

Chapter 6

Working with WebSphere Business Modeler

The process flow modeling capability available in the Fabric Authoring widgets is restricted to defining simple sequences of tasks and is not designed to be a replacement for Modeler. Instead, the application flow represents a simplified, high-level outline of the process which provides visual context for a business user to understand where policies are used in a solution.

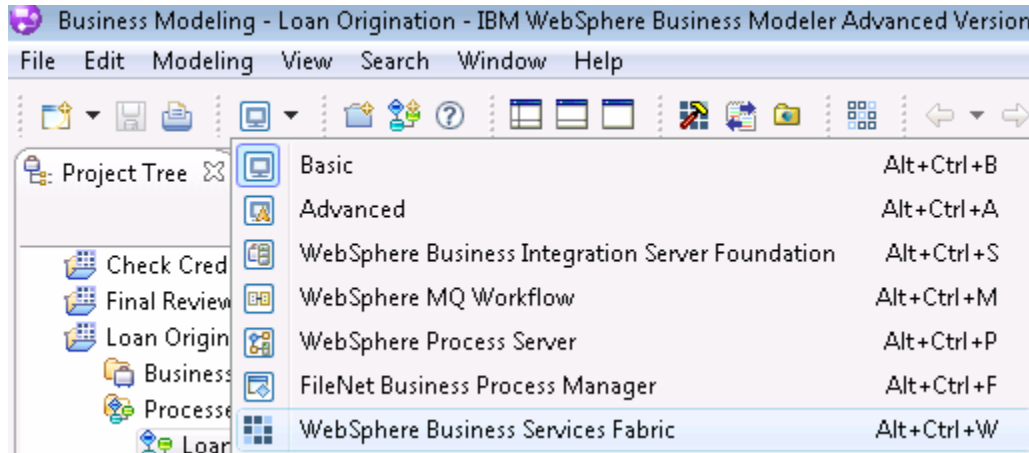
There are two basic approaches for working with Modeler and the Fabric Authoring capability. The first, and most general, approach involves disconnected use Modeler and Fabric authoring during the Model phase of development. Modeler is used to develop and simulate business processes while the authoring capability in business space is used to define business vocabulary and policies that determine the dynamic behavior of selected process steps.

The second approach, covered in this chapter, involves importing content authored in business space to establish a collection of Modeler projects with skeletal process flows. Once established, the modeler projects can be updated to pick up certain changes introduced in business space. However, there is currently no reverse capability to import or update the content in business space with changes introduced in Modeler. Also, it is important to consider certain limitations that come along with using this approach.

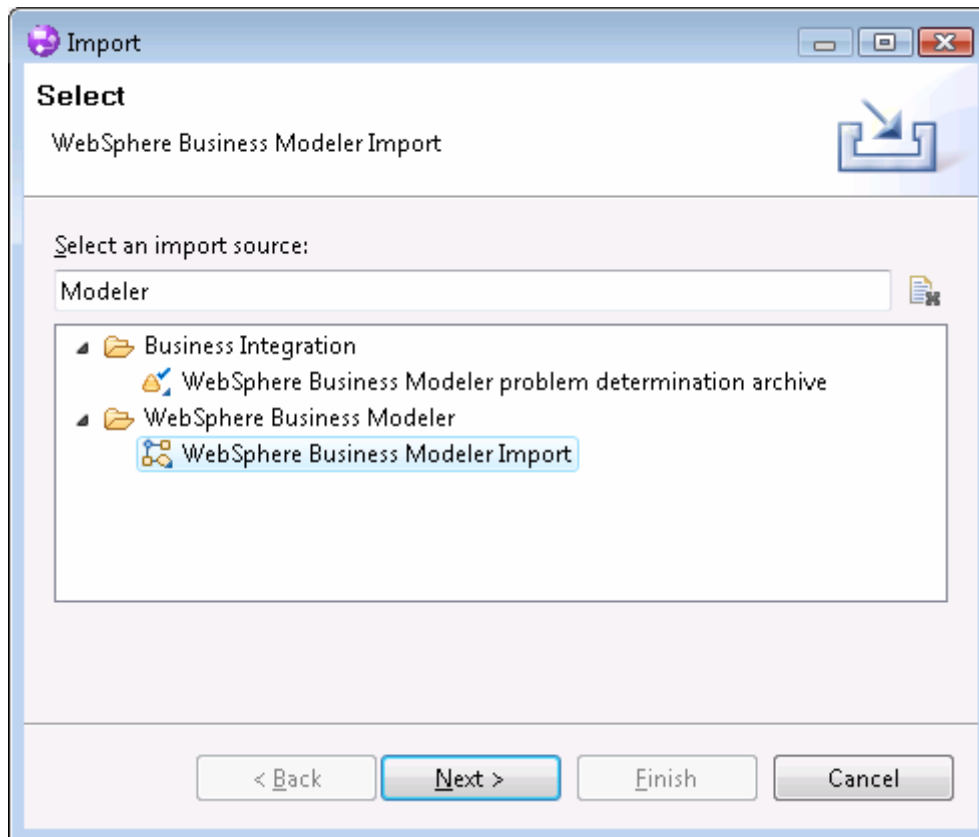
Importing an Fabric application into Modeler

Modeler features that support Fabric integration are only enabled in the Fabric mode. This mode includes all of the capabilities of the WebSphere Process Server mode and adds the ability to import from Fabric. It also adds the ability to set a technical attribute that indicates a task should be implemented with a dynamic assembler component.

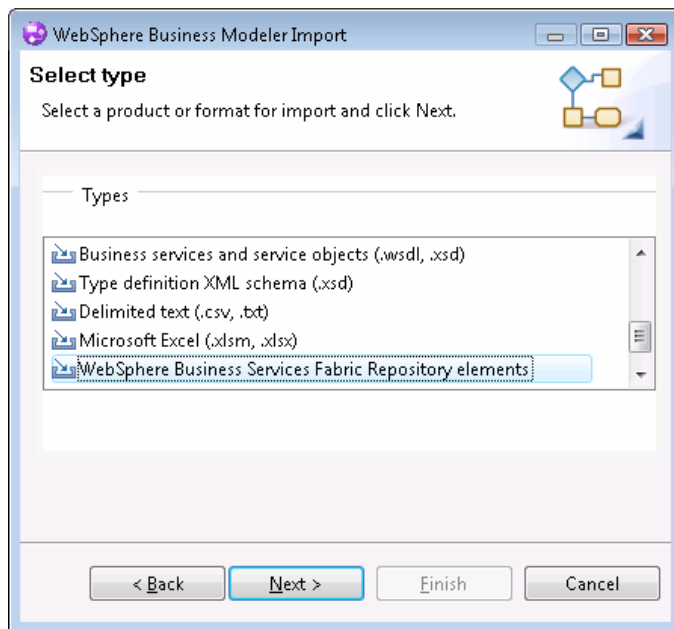
1. Switch to Fabric Mode.



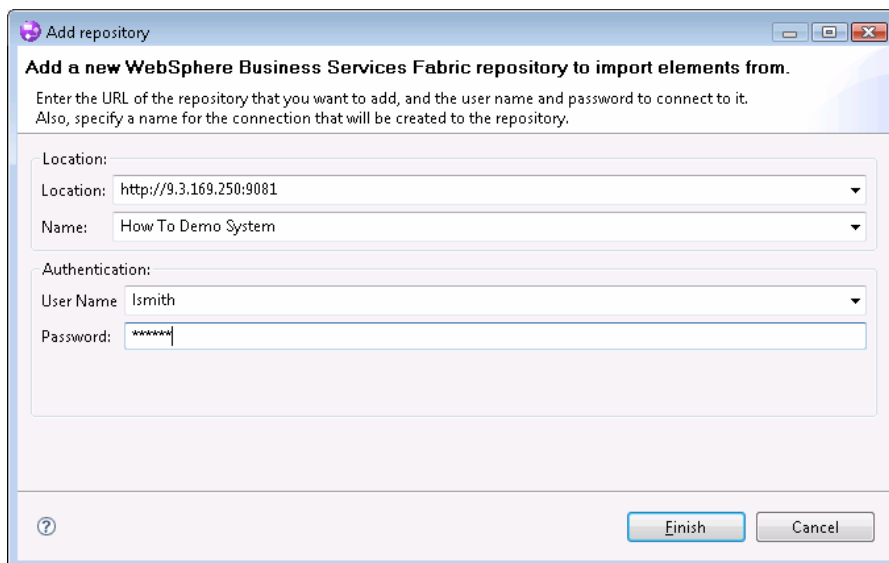
2. From the main menu, select File→Import... Select WebSphere Business Modeler Import and click Next.



3. Select WebSphere Business Services Fabric Repository Elements and click Next.



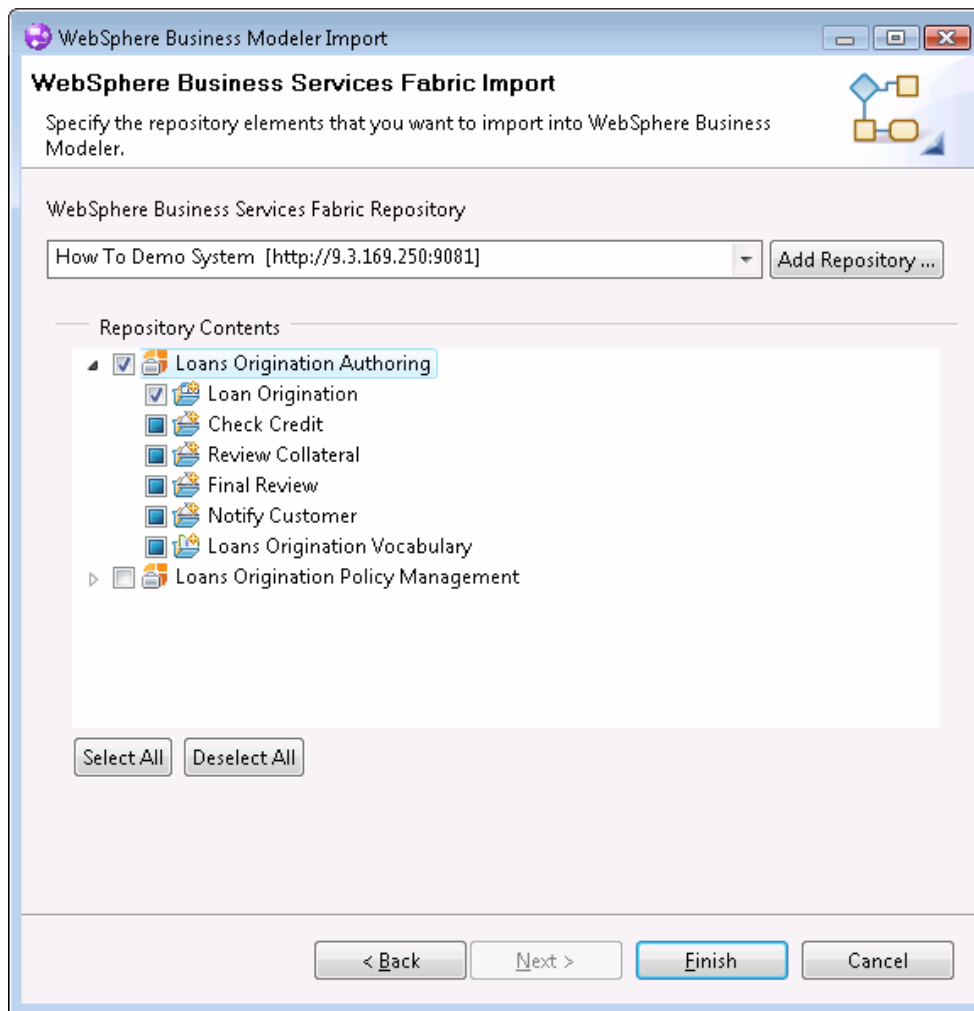
4. Click the Add Repository... button. Define connection details for the Fabric server to import from. Use the server url, user name, and password used to log into business space.



The Repository Contents section shows the applications, business services, and vocabularies grouped by the business spaces they are

associated with. A user will only see the content in those fabric authoring spaces that the users is authorized to see in business space.

5. Select the Loans Origination Authoring space and all of its contents. Click Finish.



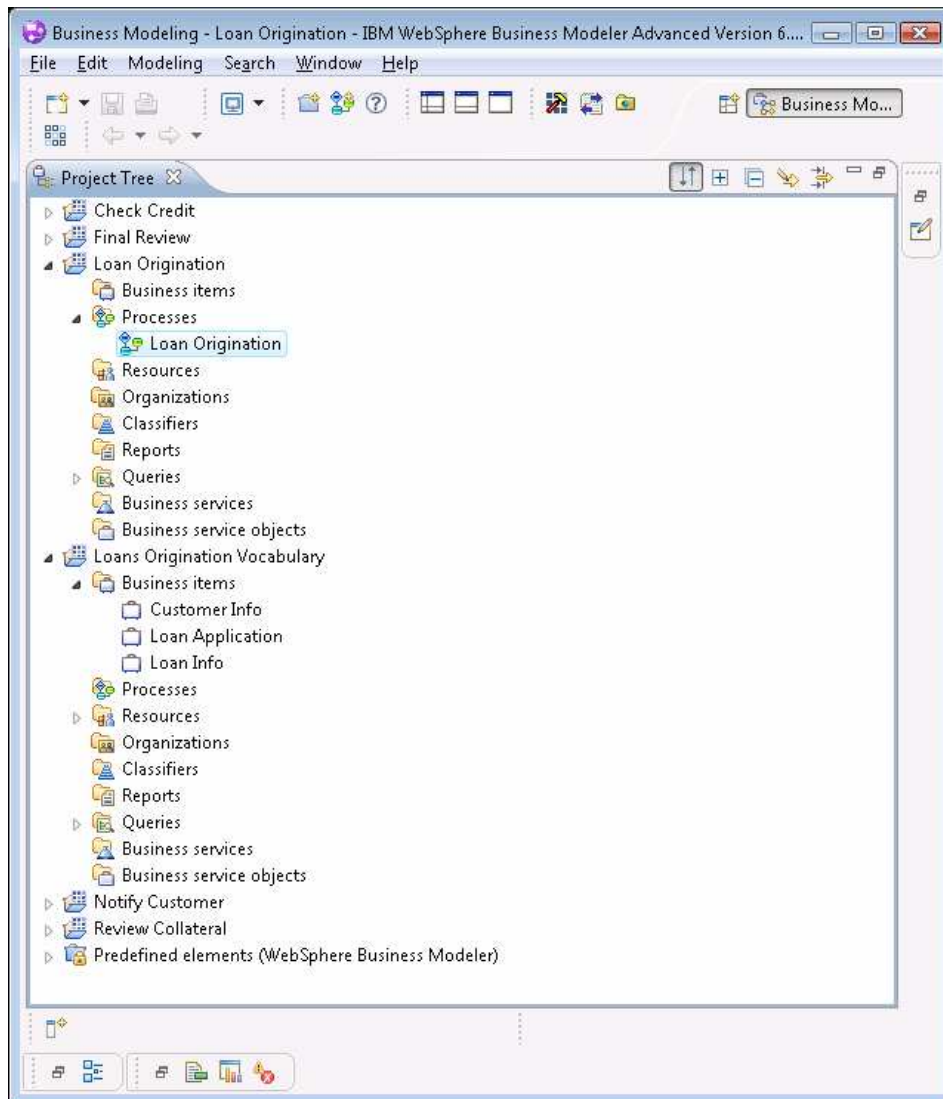
The import operation creates several Modeler projects. Each imported vocabulary becomes a project that contains business items derived from the object concepts that were defined in the authoring space.

Each business service becomes a modeler project with an empty process flow created for each process variation that was defined. A service is

defined for the business service itself with technical attributes automatically set to indicate implementation with a dynamic assembly component.

Finally, an application becomes a modeler project with a process flow that captures the process modeled in the application flow. Each task in the flow uses an abstract service defined in a business service project.

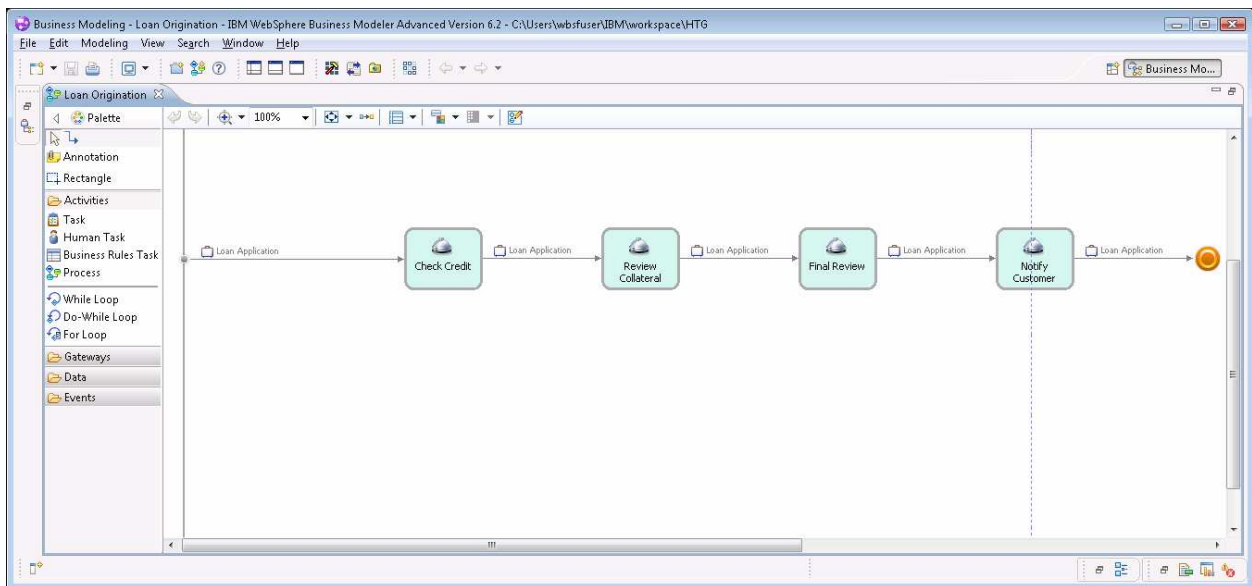
Vocabularies and business services are designed to be re-used in solutions that are developed. Multiple modeler projects are created for the same reason. Instead of duplicating business items or business service logic each time it is needed, this content is grouped into different modeler projects that can be re-used.



Working with the main process flow

Starting at the top, look at the process flow created for the Loan Origination application.

1. Open the Loan Origination process in the Loan Origination project.



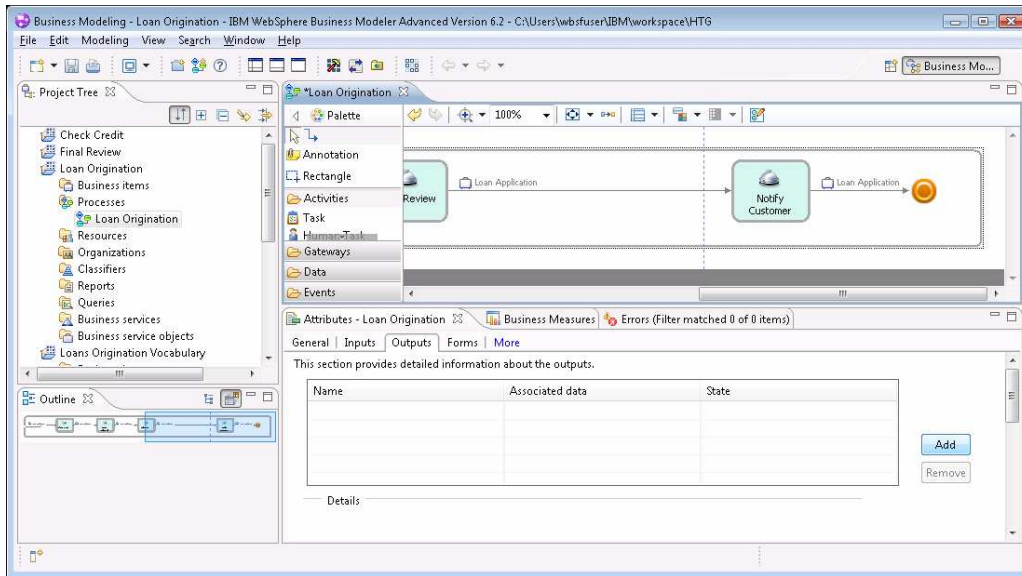
In the Fabric authoring space, you defined two channels that can invoke the process including “Customer Care Call Center” and “Better Financials Online”. Since there is no concept of a channel in Business Modeler, this information does not appear in the process flow.

When defining the business services in the authoring space, you were careful to specify the loan application as the input and output of every business service. When the process is imported, the orchestration from one task to the next is only preserved when the output of one business service is the same as the input to the next.

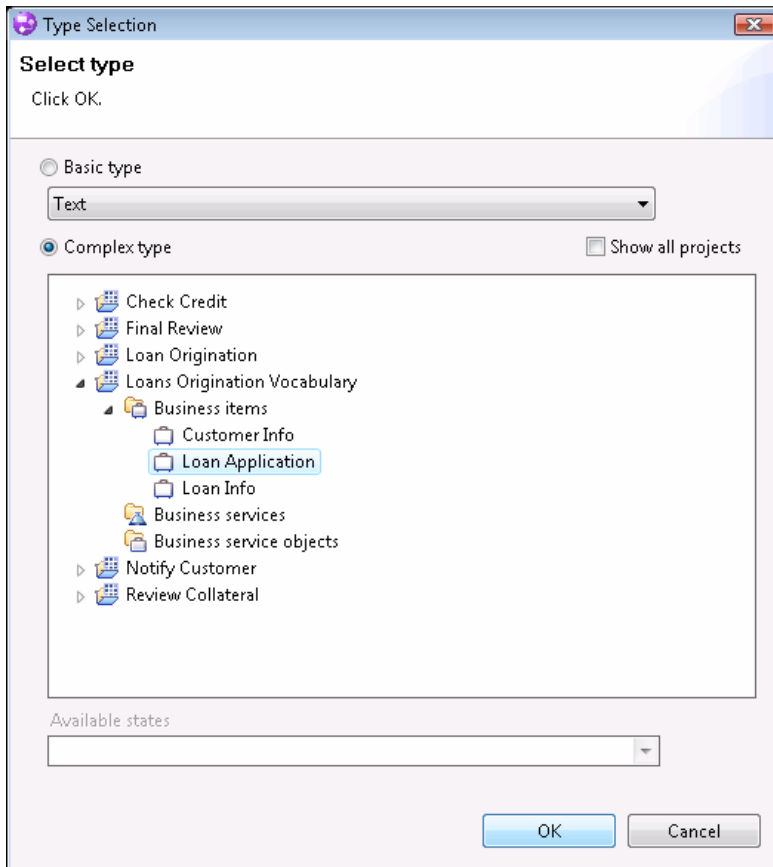
For the purposes of authoring policies in a Fabric authoring space, it is sufficient to add inputs to a business service, represent the business concepts that should be available to evaluate policies for that step at runtime. When the modeler import is brought into the picture, business service inputs and outputs need to be modeled more completely and with greater restrictions since these are used to define the business items and flow from one task to the next.

The output of the final step in your Loan Origination process is wired to the process terminal. If you want your process to yield an output you will need to configure and output on the process.

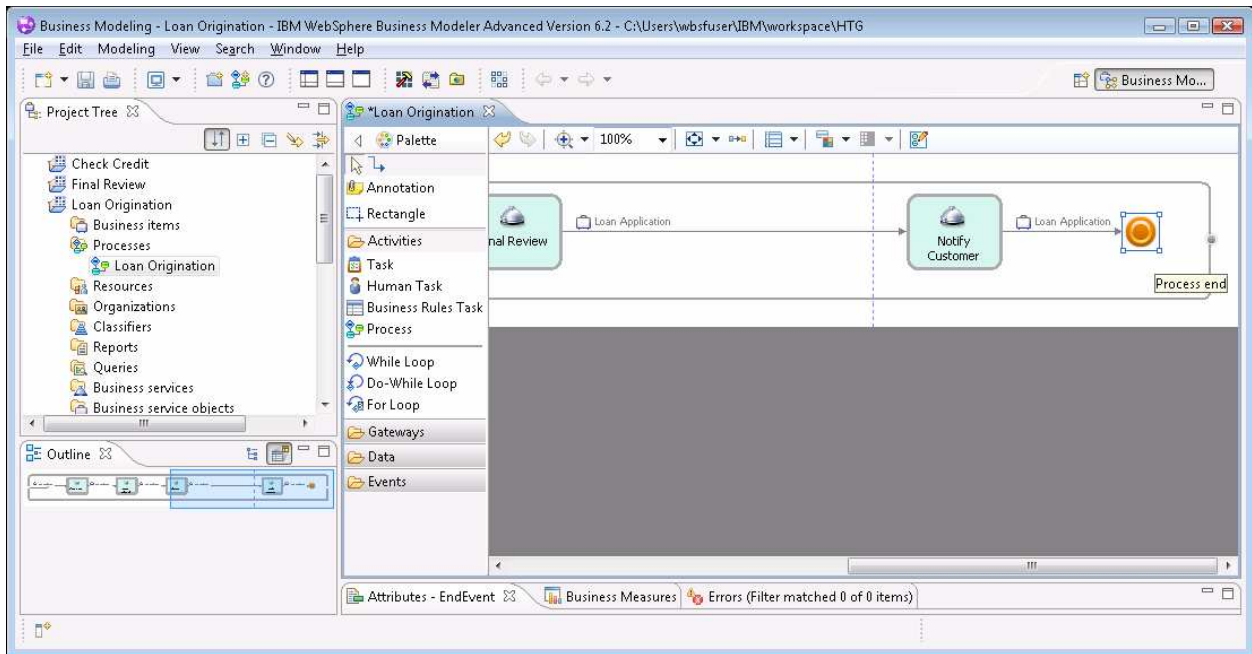
2. Open the attributes view for the process, go to the outputs tab.



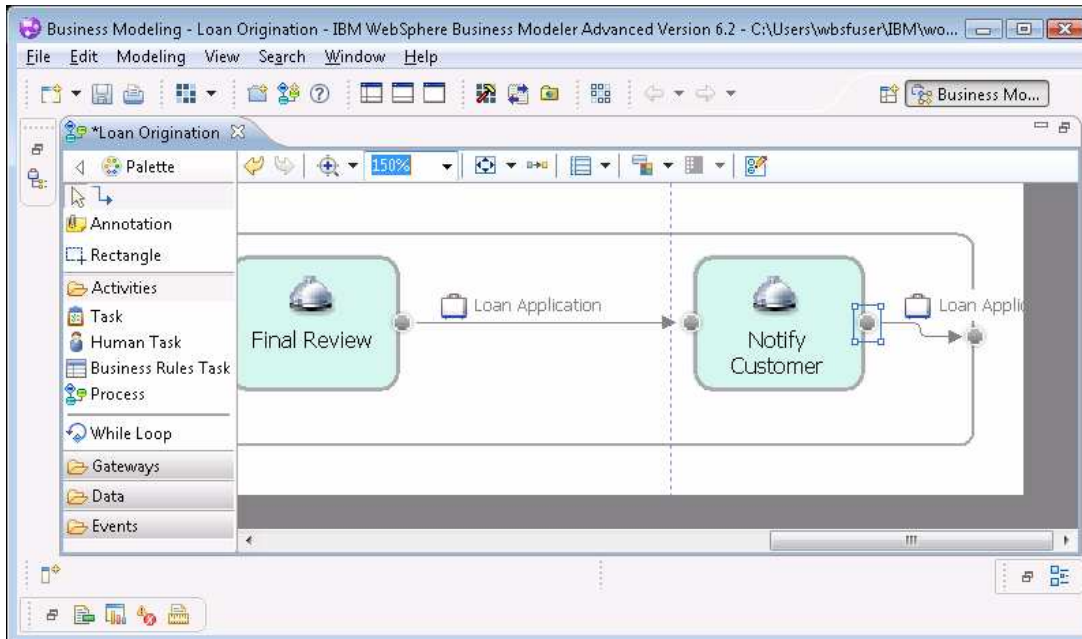
3. Click the Add button. Select the Loan Application business item as the type for the output.



4. Highlight and delete the terminal attached to the output of Notify Customer.



5. Wire the output of Notify Customer to the newly added output of the process.

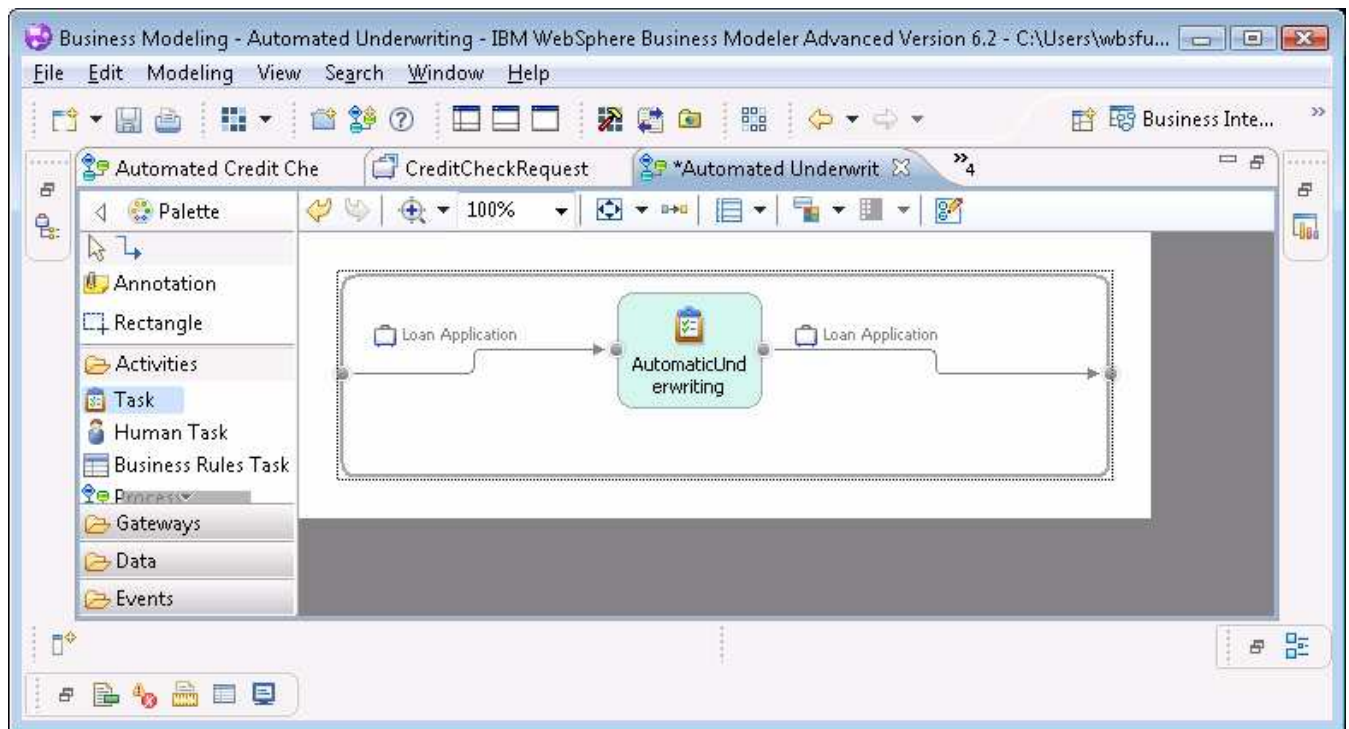


Flesh out Processes for each Process Variation

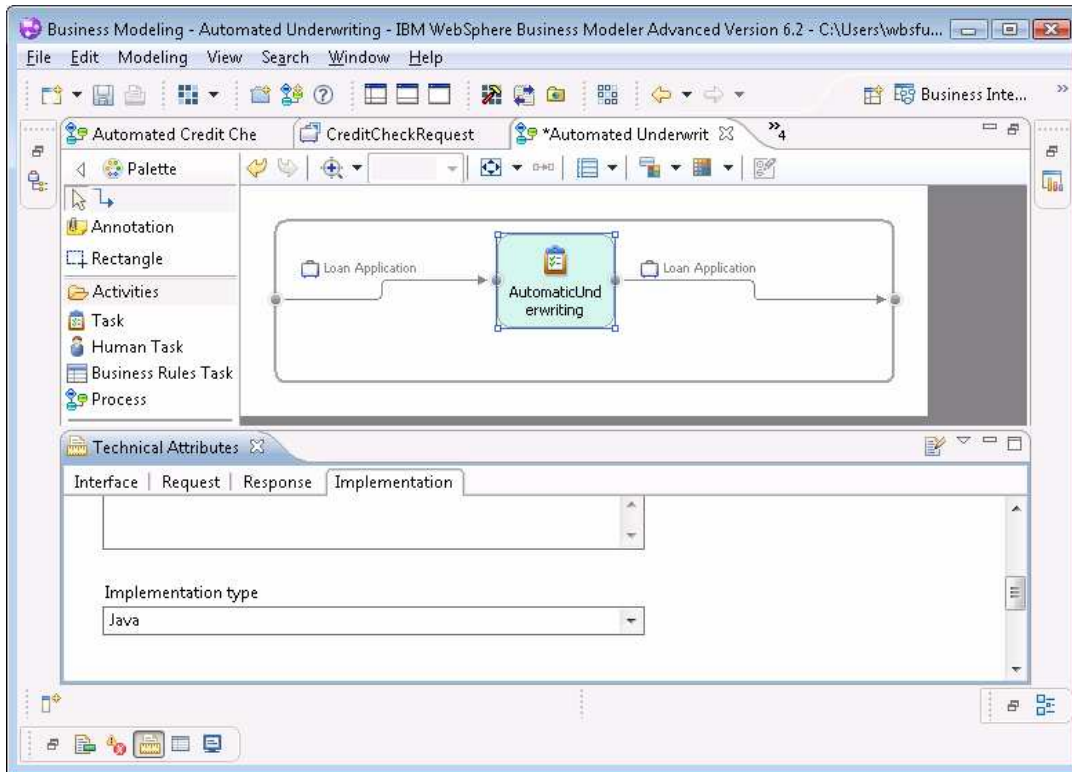
Each process variation is created as an empty process flow with the inputs and outputs derived from the definition of the Business Service in the authoring space. Open each process and model the process for each step.

Since the purpose of this guide is to illustrate how to build applications with the Fabric, you will keep these processes very simple with very simple implementations of the tasks.

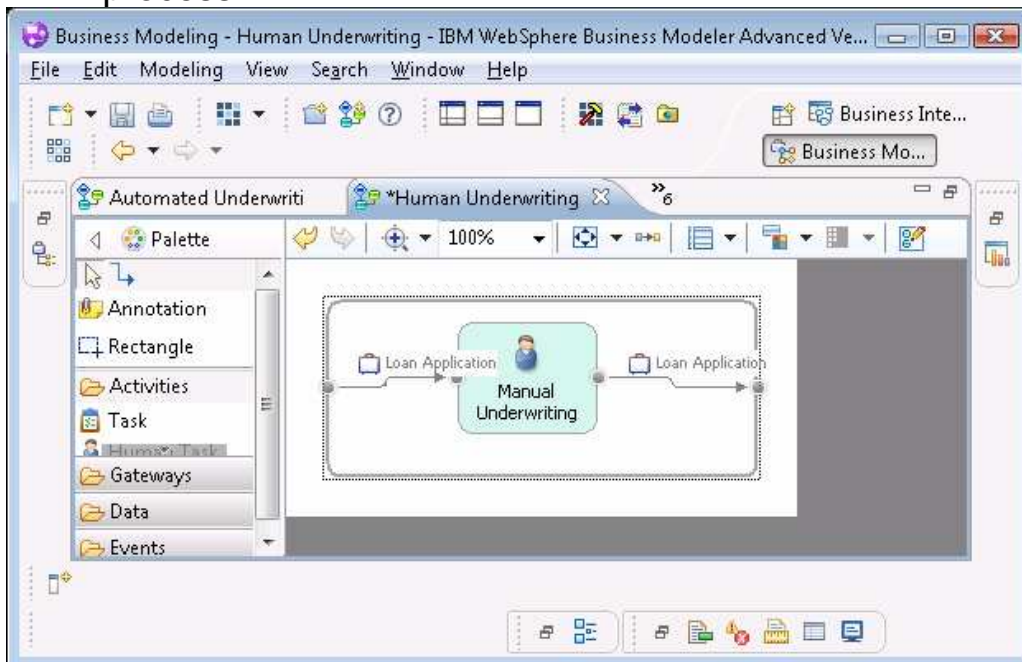
1. Open the Automated Underwriting Process. Add a task named “AutomaticUnderwriting” and wire it to the input and output.



2. Using the technical attributes view, set the implementation type for this task to Java. When exporting to WID, this will cause the task to be implemented with a Java component.



3. Open the Human Underwriting process. Add a human task into the process.

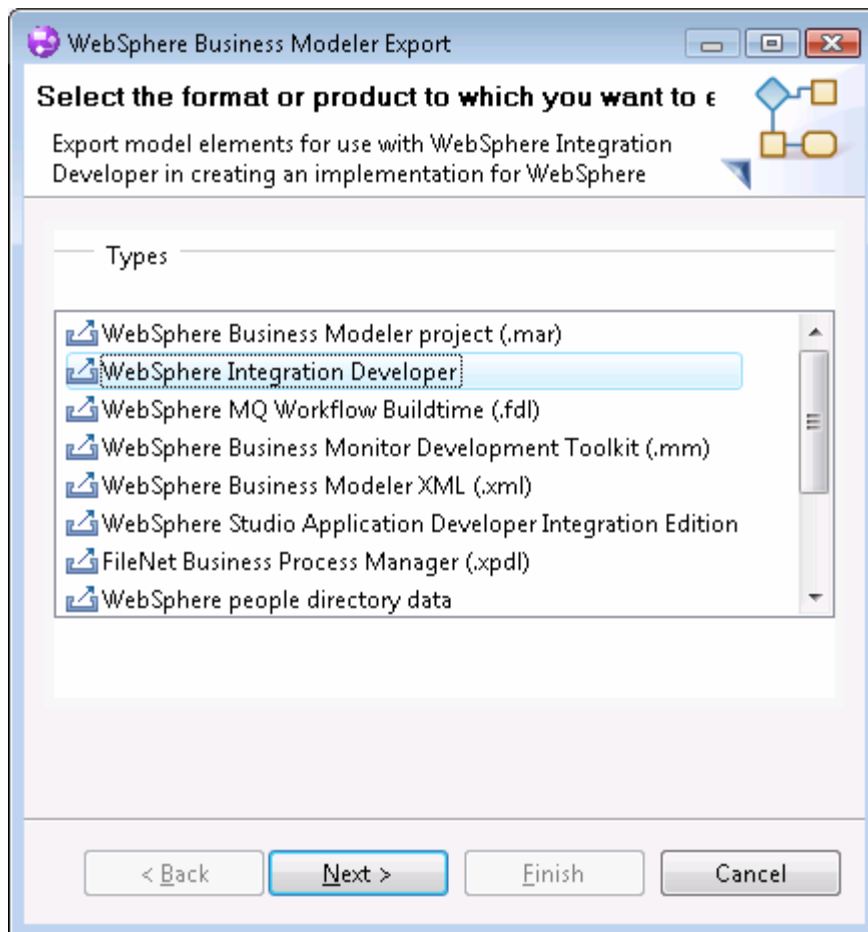


4. Fill in remaining process variations.

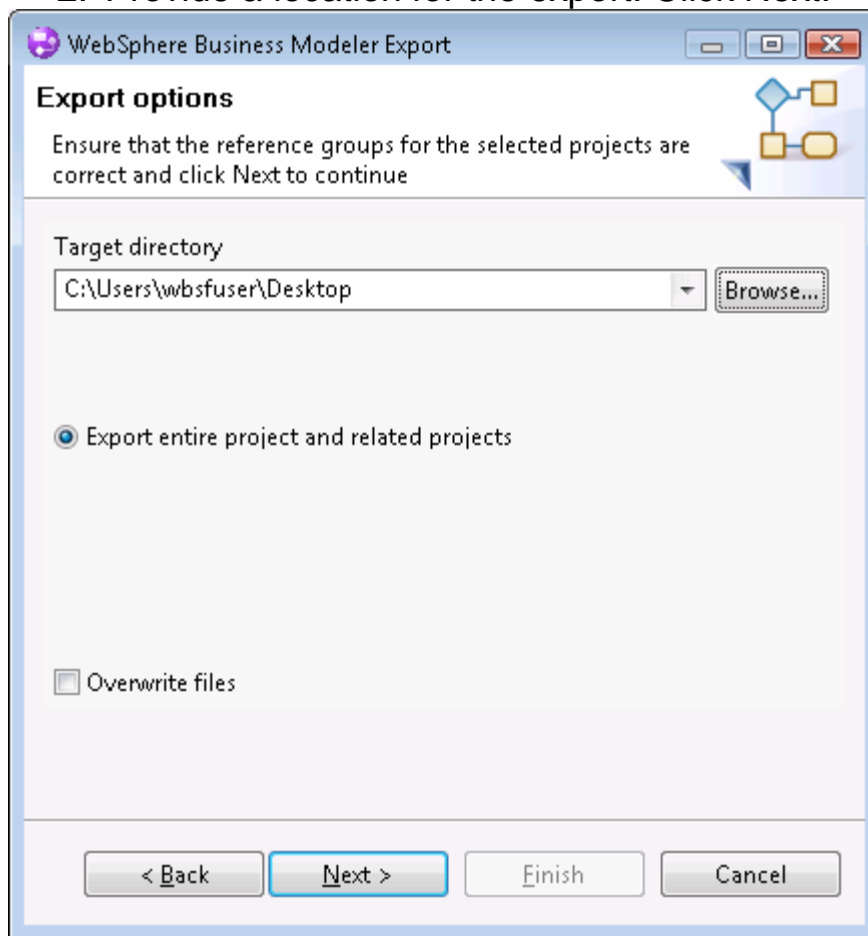
Export for WebSphere Integration Developer

When modeling is complete you can export the project interchange file with SCA modules and libraries for further assembly in WebSphere Integration Developer.

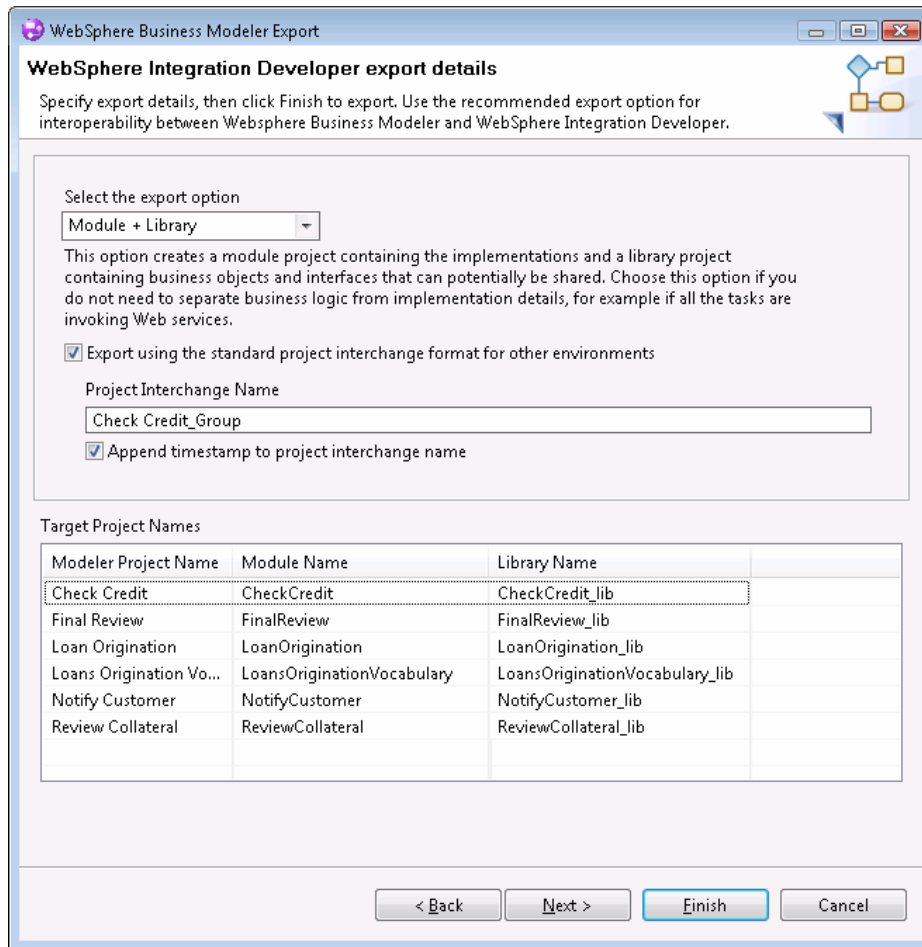
1. Select File-> Export ... Select WebSphere Integration Developer and click next.



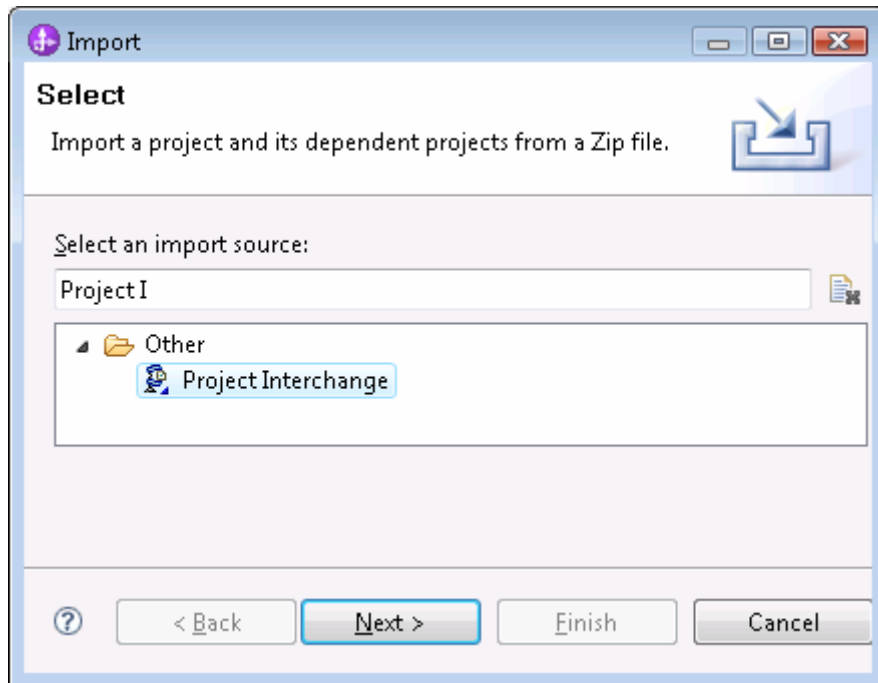
2. Provide a location for the export. Click Next.



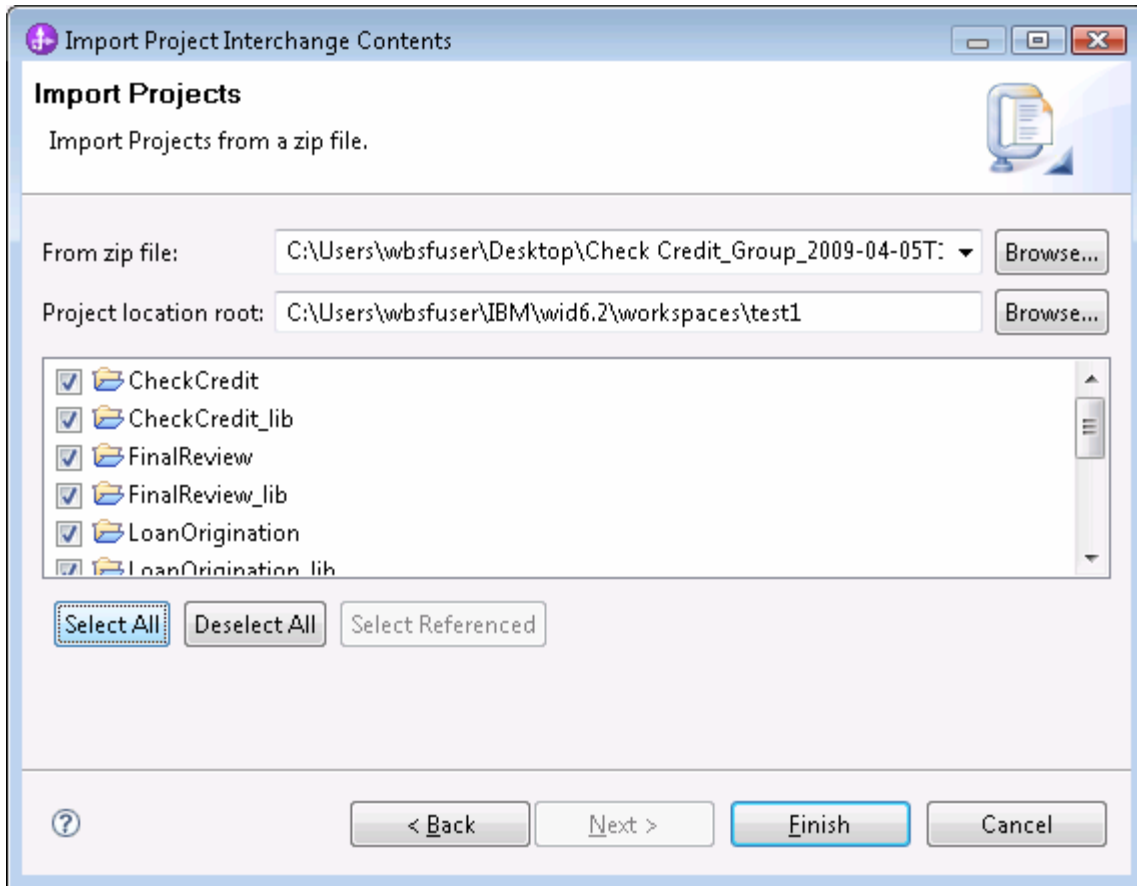
3. Select the Module + Library option. Click Finish.



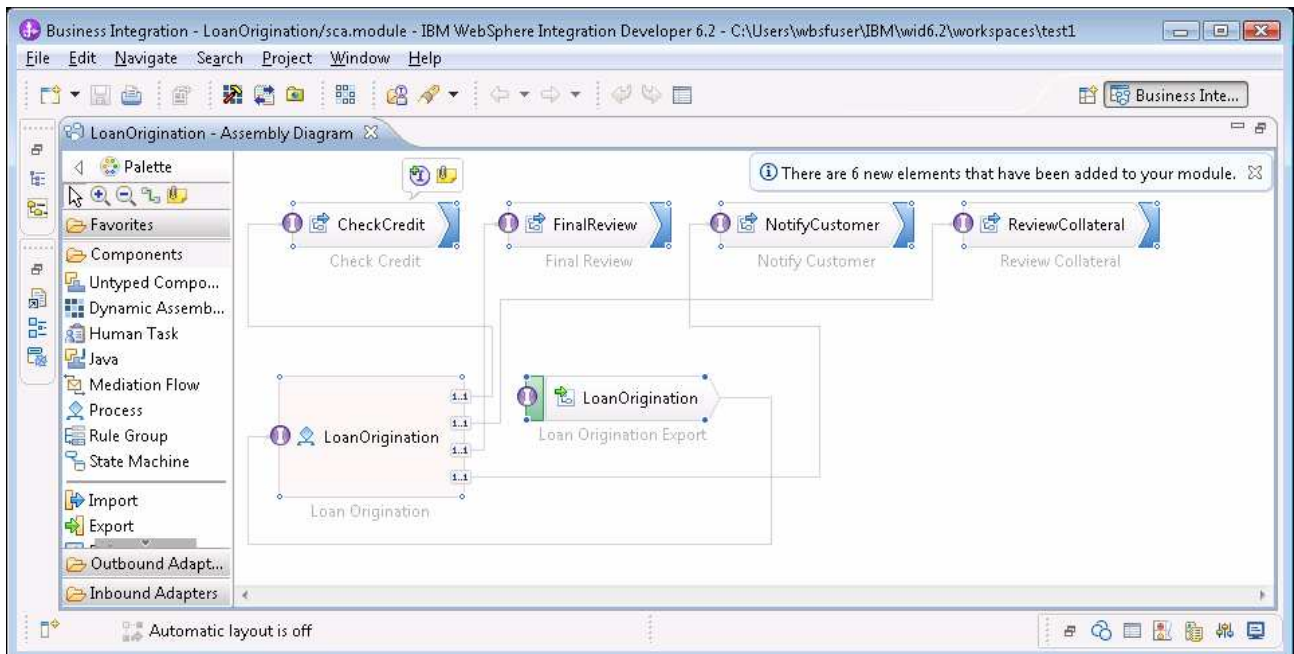
4. In the Business Integration perspective in WebSphere Integration Developer, select File -> Import. Choose the Project Interchange option, and click Next.



5. Provide the location of the compressed file. Select all projects, and click Finish.

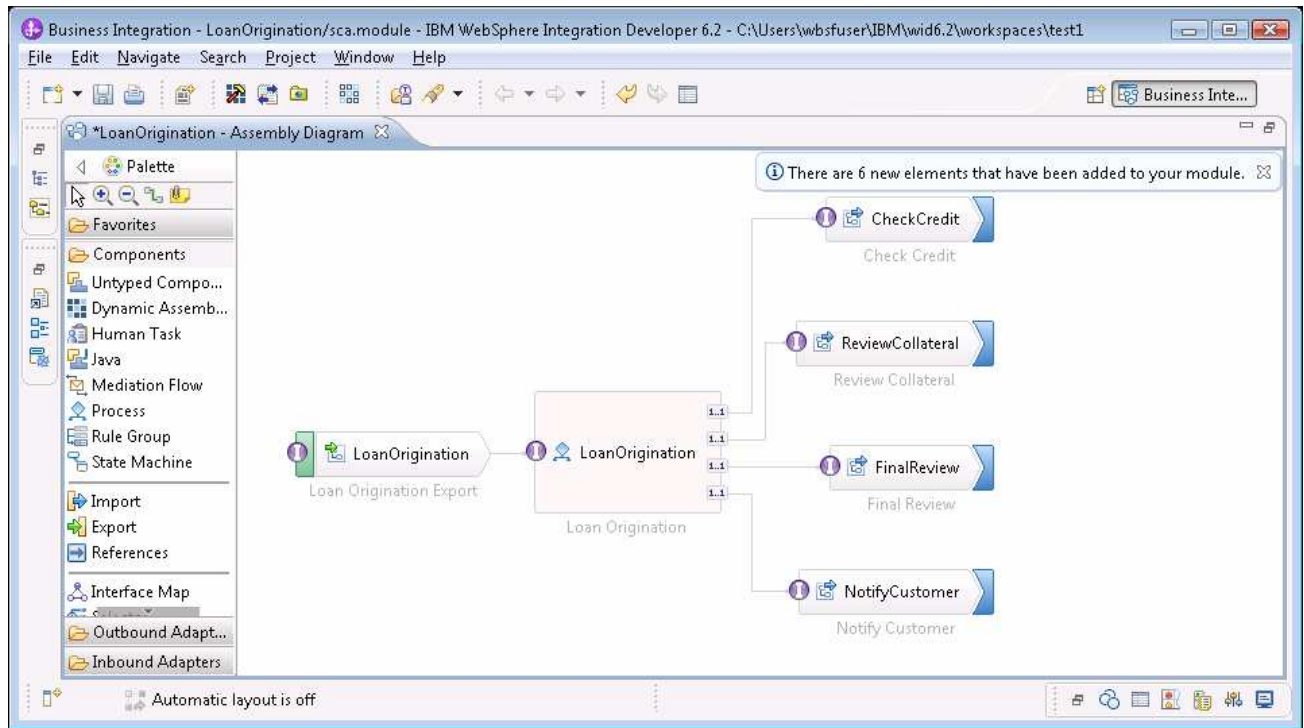


6. Open the assembly diagram for the Loan Origination module.



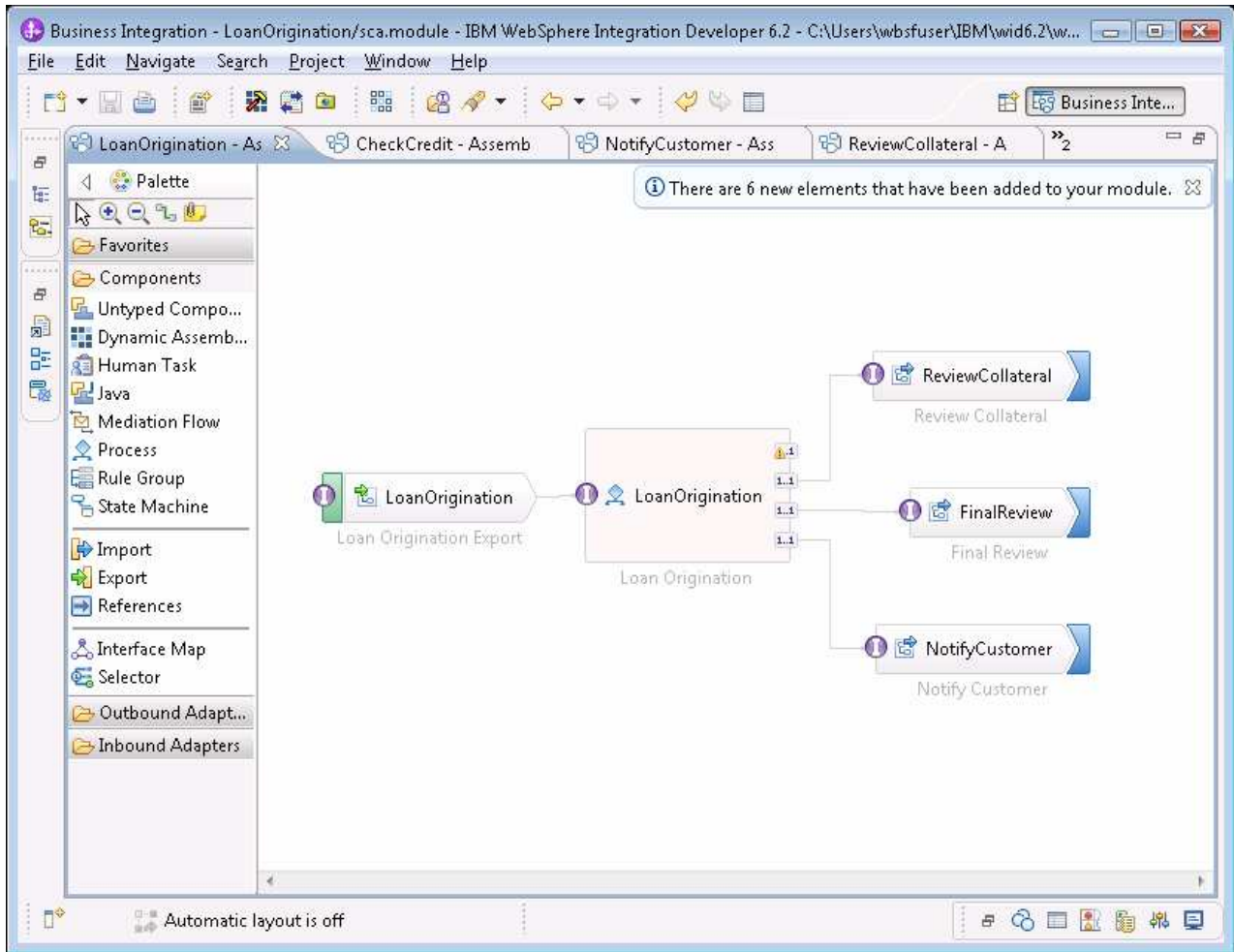
7. Right click in the assembly editor and select Layout contents. The

resulting layout makes more sense to most people.



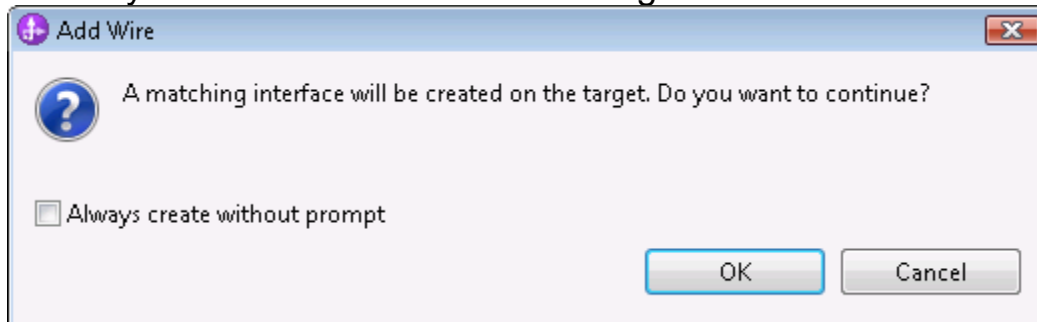
Typically, a dynamic assembly component is used in place of an import. When modeler generates the implementation for the process variations, it also puts the dynamic assembler component in that module. Imports wired off the process are linked to the export of the dynamic assembly component. The next few steps cover how to remove this extra layer of indirection.

8. Delete the Check Credit import.



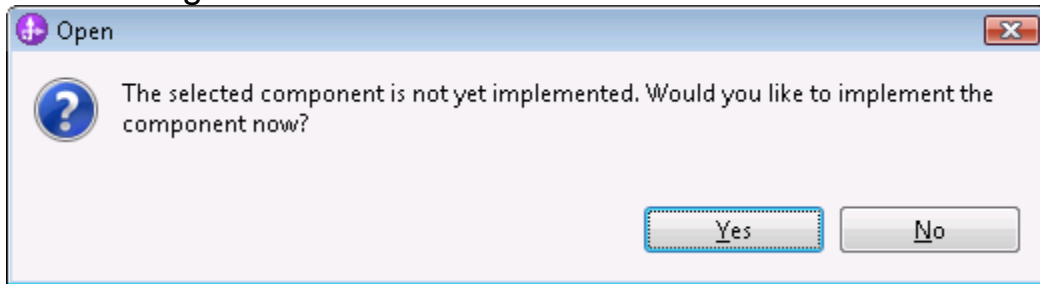
9. Select the Dynamic Assembler component on the palette and drop it in the place where the import used to be and name it “CreditCheck”. Draw a wire from the open partner link to the new Dynamic Assembler.

10. The system will ask about associating an interface. Click OK.

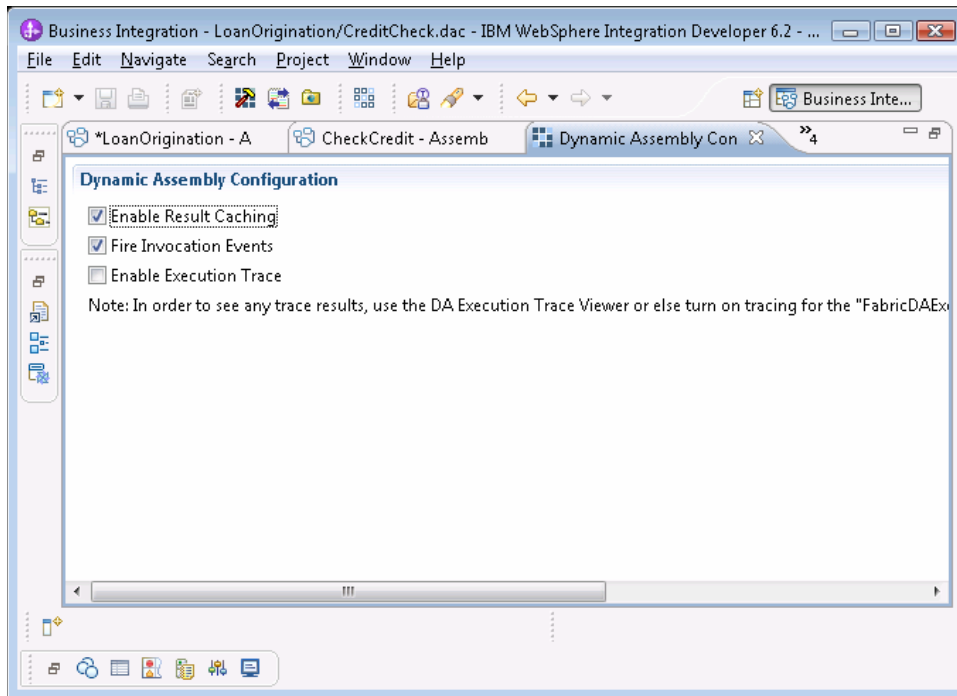


11. Double-Click the component to implement it. Click Yes for the

dialog.



12. The editor provides three configuration options for the dynamic assembler. The defaults are acceptable.



13. Open the assembly diagram for the Check Credit module and delete the dynamic assembly component and the export associated with it.
14. Perform the same steps for the other three imports. This Loan Origination assembly diagram should then look something like this:

