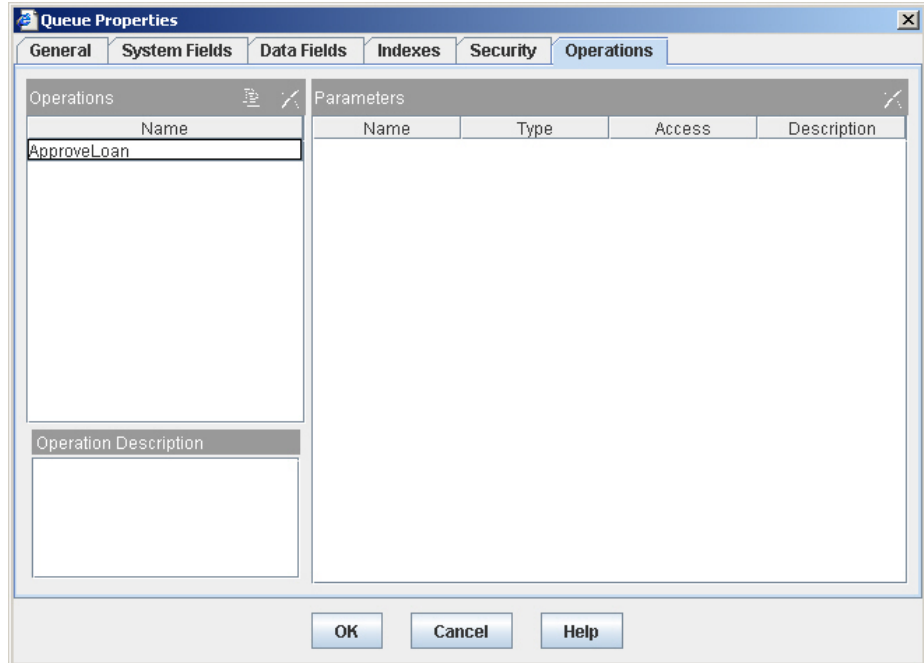
d. Create another operation called RejectLoan.

e. Select the ApproveLoan operation and then create the following parameters:

Name	Type	Access
LoanAmount	Integer	Read
LoanType	String	Read
Cost	Float	Read

The parameters for the ApproveLoan operation should look like this:
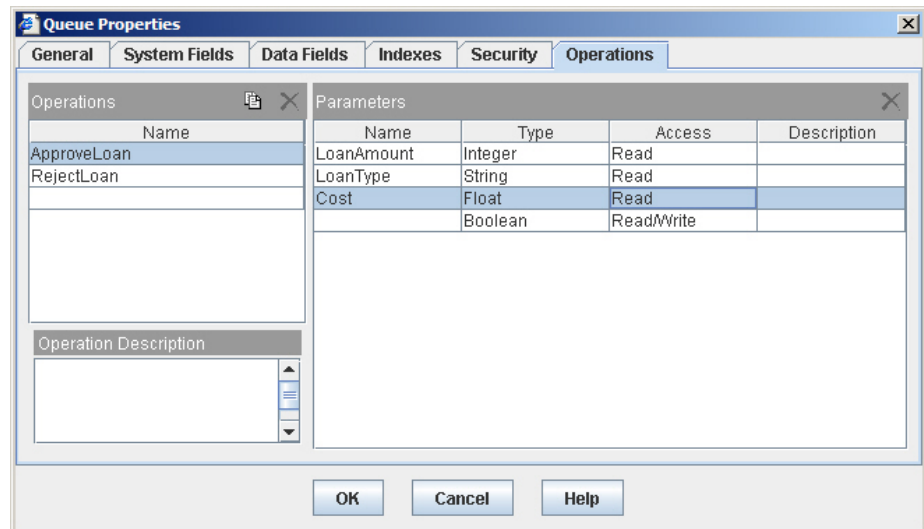
f. Select the RejectLoan operation and then create the following parameters:

Name	Type	Access
LoanAmount	Integer	Read
LoanType	String	Read

g. Click **OK**.

5. Now that you have created the work queues and their operations, you can commit these changes. In the Process Configuration Console, click the Commit Changes button in the toolbar. The Commit Changes window opens.
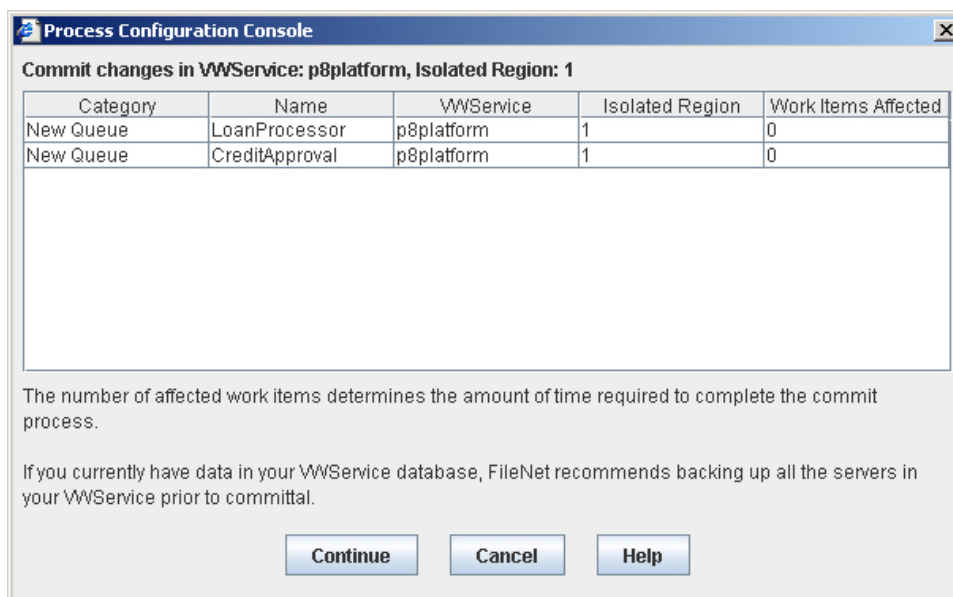
6. Click **Continue**.
7. After the console has committed the changes, click **Close**.
8. Close the Process Configuration Console by selection **File** → **Exit**.

You have now configured the workflow environment. You are now ready to integrate the process with the workflow environment.

Integrating the process with the workflow environment

You are now ready to integrate the loan application process from the sample with the workflow environment that you created.

This part of the lesson builds upon the example in "Configuring the workflow environment" on page 44 using the same names . If you following the tutorial using your own workflow environment, substitute your names for the names contained the steps below.

To integrate the process with the workflow environment, complete the following steps:

1. In the FileNet P8 Workplace, expand the **Author** node and select **Advanced Tools**.
2. Click **Process Designer**.
3. From the File menu in the Process Designer, select **File** → **Open**
4. In the window that opens, browse to the Loan Application sample.xpdl and click **Open**.
5. Assign steps to work queues.
   a. Select the **Credit Approval** step.
   b. In the Step Destination section of the Properties, select the **Work Queue** radio button. In the drop down list, select **CreditApproval**. The icon for the

step in the Workflow changes because the step has been assigned to the CreditApproval workflow.
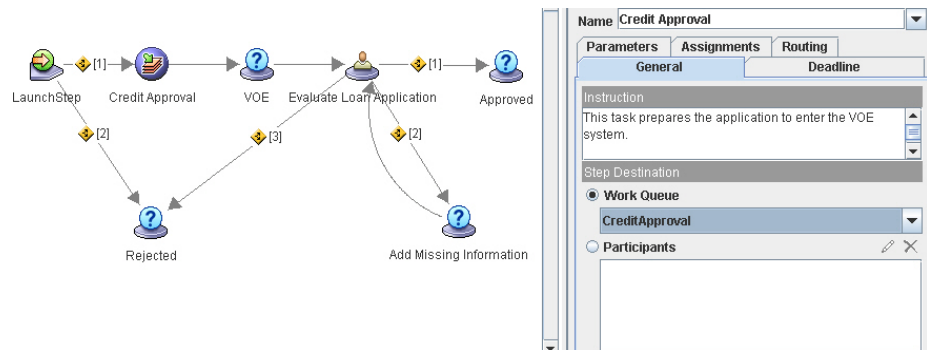
c. Using the same technique, assign the Add Missing Information step to the CreditApproval workflow.

d. Assign the Rejected step to the LoanProcessor workflow.
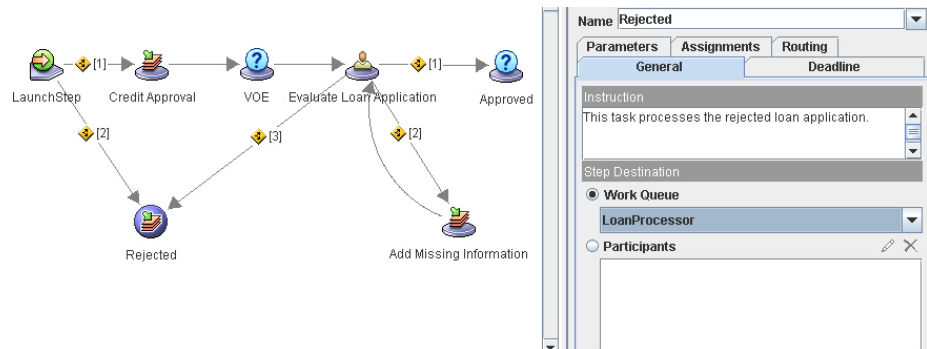
e. In the workflow environment, there is a RejectLoan operation in the LoanProcessor work queue. To assign the step to the operation, click the **Parameters** tab.

f. From the Operation drop down list, select the RejectLoan operation. The parameters for the operation display. The values for the parameter expressions should be automatically defined because the loan process contains data fields with the same name and type. If they are not defined, set the expression for each parameter to the matching data field.
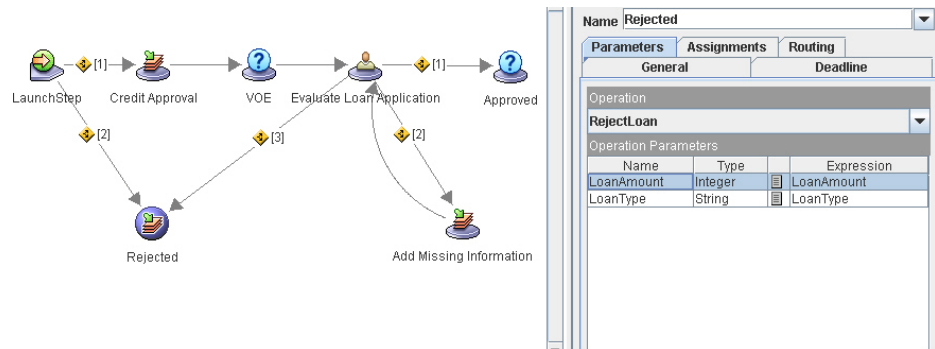
g. Assign the Approved step to the LoanProcessor workflow using the General properties page.

h. Assign the Approved step to the AcceptLoan operation using the parameters page.
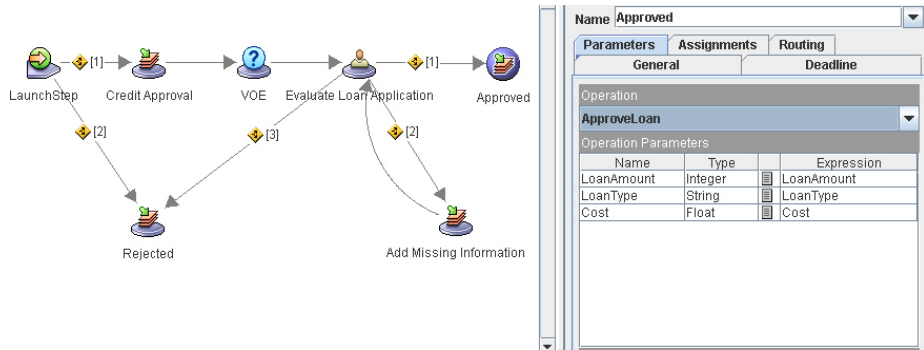
6. In the loan application process, one of the things the VOE task does is update the cost of the loan. For example, say the cost of the loan is equal to 15% of the loan amount. To do this in the workflow:

   a. Select the **VOE** step.

   b. In the properties of the VOE step, select the **Assignments** tab.

   c. Click the empty cell in the Field Assignments and select **Cost** from the drop down list.
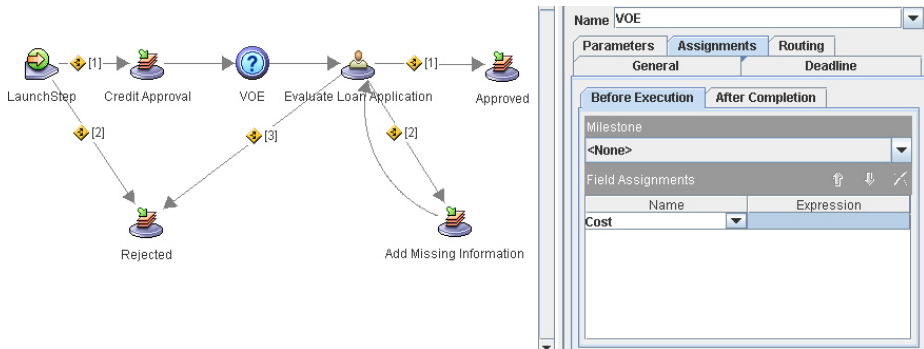
   d. In the Expression cell, type: `LoanAmount*0.15`

   Note: An alternative implementation would be to use a data field to store the percentage so that it could be updated when the process runs.

   Tip: You can also click the **...** button to open the Expression Builder to create the expression.

7. The loan application process has the Underwriter workflow group assigned as a participant, which you can see if you select the Evaluate Loan Application step and then select the General tab. You need to assign a user as a participant of the Underwriter workflow group.

   a. In the toolbar, click the Workflow Properties button [icon]. In the Workflow Properties window that opens, select the **Workflow Groups** tab.
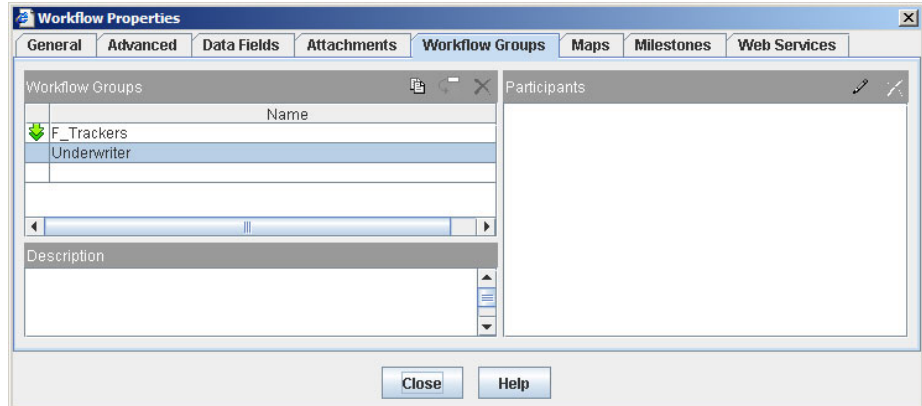
b. Select the Underwriter workflow group and then click the Modify button ✎ in the Participants section. The Participant Selection window opens.
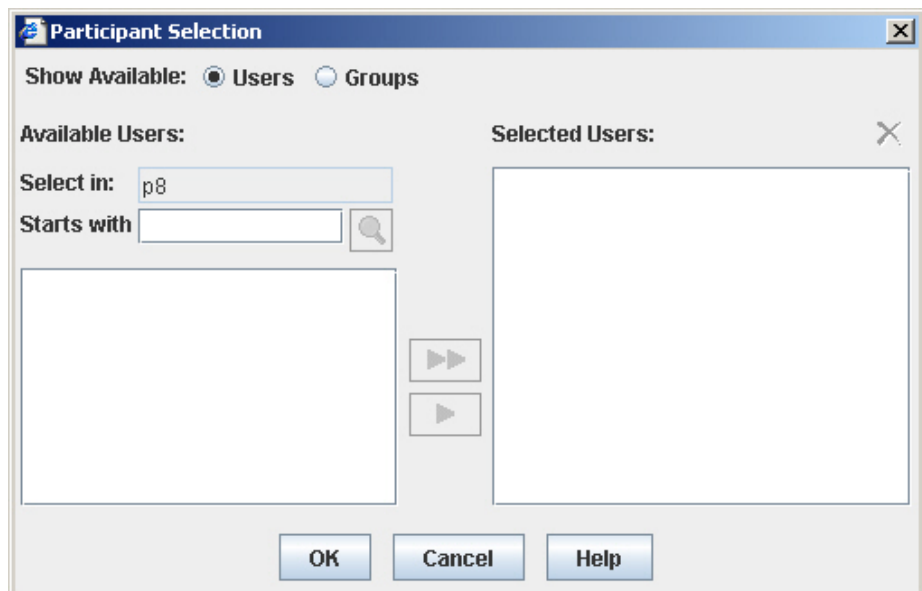
c. In the **Starts with** field, type: admin The Search button enables.

d. Click the Search button. The Administrator user displays in the search results.
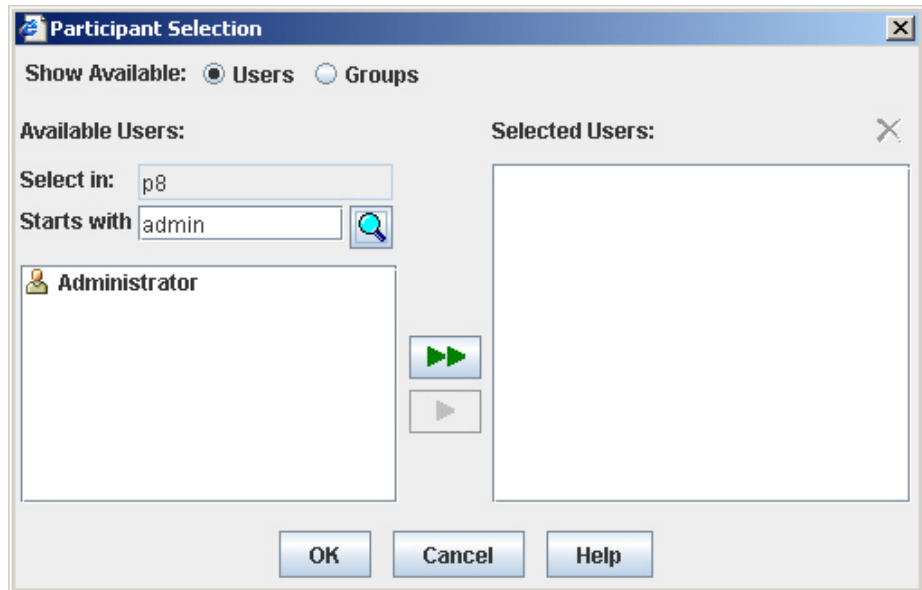
e. Select the Administrator user and then click the double arrow button to move the Administrator user into the **Selected Users** field.

f. Click **OK** In the Workflow Properties window, the Participants section now lists the Administrator user.
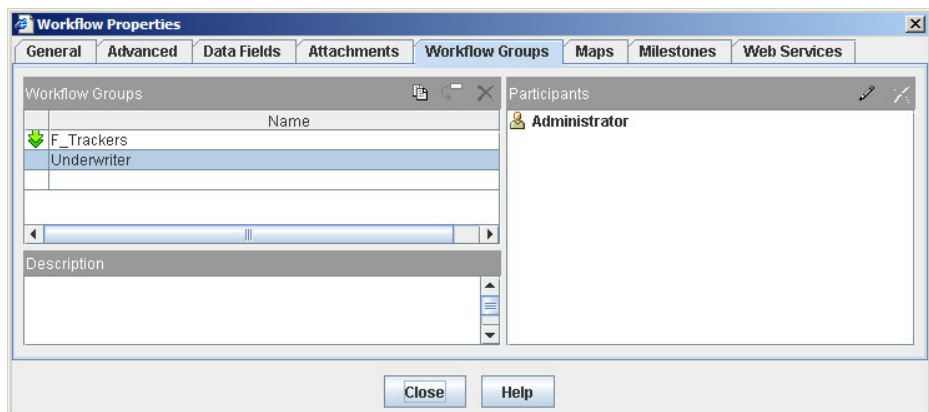
g. Click **Close**.

You have now completed integrating the process with the workflow environment. You are now ready to save the process definition

Validating and saving the process definition

With the workflow environment defined and the steps in the process associated with the appropriate work queues, operations, and participants, you are ready to validate the process and then save the process definition.

To validate the process and then save the process definition, complete the following steps:

1. At this point, you have fully defined the process. You should now validate it to check for problems. Select **File → Validate**. After the validation completes, you will see a message window informing you of the results of the validation. Close the message window.

> **Tip:** If your receive a validation error, check that the operation parameters for the Approved and Rejected steps have expressions. The expressions should match the name of the parameters.

2. With a valid process, you are now ready to deploy it.

   a. Select **File → Transfer**. The wizard that saves the workflow definition opens.
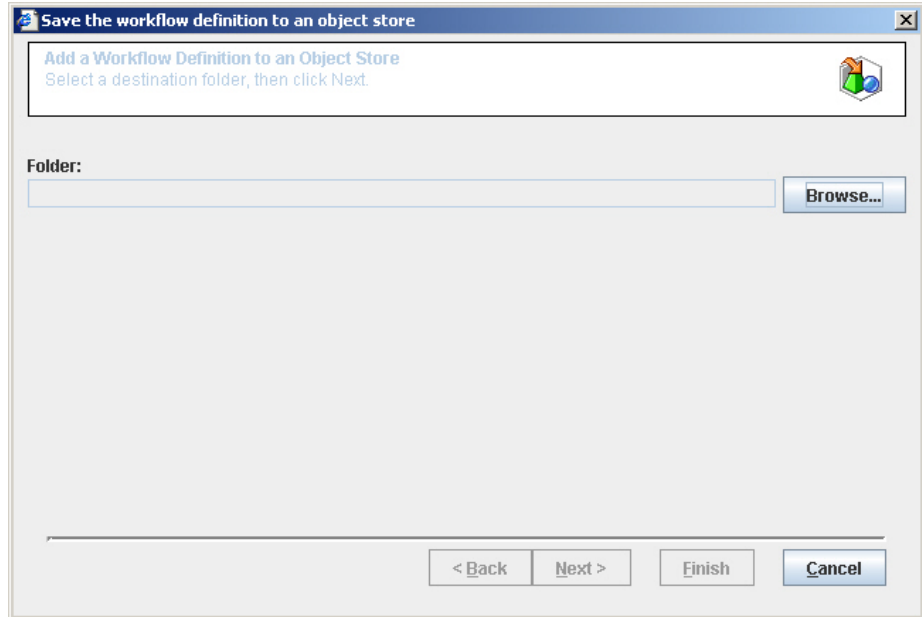
   b. Browse to where you want to save the workflow definition. If you need to, you can create the folder. If you created a folder, select the folder and then click **Select** to close the Select a Folder window.
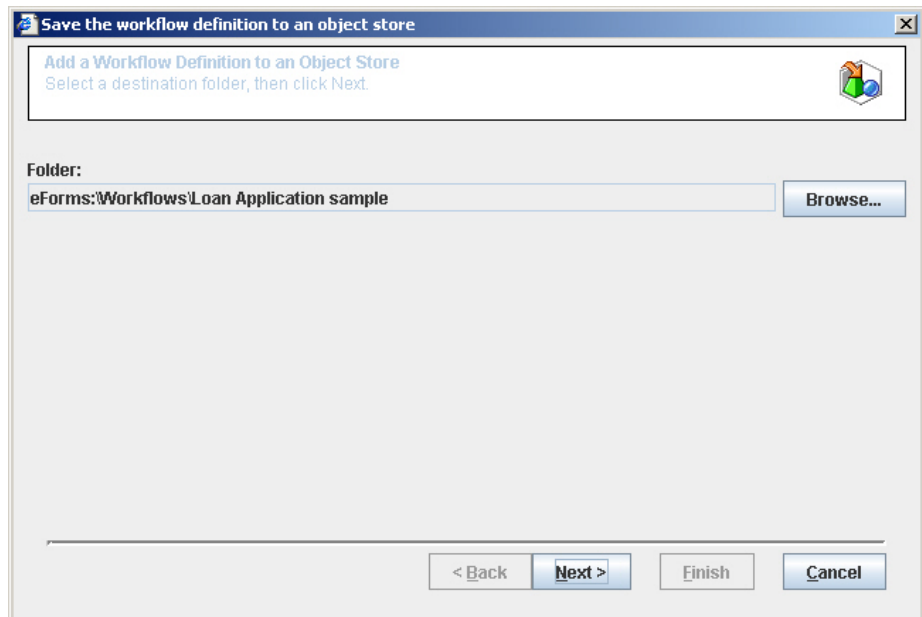
   c. Click **Next**.
   d. In the Document Title field, type: Loan Application
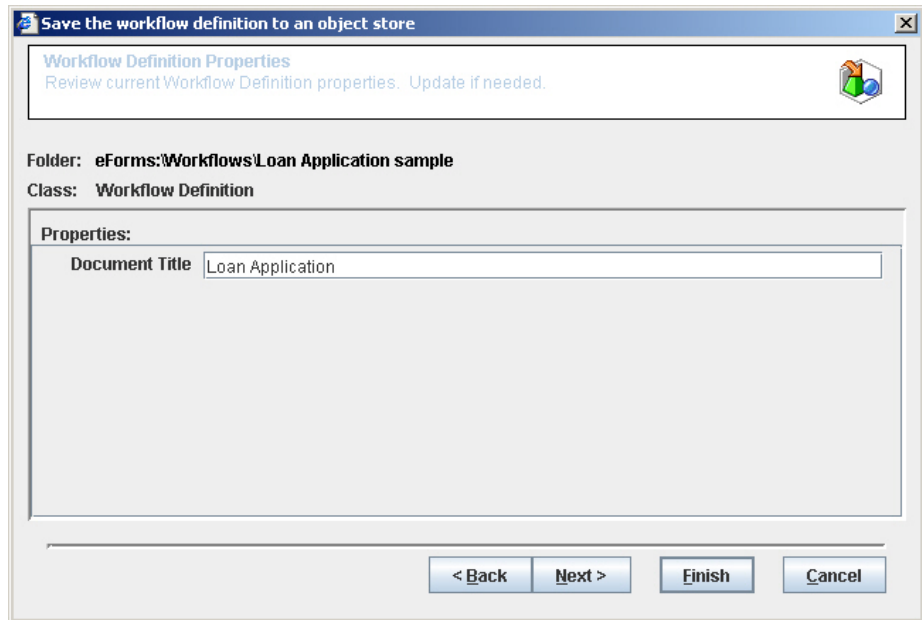
This step names the file that contains the workflow definition. The wizard places the file in the folder you specified. The document has default security applied.

   e. Click **Finish**. When the wizard has completed saving the document, it opens a message window with the results of the save. Click **Close**.

3. Close the Process Designer by selecting **File → Close**. A window opens prompting you to check in your workflow definition.

4. Because you have just saved the workflow definition, you do not need to check it in. Select **Cancel the checkout** and then click **OK**.

The Loan Application process is now deployed and is ready for launching and processing. For information on launching and processing, refer to the FileNet P8 documentation.

Lesson checkpoint

You have created and saved a process definition that is ready for launching and processing.

In this lesson, you have learned how to implement the process in the FileNet P8 Platform. The lesson demonstrated:

- How to set up the workflow environment

- How to integrate the process with the workflow environment
- How to validate the process, and then save the process definition

XPL messages (FileNet Business Process Manager export)

XPL messages can appear when you are exporting using the FileNet Business Process Manager export.

XPL010100W The "*element_or_attribute_name*" is not supported. It was not exported.

Explanation:
The FileNet Business Process Manager mode does not support this element or attribute. When you import the process into FileNet P8, the element or attribute will not be in the workflow.

User response:
Create the equivalent to the missing element or attribute in FileNet P8 using the Process Designer.

XPL010110W The "*element_name*" is not supported. A generic XPDL Activity was generated.

Explanation:
The FileNet Business Process Manager mode does not support this element.

XPL010120E There is more than one process called "*process_name*" being exported. Process names must be unique when you are exporting using the FileNet Business Process Manager export.

Explanation:
While WebSphere Business Modeler allows you to reuse names within separate process catalogs, each process name must be unique within an XPDL file. When you use the FileNet Business Process Manager export, it flattens the structure you created in WebSphere Business Modeler when it generates the WorkflowProcess elements in the XPDL file. If the export encounters two or more processes with the same name, it cancels the creation of the XPDL file and generates this error.

User response:
Rename processes until each process has a unique name.

XPL010140W The file "*file_name*" already exists and has not been overwritten.

Explanation:
The export detected that a file by that name already exists in the directory. When you were asked whether to overwrite the existing file, you chose to not overwrite it.

User response:
Save the XPDL file into a different directory.

XPL010150W An error occurred while saving the file "*file_name*."

User response:
Make sure that the file is not locked or in use and that you have the correct permissions to write to the directory you are trying to save to, then try to export again.

XPL010160W An error occurred while transforming the condition on "*element_name*" in "*process_name*." The condition was not mapped.

User response:
Create the missing condition in FileNet P8 using the Process Designer.

XPL010170W Synchronized call to process "*process_name*" has been partially exported. You will need to complete the call in FileNet P8.

Explanation:
WebSphere Business Modeler can only provide a partial implementation of the call.

User response:
To complete the call in FileNet P8:

1. Display the called workflow process by clicking **View** → **Edit Workflow** → *process name*.

2. In the Workflow Properties window, click the **Advanced** tab.

3. Specify a unique identifier expression in the **Condition Identifier** field.

XPL010190W The name "*name*" exceeds the maximum allowed length of *name_length_maximum*. The name was truncated.

Explanation:
FileNet P8 imposes a limit on the number of characters that a name can have. The maximum number of characters depends on the type of the FileNet P8 element. To prevent elements from exceeding the maximum, the FileNet Business Process Manager export truncates the names at the following maximums:

XPL010200W

FileNet P8 element	Maximum number of characters
Activity or connection name	246
Data field name	27
Role name	20

These maximums are at three characters less than the FileNet P8 maximum to allow for the appending of a number used to ensure that exported elements have unique names.

XPL010200W The name "*name*" is not unique. It was replaced with "*name_length_maximum*."

Explanation:
The FileNet Business Process Manager export has processed another element has the same name. Because names must be unique in the XPDL file, the export has renamed the element to the specified name.

ZXP messages (FileNet Business Process Manager mode)

ZXP messages can appear when you are modeling in the FileNet Business Process Manager mode.

ZXP100001E **Notification broadcaster**
"notification_broadcaster_name" **in**
"process_name" **is not supported.**

Explanation:
The FileNet Business Process Manager mode does not support notification broadcaster elements.

User response:
Delete the notification broadcaster. If you do not delete the notification broadcaster and import the process into FileNet Business Process Manager, the notification broadcaster displays as a connected general step.

ZXP100002E **Notification receiver**
"notification_receiver_name" **in**
"process_name" **is not supported.**

Explanation:
The FileNet Business Process Manager mode does not support notification receiver elements.

User response:
Delete the notification receiver. If you do not delete the notification receiver and import the process into FileNet Business Process Manager, the notification receiver displays as a connected general step.

ZXP100003E **Timer** *"timer_name"* **in** *"process_name"* **is not supported.**

Explanation:
The FileNet Business Process Manager mode does not support timer elements.

User response:
Delete the timer. If you do not delete the timer and import the process into FileNet Business Process Manager, the timer displays as a connected general step.

ZXP100004E **Observer** *"observer_name"* **in**
"process_name" **is not supported.**

Explanation:
The FileNet Business Process Manager mode does not support observer elements.

User response:
Delete the observer. If you do not delete the observer and import the process into FileNet Business Process Manager, the observer displays as a connected general step.

ZXP100005E **Map** *"map_name"* **in** *"process_name"* **is not supported.**

Explanation:
The FileNet Business Process Manager mode only supports one map element per process and it must be the first element in the flow. The purpose of the map is to populate repositories with data that is entering the process. The FileNet Business Process Manager mode does not support additional maps or maps that are not the first element in the process.

User response:
Delete the map. If you do not delete the map and import the process into FileNet Business Process Manager, the map displays as a connected general step.

ZXP100006E **The process** *"process_name"* **is referencing a global task** *"global_task_name."* **References to global tasks are not supported.**

Explanation:
The FileNet Business Process Manager mode does not support global task elements.

User response:
Delete the global task reference from the process. If you do not delete the reference and import the process into FileNet Business Process Manager, the global task reference displays as a connected general step.

ZXP100007E **The process** *"process_name"* **is referencing a global service** *"global_service_name."* **References to global service are not supported.**

Explanation:
The FileNet Business Process Manager mode does not support global service elements.

User response:
Delete the global service reference from the process. If you do not delete the reference and import the process into FileNet Business Process Manager, the global service reference displays as a connected general step.

ZXP100008E **The process** *"process_name"* **is referencing a business service** *"business_service_name."* **References to business service are not supported.**

Explanation:

The FileNet Business Process Manager mode does not support business service elements.

User response:
Delete the business service reference from the process. If you do not delete the reference and import the process into FileNet Business Process Manager, the business service reference displays as a connected general step.

ZXP102500E Element "*element_name*" must have a type.

Explanation:
In FileNet Business Process Manager mode, the element must have a type.

User response:
In the attributes of the element, specify its type.

ZXP102501W The basic type "*type_name*" is not supported for "*element_name*."

Explanation:
The FileNet Business Process Manager mode does not support all of the basic types available in WebSphere Business Modeler. The supported basic types are Boolean, Date, DateTime, Float, Integer, String, and Time.

User response:
Select the element and then, using the Attributes view, change the type of the element to one of the supported types.

ZXP102502E The type "*type_name*" is not supported for "*element_name*."

Explanation:
The FileNet Business Process Manager mode does not support complex data types. The supported types are the following basic types: Boolean, Date, DateTime, Float, Integer, String, and Time.

User response:
Select the element and then, using the Attributes view, change the type of the element to one of the supported types.

ZXP103001W Bulk resource "*bulk_resource_name*" is not supported for "*element_name*."

Explanation:
The FileNet Business Process Manager mode does not support bulk resource assignment. The bulk resource assignment will not be exported.

ZXP103002W Resource assignment "*resource_assignment_name*" for "*element_name*" is not supported.

Explanation:

The FileNet Business Process Manager mode only supports resource assignments in which the associated resource instances are based on the predefined individual resource definitions (person and staff) and resource assignments associated with roles. Other resource assignments are not supported and the FileNet Business Process Manager export does not export them.

ZXP110001E "*Element_name*" has more than one input criterion.

Explanation:
The FileNet Business Process Manager mode does not support more than one input criterion per element.

User response:
To correct the error:
1. Select the element with the error.
2. in the Attributes view, click the **Input Logic** tab.
3. Expand the **Input Criteria** section.
4. From the table, select an input criterion and click **Remove**.
5. Keep removing input criteria until there is only one.
6. Save the process.

ZXP110002E "*Element_name*" has more than one output criterion.

Explanation:
The FileNet Business Process Manager mode does not support more than one output criterion per element.

User response:
To correct the error:
1. Select the element with the error.
2. in the Attributes view, click the **Output Logic** tab.
3. Expand the **Output Criteria** section.
4. From the table, select an output criterion and click **Remove**.
5. Keep removing output criteria until there is only one.
6. Save the process.

ZXP110003W The minimum number of items needed for data input or output "*input_or_output_name*" on "*element_name*" must be greater than zero.

Explanation:
In FileNet Business Process Manager mode, an input or output containing data must have at least one item. Note that in FileNet Business Process Manager mode, the recommended way of passing data from one element to another is through connections to a repository rather than a direct data connection between the two elements.

**ZXP110004E A loop condition expression on loop
"*loop_name*" has not been defined.**

Explanation:
In FileNet Business Process Manager mode, a loop must have a condition that determines when the loop ends.

User response:
To correct this error:

1. Select the loop with the error.
2. In the Attributes view, click the **Loop condition** tab.
3. Click **Edit**.
4. Using the Expression Builder, create the expression for the loop.
5. When you have created the expression, click **OK**.
6. Save the process.

**ZXP110005E The output condition on branch
"*branch_name*" of decision
"*decision_name*" is not defined.**

Explanation:
In FileNet Business Process Manager mode, each output branch of a decision must have a condition.

User response:
To correct this error:

1. Select the decision with the error.
2. In the Attributes view, click the **Output branches** tab.
3. Select a branch.
4. Near the **Expression** field, click **Edit**.
5. Using the Expression Builder, create the expression for the branch.
6. When you have created the expression, click **OK**.
7. Save the process.

**ZXP110006E The source and target of connection
"*connection_name*" have different
minimum and maximum numbers of
items.**

Explanation:
The numbers in the minimum and maximum number of data items in the output of the source element must match the numbers in the minimum and maximum of the input of the target element.

User response:
To correct this error:

1. Select the output of the source element.
2. Check the values of the minimum and maximum fields.
3. Select the input of the target element.
4. Check the values of the minimum and maximum fields.

5. Change the values in the output or in the input so that the two match.
6. Save the process.

**ZXP110007E Data input or output
"*input_or_output_name*" on
"*element_name*" must reference a
repository.**

Explanation:
In FileNet Business Process Manager mode, an input containing data whose input source is set to Repository must reference a repository or an output containing data whose output source is set to Repository must reference a repository.

User response:
To correct this error, connect the data to a repository. The associated data for the repository must match the data type of the input or output.

**ZXP110010W The input connection to input criterion
"*input_criterion_name*" of element
"*element_name*" is not connected to
another element. The element will be
ignored when you export the model.**

Explanation:
When you export the process using the FileNet Business Process Manager export, the unreachable element will not be included in the export. Elements that trace their process flow back to this element will also not be included in the export.

**ZXP110011E Data input or output
"*input_or_output_name*" on
"*element_name*" must not reference a
global repository.**

Explanation:
The FileNet Business Process Manager mode does not support references to global repositories.

User response:
To correct this error:

1. Create a local repository.
2. If the data for the input or output is not String, change the data type associated with the local repository.
3. Delete the reference to the global repository.
4. Connect the output or input to the local repository.
5. Save the process.

**ZXP110012E The input criterion
"*input_criterion_name*" on join
"*join_name*" must not have data input.**

Explanation:

ZXP110013E • ZXP118004E

The FileNet Business Process Manager mode does not support joins that join data flow. Use joins to control process flow only.

User response:
To correct this error, use repositories to pass data from one element to another.

ZXP110013E **Constant data input "***input_name***" on "***element_name***" requires a constant value.**

Explanation:
The data input has been set to "Constant" but it does not have a constant value.

User response:
To correct this error:

1. Select the element with the error.
2. In the Attributes view, click the **Inputs** tab.
3. Select the input with its input source set to **Constant**.
4. Scroll down the **Constant values** field and type a value.
5. Save the process.

ZXP110014E **Constant data input "***input_name***" on "***element_name***" must be a basic type.**

Explanation:
The FileNet Business Process Manager mode only supports specific basic types. The supported types are Boolean, Date, DateTime, Float, Integer, String, and Time.

User response:
To correct this error, replace the data type of the input with a supported basic data type.

ZXP110015W **Constant data input "***input_name***" on "***element_name***" is an unsupported basic type.**

Explanation:
The FileNet Business Process Manager mode only supports specific basic types. The supported types are Boolean, Date, DateTime, Float, Integer, String, and Time.

ZXP118000E **The expression for element "***element_name***" uses the variable "***variable_name***."**

Explanation:
The FileNet Business Process Manager mode does not support variables within expressions.

User response:
To correct this error, edit the expression and remove the variable.

ZXP118001E **The expression for element "***element_name***" uses the union operator.**

Explanation:
The FileNet Business Process Manager mode does not support union operators within expressions.

User response:
To correct this error, edit the expression and replace the union operator with another operator.

ZXP118002E **Element "***element_name***" contains an expression with an unsupported function. The function references multiple local repositories.**

Explanation:
The FileNet Business Process Manager mode does not support a function referencing more than one local repository.

User response:
To correct this error, edit the expression and remove references to local repositories until the function only references one local repository.

ZXP118003E **The function expression for element "***element_name***" is not supported.**

Explanation:
The FileNet Business Process Manager mode does not support the function expression except for the String functions and the count numeric function.

Note: WebSphere Business Modeler evaluates both sides of an expression. You may see this validation error issued for both sides of the expression if there is an unsupported function expression on each side.

User response:
To correct this error, edit the expression and remove the function from the expression.

ZXP118004E **The expression for element "***element_name***" references a modeling artifact with an unsupported type. The supported types are Boolean, Date, DateTime, Float, Integer, String, and Time.**

Explanation:
The FileNet Business Process Manager mode only supports some of the basic data types that are available. The supported basic types are: Boolean, Date, DateTime, Float, Integer, String, and Time.

Note: WebSphere Business Modeler evaluates both sides of an expression. You may see this validation error issued for both sides of the expression if there is a reference to an unsupported type on each side.

User response:
To correct this error, edit the expression and change the
data type for the modeling artifact to one of the
supported basic types.

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 documentation 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 documentation. 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 (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:

Lab Director
IBM RTP Laboratory
3039 Cornwallis Road
P.O. BOX 12195
Raleigh, NC 27709-2195
U.S.A

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.

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:

(C) (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. (C) Copyright IBM Corp. 20004, 2005. All rights reserved.

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

Trademarks and service marks

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

IBM
IBM (logo)
SupportPac
WebSphere
Workplace

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

Intel, Intel Inside (logos), and Pentium are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

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

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

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

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

IBM WebSphere Business Modeler FileNet Integration
SupportPac-BA77