Chapter 7

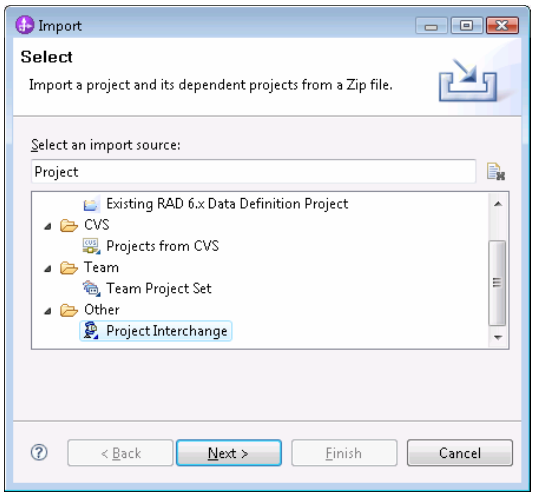
Assembly in WebSphere Integration Developer

In this chapter, you will review the SCA modules created for this scenario. Instead of walking step-by-step through the definition of BPELs, services interfaces, and your mock backend service implementations, you will start by importing a project interchange that contains several SCA modules. Then you will walk through these modules and see how it works.

Importing the Loan Origination SCA modules

1. Select File -> Import … from the main menu.

2. Select Project Interchange. Click Next.



Import

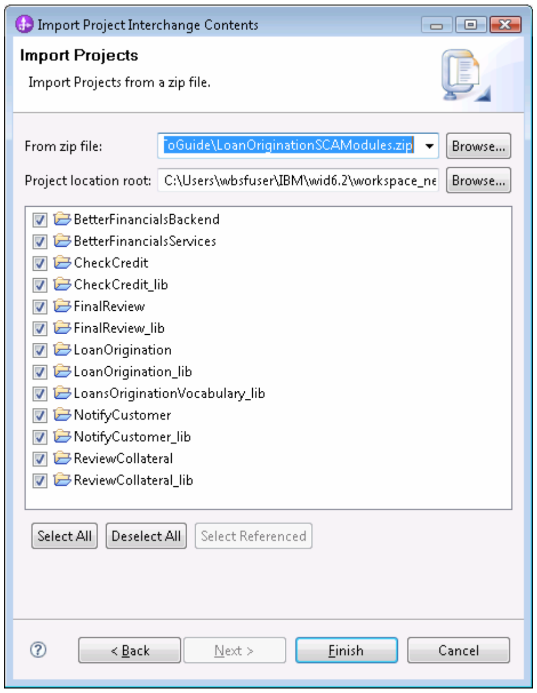
Select

Import a project and its dependent projects from a Zip file.

Select an import source:

Project

3. Browse to the location of LoanOriginationSCAModules.zip. The set of projects in the zip show up. Click Select All and then Finish.


Import Project Interchange Contents

Import Projects

Import Projects from a zip file.

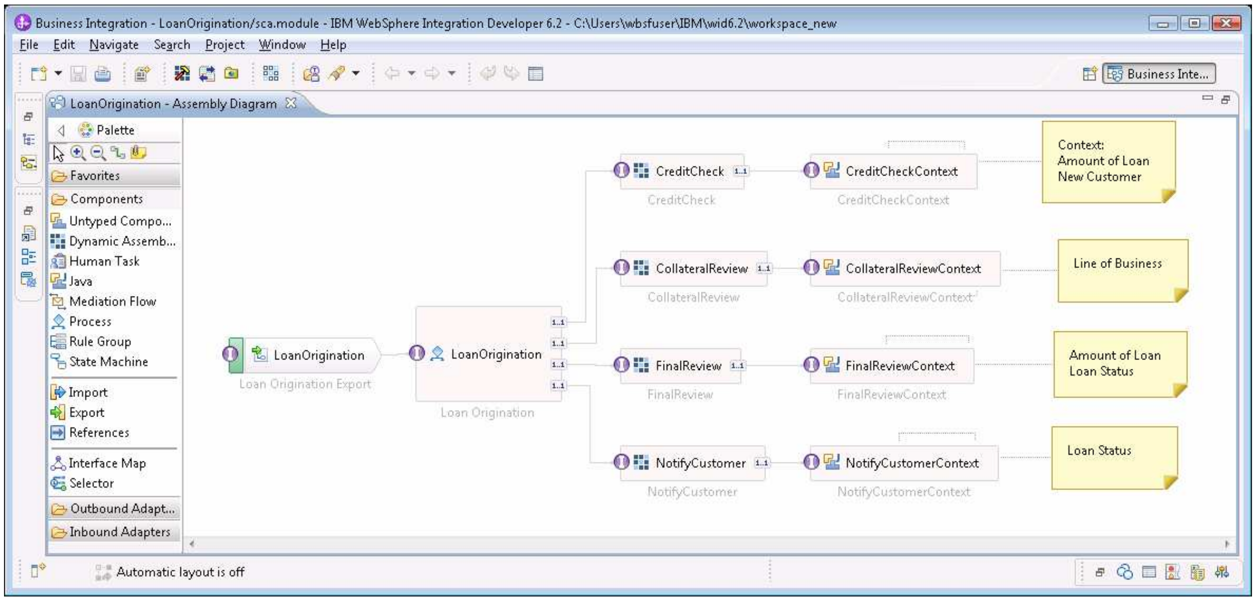
From zip file: oGuide\LoanOriginationSCAModules.zip

Project location root: C:\Users\wbsfuser\IBM\wid6.2\workspace_ne

- [✓] BetterFinancialsBackend
- [✓] BetterFinancialsServices
- [✓] CheckCredit
- [✓] CheckCredit_lib
- [✓] FinalReview
- [✓] FinalReview_lib
- [✓] LoanOrigination
- [✓] LoanOrigination_lib
- [✓] LoansOriginationVocabulary_lib
- [✓] NotifyCustomer
- [✓] NotifyCustomer_lib
- [✓] ReviewCollateral
- [✓] ReviewCollateral_lib

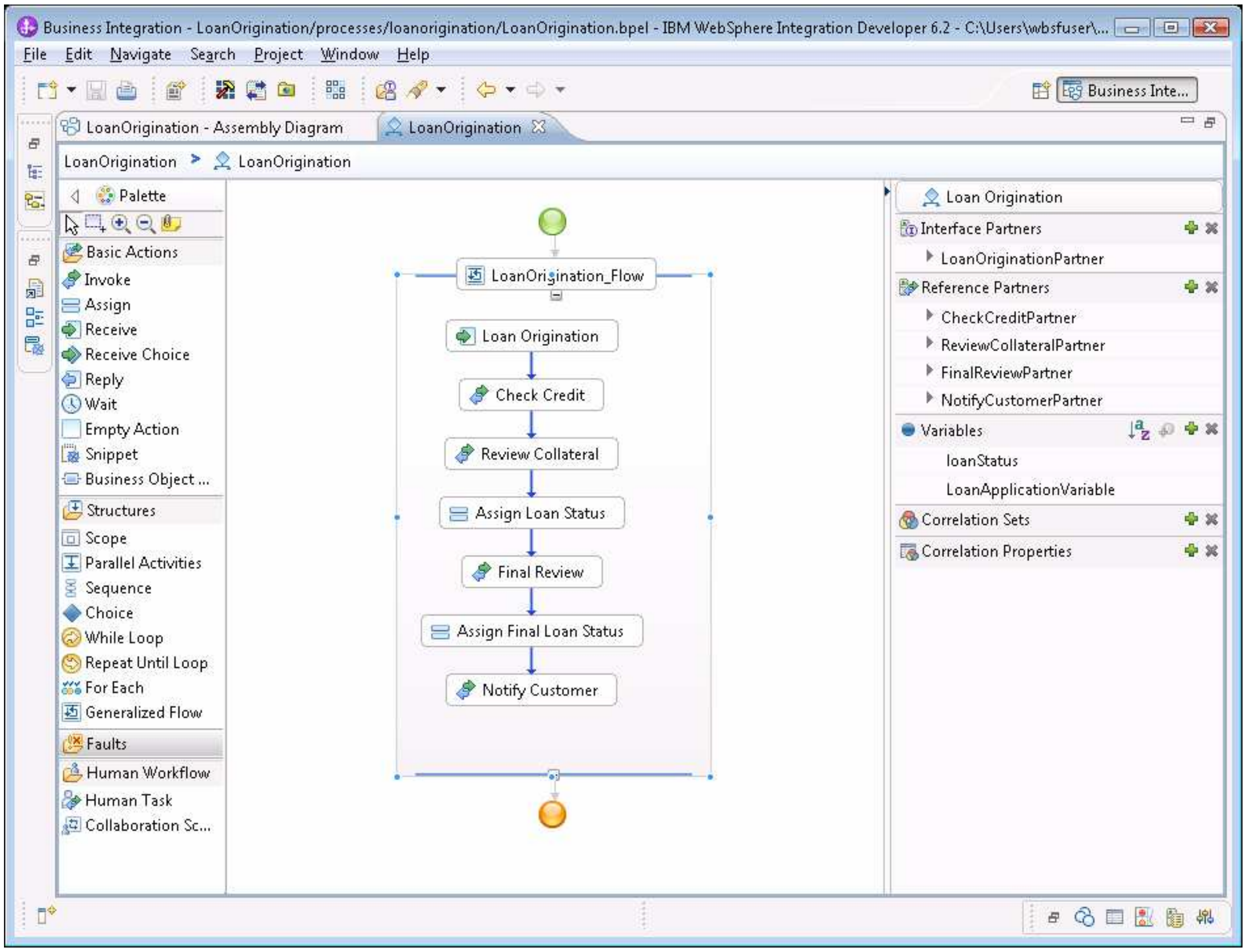
Exploring the Loan Origination assembly

1. Open the assembly diagram for the LoanOrigination SCA module.


Business Integration - LoanOrigination/sca.module - IBM WebSphere Integration Developer 6.2 - C:\Users\wbsfuser\IBM\wid6.2\workspace_new

This module exports the Loan Origination process and plays the role of the Business Application module. The process has one partner link for each business service that was modeled in the Fabric authoring space and these are wired to dynamic assembly components. At each step in the process, the dynamic assembly component is invoked. The dynamic assembler will use business policies and other configuration to dynamically select and invoke the most appropriate process variation to select for each business service. Wired to each dynamic assembler is a Java component that implements the ContextExtractor interface. These components are invoked by the dynamic assembler to provide context elements needed for evaluating policies. Notes attached to the context extractors are used to indicate the context provided for each dynamic assembler. You will cover the steps for defining context extractors in the next chapter.

2. Double click the LoanOrigination process to open the BPEL editor.



After an initial receive step, the Check Credit action invokes the check credit partner with a loan application and receives the loan application back with a credit score filled in.

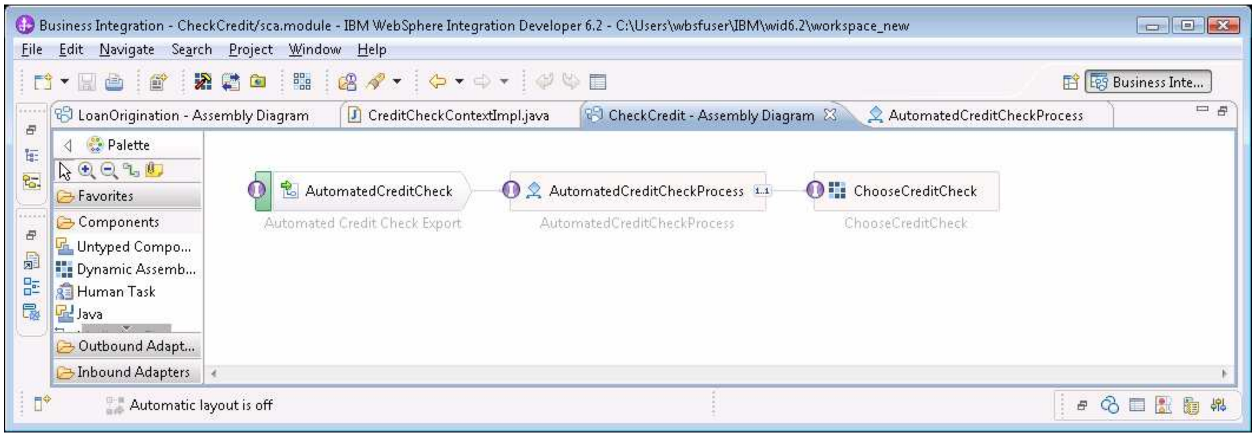
The next invokes the review collateral partner with the loan application. The response is an updated loan status which is saved to the loanStatus variable. The following assign action updates the loan application with the value of loanStatus returned from the collateral review.

The final review step is similar in that the request is again a loan application and the response is an updated loanStatus.

The final action in the process invokes the notify customer partner with the loan application. The NotifyCustomer interface is modeled as a one-way operation.

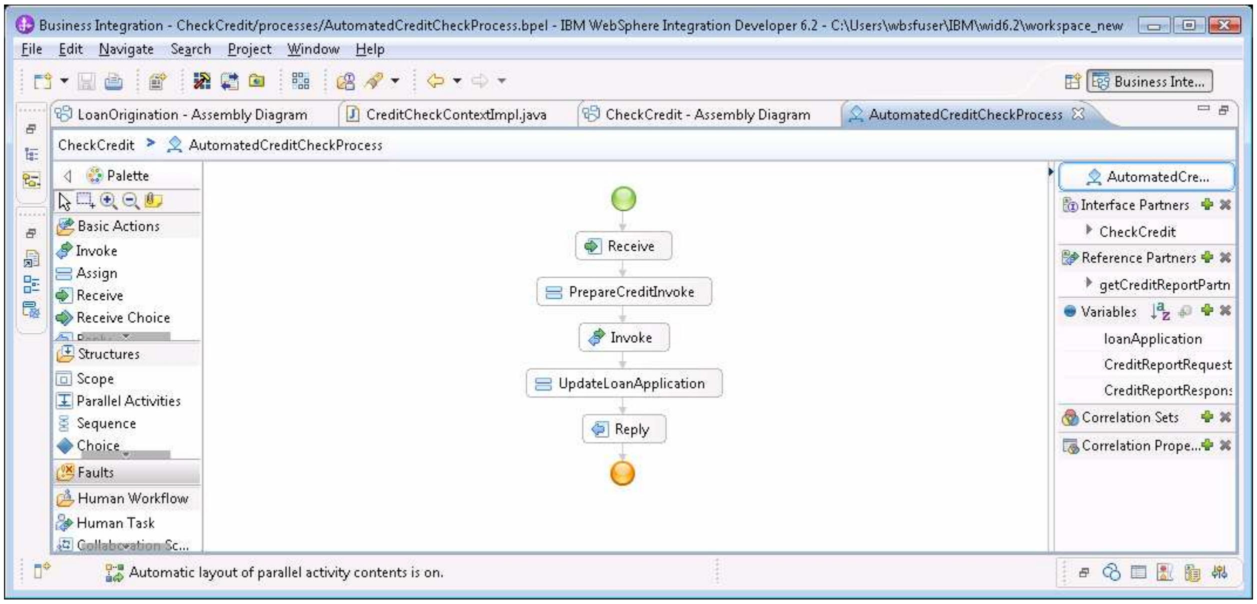
Exploring the CheckCredit assembly

1. Open the assembly diagram for the CheckCredit module.


Business Integration - CheckCredit/sca.module - IBM WebSphere Integration Developer 6.2 - C:\Users\wbsfuser\IBM\wid6.2\workspace_new
File Edit Navigate Search Project Window Help
Business Inte...
LoanOrigination - Assembly Diagram
CreditCheckContextImpl.java
CheckCredit - Assembly Diagram
AutomatedCreditCheckProcess
Palette
Favorites
Components
Untyped Compo...
Dynamic Assemb...
Human Task
Java
Outbound Adapt...
Inbound Adapters
AutomatedCreditCheck
Automated Credit Check Export
AutomatedCreditCheckProcess
AutomatedCreditCheckProcess
ChooseCreditCheck
ChooseCreditCheck
Automatic layout is off

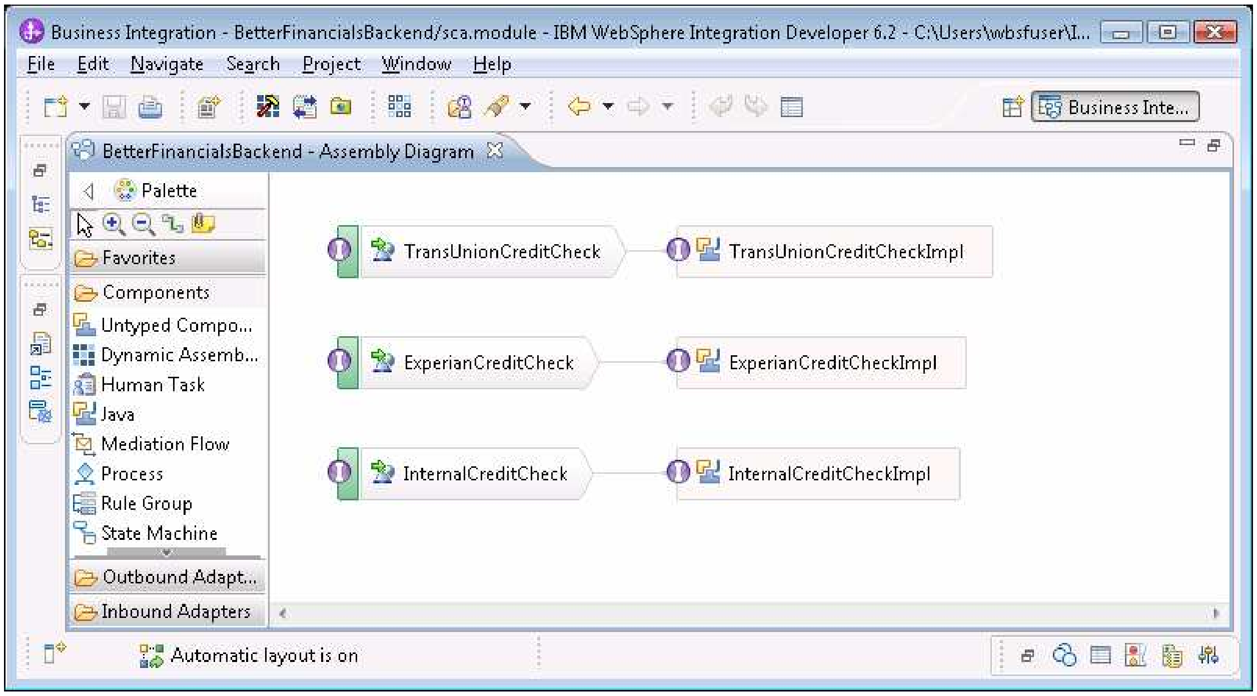
The ChooseCreditCheck dynamic assembly component uses policies to dynamically bind to the most appropriate of three backend credit report services that implement the GetCreditReport interface. Note that no context extraction is needed here. In this case, the necessary context is established by the CreditCheck dynamic assembler in the loan origination process. At runtime, the necessary context elements are propagated so that they can be accessed anywhere within the automated credit check process variation.

2. Double-click to open the AutomatedCreditCheckProcess.


Business Integration - CheckCredit/processes/AutomatedCreditCheckProcess.bpel - IBM WebSphere Integration Developer 6.2 - C:\Users\wbsfuser\IBM\wid6.2\workspace_new
File Edit Navigate Search Project Window Help
Business Inte...
LoanOrigination - Assembly Diagram
CreditCheckContextImpl.java
CheckCredit - Assembly Diagram
AutomatedCreditCheckProcess
CheckCredit > AutomatedCreditCheckProcess
Palette
Basic Actions
Invoke
Assign
Receive
Receive Choice
Structures
Scope
Parallel Activities
Sequence
Choice
Faults
Human Workflow
Human Task
Collaboration Sc...
Receive
PrepareCreditInvoke
Invoke
UpdateLoanApplication
Reply
AutomatedCre...
Interface Partners
CheckCredit
Reference Partners
getCreditReportPartn
Variables
loanApplication
CreditReportRequest
CreditReportRespons
Correlation Sets
Correlation Prope...
Automatic layout of parallel activity contents is on.

The interface to this process takes a loan application and returns a loan application with a credit score. The process is used to create a CreditReportRequest, invoke the getCreditReport reference partner, and update the loan application with the credit score from the CreditReportResponse.

3. Open the assembly diagram for the BetterFinancialsBackend module.

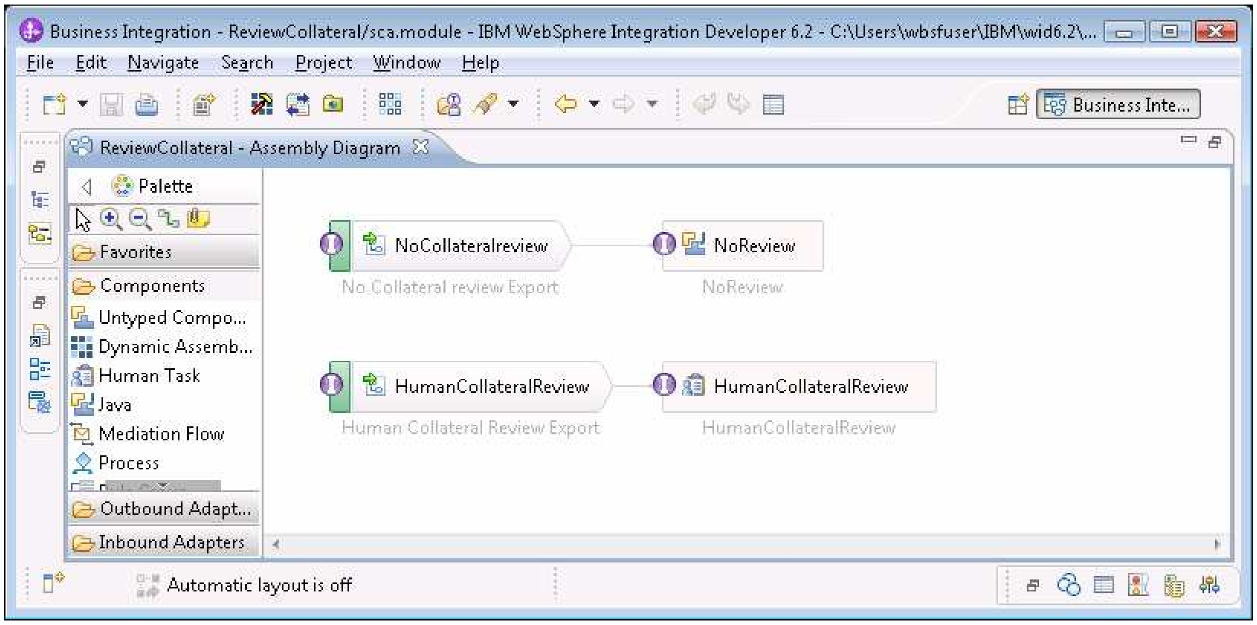


The BetterFinancialsBackend module provides simple mock implementations of the three backend credit services exposed through Soap/HTTP exports. To make it easier to distinguish which service is selected, each mock implementation returns a different fixed credit score. The following table shows the credit score returned by each implementation:

Credit Service	Mock Credit Score
Trans Union	500
Experian	600
Internal Credit Check	700

Exploring the Review Collateral assembly

1. Open the Assembly Editor for the ReviewCollateral SCA Module.


Business Integration - ReviewCollateral/sca.module - IBM WebSphere Integration Developer 6.2 - C:\Users\wbsfuser\IBM\wid6.2\...
File Edit Navigate Search Project Window Help
Business Inte...
ReviewCollateral - Assembly Diagram
Palette
Favorites
Components
Untyped Compo...
Dynamic Assemb...
Human Task
Java
Mediation Flow
Process
Outbound Adapt...
Inbound Adapters
NoCollateralreview
No Collateral review Export
NoReview
NoReview
HumanCollateralReview
Human Collateral Review Export
HumanCollateralReview
HumanCollateralReview
Automatic layout is off

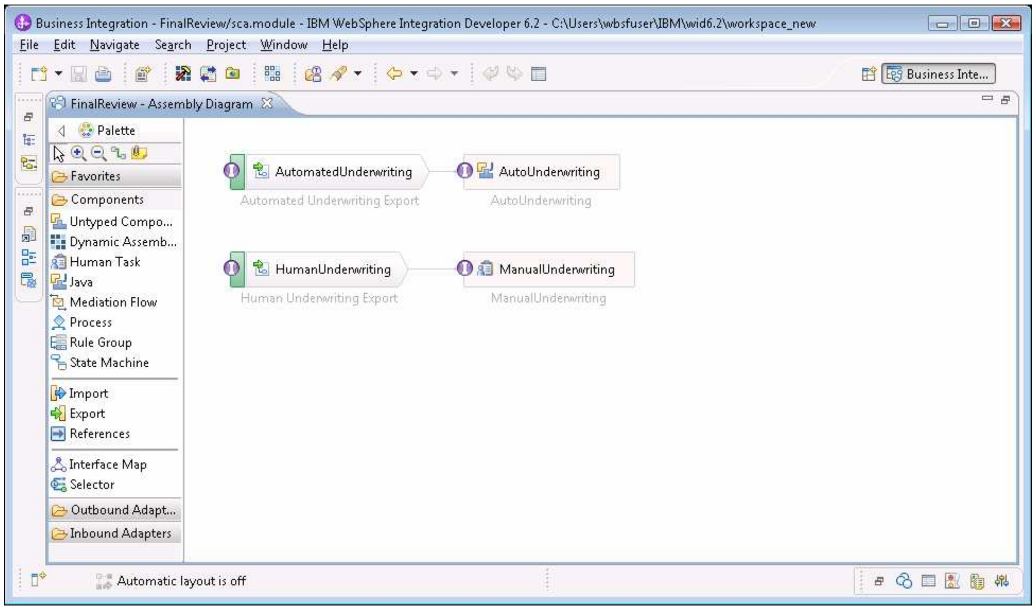
This module provides implementations of two process variations for the Review Collateral business service. Although this example and others in this guide place variations in the same module, consider putting each variation in its own module. That way, a new service can be introduce or an old one modified by deploying a new or updated module for that one process variation.

The No Collateral Review implementation will update the loanStatus to the INPROGRESS state if no loan status is provided.

The human collateral review launches a human task for someone to review the collateral and update the status of the loan. The human task is configured so that the work can be performed in the Business Process Choreographer.

Exploring the Final Review assembly

1. Open the assembly editor for the Final Review SCA module.



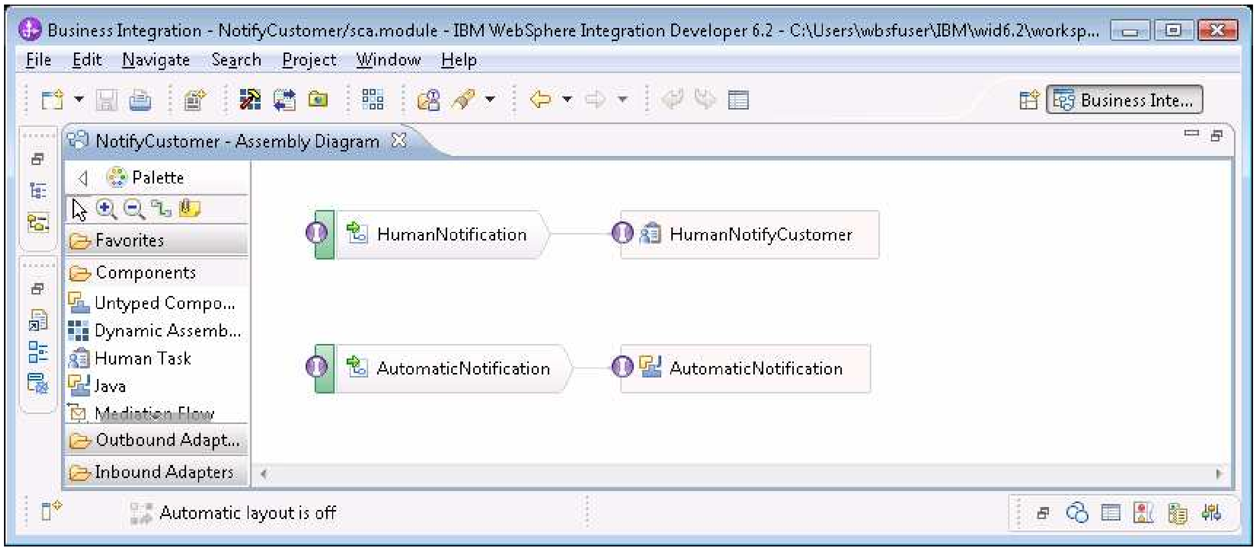
This module provides implementation for the human and automated process variations for the Final Review business service.

The automatic underwriting Java implementation will set the loan status to APPROVED unless the loan has already been rejected or flagged for needing additional information.

The manual underwriting human task will allow an underwriter to review the loan and determine the final status.

Exploring the Notify Customer assembly

1. Open the assembly diagram for the NotifyCustomer SCA module.



The Notify Customer SCA Module provides implementations for both the human and automated process variations of the Notify Customer business service.

The java implementation of AutomaticNotification “notifies” the customer by printing the customer name and final loan status as a system out message that opens in the server console at runtime.

The human implementation will create a human task so that customers can be directly contacted by a bank representative.