iWay

iWay Adapter for SWIFT for BEA WebLogic
User's Guide
Version 5 Release 5

DN3501514.0205

Preface

This documentation describes how to use the iWay Adapter for SWIFT for BEA WebLogic. It is intended for developers to enable them to parse, transform, and integrate financial information into the existing enterprise and pass information electronically to partners in SWIFT mandated form.

How This Manual Is Organized

The following table lists the numbers and titles of the chapters and appendix for this manual with a brief description of the contents of each chapter and appendix.

Chapter/Appendix		Contents
1	Introducing the iWay Adapter for SWIFT for BEA WebLogic	Introduces the iWay Adapter for SWIFT for BEA WebLogic.
2	Creating XML Schemas or Web Services for the iWay Adapter for SWIFT	Describes how to open a connection to SWIFT, create XML request and response schemas for SWIFT services, and create business services (or Web services).
3	Listening for Events in SWIFT	Describes how to use the adapter to listen, react, and dispose of event data.
4	Using Web Services Policy-Based Security	Describes how to configure Web services policy-based security.
5	Management and Monitoring	Describes how to configure and use managing and monitoring tools provided by iBSE and JCA to gauge the performance of your run-time environment.
6	Troubleshooting and Error Messages	Explains limitations and workarounds when connecting to SWIFT.
A	Using Application Explorer in BEA WebLogic Workshop to Create XML Schemas and Web Services	Describes how to use iWay Java Swing Application Explorer running in BEA WebLogic Workshop to create XML schemas for SWIFT.

B	Using Application Explorer in BEA WebLogic Workshop for Event Handling	Describes how to use iWay Java Swing Application Explorer running in BEA WebLogic Workshop to create events for SWIFT. Provides information on deploying components in a clustered BEA WebLogic environment.

Documentation Conventions

The following table lists the conventions that apply in this manual and a description of each.

Convention	Description
`THIS TYPEFACE` or `this typeface`	Denotes syntax that you must enter exactly as shown.
`this typeface`	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
`underscore`	Indicates a default setting.
this typeface	Represents a placeholder (or variable) in a text paragraph, a cross-reference, or an important term.
this typeface	Highlights a file name or command in a text paragraph that must be lowercase.
this typeface	Indicates a button, menu item, or dialog box option you can click or select.
Key + Key	Indicates keys that you must press simultaneously.
`{ }`	Indicates two or three choices; type one of them, not the braces.
`|`	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
`...`	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (…).
`.` `.` `.`	Indicates that there are (or could be) intervening or additional commands.

Related Publications

Visit our World Wide Web site, http://www.iwaysoftware.com, to view a current listing of our publications and to place an order. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have questions about the iWay Adapter for SWIFT for BEA WebLogic?

If you bought the product from a vendor other than iWay Software, contact your distributor.

If you bought the product directly from iWay Software, call Information Builders Customer Support Service (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 a.m. and 8:00 p.m. EST to address all your iWay Adapter for SWIFT for BEA WebLogic questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code number (*xxxx.xx*) when you call.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our World Wide Web site, http://www.informationbuilders.com. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of www.informationbuilders.com also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Help Us to Serve You Better

To help our consultants answer your questions effectively, please be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following tables list the specifications our consultants require.

Platform	
Operating System	
OS Version	
Product List	
Adapters	
Adapter Deployment	For example, JCA, Business Services Engine, iWay Adapter Manager
Container Version	

The following table lists components. Specify the version in the column provided.

Component	**Version**
iWay Adapter	
EIS (DBMS/APP)	
HOTFIX / Service Pack	

The following table lists the types of Application Explorer. Specify the version (and platform, if different than listed previously) in the columns provided.

Application Explorer Type	**Version**	**Platform**
Swing		
Servlet		
ASP		

In the following table, specify the JVM version and vendor in the columns provided.

Version	**Vendor**

The following table lists additional questions to help us serve you better.

Request/Question	**Error/Problem Details or Information**
Provide usage scenarios or summarize the application that produces the problem.	
Did this happen previously?	
Can you reproduce this problem consistently?	
Any **change in the application environment**: software configuration, EIS/ database configuration, application, and so forth?	
Under what circumstance does the problem *not* occur?	
Describe the **steps** to reproduce the problem.	
Describe the **problem**.	
Specify the **error** message(s).	

The following table lists error/problem files that might be applicable.

XML schema
XML instances
Other input documents (transformation)
Error screen shots
Error output files

XML schema
Trace and log files
Log transaction

User Feedback

In an effort to produce effective documentation, the Documentation Services staff welcomes your opinions regarding this manual. Please use the Reader Comments form at the end of this manual to communicate suggestions for improving this publication or to alert us to corrections. You also can go to our Web site, http://www.iwaysoftware.com and use the Documentation Feedback form.

Thank you, in advance, for your comments.

iWay Software Training and Professional Services

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our World Wide Web site, http://www.iwaysoftware.com or call (800) 969-INFO to speak to an Education Representative.

Interested in technical assistance for your implementation? Our Professional Services department provides expert design, systems architecture, implementation, and project management services for all your business integration projects. For information, visit our World Wide Web site, http://www.iwaysoftware.com.

Contents

CHAPTER 1

Introducing the iWay Adapter for SWIFT for BEA WebLogic

Topics:

- Overview
- How the iWay Adapter for SWIFT for BEA WebLogic Works
- Deployment Information for the iWay Adapter for SWIFT for BEA WebLogic

This section presents an overview of the iWay Adapter for SWIFT for BEA WebLogic and explains how the adapter works.

Overview

The iWay Adapter for SWIFT transforms documents into XML format and transforms XML representations of SWIFT documents back into SWIFT format.

After the adapter transforms the documents into XML format, the information can be integrated into back or front office systems using the iWay application and data adapters that are available from the iWay Adapter Suite of products.

J2EE™ standard interfaces and protocols such as JCA, JDBC™, and JMS™ are also supported with the iWay Adapter for SWIFT. The same adapters can be used to obtain the information required to populate SWIFT messages. For example, you can use the SQL Adapter updates in an RDBMS to trigger a SQL query that returns a XML formatted answer set that can be mapped to a SWIFT message.

Data dictionaries, which describe the XML format, are used to enable the mapping of XML documents to SWIFT format and SWIFT messages to XML format.

After structural integrity has been verified during the transformation stage, the BEA WebLogic Server performs validation using a set of rules defined in an XML formatted rules file.

Where applicable, acknowledgment documents are returned to the sending application, but only if the incoming document is structurally valid. If the content validation fails, an error code is returned in the acknowledgement document, if one is expected.

How the iWay Adapter for SWIFT for BEA WebLogic Works

The iWay Adapter for SWIFT supports bi-directional transformation using XML as the intermediate protocol. The adapter is supplied with templates to convert SWIFT messages to XML format and XML to SWIFT format. The SWIFT documents are described using data dictionaries supplied with the adapter. The data dictionaries conform to the SWIFT green books, and the format for describing data elements conform to the standards in the books. These dictionaries are in XML format and you can edit them to tailor messages to individual bank and/or market standards.

Validating SWIFT Messages

The SWIFT messages are validated in two ways. The first way is by using the dictionaries to parse the document and validate the structure of the message type and tag structure.

The second method of validating messages uses the rules engine to provide content (domain and network) validation. The SWIFT document is validated by use of a rule file in XML format that applies pre-built rules based on the XML tag. These are can be customized, and you can write your own rules to apply extra business logic.

Depending on the direction (XML to SWIFT or SWIFT to XML), the content validation occurs before or after structural validation. The following section explains how transformation and validation works for both inbound (receiving a SWIFT message) and outbound (creating a SWIFT message) processes.

Inbound Processing

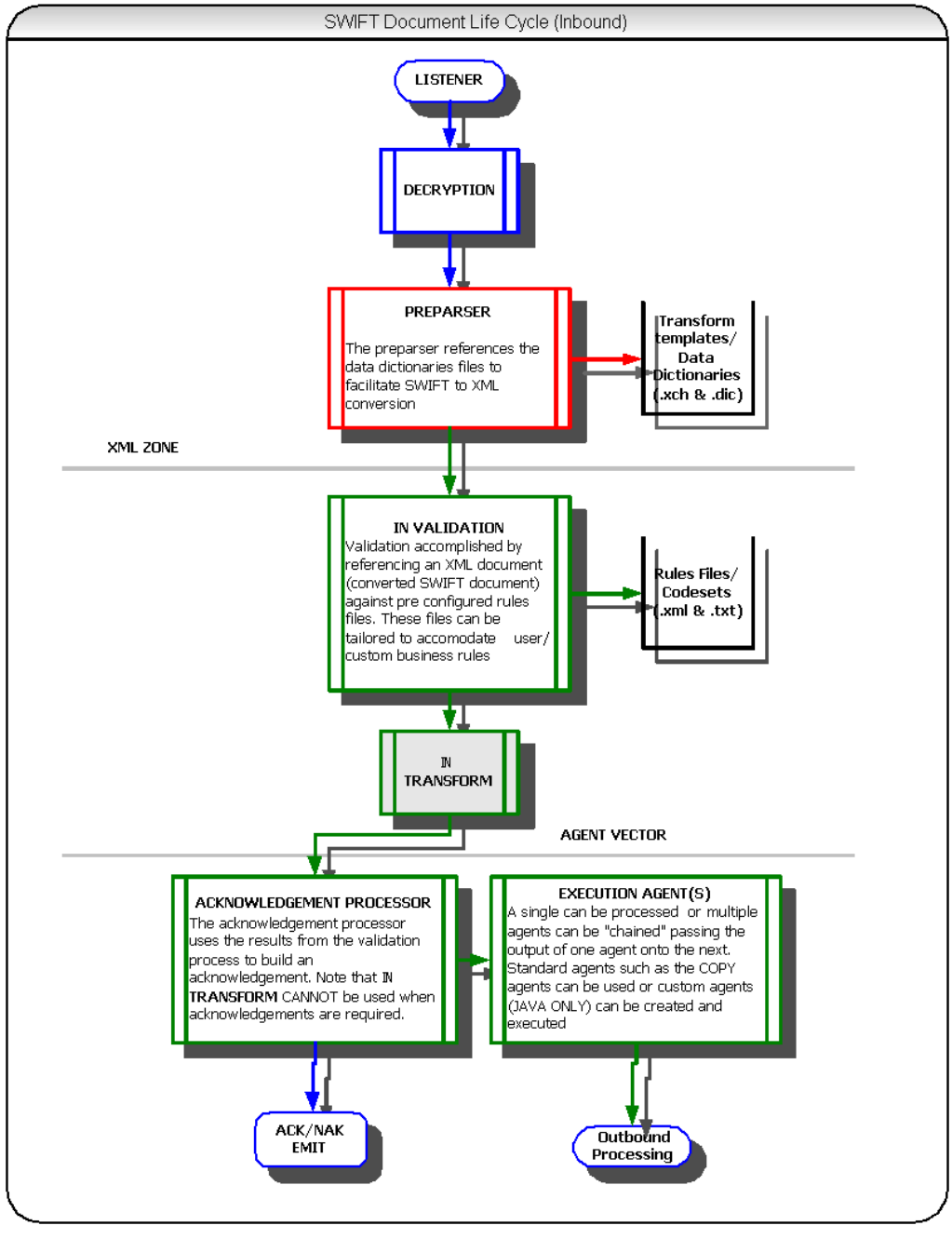
A configured listener picks up an inbound SWIFT message. The first step in the process is to apply an encryption/decryption algorithm. This is a configurable process and you can use any Java™-based encryption processor.

The next step is the pre-parse stage. Because the BEA WebLogic Server maps input and output using XML format, the document (if not in XML format at this stage) must be converted to XML. This is when the pre-built (or customized) transformation templates are called to convert SWIFT format to XML.

When the documents are in XML format, the content of the SWIFT document can be validated based on supplied (or customized) rules files. iWay supplies pre-built rules files that apply validation rules to tags or groups of tags in the XML document, specific to the SWIFT green books.

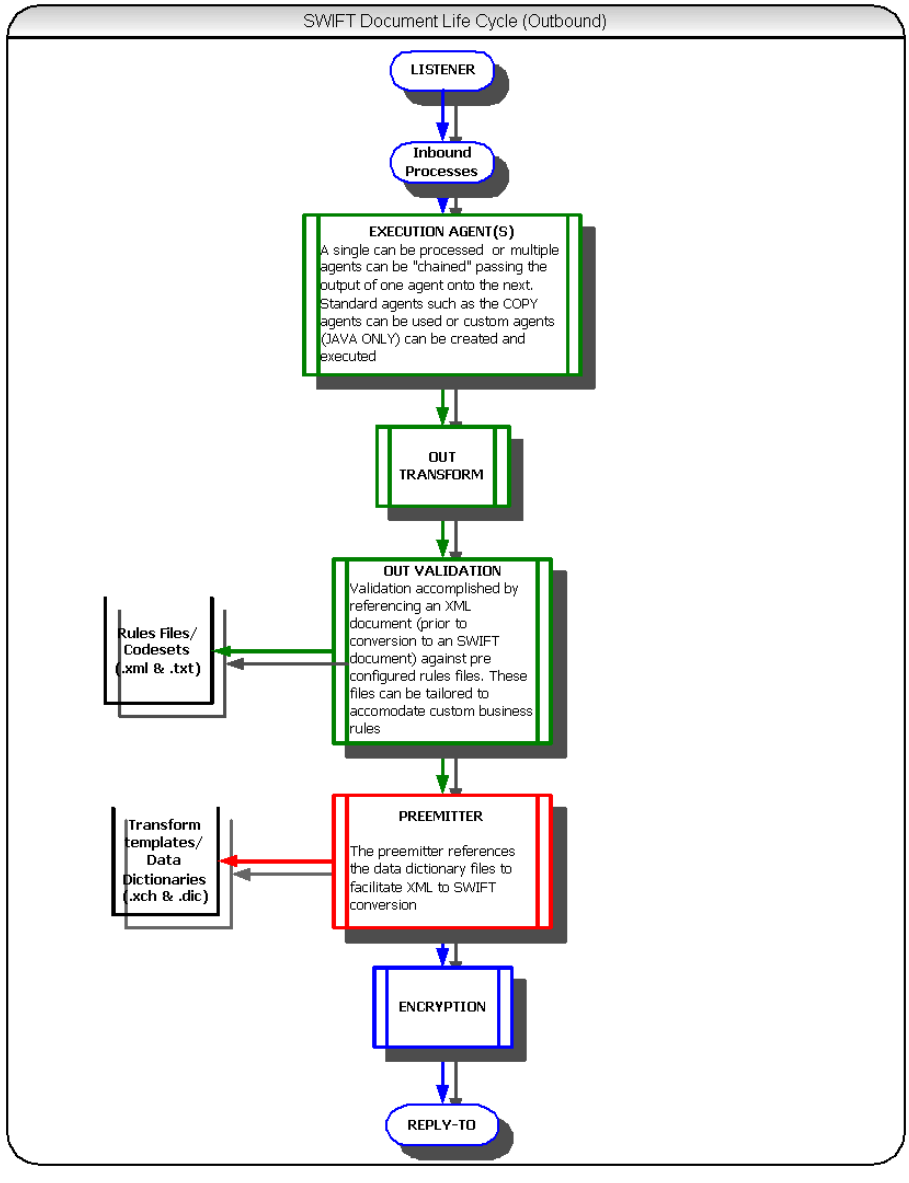
After the XML document is parsed and validated, it can be processed according to your site specification. The process flow can end here (for example, parse, convert, validate, and apply business logic). You also can configure all steps for the outbound process, which is the reverse to the inbound process.

The following diagram illustrates the inbound process:



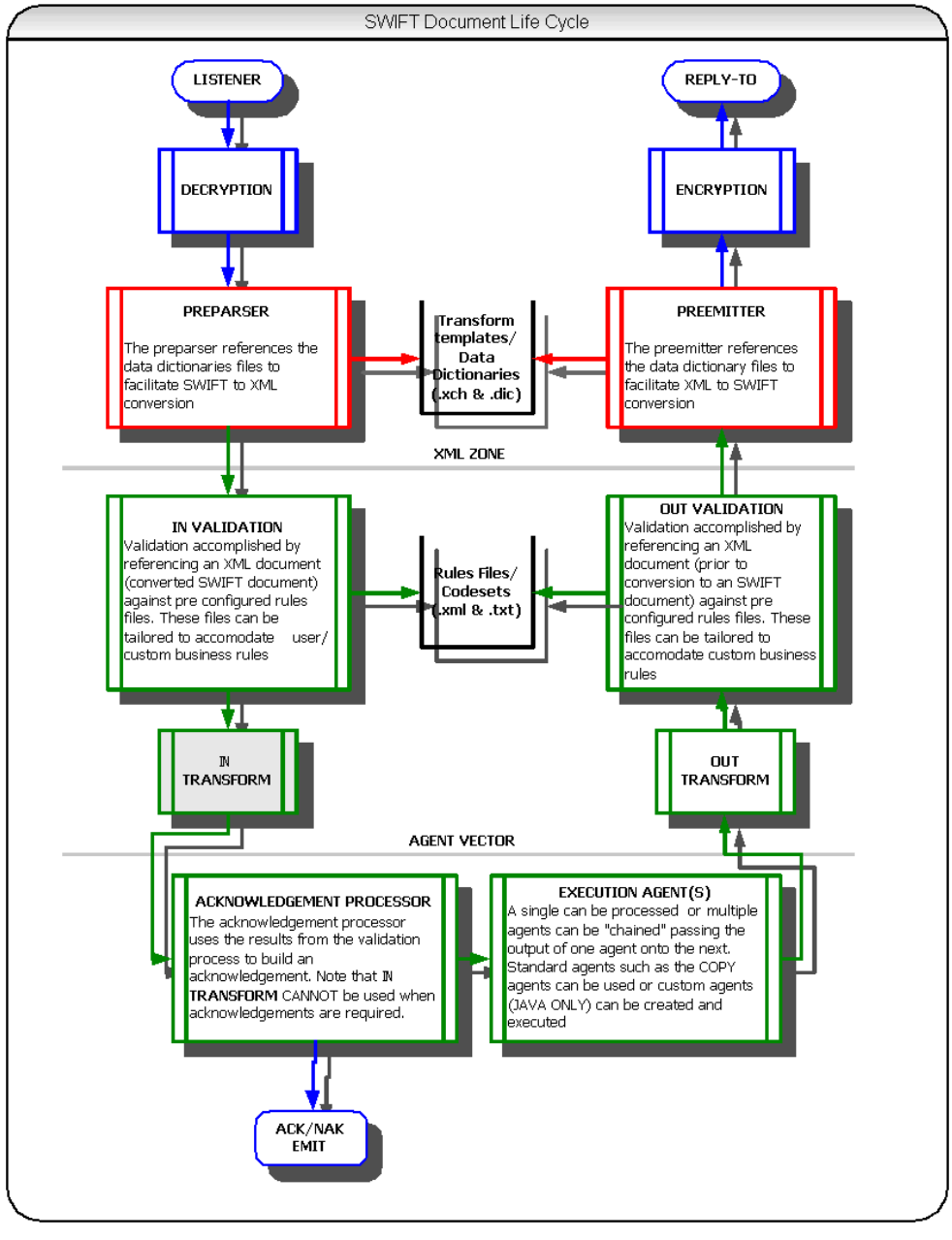
Outbound Processing

The outbound process is the reverse of the inbound process. A document can be received in XML format and have business logic applied to it. The rules engine then validates the document and transforms it into a SWIFT document at the pre-emit stage in the process. The following diagram illustrates outbound processing.



Process Flow

Both inbound and outbound processes can be run in one pass as illustrated in the following diagram:



Deployment Information for the iWay Adapter for SWIFT for BEA WebLogic

The iWay Adapter for SWIFT works in conjunction with the following components:

- iWay Application Explorer

with either

- Integration Business Services Engine (iBSE)

or

- iWay Enterprise Connector for J2EE™ Connector Architecture (JCA)

Application Explorer, used to configure connections and create Web services and events, can be configured to work in a Web services environment in conjunction with the Integration Business Services Engine or with the iWay Enterprise Connector for J2EE Connector Architecture (JCA). When working in a JCA environment, the connector uses the Common Client Interface (CCI) to provide fast integration services using iWay Adapters instead of using Web services.

Both iBSE and the iWay connector for JCA are deployed to an application server such as BEA WebLogic Server with iWay Application Explorer and the adapters.

Deployment Information Roadmap

The following table lists the location of deployment information for the iWay Adapter for SWIFT and iWay Application Explorer. A description of the Integration Business Services Engine (iBSE) and the iWay Enterprise Connector for J2EE Connector Architecture (JCA) follow the table.

Deployed Component	For more information, see
iWay Application Explorer	• Chapter 2 and 3 of this guide • *iWay Installation and Configuration for BEA WebLogic* • *iWay Servlet Application Explorer for BEA WebLogic*
Integration Business Services Engine (iBSE)	• *iWay Installation and Configuration for BEA WebLogic*

Deployed Component	For more information, see
iWay Enterprise Connector for J2EE Connector Architecture (JCA)	• *iWay Connector for JCA for BEA WebLogic User's Guide* • *iWay Installation and Configuration for BEA WebLogic*

The Integration Business Services Engine

The Integration Business Services Engine (iBSE) exposes—as Web services—enterprise assets that are accessible from adapters regardless of the programming language or the particular operating system.

iBSE simplifies the creation and execution of Web services when running:

- Custom and legacy applications
- Database queries and stored procedures
- Packaged applications
- Terminal emulation and screen-based systems
- Transactional systems

Web services is a distributed programming architecture that overcomes hurdles with Enterprise Application Integration (EAI) that other programming models cannot. It enables programs to communicate with one another using a text-based but platform- and language-independent message format called XML.

Coupled with a platform- and language-independent messaging protocol called SOAP (Simple Object Access Protocol), XML enables application development and integration by assembling previously built components from multiple Web services.

The iWay Enterprise Connector for J2EE Connector Architecture

The iWay Enterprise Connector for J2EE Connector Architecture (JCA) enables developers of JCA-compliant applications to deploy iWay adapters as JCA resources. The connector is supported on iWay 5.5 run-tim environment.

The iWay Connector for JCA is distributed as a standard Resource Adapter Archive (RAR) for deployment to the application server. Thus, the connector can be used in systems that are non-compliant, although services such as pooled connections are not available.

CHAPTER 2

Creating XML Schemas or Web Services for the iWay Adapter for SWIFT

Topics:

- Overview
- Starting iWay Servlet Application Explorer
- Establishing a Target for SWIFT
- Creating a Schema
- Creating a Web Service

This section describes how to use iWay Servlet Application Explorer to create XML schemas or Web services for the iWay Adapter for SWIFT.

Overview

External applications that access SWIFT through the adapter use either XML schemas or Web services to pass data between the external application and the adapter. You can use iWay Servlet Application Explorer to create the required XML schemas and Web services.

Application Explorer is a Web application running within a servlet container that is accessible through a Web browser. For more information on installing and configuring the iWay Servlet Application Explorer, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

Starting iWay Servlet Application Explorer

Before you can use iWay Servlet Application Explorer, you must start BEA WebLogic. Then, you can open Application Explorer.

Procedure How to Start BEA WebLogic Server on Windows or on UNIX

1. To start the BEA WebLogic Server on Windows:
   a. Click the *Windows Start menu*.
   b. Select *Programs*, *BEA WebLogic Platform 8.1*, *User Projects, your domain for iWay,* and then, click *Start Server.*
2. To start BEA WebLogic Server on UNIX or from a command line, type the following at the prompt:

   ```
   BEA_HOME\user_projects\domains\DOMAIN_NAME\startWebLogic.cmd
   ```

   where:

   BEA_HOME
   Is the directory where BEA WebLogic is installed.

   DOMAIN_NAME
   Is the domain you are using for iWay.

Procedure How to Open iWay Servlet Application Explorer

To open Application Explorer:

1. Enter the following URL in your browser window:

   ```
   http://hostname:port/iwae/index.html
   ```

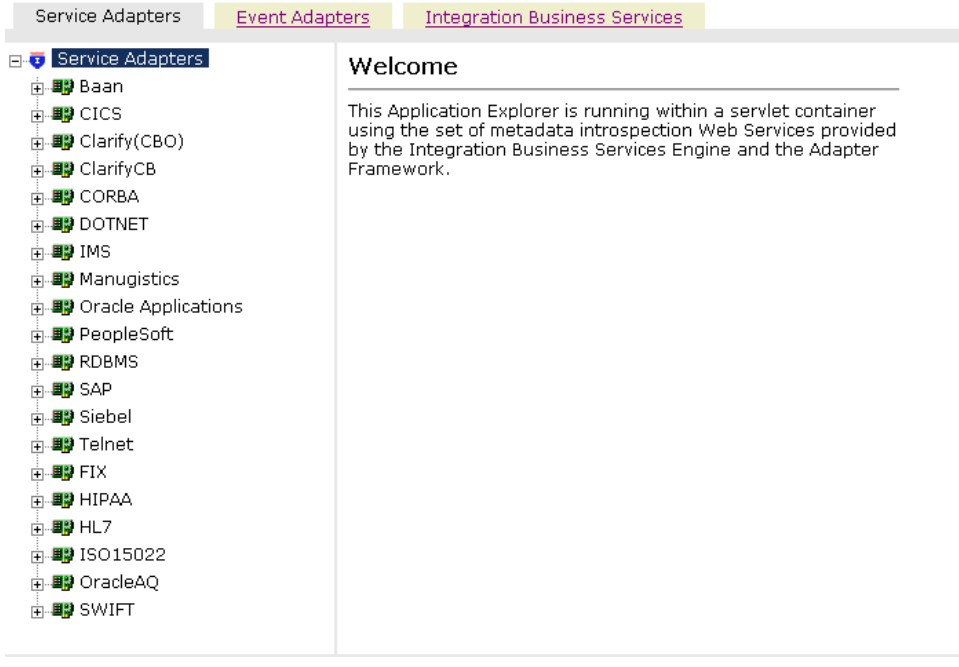
   where:

   hostname
   Is the name of the machine where your application server is running.

port

Is the port where the application server is listening.

After you start Application Explorer, the following Welcome window opens, showing the Service Adapters, Event Adapters, and Integration Business Services tabs. The Service Adapters node is highlighted in the left pane.



The Available Hosts drop-down menu in the upper right lists the iWay Connector for JCA or Servlet iBSE instance you can access.

For more information on adding instances, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

You are now ready to create new targets for SWIFT.

Establishing a Target for SWIFT

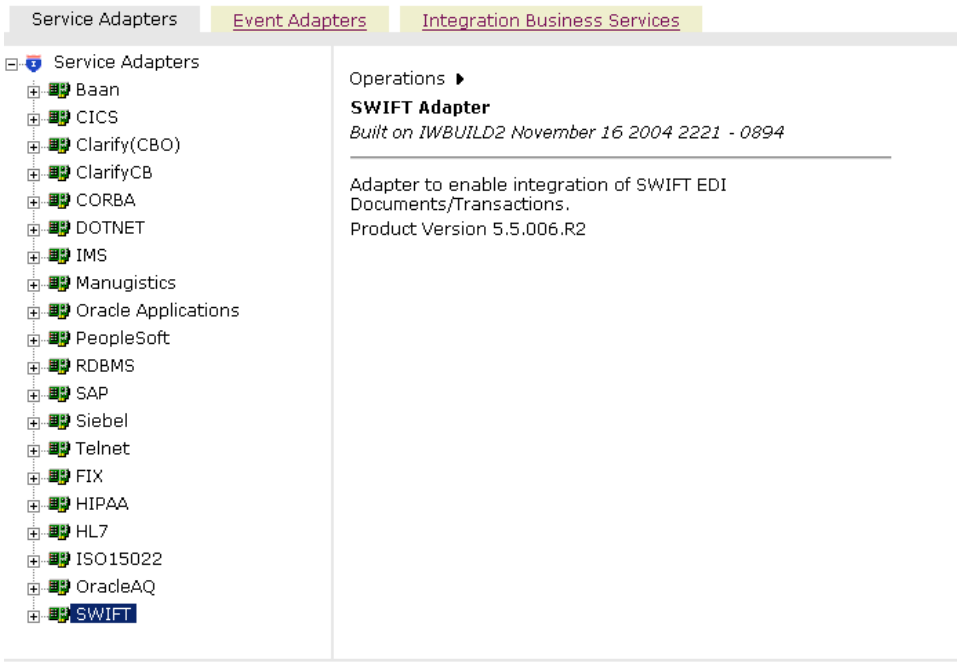
A target serves as your connection point and is automatically saved after you create it. You must establish a connection to SWIFT every time you start iWay Application Explorer or after you disconnect from the system.

Creating a New Target

To connect to SWIFT for the first time, you must create a new target.

Procedure How to Create a New Target

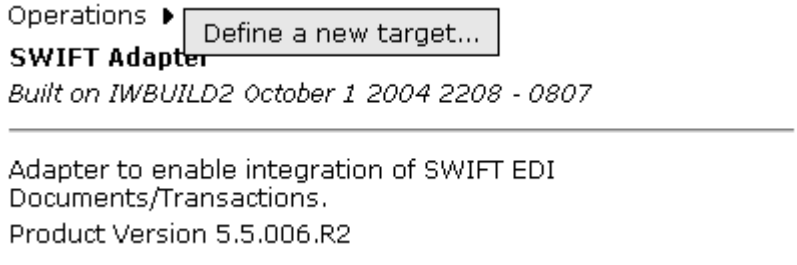
The following graphic shows the list of supported adapters in the left pane and information about the selected adapter in the right pane.



To create a new target:

1. In the left pane, click the *SWIFT* node.

   Descriptive information (for example, title and product version) regarding the iWay Adapter for Swift appears in the right pane.



2. In the right pane, move the pointer over *Operations* and select *Define a new target*.

   The Add a new SWIFT target pane opens on the right. The following illustration shows the fields in the right pane where you enter connection information for the target.

3. Specify the following information for the SWIFT target you are defining.

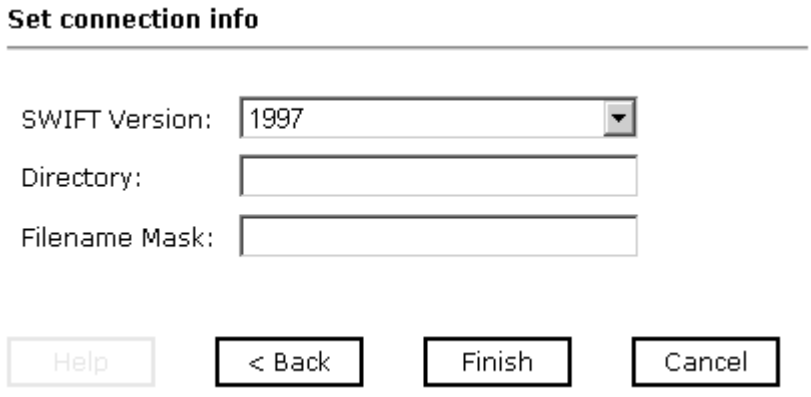
   a. Type a descriptive name and a brief description for the new target.

   b. From the Target Type drop-down list, select one of the following transports from the drop-down list:

   - File System Write. For more information on the properties required, see *File System Write Properties* on page 2-6.
   - File Transfer Protocol (FTP). For more information on the properties required, see *File Transfer Protocol Properties* on page 2-7.
   - HyperText Transfer Protocol (HTTP). For more information on the properties required, see *HyperText Transfer Protocol Properties* on page 2-8.
   - IBM MQSeries (MQ). For more information on the properties required, see *MQSeries Properties* on page 2-9.
   - TCP Session. For more information on the properties required, see *TCP Properties* on page 2-10.

4. Click *Next*.

The Set connection info pane appears on the right and includes fields that are specific to the type of transport you selected.



5. Provide the appropriate information that is specific to the transport you selected.
6. Click *Finish*.

   The following graphic shows the SWIFT target (SWIFTTarget) that appears below the SWIFT node in the left pane.

You are now ready to connect to your SWIFT target.

Reference File System Write Properties

The following table provides definitions for the properties required for the File System Write target type.

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Directory	The directory to which output messages are emitted.

Property	Definition
Filename Mask	The output file name (can contain an asterisk), which gets expanded to a timestamp. A pound sign can be used as a mask for a sequence count. Each pound symbol represents a whole number integer value. For example, File## counts up to 99 before restarting at 0, File### counts up to 999 before restarting at 0, and so on.

Reference

File Transfer Protocol Properties

The following table provides definitions for the properties required for the File Transfer Protocol target type.

Settings tab

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Host	FTP target system.
Port	FTP target system port.
User	User ID to use when connecting to the FTP host.
Password	Password associated with the user ID.
Directory	The directory to which output messages are emitted.

Property	Definition
Filename Mask	The output file name (can contain an asterisk), which gets expanded to a timestamp. A pound sign can be used as a mask for a sequence count. Each pound symbol represents a whole number integer value. For example, File## counts up to 99 before restarting at 0, File### counts up to 999 before restarting at 0, and so on.

Advanced tab

Property	Definition
Retry Interval	The maximum wait interval between retries when a connection fails. Retry interval duration in *xx*H:*xx*M:*xx*S format. For example, 1H:2M:3S is 1 hour 2 minutes and 3 seconds.
Maxtries	Maximum number of retry attempts if a write failure occurs.

Reference HyperText Transfer Protocol Properties

The following table provides definitions for the properties required for the File Transfer Protocol target type.

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
HTTP URL	The HTTP URL.
Header	The HTTP header field.

Reference MQSeries Properties

The following table provides definitions for the properties required for the MQSeries target type.

Settings tab

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Queue Manager	Name of the MQSeries queue manager to be used.
Queue Name	Queue on which request documents are received.
Correlation ID	The correlation ID to set in the MQSeries message header.

MQ Client tab

Property	Definition
Host	Name of the MQSeries queue manager to be used.
Port	Queue on which request documents are received.
Channel	The correlation ID to set in the MQSeries message header.

Reference TCP Properties

The following table provides definitions for the properties required for the TCP target type.

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Host	Host name or host address.
Port	TCP listening port.
Encoding	Document character set.

Connecting to a Target

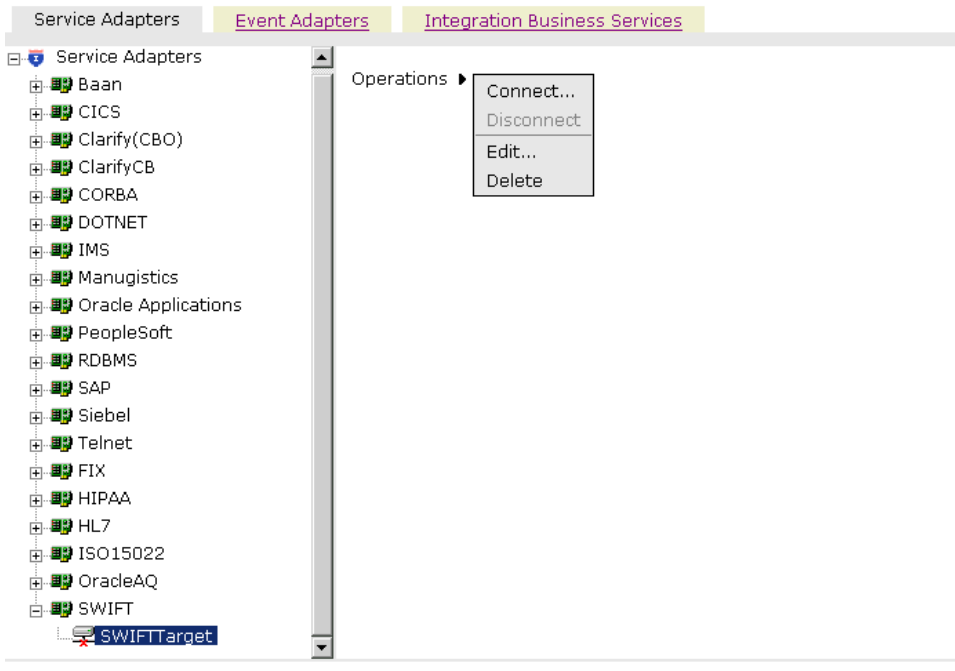
You must use the target you defined to connect to SWIFT.

Procedure How to Connect to a Target

To connect to a target:

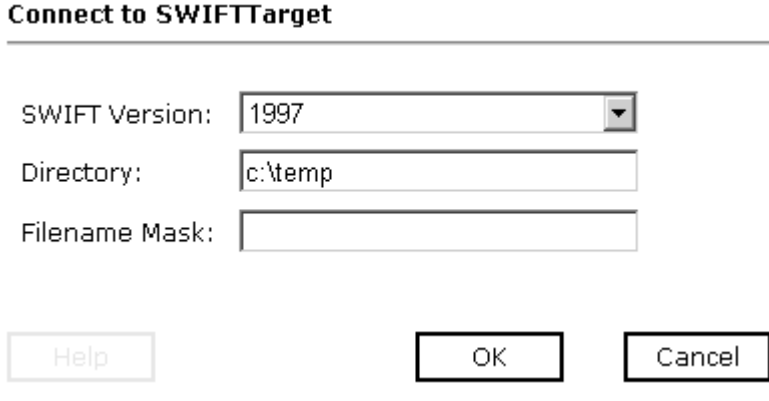
1. In the left pane, expand the *SWIFT* node and select the target you defined, for example, SWIFTTarget.

The following graphic shows the SWIFTTarget node selected in the left pane and the Operations menu expanded in the right pane.



2. Move the pointer over *Operations* and select *Connect*.

The following graphic shows that the Connect to SWIFTTarget pane opens on the right.



3. Click *OK*.

The following graphic shows that the x icon that appeared previously to the left of the SWIFTTarget node has disappeared, indicating that the node is now connected.

The following graphic shows the expanded SWIFTTarget node.

Disconnecting From a Target

Although you can maintain multiple open connections to different application systems, it is a good practice to close connections when you are not using them.

Procedure How to Disconnect From a Target

To disconnect from a target:

1. From the left pane, click the target, for example, SWIFTTarget, to which you are connected.
2. Move the pointer over *Operation* and select *Disconnect*.

   Disconnecting from the application system drops the connection, but the node remains.

Modifying a Target

After you create a target for SWIFT using iWay Servlet Application Explorer, you can edit any of the information that you provided previously.

Procedure How to Edit a Target

To edit a target:

1. In the left pane, click the target, for example, SWIFTTarget.
2. Move the pointer over *Operations* and select *Edit*.
3. Modify the connection information.
4. Click *Next* to continue editing additional fields.

5. When you have completed your edits, click *Finish*.

Deleting a Target

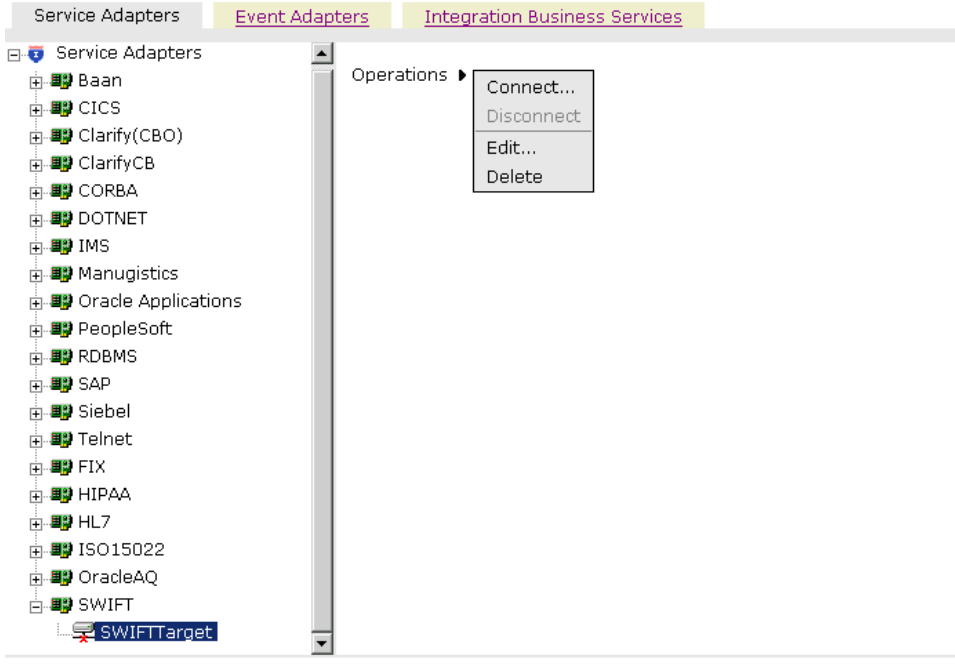
In addition to closing a target, you can delete a target that is no longer required. You can delete it whether or not it is closed. If open, the target automatically closes before it is deleted.

Procedure How to Delete a Target

To delete a target:

1. In the left pane, click the target, for example, SWIFTTarget.

   The following graphic shows the target selected in the left pane, and the operations menu expanded in the right pane.



2. Move the pointer over *Operations* and select *Delete*.
3. To delete the target you selected, click *OK*.

   The SWIFTTarget node disappears from the left pane.

Creating a Schema

You can create service schemas for Business Services and Business Components using iWay Application Explorer.

The following topic, *Creating an XML Schema*, describes how to create schemas for the adapter when you deploy the iWay Adapter for Swift for use either in a JCA (iWay Enterprise Connector for J2EE Connector Architecture) environment or a Web services environment.

If you plan to deploy the iWay Adapter for Swift in a Web services environment, see also *Creating a Web Service* on page 2-16.

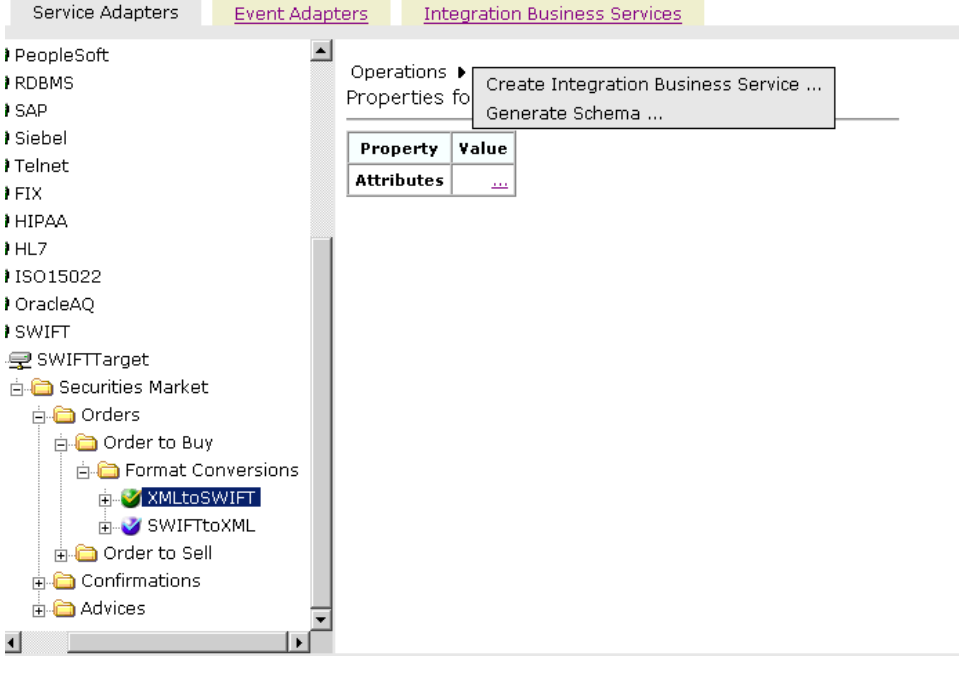
Creating an XML Schema

You create schemas for SWIFT using iWay Application Explorer.

Procedure How to Create an XML Schema

To generate service request and response schemas:

1. If you have not started the explorer, start Application Explorer and connect to your SWIFT system.
2. In the left pane, expand the SWIFTTarget node.
3. Continue expanding nodes to get to the SWIFT document level.



4. In the right pane, move the cursor over *Operations* and select *Generate Schema*.

Application Explorer builds schemas. A schemas table similar to the following appears in the right pane. This table contains three columns labeled Part, Root Tag, and Schema. The Schema column provides hyperlinks to the different schemas.

Schemas

Part	Root Tag	Schema
Request	SWIFTMT500	...
Response	emitStatus	...
Event	N/A	N/A
EventReply	N/A	N/A

Help OK Cancel

5. To view a schema, click the ellipsis (…) in the row corresponding to the schema you want to view.

The following is an example of a request schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!--  Generated by the iBSE 2004-10-05T20:52:10Z
  -->
- <xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema">
  - <xs:element name="SWIFTMT500">
    - <xs:complexType>
      - <xs:sequence>
        - <xs:element maxOccurs="1"
            minOccurs="1" name="BASIC">
          - <xs:complexType>
            - <xs:sequence>
                <xs:element minOccurs="1"
                  name="APPLICATION_ID__"
                  type="xs:string" />
                <xs:element minOccurs="1"
                  name="SERVICE_ID__"
                  type="xs:string" />
                <xs:element minOccurs="1"
```

For more information on where the schemas are stored, see the following topic, *Schema Location*.

Reference Schema Location

Application Explorer stores the schemas it creates in subdirectories under the iWay home directory of the machine where it is installed. The exact location of the schemas differs depending on whether you deploy Application Explorer with an iBSE or a JCA configuration.

- When using the adapter with an iBSE configuration, the schemas are stored under a \schemas subdirectory of the iWay home directory, for example,

  ```
  C:\Program
  Files\iway55\bea\ibse\wsdl\schemas\service\SWIFT\SWIFTTarget
  ```

  where:

  SWIFTTarget

  Is the name of the connection to SWIFT as defined in Application Explorer. Under this directory, Application Explorer creates subdirectories containing schemas.

- When using the adapter with a JCA configuration, the schemas are stored under a \schemas subdirectory of the iWay home directory, for example,

  ```
  C:\Program Files\iWay55\config\base\schemas\SWIFT\SWIFTTarget
  ```

  where:

  SWIFTTarget

  Is the name of the connection to SWIFT as defined in Application Explorer. Application Explorer stores the schemas in this directory.

Creating a Web Service

You can generate a business service (also known as a Web service) for SWIFT operations.

Ensure you properly configure the servlet iBSE. For more information on installing and deploying iWay components, see the *iWay Installation and Configuration for BEA WebLogic* manual.

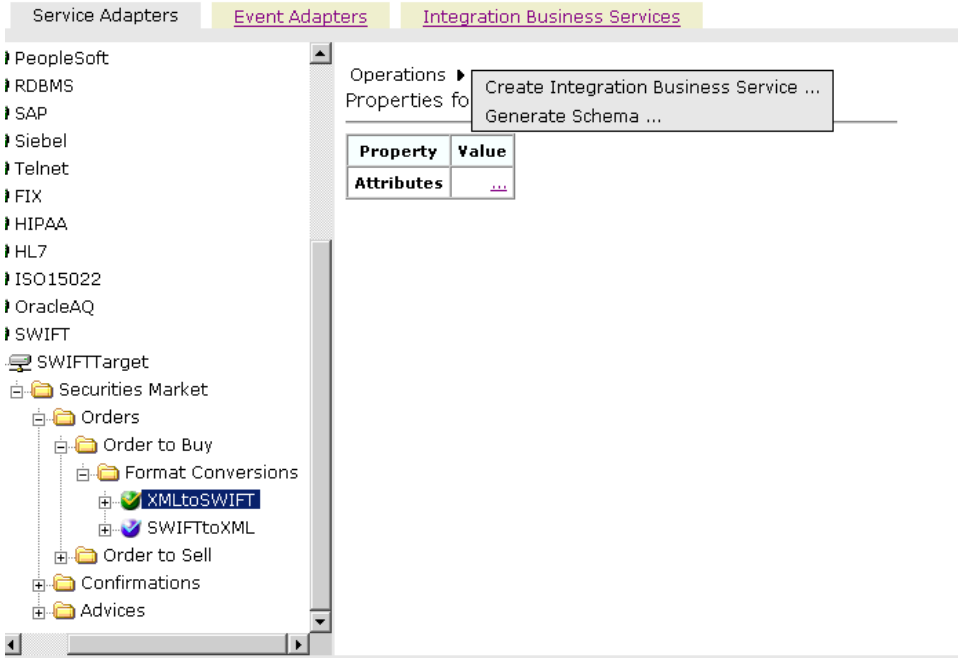
Note: In a J2EE Connector Architecture (JCA) implementation of iWay adapters, Web services are not available. When the adapters are deployed to use the iWay Connector for JCA, the Common Client Interface provides integration services using the iWay adapters. For more information, see the *iWay Installation and Configuration for BEA WebLogic* manual and the i*Way Connector for JCA for BEA WebLogic Server User's Guide*.

Procedure How to Generate a Web Service

To generate a Web service:

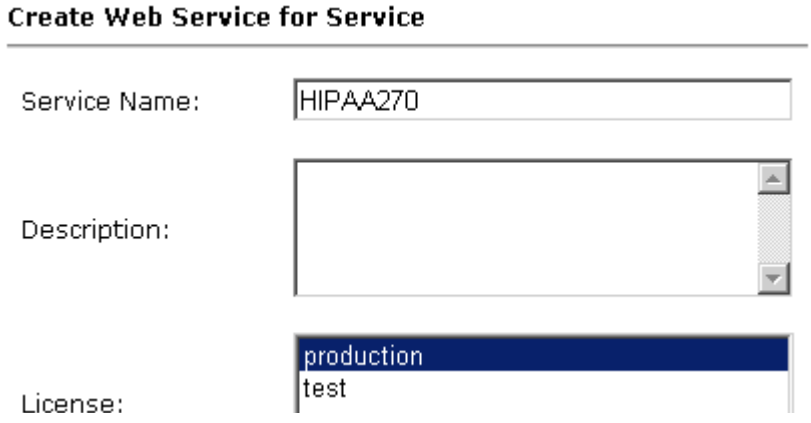
1. If you have not already connected, connect to SWIFT.
2. Expand the *SWIFT* node.
3. Continue expanding nodes down to the *Service* level.

   The following graphic shows the left pane with the Service node selected.



4. In the right pane, move the pointer over *Operations* and select *Create Integration Business Services*.

- If this is not the first Web service you want to create and use, choose whether to create a new service or use an existing service.

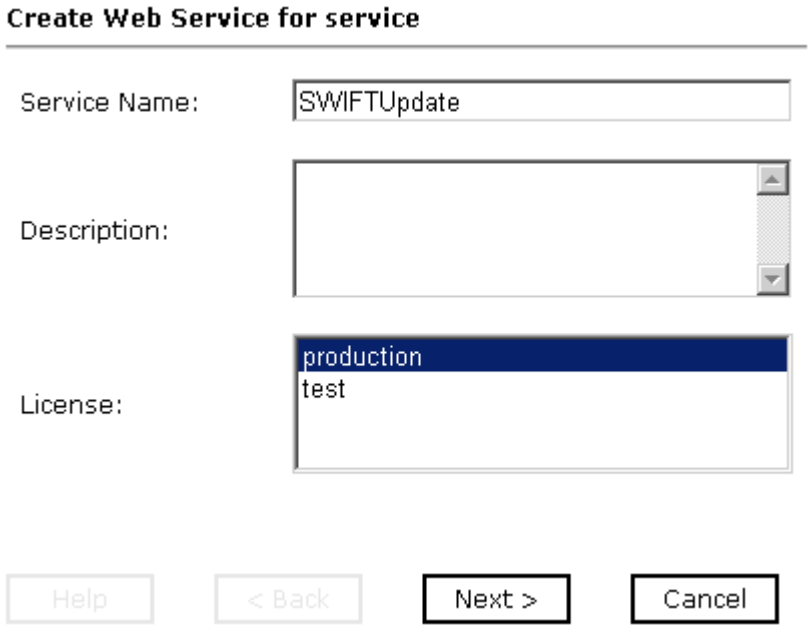


a. To use a previously created service, select the option to use an existing service and click *Next*.

A drop-down list appears.

b. Select the business service to which you want to add the new service and click *Next*.

- If this is the first Web service you are creating or if you select to create a new service, the Create Web Service pane appears. This pane provides three fields followed by a help button and three action buttons.



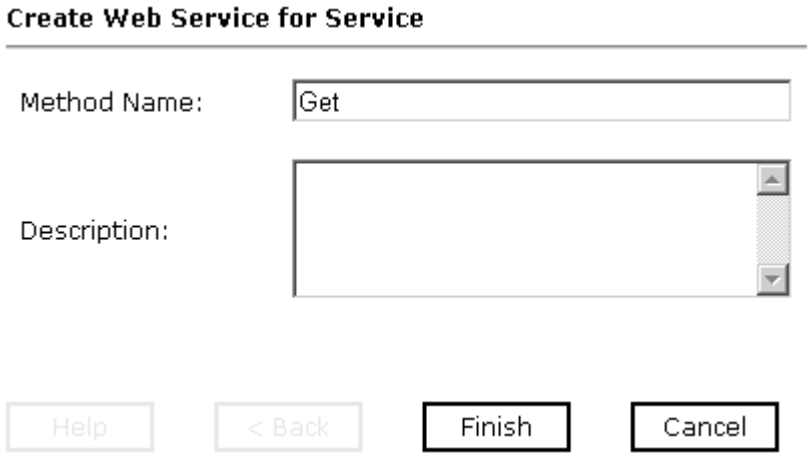
  a. In the Service Name field, type a name to identify the Web service (under the Service node in the left pane of the Integration Business Services tab).

  b. In the Description field, type a brief description of the Web service.

  c. In the License field, select the license(s) with which you want to associate this business service. To select more than one, hold down the *Ctrl* key and click the licenses.

5. Click *Next*.

The right pane displays the next Create Web Service pane, which prompts you for information about the method of the service. It includes two fields, a help button, and three action buttons.

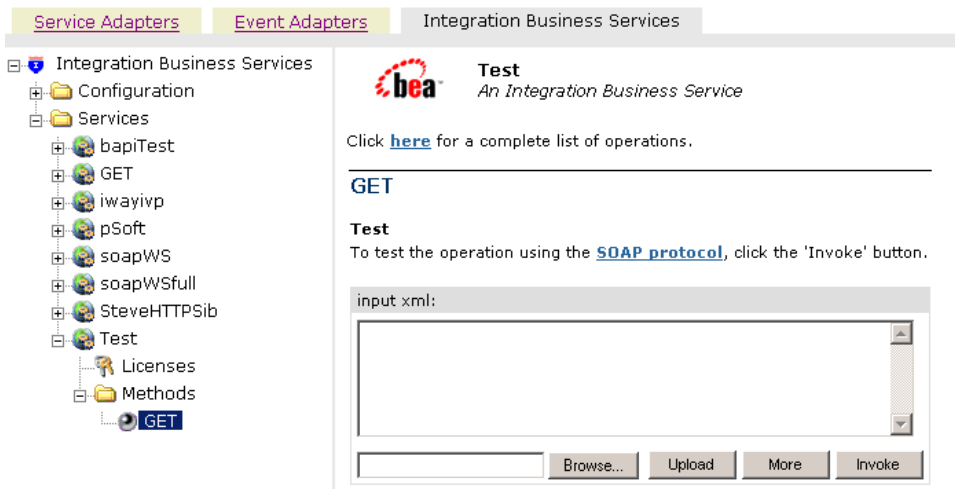


a. In the Method Name field, type a name to specify the name of the method to be added to the business service.

b. In the Description field, type a brief description of the method.

6. Click *Finish*.

Application Explorer switches the view to the Integration Business Services tab, and the new business service appears in the left pane.



Testing a Web Service for a Business Object

After you create a Web service, test it to ensure it functions properly. Application Explorer includes a test tool for testing a Web service.

Procedure How to Test a Web Service for a Business Object

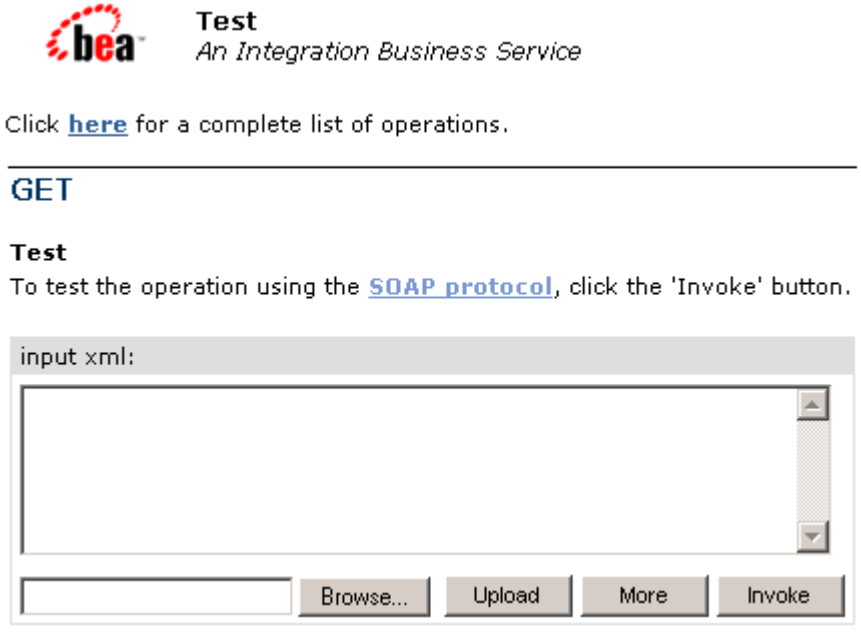
To test a Web service:

1. If you are not on the Integration Business Services tab of Application Explorer, click the tab to access business services.
2. If it is not expanded, expand the *Integration Business Services* node.
3. Expand the *Services* node.
4. Select the name of the business service you want to test.

The business service name appears as a link in the right pane, as shown in the following graphic.

5. In the right pane, click the named business services link, for example, *Get*.

The test option appears in the right pane. This pane provides a text field in which to paste the XML input or browse to a file that can be uploaded.



6. Provide the appropriate XML input.
7. Click *Invoke*.

   Application Explorer displays the results in the results pane on the right.

CHAPTER 3

Listening for Events in SWIFT

Topics:

- Understanding iWay Event Functionality
- Creating an Event Port
- Creating a Channel

The following section describes how to use the iWay Adapter for SWIFT for BEA WebLogic to listen, react, and dispose of event data. iWay Servlet Application Explorer deployed to a BEA WebLogic Server enables you to listen for events.

Understanding iWay Event Functionality

Events are generated as a result of a SWIFT document arriving at a particular queue. You can use documents arriving at a queue to trigger an action in your application. For example, information in a message arriving at a queue can be used to update customer information in a database. If your application must perform an action when this happens, your application is a consumer of this event.

After you create a connection to your application system, you can add events using iWay Servlet Application Explorer. To create an iWay event, you must create a port and a channel.

- Port

  A port associates a particular business object exposed by an adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption. For more information, see *Creating an Event Port on page 3-2*.

- Channel

  A channel represents configured connections to particular instances of back-end or other types of systems. A channel binds one or more event ports to a particular listener managed by an adapter. For more information, see *Creating a Channel* on page 3-14.

Creating an Event Port

The following procedures describe how to create an event port from the Event Adapters tab for various dispositions using Application Explorer.

The following dispositions are available when using the servlet Application Explorer in conjunction with an iBSE implementation. You can switch between an iBSE and a JCA implementation by choosing one or the other from the drop-down menu in the upper right of the Application Explorer.

- File
- iBSE
- MSMQ
- JMS queue
- SOAP
- HTTP
- MQ Series
- MAIL

Note: The MAIL disposition option will be supported in a future release.

The following dispositions are available when using Application Explorer in conjunction with a JCA connector implementation.

- File
- HTTP
- JMS queue
- MQ Series

Procedure **How to Create an Event Port for the File Disposition**

To create a specific event port for the File disposition:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port fields appear on the right, as shown in the following graphic. This pane provides four fields to define the new port, a help button, and two action buttons.

   a. Type a name for the event port and provide a brief description.

   b. From the Disposition Protocol drop-down list, select *FILE*.

c. In the Disposition field, provide a destination where the event data is written.

When pointing Application Explorer to an **iBSE** deployment, use the following format:

```
ifile://[location];errorTo=[pre-defined port name or another disposition url]
```

For example:

```
ifile://D:\in\x.txt;errorTo=ifile://D:\error
```

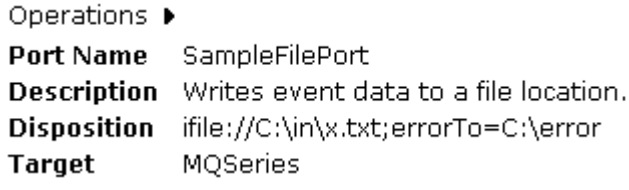
When pointing Application Explorer to a **JCA** deployment, provide the full path to the directory.

The following table defines the parameters for the File disposition.

Parameter	Description
location	The destination and filename of the document where event data is written. For example, C:\in\x.txt.
errorTo	Predefined port name or another disposition URL to which error logs are sent. Optional.

5. Click *OK*.

The event port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created. The summary is shown in the following graphic.



You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page 3-14.

Procedure How to Create an Event Port for iBSE

You can call a Web Service created through Integration Business Services Engine (iBSE).

To create an event port for iBSE:

1. Click the *Event Adapters* tab.

The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port pane opens on the right.

   a. In the Port Name field, type a name for the connection.

      The name is used to build a repository entry as well as to identify the connection.

   b. In the Description field, type a description for the target name you just created.

   c. From the Disposition Protocol drop-down list, select *iBSE*.

   d. In the Disposition field, enter an iBSE destination in the form of:

```
ibse:svcName.mthName;responseTo=[pre-defined port name or another
disposition url];errorTo=[pre-defined port name or another
disposition url]
```

   The following table defines the parameters for the disposition.

Parameter	Description
svcName	Name of the service created with iBSE.
mthName	Name of the method created for the Web service.
responseTo	Location where responses to the Web service are posted. A predefined port name or another full URL. Optional.
errorTo	Location where error documents are sent. A predefined port name or another full URL. Optional.

5. Click *OK*.

   In the right pane, a table appears that summarizes the information associated with the event port you created. The event port also appears under the ports node in the left pane.

   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page 3-14.

Procedure How to Create an Event Port for a JMS Queue

To create an event port for a JMS queue:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port pane opens on the right.

   a. Type a name for the event port and provide a brief description.

   b. From the Disposition Protocol drop-down list, select *JMSQ*.

   c. In the Disposition field, enter a JMS destination.

      When pointing Application Explorer to an **iBSE** deployment, use the following format:

      ```
      jmsq:myQueueName@myQueueFac;jndiurl=[myurl];jndifactory=[myfactory
      ];user=[user];password=[xxx];errorTo=[pre-defined port name or
      another disposition url]
      ```

      When pointing Application Explorer to a **JCA** deployment, use the following format:

      ```
      jms:jmsqueue@jmsfactory;jndiurl=;jndifactory=;
      ```

      The following table defines the parameters for the disposition.

Parameter	Description
queue	JNDI name of a queue to which events are emitted.
Connection Factory	A resource that contains information about the JMS Server. The WebLogic connection factory is: `javax.jms.QueueConnectionFactory`

Parameter	Description
jndiurl	The URL to use to contact the JNDI provider. The syntax of this URL depends on which JNDI provider is being used. This value corresponds to the standard JNDI property, `java.naming.provider.url` For BEA WebLogic Server this is `t3://host:port` where: *host* Is the machine name where WebLogic Server is installed. *port* Is the port on which WebLogic server is listening. The default port if not changed at installation is 7001.
jndifactory	Is JNDI context.INITIAL_CONTEXT_FACTORY and is provided by the JNDI service provider. For WebLogic Server, the WebLogic factory is `weblogic.jndi.WLInitialContextFactory`
user	A valid user name required to access a JMS server.
password	A valid password required to access a JMS server.
errorTo	Location where error documents are sent. A predefined port name or another full URL. Optional.

5. Click *OK*.

   The event port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created. The port listing and summary are shown in the following graphic.

   You are now ready to associate the event port with a channel. For more information, see *Creating a Channel* on page 3-14.

Procedure How to Create an Event Port for MSMQ

To create an event port for MSMQ:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port pane opens on the right.

   a. In the Port Name field, type a name for the connection, for example, Queue1_on_NTK.

      The name is used to build a repository entry as well as to identify the connection.

   b. In the Description field, type a description for the target name you just created.

   c. From the Disposition Protocol drop-down list, select *MSMQ*.

   d. In the Disposition field, enter a MSMQ destination in the form of:

      ```
      msmq:/host/private$/qName;errorTo=[pre-defined port name or another disposition url]
      ```

   The following table defines the parameters for the disposition.

Parameter	Description
host	Machine name where the Microsoft Queuing system is running.
Queue Type	Private queues are queues that are not published in Active Directory. They appear only on the local computer that contains them. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue. For private queues, enter *Private$*.
qName	Name of the private queue where messages are placed.
errorTo	Location where error documents are sent. A predefined port name or another full URL. Optional.

5. Click *OK*.

   The event port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created. The summary is shown in the following graphic.

   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page 3-14.

Procedure How to Create a Port for the SOAP Disposition

To create a port for a SOAP disposition:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.

3. Select the *ports* node.

4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port window opens in the right pane.

   a. Type a name for the event port and provide a brief description.

   b. From the Disposition Protocol drop-down list, select *SOAP*.

   c. In the Disposition field, enter an SOAP destination, using the following format:

   ```
   soap:[wsdl-url];soapaction=[myaction];method=[web service
   method];namespace=[namespace];responseTo=[pre-defined port name or
   another disposition URL];errorTo=[pre-defined port name or another
   disposition url]
   ```

   The following table defines the parameters for the disposition.

Parameter	Description

wsdl-url	The URL to the WSDL file that is required to create the SOAP message. For example:
	`http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl`
	where:
	`webservice`
	Is the name of the Web service you created using Application Explorer.
	This value can be found by navigating to the Integration Business Services tab and opening the *Service Description* link in a new window. The WSDL URL appears in the Address field.
	You can also open the WSDL file in a third party XML editor (for example, XMLSPY) and view the SOAP request settings to find this value.

Parameter	Description
soapaction	The method that will be called by the SOAP disposition. This value can be found in the WSDL file.
method	The Web service method you are using. This value can be found in the WSDL file.
namespace	The XML namespace you are using. This value can be found in the WSDL file.
responseTo	The location to which responses are posted, which can be a predefined port name or another URL. Optional. The URL must be complete, including the protocol.
errorTo	The location to which error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

5. Click *OK*.

   The event port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created.

Procedure How to Create an Event Port for an HTTP Disposition

To create an event port for an HTTP disposition:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port pane opens on the right.

   a. Type an event port name and a brief description.
   b. From the disposition protocol drop-down list, select *HTTP*.
   c. From the Disposition field, enter an HTTP destination.

When pointing Application Explorer to an **iBSE** deployment, use the following format:

```
ihttp://[myurl];responseTo=[pre-defined port name or another
disposition url];
```

where:

url

Is the URL target for the post operation, for example,

```
http://myhost:1234/docroot
```

responseTo

Is the location where responses are posted (optional).

When pointing Application Explorer to a **JCA** deployment, use the following format:

```
http://host:port/uri
```

where:

host:port

Is the combination of the name of the host on which the Web server resides and the port on which the server is listening for the post operation.

uri

Is the universal resource identifier that completes the url specification.

5. Click *OK*.

   The port appears under the ports node in the left pane.

Procedure How to Create an Event Port for MQ Series Disposition

To create an event port for MQ Series using Application Explorer:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens.

2. In the left pane, expand the *SWIFT* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create Event Port pane opens on the right.

   a. Type an event port name and a brief description.

   b. From the disposition protocol drop-down list, select *MQ Series*.

   c. In the Disposition field, enter an MQ Series destination.

When pointing Application Explorer to an **iBSE** deployment, use the following format:

```
mqseries:/qManager/
qName;host=[hostname];port=[port];channel=[channnelname];errorTo=[
pre-defined port name or another disposition url]
```

When pointing Application Explorer to a **JCA** deployment, use the following format:

```
mq:qmanager@respqueue;host=;port=;channel=
```

The following table defines the parameters for the disposition.

Parameter	Description
qManager	Is the name of the queue manager to which the server must connect.
qName or respqueue	Name of the queue where messages are placed.
host	The host on which the MQ Server is located (for the MQ Client only).
port	The number to connect to an MQ Server queue manager (for the MQ client only).
channel	The case-sensitive name of the channel that connects with the remote MQ Server queue manager (for the MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.
errorTo	Location where error documents are sent. This can be a predefined port name or another full URL. Optional.

5. Click *OK*.

   The newly created event port appears under the port section of the event adapter in the left pane.

Editing or Deleting an Event Port

The following procedures provide information on how to modify and delete an event port.

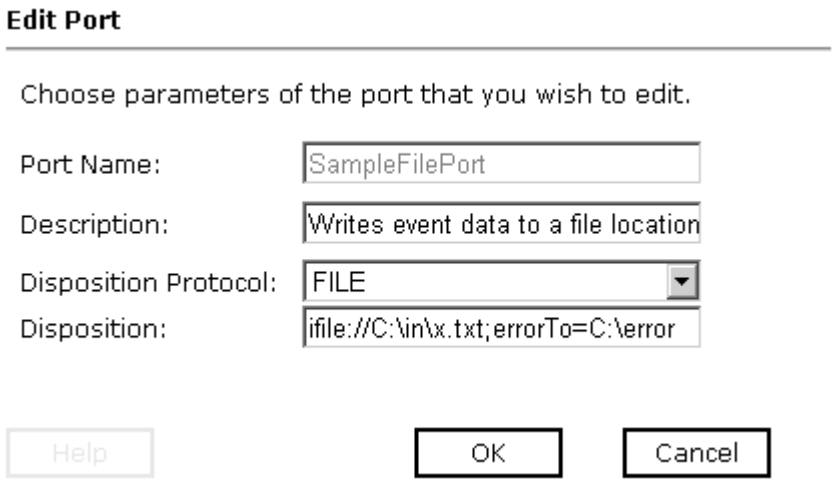
Procedure How to Edit an Event Port

To edit an existing event port:

1. In the left pane, select the event port you want to edit.

2. In the right pane, move the pointer over *Operations* and select *Edit*.

   The Edit Port window opens. This pane provides four fields, a help button, and two action buttons.



3. Make the required changes to the Description, Disposition Protocol, or Disposition fields, and click *OK*.

   Note: The Edit Port pane does not allow you to change the name of the port, only the parameters.

Procedure How to Delete an Event Port

To delete an existing event port:

1. Select the event port you want to delete.
2. In the right pane, move the pointer over *Operations* and select *Delete*.

   A confirmation dialog box opens.

3. To delete the event port you selected, click *OK*.

   The event port disappears from the list in the left pane.

Creating a Channel

The following procedure describes how to create a SWIFT channel for your event. You must associate a port to a channel before you can make the channel active.

Procedure How to Create a SWIFT Channel

To create a channel using iWay Application Explorer:

1. Click the *Event Adapters* tab.

   The Event Adapters window opens. The adapters that appear in the left pane support events.

2. Expand the *SWIFT* node.

   The ports and channels nodes appear in the left pane.

3. Click the *channels* node.
4. In the right pane, move the pointer over *Operations* and select *Add a new channel*.

   The Add a new channel window opens.

   a. Type a name for the channel, for example, SWIFTChannel.

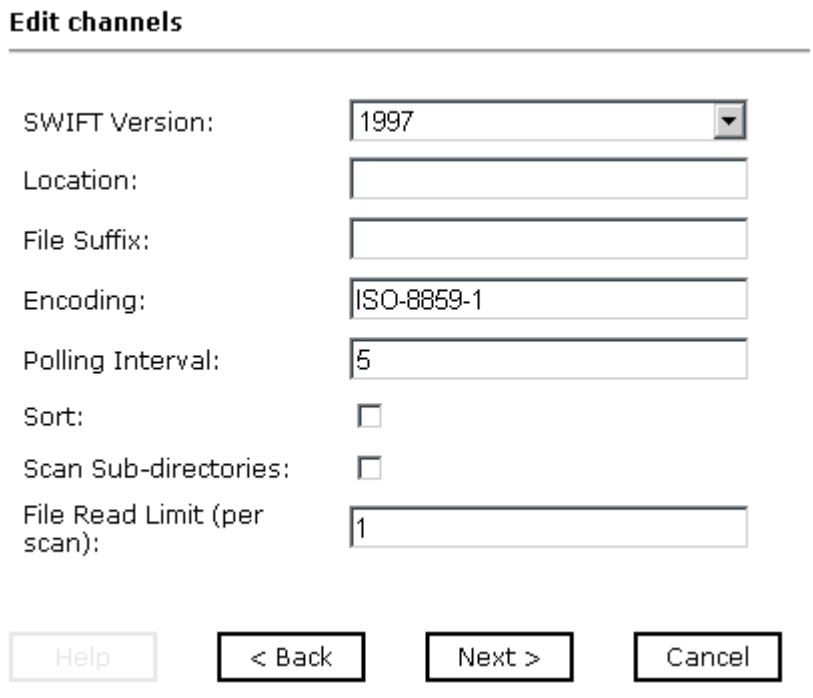
   b. Type a brief description.

c. From the drop-down list, select a type of listener:

- File System Listener (FILE)
- HyperText Transfer Protocol
- TCP Listener (TCP)
- IBM MQ Series (MQ)
- File Transfer Protocol (FTP)

5. Click *Next*.

The Edit Channels window opens in the right pane and includes fields that are specific to the type of listener you selected.



6. Provide the appropriate information that is specific to the listener you selected.

For information on the parameters for a File System Listener (FILE) listener, see *File System Listener (FILE) listener Configuration Parameters* on page 3-17.

For information on the parameters for a HyperText Transfer Protocol listener, see *HyperText Transfer Protocol Listener Configuration Parameters* on page 3-18.

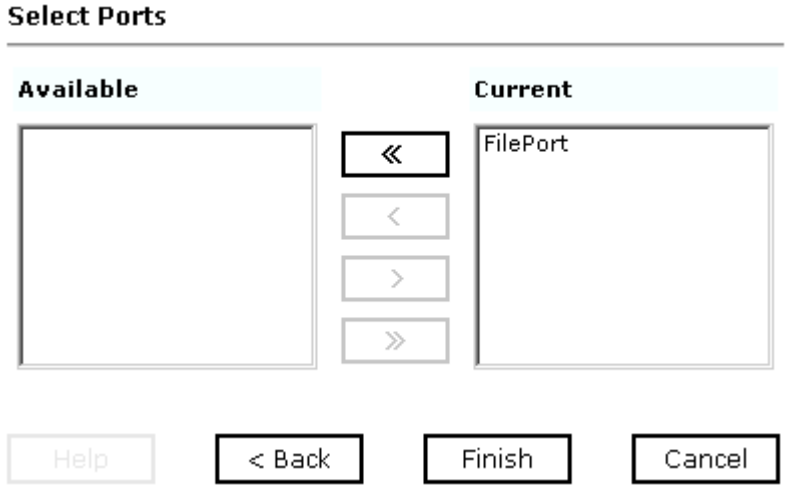
For information on the parameters for a TCP Listener, see *TCP Listener Configuration Parameters* on page 3-19.

For information on the parameters for an IBM MQ Series (MQ) listener, see *IBM MQ Series (MQ) Listener Configuration Parameters* on page 3-20.

For information on the parameters for a File Transfer Protocol (FTP) listener, see *File Transfer Protocol (FTP) Listener Configuration Parameters* on page 3-21.

7. Click *Next*.

The Select Ports pane opens, as shown in the following graphic. A list of available ports appear in Available field on the left, and the ports that are currently associated appear in the Current field on the right. This pane also contains a help button and three action buttons.



a. Select an event port from the list of current ports.

b. Click the single right (>) arrow button to transfer the port to the list of available ports. To associate all the event ports, click the double right (>>) arrow button.

8. Click *Finish*.

A summary window opens in the right pane, showing the channel description, channel status, and available ports.

All the information in the summary is associated with the channel you created.

The channel also appears under the channels node in the left pane. The following graphic shows a sample listing of a channel. An X over the icon, also shown in this graphic, indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

9. In the right pane, move the pointer over *Operations* and select *Start the channel*.

The channel you created becomes active.

The X that was over the icon disappears.

10. To stop the channel, move the pointer over *Operations* and select *Stop the channel*.

Reference **File System Listener (FILE) listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Location	The directory where messages are received. DOS-style file patterns are valid for this parameter. You can specify a file pattern as well as a directory. For example, c:\xyz\ab*cd (without a file suffix) takes the file suffix from that parameter. If you use a pattern, files are selected based on the suffix and then the pattern. AB?CD selects ABxCD. AB*CD selects ABxxxCD.
File Suffix	File extension for the file event. This limits input files to those with the specified extensions. The "." is not required. The minus sign ("-") indicated that there is no extension. If the file extension is zip, the unzipped files must conform to the event schema, or they will fail. This function also works with transform configured.
Encoding	The host on which the MQ Server is located (for the MQ Client only).
Polling Interval	This is a time, expressed as xxH:xxM:xxS For example 1 hour, 2 minutes, and 3 seconds is: 1H:2M:3S The maximum interval between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. The side-effect of a high value is that a worker thread cannot respond to a stop command. If this value is set to 0, the listener runs once and terminates. The default value is 2 seconds.
Sort	The case-sensitive name of the channel that connects with the remote MQ Server queue manager (for the MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.

Parameter	Description
Scan Sub-directories	Location where error documents are sent. This can be a predefined port name or another full URL. Optional.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference **HyperText Transfer Protocol Listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Port	The port where the adapter listens for the HTTP transfer.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference **TCP Listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Port	The port where the adapter listens for the TCP transfer.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.
Allowable Client Host	The name or address of the client restricted to accessing this adapter.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.

Parameter	Description
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference IBM MQ Series (MQ) Listener Configuration Parameters

On the Settings tab:

Parameter	Description
Queue Manager	The name of the MQ queue manager to be used.
Queue Name	The name of the MQ Series or WebSphere MQ queue that the SWIFT system polls.
Polling Interval	The maximum wait interval (in the format *nn*H:*nn*M:*nn*S) between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. However, with a high value, the worker thread cannot respond to a stop command. If timeout is set to 0, the listener runs once and terminates. The default is 2 seconds.

On the MQ Client tab:

Parameter	Description
Host	The host where the MQ Server is located.
Port	The port number used to connect to an MQ Server.
Channel	The channel between an MQ Client and an MQ Server.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.

Parameter	Description
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference **File Transfer Protocol (FTP) Listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Host	The name of the FTP host.
Port	The port where the adapter listens on the FTP transfer.
User	The user name to log onto the FTP Server.
Password	The password for the FTP user.
Location	The directory where messages are received. DOS-style file patterns are available for this parameter. You can specify a file pattern as well as a directory. For example, c:\xyz\ab*cd (without a file suffix) takes the file suffix from that parameter. If you use a pattern, files are selected based on the suffix and then the pattern. AB?CD selects ABxCD. AB*CD selects ABxxxCD.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.
Polling Interval	The maximum wait interval (in the format *nn*H:*nn*M:*nn*S) between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. However, with a high value, the worker thread cannot respond to a stop command. If timeout is set to 0, the listener runs once and terminates. The default is 2 seconds.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Procedure How to Edit a Channel

To edit an existing channel:

1. In the left pane, select the channel you want to edit.
2. In the right pane, move the pointer over *Operations* and select *Edit*.

   The Edit channels window opens.
3. Make the required changes to the channel configuration and click *Finish*.

Procedure How to Delete a Channel

To delete an existing channel:

1. In the left pane, select the channel you want to delete.
2. In the right pane, move the pointer over *Operations* and select *Delete*.

   A confirmation dialog box opens.
3. To delete the channel you selected, click *OK*.

   The channel disappears from the list in the left pane.

CHAPTER 4

Using Web Services Policy-Based Security

Topics:

- Integration Business Services Policy-Based Security
- Configuring Integration Business Services Policy-Based Security

Servlet Application Explorer provides a security feature called Integration Business Services policy-based security. The following topics describe how this feature works and how to configure it.

Note: For the iWay 5.5 RG2 Release, it is recommended that policy-based security not be enabled.

Integration Business Services Policy-Based Security

Integration Business Services provide a layer of abstraction between the back-end business logic they invoke and the user or application running the business service. This enables easy application integration but raises the issue of controlling the use and execution of critical and sensitive business logic that is run as a business service.

Servlet Application Explorer controls the use of business services that use adapters with a feature called policy-based security. This feature enables an administrator to apply *policies* to Integration Business Services (iBS) to deny or permit their execution.

A *policy* is a set of privileges associated with the execution of a business service that can be applied to an existing or new iBS. When you assign specific rights or privileges inside a policy, you need not recreate privileges for every iBS that has security issues in common with other Integration Business Services. Instead, you can use one policy for many Integration Business Services.

The goal is to secure requests at both the transport and the SOAP request level that is transmitted on the wire. Some policies do not deal with security issues directly but affect the run-time behavior of the business services to which they are applied.

The iBSE administrator creates an instance of a policy type, names it, associates individual users and/or groups (a collection of users), and then applies the policy to one or more business services.

You can assign a policy to an iBS or to a method within an iBS. If a policy is applied only to a method, other methods in that iBS are not governed by it. However, if a policy is applied to the iBS, all methods are governed by it. At run time, the user ID and password that are sent to iBSE in the SOAP request message are checked against the list of users for all policies applied to the specific iBS. The Resource Execution policy type is supported and dictates who can or cannot execute the iBS.

When a policy is not applied, the default value for an iBS is to “grant all.” For example, anyone can execute the iBS until the Resource Execution policy is associated to the iBS. At that time, only users granted execution permission, or those who do not belong to a group that was denied execution permissions, have access to the iBS.

Configuring Integration Business Services Policy-Based Security

Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Servlet Application Explorer. For more information, see *How to Create a User to Associate With a Policy* on page 4-3 or *How to Create a Group to Associate With a Policy* on page 4-5.

An execution policy governs who can execute the business service to which the policy is applied. For more information, see *How to Create an Execution Policy* on page 4-7.

You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to iBSE and therefore, need not be applied to an individual business service. You need not create a policy, however, you must enable the Security Policy option in Servlet Application Explorer. For more information, see *How to Configure IP and Domain Restrictions* on page 4-10.

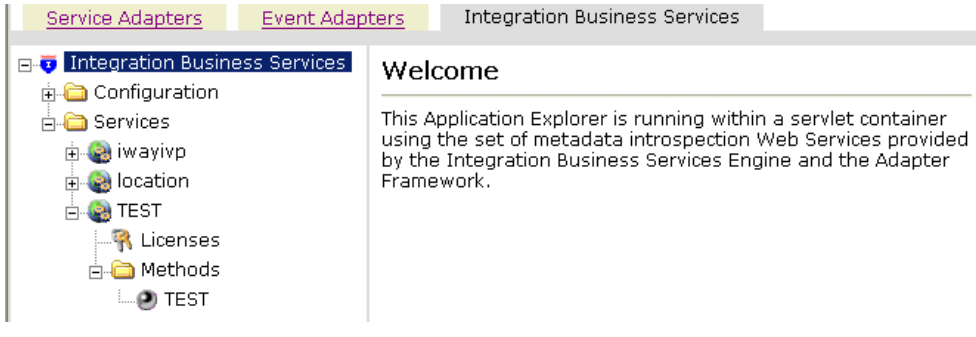
Note: For the iWay 5.5 RG2 Release, it is recommended that policy-based security not be enabled.

Procedure How to Create a User to Associate With a Policy

To create a user to associate with a policy:

1. Open *Servlet Application Explorer*.

   The following image shows the window that opens and includes three tabs corresponding to Service Adapters, Event Adapters, and Integration Business Services. The Integration Business Services tab is active and displays a Welcome screen on the right. The image shows the Integration Business Services node expanded in the left pane.



   a. Click the *Integration Business Services* tab.

   b. Expand the *Configuration* node.

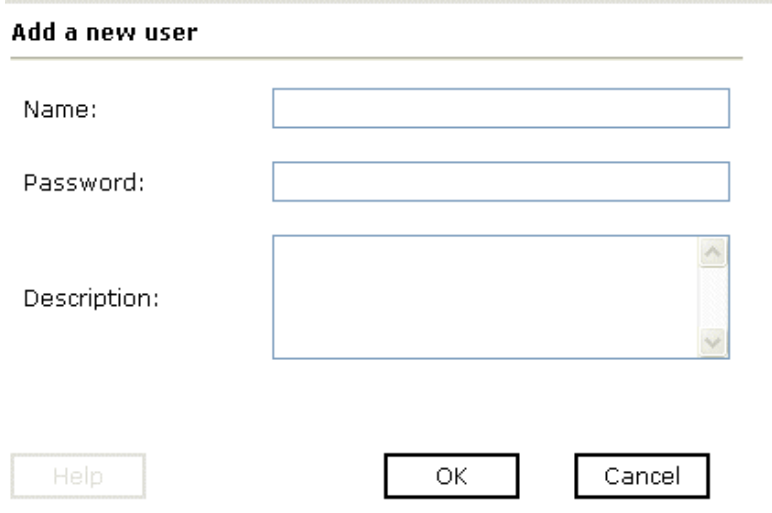
   c. Expand the *Security* node.

d. Expand the *Users and Groups* node.

e. Select *Users*.

2. In the right pane, move the pointer over *Operations* and select *Add*.

The following image shows the Add a new user pane that opens and includes fields where you enter a user name, a password, and a description of the user. The pane includes a Help button, an OK button to instruct the system to accept inputs, and a Cancel button to escape from the pane.



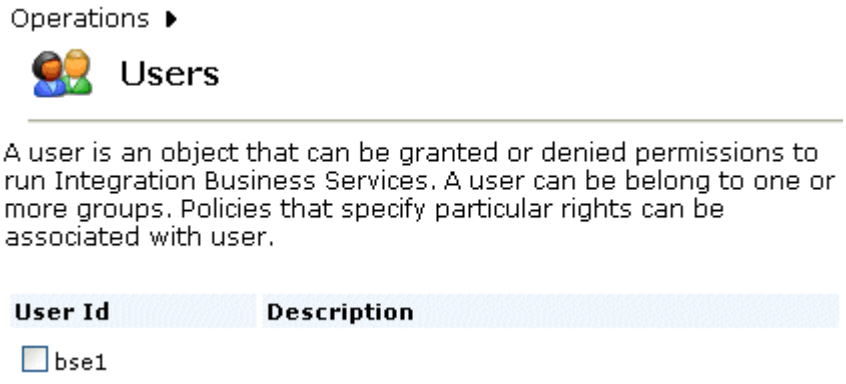
a. In the Name field, type a user ID.

b. In the Password field, type the password associated with the user ID.

c. In the Description field, type a description of the user (optional).

3. Click *OK*.

The following image opens and shows a new user added to the configuration. It includes a definition of a user and a user ID and description.

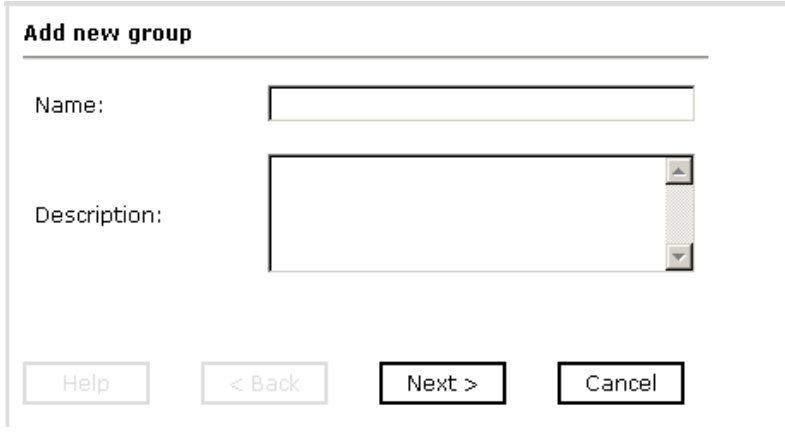


Procedure How to Create a Group to Associate With a Policy

To create a group to associate with a policy:

1. Open *Servlet Application Explorer*.
   a. Click the *Integration Business Services* tab.
   b. Expand the *Configuration* node.
   c. Expand the *Security* node.
   d. Expand the *Users and Groups* node.
   e. Select *Groups*.
2. In the right pane, move the pointer over *Operations* and click *Add*.

The following image shows the Add new group pane that opens with fields where you enter a name and a description for the group. To continue after typing inputs, click the *Next* button. The pane also includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.

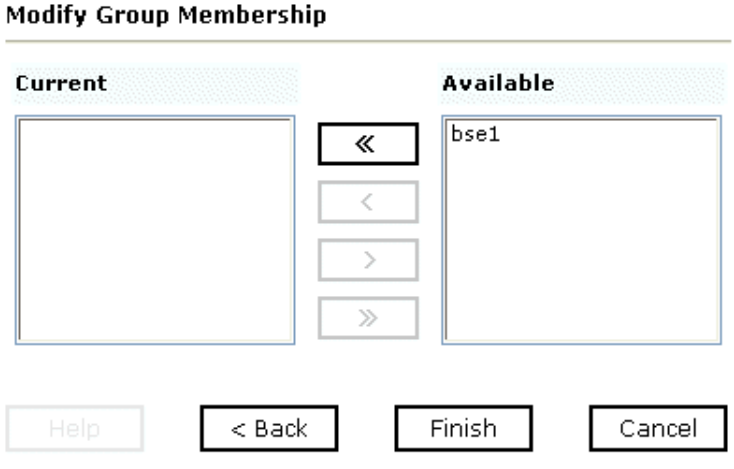


a. In the Name field, type a a name for the group.

b. In the Description field, type a description for the group (optional).

3. Click *Next*.

The following image shows the Modify Group Membership pane where you can move users to or from a group using the arrow keys to move them between the Current and Available lists and then clicking the Finish button. The pane includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.

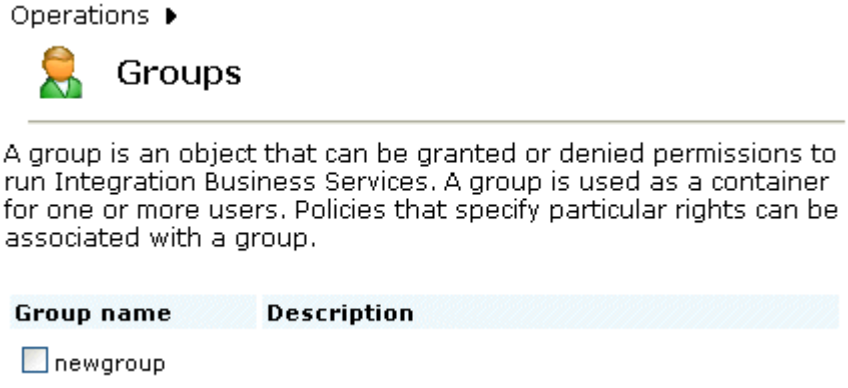


You can either highlight a single user in the list of available users and add it to the current list by clicking the left arrow, or you can click the double left arrow to add all users in the list of available users to the group.

4. After you select a minimum of one user, click *Finish*.

   The new group is added.

   The following image shows a pane with a new group added to the configuration. It includes a definition of a group and the group name and description.



Procedure How to Create an Execution Policy

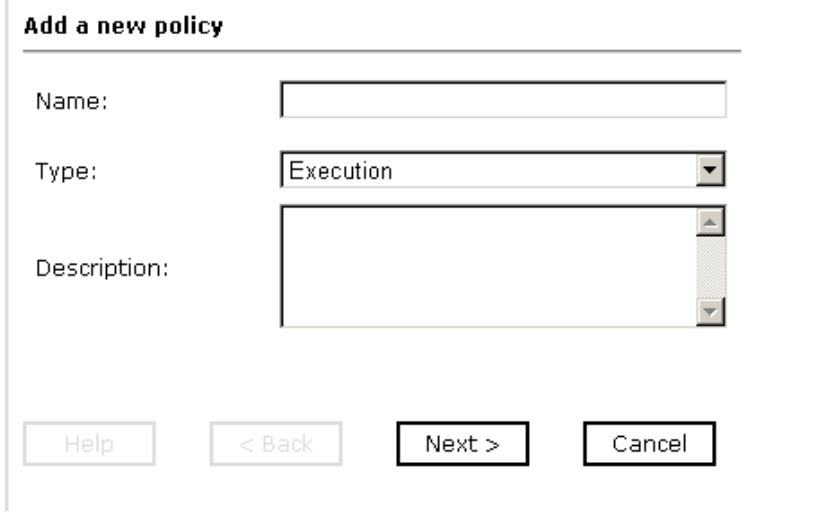
To create an execution policy:

1. Open *Servlet Application Explorer*.
   a. Click the *Integration Business Services* tab.
   b. Expand the *Configuration* node.
   c. Select *Policies*.

The following image shows the Policies pane on the right where you apply a policy. The Operations menu becomes available with three options, Build/Rebuild, Add, and Refresh.

2. Move the pointer over *Operations* and click *Add*.

The following image shows the Add a new policy pane that opens with fields for entering the name, type, and description of the policy. To continue, click the *Next* button. The pane includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.



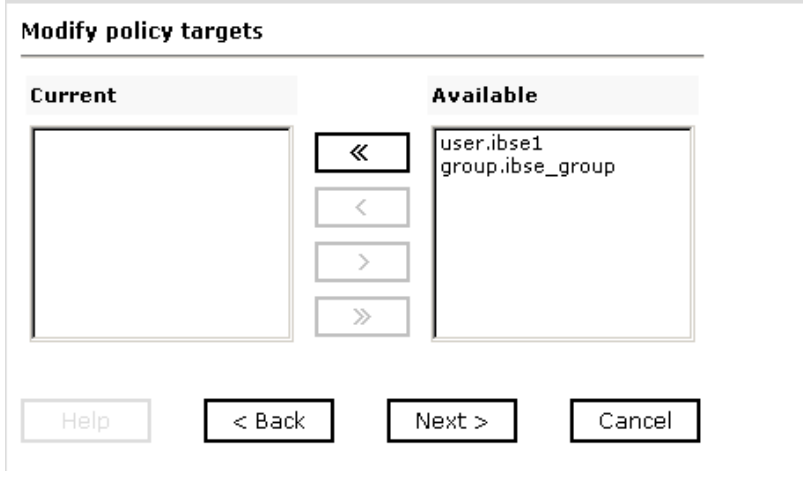
a. In the Name field, type a a name for the policy.

b. From the Type drop-down list, select *Execution*.

c. In the Description field, type a description for the policy (optional).

3. Click *Next*.

The following image shows the Modify policy targets pane that opens and includes a list of current and available targets and arrow buttons to move targets from one list to the other. The pane also includes a Help button, a Back button to return to the previous screen, a Next button to continue to the next screen, and a Cancel button to escape from the pane.

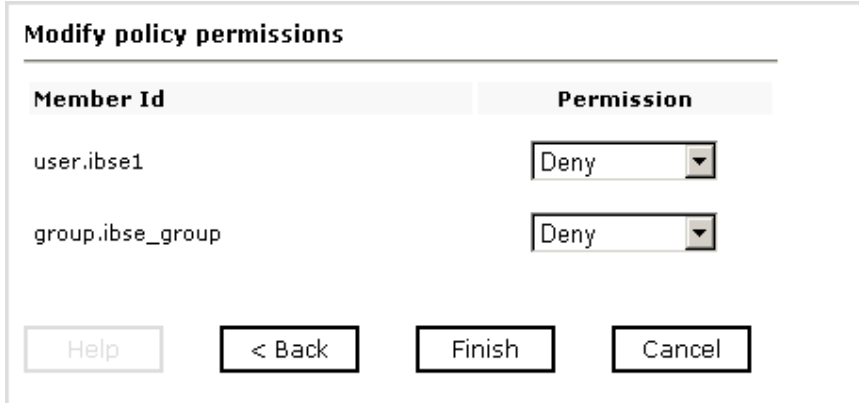


4. Select a minimum of one user or group from the Available pane.

Note: This user ID is checked against the value in the user ID element of the SOAP header sent to iBSE in a SOAP request.

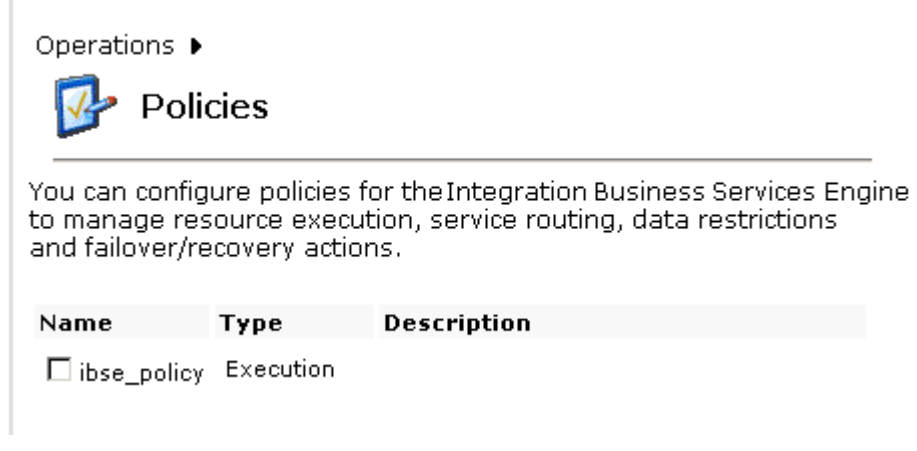
5. Click *Next*.

The following image shows the Modify policy permissions pane that opens and includes drop-down lists where you can select to grant or deny permission to members and then click a button to finish. The pane also includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.



6. To assign whether users or groups may execute the iBSE, select *Grant* to permit execution or *Deny* to restrict execution from a Permission drop-down list.
7. Click *Finish*.

   The following image shows the pane that summarizes your configuration. It includes a definition of policies and the name, type, and description of the policies.



Procedure How to Configure IP and Domain Restrictions

To configure IP and domain restrictions:

1. Open *Servlet Application Explorer.*

a. Select the *Integration Business Services* tab.

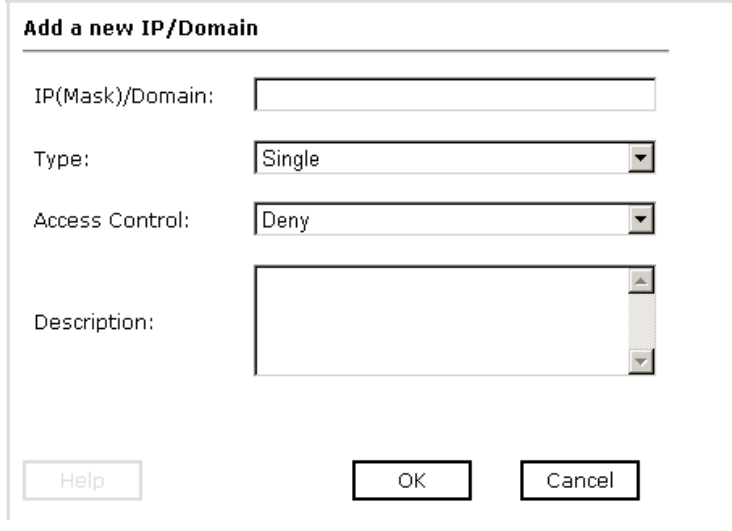
b. Expand the *Configuration* node.

c. Expand the *Security* node.

d. Select *IP and Domain*.

2. In the right pane, move the pointer over *Operations* and click *Add*.

The following image shows the Add a new IP/Domain pane that opens where you enter information for the IP/Domain in four fields. You must select a type of restriction from a drop-down list before you can enter information in the IP(Mask)/Domain field. The pane also includes a Help button, an OK button to instruct the system to accept inputs, and a Cancel button to escape from the pane.



a. From the Type drop-down list, select the type of restriction.

b. In the IP(Mask)/Domain field, type the IP or domain name using the following guidelines.

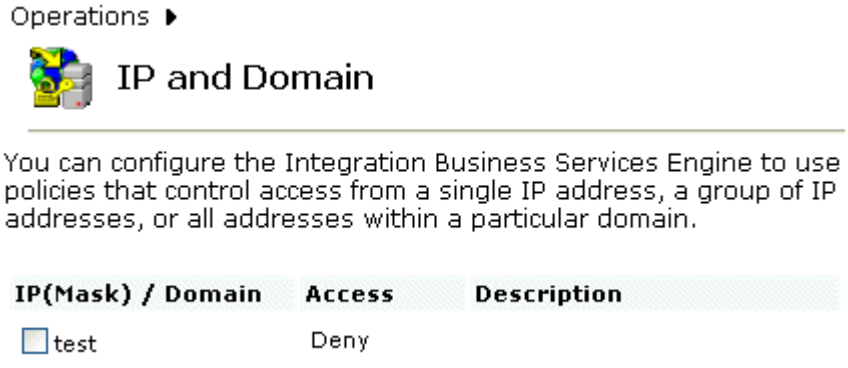
If you select Single (Computer) from the Type drop-down list, you must provide the IP address for that computer. If you only know the DNS name for the computer, click *DNS Lookup* to obtain the IP Address based on the DNS name.

If you select Group (of Computers), you must provide the IP address and subnet mask for the computer group.

If you select Domain, you must provide the domain name, for example, yahoo.com.

3. From the Access Control drop-down list, select *Grant* to permit access or *Deny* to restrict access for the IP addresses and domain names you are adding.
4. Click *OK*.

   The following image shows the pane that opens and summarizes your configuration including the domain name, whether access is granted or denied, and a description (optional).



CHAPTER 5

Management and Monitoring

Topics:

- Managing and Monitoring Services and Events Using iBSE
- Managing and Monitoring Services and Events Using the JCA Test Tool
- Setting Engine Log Levels
- Configuring Connection Pool Sizes
- Migrating Repositories
- Exporting or Importing Targets
- Retrieving or Updating Web Service Method Connection Information
- Starting or Stopping a Channel Programmatically

After you create services and events using Servlet Application Explorer, you can use managing and monitoring tools provided by the Integration Business Services Engine (iBSE) and the iWay Connector for JCA to measure the performance of your run-time environment. This section describes how to configure and use these features.

Managing and Monitoring Services and Events Using iBSE

Integration Business Services Engine (iBSE) provides a console to manage and monitor services and events currently in use and to display resource usage and invocation statistics. These indicators can help you adjust your environment for optimum efficiency.

The following monitoring levels are available for services:

- System
- Service
- Method

The following monitoring levels are available for events:

- System
- Channel
- Port

Procedure: How to Configure Monitoring Settings

To configure monitoring settings:

1. Ensure that your BEA WebLogic Server is started.
2. To access the monitoring console, enter the following URL in your Web browser:

   `http://localhost:port/ibse/IBSEConfig`

   where:

   localhost

   Is the machine where the application server is running.

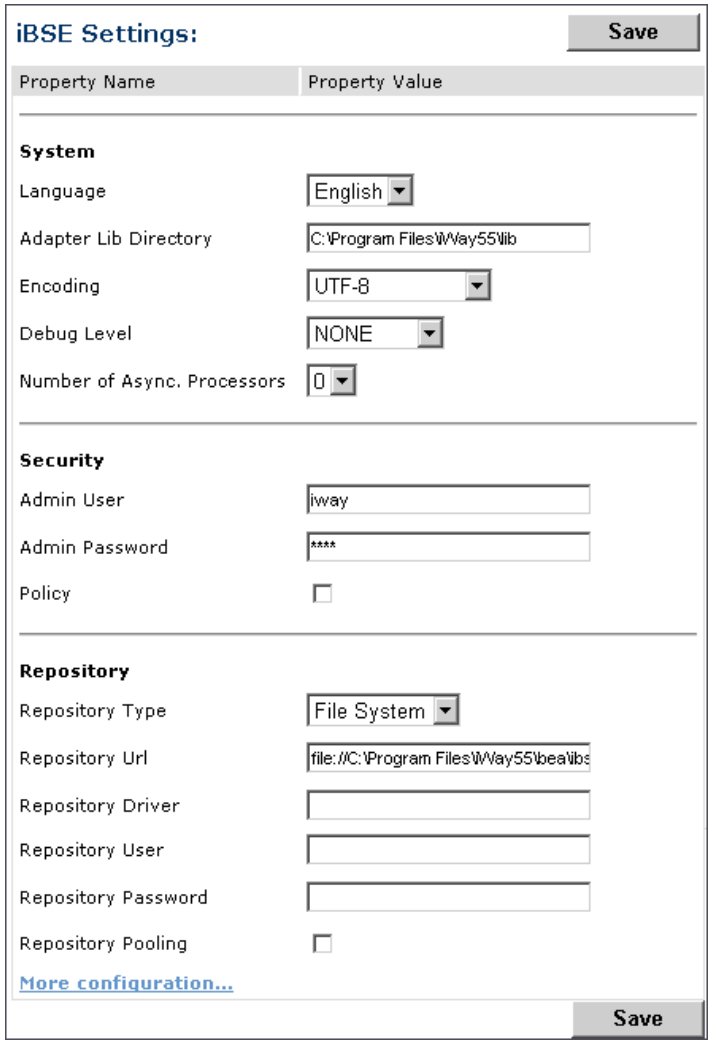
   port

   Is the HTTP port for the application server.

   The following image shows the iBSE Settings window that opens. It lists property names and includes fields where you can enter values for each property. To configure system settings, the System pane contains drop-down lists for selecting language, encoding, the debug level, and the number of asynchronous processors. It also contains a field where you can enter a path to the adapters lib directory.

   To configure security settings, the Security pane contains fields for typing the Admin User name and the associated password and a check box for specifying policy.

To configure repository settings, the Repository pane contains a drop-down list for selecting the repository type, fields to type information for the repository URL, driver, user, and password, and a check box where you can enable repository pooling. In the upper and lower right of the window is a Save button. In the lower left of the window is an option to click to access more configuration settings.



3. Click *More configuration*.

Tip: To access the monitoring console directly, enter the following URL in your Web browser:

`http://localhost:port/ibse/IBSEStatus`

where:

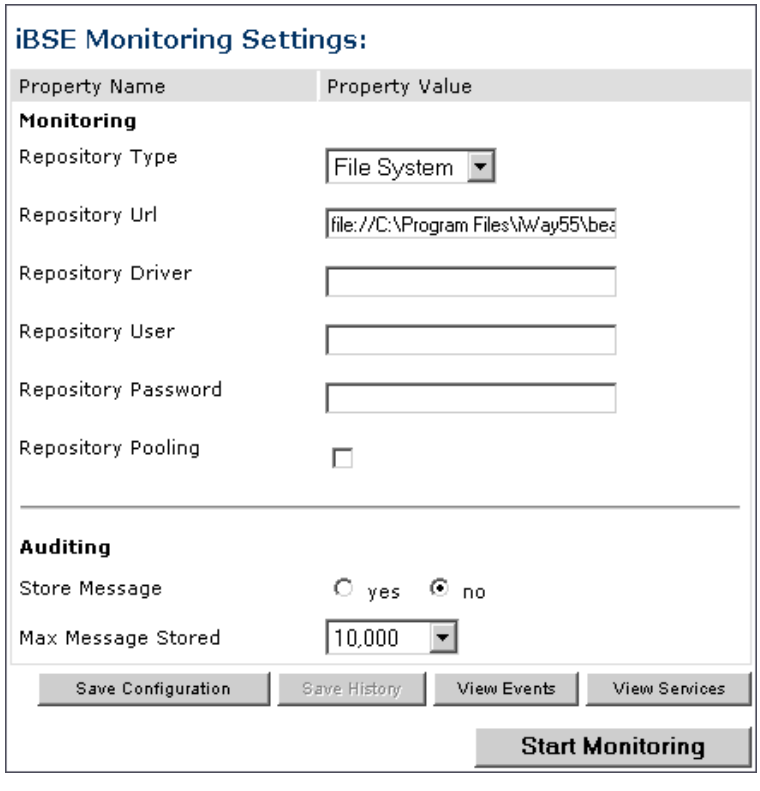
localhost

Is the machine where the application server is running.

port

Is the HTTP port for the application server.

The following image shows the iBSE Monitoring Settings window that opens. It lists property names and includes a corresponding field where you can enter values for each property. The Monitoring pane contains a drop-down list for selecting the repository type, fields to type information for the repository URL, driver, user, and password, and a check box where you can enable repository pooling. The Auditing pane contains an option button to click to specify whether to store a message and a drop-down list where you can select the maximum messages to store. At the bottom of the window is a row of buttons that you can click to save your configuration, view events, or view services. The Save History button is inactive. After you enter properties and choose whether to save or view, you can click the Start Monitoring button.



a. In the Monitoring pane, from the Repository Type drop-down list, select the type of repository you are using.

b. To connect to the database in the Repository Url field, type a JDBC URL.

c. To connect to the database in the Repository Driver field, type a JDBC Class.

d. To access the monitoring repository database, type a user ID and password.

e. To enable pooling, click the *Repository Pooling* check box.

f. In the Auditing pane, select *yes* if you want to store messages.

This option is disabled by default.

Note: You must start and then, stop monitoring to enable this option.

g. Select the maximum number of messages you want to store.

By default, 10,000 is selected.

Note: Depending on your environment and the number of messages that are exchanged, storing a large number of messages may affect system performance. If you need more information about your system resources, consult your system administrator.

h. Click *Save Configuration*.

4. Click *Start Monitoring*.

iBSE begins to monitor all services and events currently in use. If you selected the option to store messages, iBSE stores messages.

5. To stop monitoring, click *Stop Monitoring*.

Procedure: How to Monitor Services

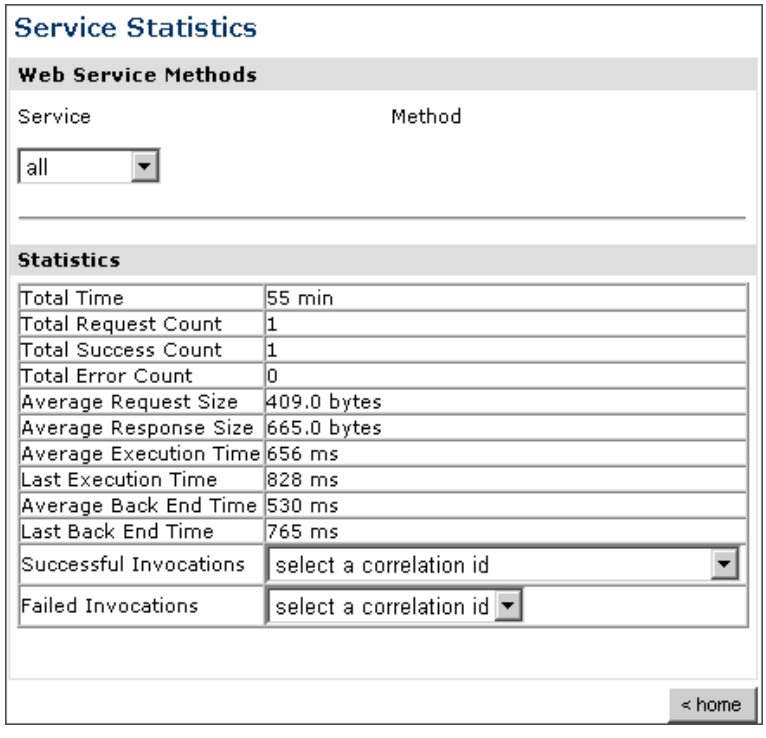
To monitor services:

1. Ensure that your BEA WebLogic Server is started.

2. From the iBSE Monitoring Settings window, click *Start Monitoring*.

3. Click *View Services*.

The following image shows the System Level Summary (Service Statistics) window that opens. The Web Service Methods pane contains a drop-down list where you select a service. On the right, space is reserved for a drop-down list of methods that will appear. The Statistics pane contains a table with a summary of service statistics and two drop-down lists where you can select a successful or failed invocation to view more information about that service. At the bottom of the window is a home button to click to return to the iBSE Monitoring Settings window.



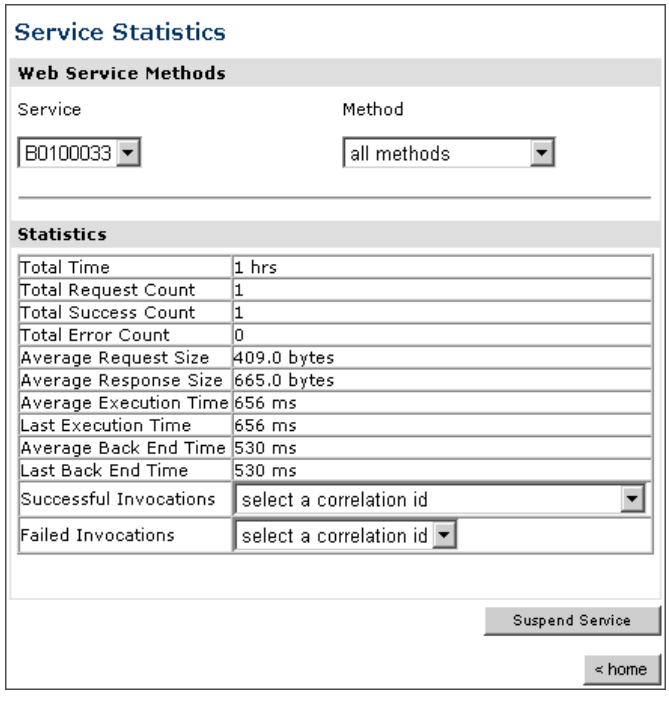
The system level summary provides services statistics at a system level.

The following table consists of two columns, one that lists the name of each statistic and the other that describes the corresponding service statistic.

Statistic	Description
Total Time	Total amount of time iBSE monitors services. The time starts after you click Start Monitoring in the iBSE Monitoring Settings window.
Total Request Count	Total number of services requests that were made during the monitoring session.
Total Success Count	Total number of successful service executions.
Total Error Count	Total number of errors that were encountered.
Average Request Size	Average size of an available service request.
Average Response Size	Average size of an available service response size.
Average Execution Time	Average execution time for a service.
Last Execution Time	Last execution time for a service.
Average Back End Time	Average back end time for a service.
Last Back End Time	Last back end time for a service.
Successful Invocations	A list of successful services arranged by correlation ID. To retrieve more information for a service, you can select the service from the drop-down list.
Failed Invocations	A list of failed services arranged by correlation ID. To retrieve more information for a service, you can select the service from the drop-down list.

4. Select a service from the drop-down list.

The following image shows the System Level Summary (Service Statistics) window that opens. The Web Service Methods pane contains a drop-down list on the left where you select a service and a drop-down list on the right where you select a service method. The Statistics pane contains a table with a summary of service statistics and two drop-down lists. To view more information about that service, you can select it from the Successful Invocations or Failed Invocations drop-down list. To suspend or resume a service, you can click a button in the lower right. To return to the iBSE Monitoring Settings window, you click the home button (also located in the lower right).

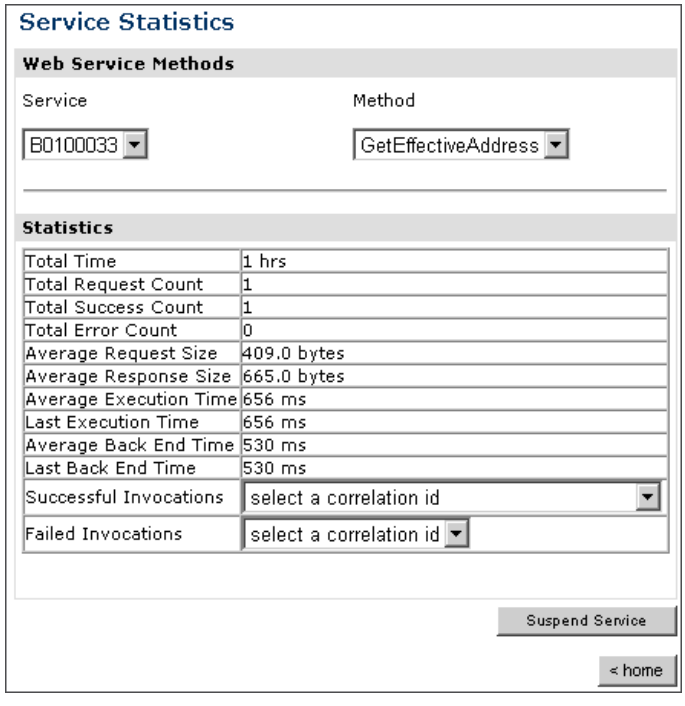


a. To stop a service at any time, click *Suspend Service*.

b. To restart the service, click *Resume Service*.

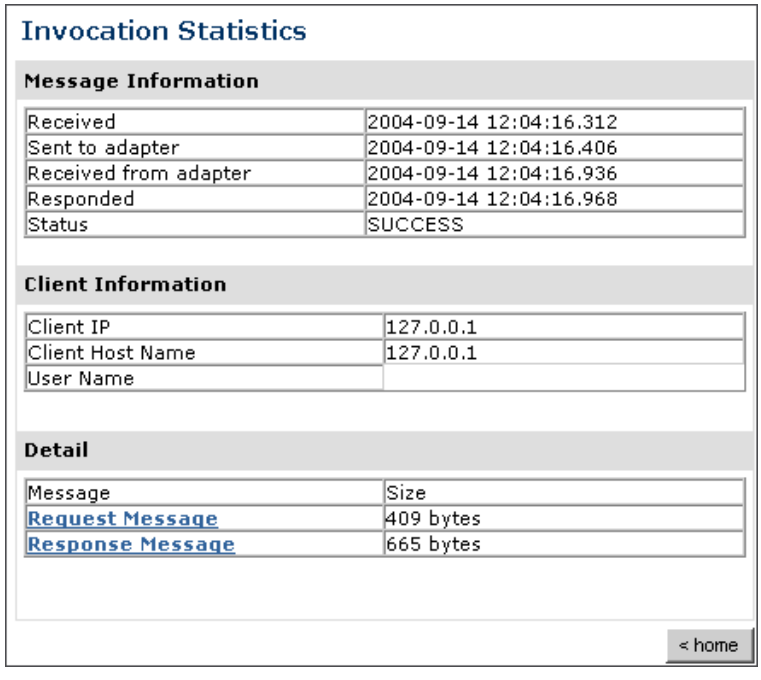
5. Select a method for the service from the Method drop-down list.

The following image shows the Method Level Summary (Service Statistics) window that opens. The Web Service Methods pane contains a drop-down list on the left where you select a service and a drop-down list on the right where you select a service method. The Statistics pane contains a table with a summary of service statistics and two drop-down lists. To view more information about that service, you can select it from the Successful Invocations or Failed Invocations drop-down list. To suspend or resume a service, you can click a button in the lower right. To return to the iBSE Monitoring Settings window, you click the home button (also located in the lower right).



6. For additional information about a successful service and its method, select a service based on its correlation ID from the Successful Invocation drop-down list.

The following image shows the Invocation Level Statistics window that opens. The Message Information pane contains a table of information about the message. The Client Information pane contains a table of information about the client. The Detail pane contains a table that shows the size of the request and response messages, with options to click to view the respective XML documents. In the lower right of the window is a home button to click to return to the iBSE Monitoring Settings window.



7. To view the XML request document in your Web browser, click *Request Message*.

   You can also view the XML response document for the service.

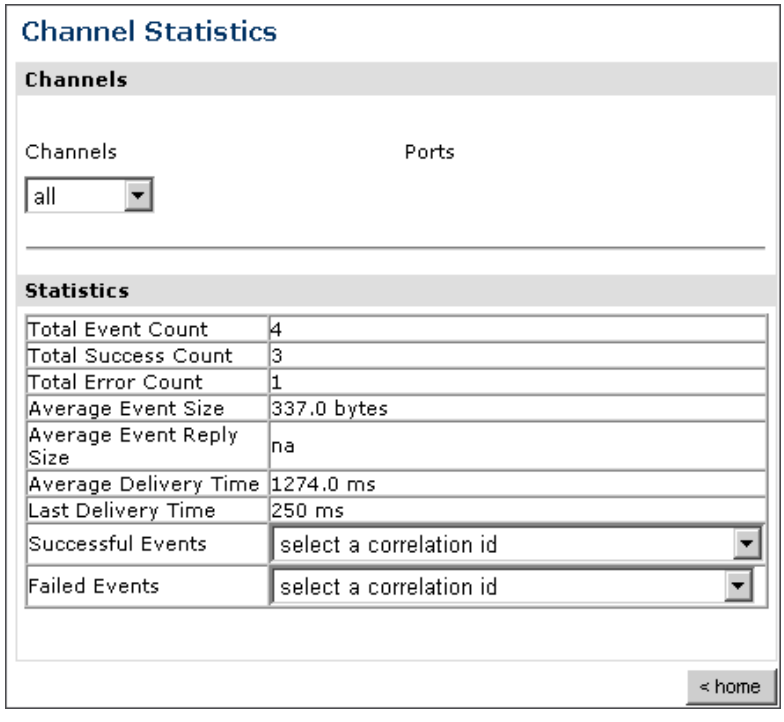
8. To return to the iBSE Monitoring Settings window, click *home*.

Procedure: How to Monitor Events

To monitor events:

1. Ensure that your BEA WebLogic Server is started.
2. In the iBSE Monitoring Settings window, click *Start Monitoring*.
3. Click *View Events*.

The following image shows the System Level Summary (Channel Statistics) window that opens. The Channels pane contains a drop-down list on the left where you select a channel. On the right, space is reserved for a drop-down list of ports that will appear. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a home button to click to return to the iBSE Monitoring Settings window.



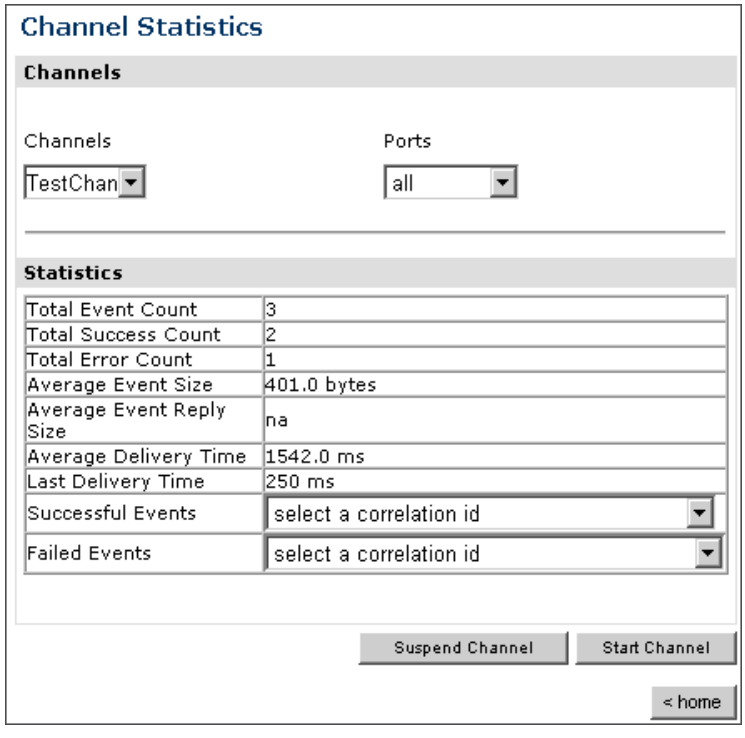
The system level summary provides event statistics at a system level.

The following table consists of two columns, one that lists the name of each statistic and the other that describes the corresponding event statistic.

Statistic	**Description**
Total Event Count	Total number of events.
Total Success Count	Total number of successful event executions.
Total Error Count	Total number of errors that were encountered.
Average Event Size	Average size of an available event request.
Average Event Reply Size	Average size of an available event response.
Average Delivery Time	Average delivery time for an event.
Last Delivery Time	Last delivery time for an event.
Successful Events	List of successful events arranged by correlation ID. To retrieve more information for an event, select the event from the drop-down list.
Failed Events	List of failed events arranged by correlation ID. To retrieve more information for an event, select the event from the drop-down list.

4. Select a channel from the drop-down list.

The following image shows the Channel Level Event Summary (Channel Statistics) window that opens. The Channels pane contains a drop-down list on the left where you select a channel and a drop-down list on the right where you select a port. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a button to click to suspend or resume a channel and a home button to click to return to the iBSE Monitoring Settings window.

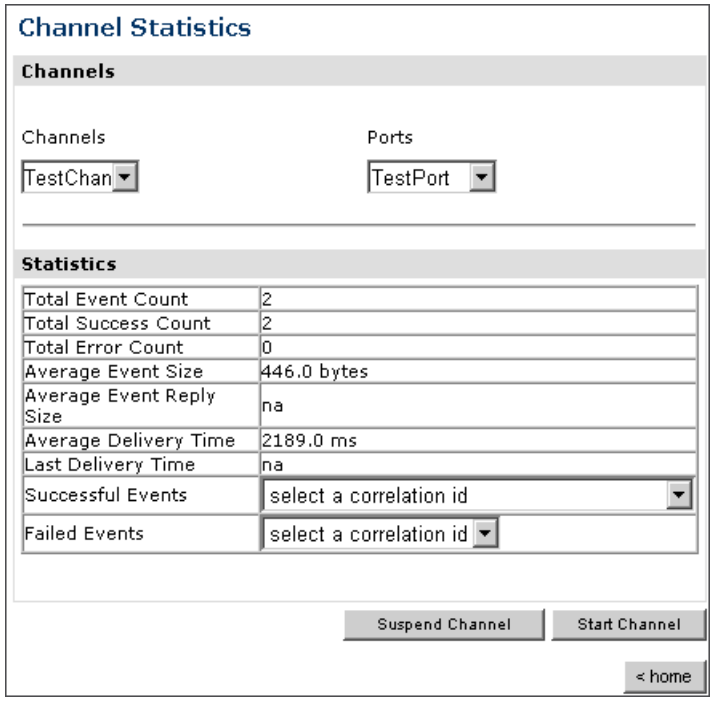


a. To stop a channel at any time, click *Suspend Channel*.

b. To start the channel, click *Start Channel*.

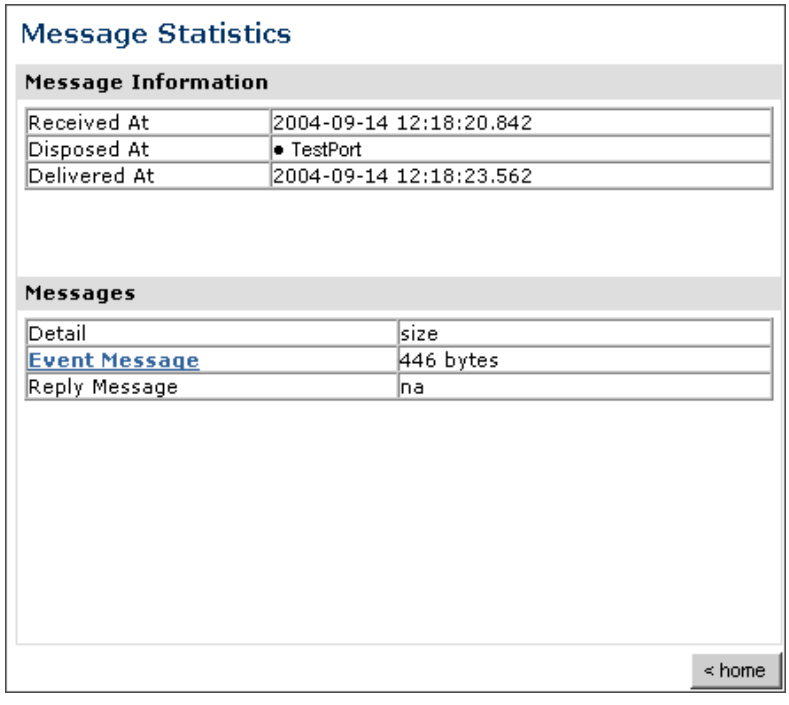
5. From the Ports drop-down list, select a port for the channel.

The following image shows the Port Level Event Summary (Channel Statistics) window that opens. The Channels pane contains a drop-down list on the left where you select a channel and a drop-down list on the right where you select a port. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a button to click to suspend or resume a channel and a home button to click to return to the iBSE Monitoring Settings window.



6. For more information about a successful event and its port, select an event based on its correlation ID from the Successful Events drop-down list.

The following image shows the Event Level Statistics (Message Statistics) window that opens. The Message Information pane contains a table of information pertaining to the event message. The Messages pane contains a table that shows the size of the event and reply messages, with an option to view an XML document of the event message. In the lower right of the window is a home button to click to return to the iBSE Monitoring Settings window.



a. To view the XML event document in your Web browser, click *Event Message*.

b. To return to the iBSE Monitoring Settings window, click *home*.

Managing and Monitoring Services and Events Using the JCA Test Tool

The JCA Test Tool, which is also known as the JCA Installation Verification Program (IVP), provides a console to manage and monitor services and events currently in use and to display resource usage and invocation statistics. These indicators can help you adjust your environment for optimum efficiency.

Procedure: How to Manage and Monitor Services Using the JCA Test Tool

To manage and monitor services using the JCA Test Tool:

1. Open a Web browser to:

   `http://localhost:port/iwjcaivp`

   where:

   localhost

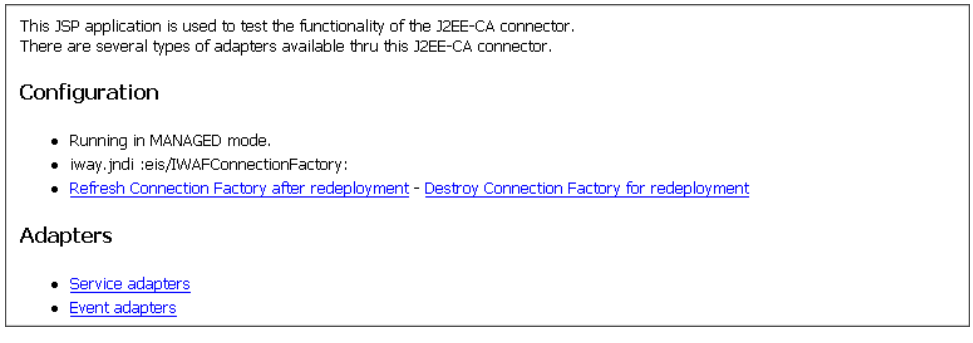
   Is the name of the machine where your application server is running.

   port

   Is the port for the domain you are using. The port for the default domain is 7001.,for example:

   `http://localhost:7001/iwjcaivp`

   The following image shows the JCA Test Tool page that opens. The page contains a description of the function of the tool and configuration information, including options to change your connection settings. It also provides options for viewing service or event adapters.



   The JCA Test Tool runs in managed mode by default.

2. Perform the following steps to monitor the latest service adapter configuration.

   Note: You must perform these steps for every new adapter target that is created using a JCA implementation of Application Explorer. In addition, you also must perform these steps for every new JCA configuration that is created using Application Explorer.

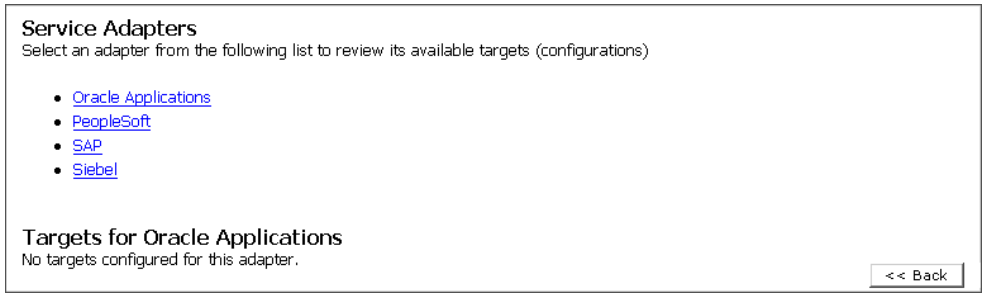
a. Click *Destroy Connection Factory for redeployment*.

b. Redeploy the JCA connector module using the BEA WebLogic Server console.

c. In the JCA Test Tool, click *Refresh Connection Factory after redeployment*.

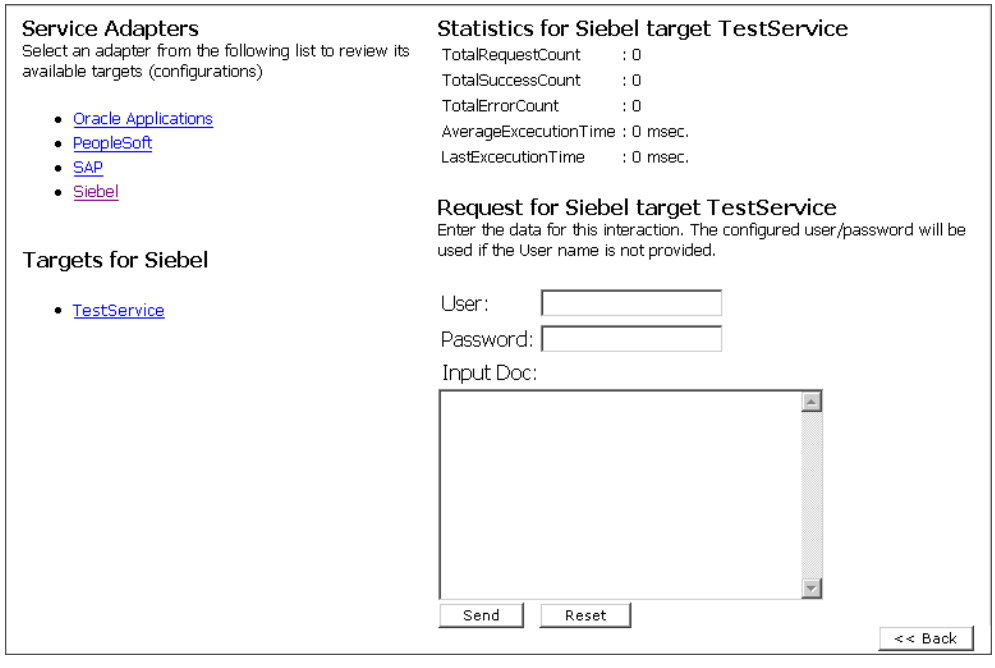
3. Click *Service adapters*.

The following image shows the Service Adapters page that opens. The page provides a live list of available service adapters and a list of targets configured for a specific adapter. In the lower right is a Back button to click to return to the previous page.



4. Select a service adapter to monitor.

The following image shows the page that opens. The left side provides a live list of available service adapters and a list of any targets configured for a specific adapter. The upper right side shows statistics for a selected target. The middle right has a User field and a Password field. The lower right contains a box where you type or paste an input document. Below the input box is a Send button to click to send a request for a test service and a Reset button to click to reset the fields. In the lower right is a Back button to click to return to the previous page.



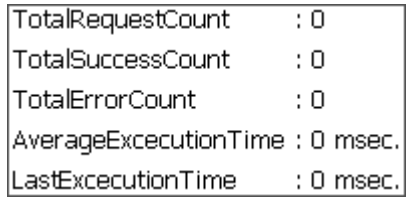
a. Click the desired target for your service adapter.

b. In the Request area, enter a user name and password.

c. In the Input Doc area, enter a request document that was created from the request schema for your service.

5. Click *Send*.

The following image shows the updated statistics that appear for your service if the request is successful. The statistics include the total number of requests, successes, and errors and the average and last execution time in milliseconds.



Procedure: How to Manage and Monitor Events Using the JCA Test Tool

To manage and monitor events using the JCA Test Tool:

1. Open a Web browser to:

   `http://localhost:port/iwjcaivp`

   where:

   localhost

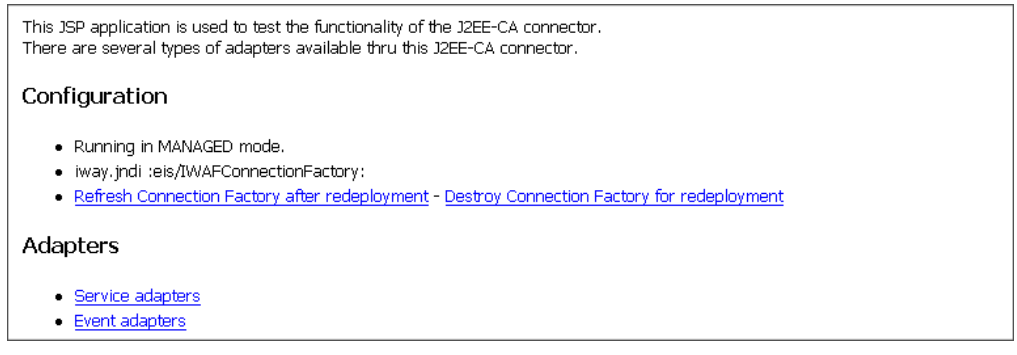
   Is the name of the machine where your application server is running.

   port

   Is the port for the domain you are using. The port for the default domain is 7001, for example:

   `http://localhost:7001/iwjcaivp`

   The following image shows the JCA Test Tool page that opens. The page contains a description of the function of the tool and configuration information, including options to change your connection settings. It also provides options for viewing service or event adapters.



   The JCA Test Tool runs in managed mode by default.

2. Perform the following steps to monitor the latest event adapter configuration.

   Note: You must perform these steps for every new adapter target that is created using a JCA implementation of Application Explorer. In addition, you must also perform these steps for every new JCA configuration that is created using Application Explorer.

   a. Click *Destroy Connection Factory for redeployment*.

   b. Redeploy the JCA connector module using the BEA WebLogic Server console.

   c. In the JCA Test Tool, click *Refresh Connection Factory after redeployment*.

3. Click *Event adapters*.

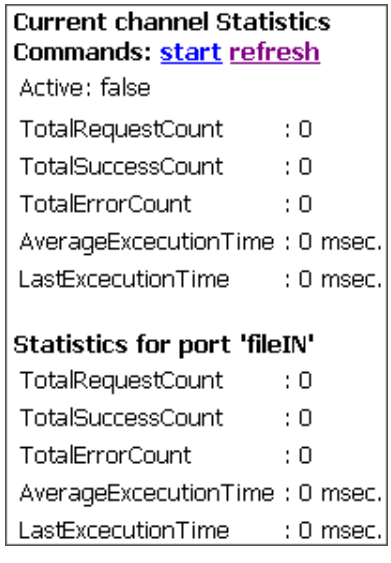
   The Event Adapters page opens.

4. Select the event adapter to monitor.

5. Click the desired channel for your event adapter.

6. Click *start*.

   The following image shows the updated statistics for your channel and the port. The statistics include the total number of requests, successes, and errors and the average and last execution time in milliseconds. There are options to click in the upper right of the page to start or refresh the channel.



Setting Engine Log Levels

The following section describes how to set engine log levels for Servlet iBSE and JCA. For more information, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

Procedure: How to Enable Tracing for Servlet iBSE

To enable tracing for Servlet iBSE:

1. Open the Servlet iBSE configuration page at:

   ```
   http://localhost:port/ibse/IBSEConfig
   ```

   where:

   localhost

   Is the name of the machine where your application server is running.

   port

   Is the port for the domain you are using. The port for the default domain is 7001, for example:

   ```
   http://localhost:7001/ibse/IBSEConfig
   ```

2. In the System pane, from the Debug drop-down list, select the level of tracing.
3. Click *Save*.

The default location for the trace information on Windows is:

```
C:\Program FIles\bea\ibse\ibselogs
```

Procedure: How to Enable Tracing for JCA

To enable tracing for JCA:

1. Open the extracted ra.xml file in a text editor.
2. Locate and change the following setting:

   LogLevel. This setting can be set to DEBUG, INFO, or ERROR.

   ```
   <context-param>
   <config-property>
       <config-property-name>LogLevel</config-property-name>
       <config-property-type>java.lang.String</config-property-type>
       <config-property-value></config-property-value>
   </config-property>
   ```

   For example:

   ```
   <config-property-value>DEBUG</config-property-value>
   ```

A directory in the configuration directory contains the logs.

a. Review the logs generated by your application server.

b. Leave the remainder of the previous file unchanged.

3. Save the file and exit the editor.

4. Redeploy the connector.

Configuring Connection Pool Sizes

The following topic describes how to configure connection pool sizes for the JCA connector.

Procedure: How to Configure Connection Pool Sizes

To configure connection pool sizes:

1. Open the extracted ra.xml file in a text editor.

2. Locate and change the following setting:

pool-params. The JCA Resource Connector has an initial capacity value of 0 by default and cannot be changed. The maximum capacity value is 10 by default and can be changed to a higher value.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE weblogic-connection-factory-dd (View Source for full
doctype...)>
- <weblogic-connection-factory-dd>
  <connection-factory-name>IWAFJCA</connection-factory-name>
  <jndi-name>eis/IWAFConnectionFactory</jndi-name>
  - <pool-params>
    <initial-capacity>0</initial-capacity>
    <max-capacity>10</max-capacity>
    <capacity-increment>1</capacity-increment>
    <shrinking-enabled>false</shrinking-enabled>
    <shrink-period-minutes>200</shrink-period-minutes>
  </pool-params>
  <security-principal-map />
</weblogic-connection-factory-dd>
```

3. Save the file and exit the editor.

4. Redeploy the connector.

Migrating Repositories

During design time, a repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. For more information on configuring repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

The information in the repository also is referenced at run time. For management purposes, you can migrate iBSE and JCA repositories to new destinations without affecting your existing configuration. For example, you may want to migrate a repository from a development environment to a production environment. The BEA WebLogic Server must be restarted to detect new repository changes.

File Repositories

If you want to migrate a File repository to another destination, copy the ibserepo.xml file from the following path:

```
drive:\Program Files\iWay55\bea\ibse\ibserepo.xml
```

where:

drive

Is the location of your iWay 5.5 installation.

You can place the ibserepo.xml file in a new location that is a root directory of the iBSE Web application, for example:

```
drive:\ProductionConfig\bea\ibse\ibserepo.xml
```

iBSE Repositories

The following topic describes how to migrate an iBSE repository that is configured for Oracle. You can follow the same procedure if you want to migrate an iBSE repository that is configured for Microsoft SQL Server 2000, Sybase, or DB2. However, when you are configuring a new environment, you must execute the script that creates the repository tables for your database. In addition, verify that all required files and drivers for your database are in the class path. For more information on configuring repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

Note: The following procedure allows you to migrate only Web services. If migrating event handling information is one of your requirements, you must migrate at the database level. For more information, see *Migrating Event Handling Configurations* on page 5-28.

Procedure: How to Migrate an iBSE Repository Configured for Oracle

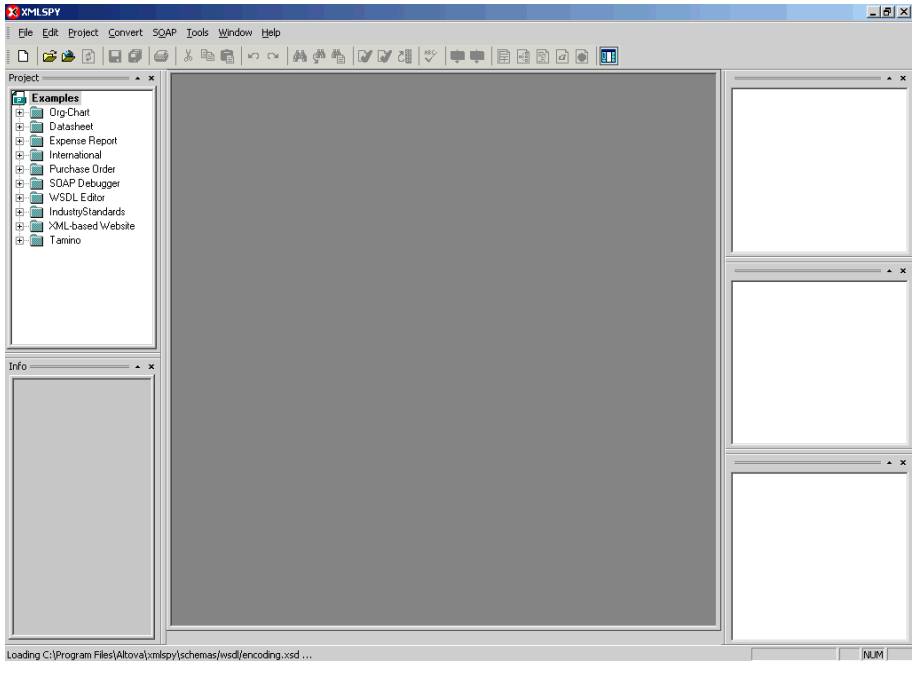
To migrate an iBSE repository that is configured for Oracle:

1. Copy the iBSE configuration service URL, for example:

   `http://localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl`

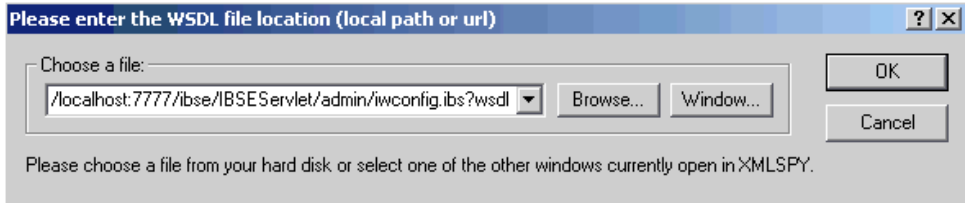
2. Open a third party XML editor, for example, XMLSPY.

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



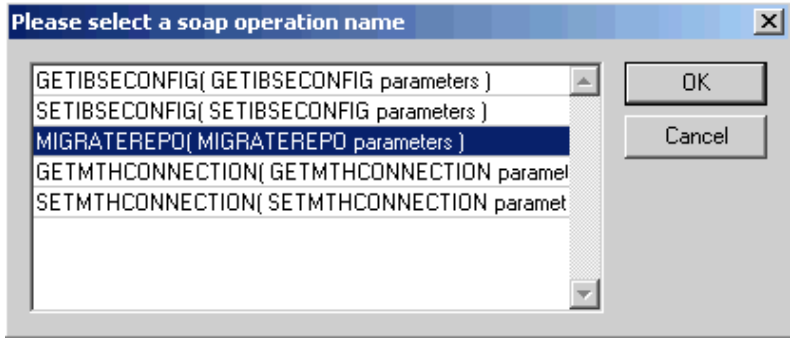
3. From the SOAP menu, select *Create new SOAP request*.

   The following image shows the WSDL file location dialog box that opens, where you enter a local path or URL. The dialog includes Browse, Window, OK, and Cancel buttons.



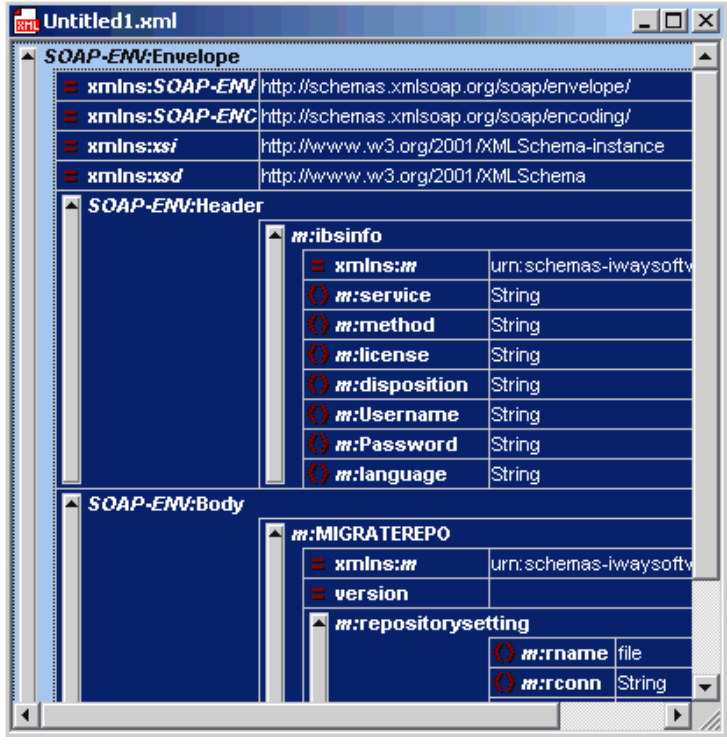
4. In the Choose a file field, paste the iBSE configuration service URL.
5. Click *OK*.

   The following image shows the soap operation name dialog box that opens with a list of available control methods. You can select from the list and click OK or to escape from the dialog box, you can click Cancel.



6. Select the *MIGRATEREPO(MIGRATEREPO parameters)* control method and click *OK*.

The following image shows a portion of the window that opens with the structure of the SOAP envelope. It includes information about location and schemas.



7. Locate the *Text view* icon in the tool bar.

In the following image, the pointer points to the Text view icon.

8. To display the structure of the SOAP envelope as text, click the *Text view* icon.

The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```
<m:MIGRATEREPO
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config" version="">
<m:repositorysetting>
<m:rname>oracle</m:rname>
<m:rconn>String</m:rconn>
<m:rdriver>String</m:rdriver>
<m:ruser>String</m:ruser>
<m:rpwd>String</m:rpwd>
</m:repositorysetting>
<m:servicename>String</m:servicename>
</m:MIGRATEREPO>
```

a. For the <m:rconn> tag, replace the String placeholder with the repository URL where you want to migrate your existing iBSE repository.

For example, the Oracle repository URL has the following format:

```
jdbc:oracle:thin:@[host]:[port]:[sid]
```

b. For the <m:rdriver> tag, replace the String placeholder with the location of your Oracle driver.

Note: This is an optional tag. If you do not specify a value, the default Oracle JDBC driver is used.

c. For the <m:ruser> tag, replace the String placeholder with a valid user name to access the Oracle repository.

d. For the <m:rpwd> tag, replace the String placeholder with a valid password to access the Oracle repository.

10. Perform one of the following migration options.

If you want to migrate a **single** Web service from the current iBSE repository, enter the Web service name in the <m:servicename> tag, for example:

```
<m:servicename>Service1</m:servicename>
```

If you want to migrate **multiple** Web services from the current iBSE repository, duplicate the <m:servicename> tag for each Web service, for example:

```
<m:servicename>Service1</m:servicename>
<m:servicename>Service2</m:servicename>
```

If you want to migrate **all** Web services from the current iBSE repository, remove the <m:servicename> tag.

11. From the SOAP menu, select *Send request to server*.

Your iBSE repository and the Web services you specified migrate to the new Oracle repository URL that you specified.

JCA Repositories

The following procedure describes how to migrate a JCA repository. For more information on configuring JCA repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

Procedure: How to Migrate a JCA Repository

To migrate a JCA repository:

1. Navigate to the location of your JCA configuration directory where the repository schemas and other information is stored, for example:

   ```
   C:\Program Files\iway55\config\base
   ```

2. Locate and copy the *repository.xml* file.

3. Place this file in a new JCA configuration directory to migrate the existing repository.

   Your JCA repository migrates to the new JCA configuration directory.

Migrating Event Handling Configurations

This topic describes how to migrate your iBSE repositories at a database level for Microsoft SQL Server 2000, Oracle, Sybase, or DB2. You can use this information to migrate event handling information, for example, port or channel configurations.

Procedure How to Migrate a Microsoft SQL Server 2000 Repository

To migrate a Microsoft SQL Server 2000 repository:

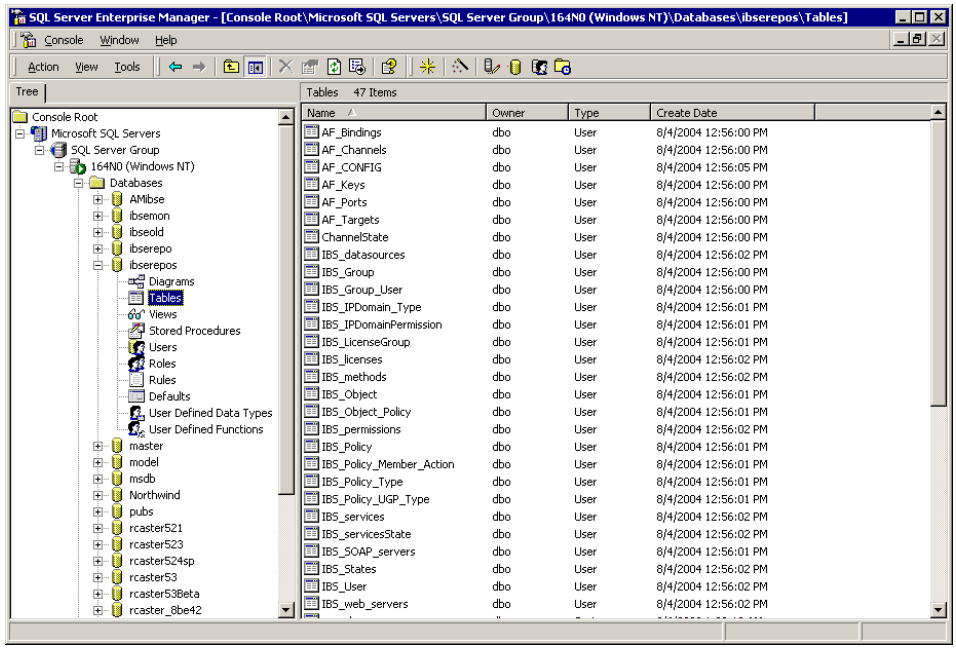
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

   ```
   C:\Program Files\iWay55\etc\setup
   ```

   This directory contains SQL to create the repository tables in the following file:

   ```
   iwse.sql
   ```

You can use iwse.sql to create the database tables that are used by iBSE. For example, the following image shows the tree in the left pane and tables in the right pane. The tables are listed by name in one column with corresponding columns for information about owner, type, and the date the table was created.



For more information on configuring the Microsoft SQL Server 2000 repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

2. To migrate the tables that were created by the iwse.sql script for iBSE, use your Microsoft SQL Server 2000 database tool set. For more information, consult your database administrator.

Procedure How to Migrate an Oracle Repository

To migrate an Oracle repository:

1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

   ```
   C:\Program Files\iWay55\etc\setup
   ```

   This directory contains SQL to create the repository tables in the following files:

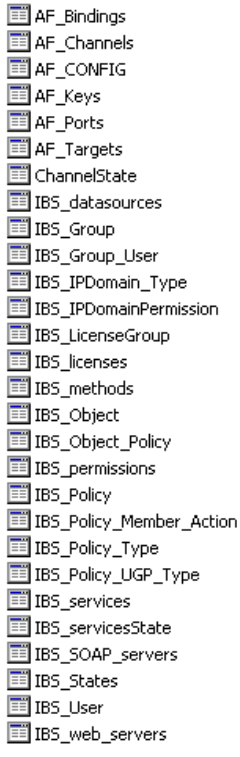
   For Oracle 8:

   ```
   iwse.ora
   ```

For Oracle 9:

```
iwse.ora9
```

2. To create the Oracle database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.



For more information on configuring the Oracle repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

3. To migrate the tables that were created by the SQL script for iBSE, use your Oracle database tool set. For more information, consult your database administrator.

Procedure How to Migrate a Sybase Repository

To migrate a Sybase repository:

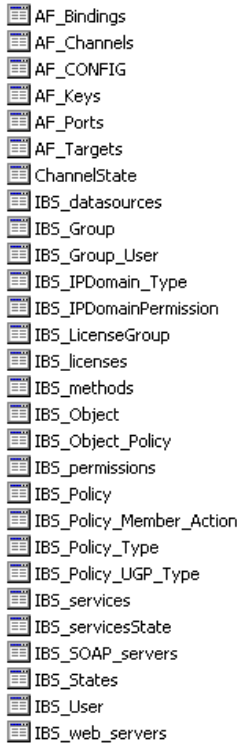
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

```
C:\Program Files\iWay55\etc\setup
```

This directory contains SQL to create the repository tables in the following file:

```
sybase-iwse.sql
```

2. To create the Sybase database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.



For more information on configuring the Sybase repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

3. To migrate the tables that were created by the SQL script for iBSE, use your Sybase database tool set. For more information, consult your database administrator.

Procedure How to Migrate a DB2 Repository

To migrate a DB2 repository:

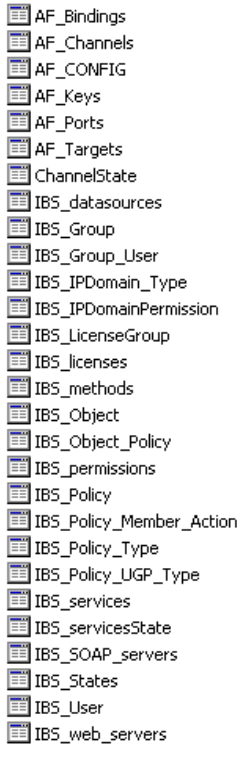
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

```
C:\Program Files\iWay55\etc\setup
```

This directory contains SQL to create the repository tables in the following file:

```
db2-iwse.sql
```

2. To create the DB2 database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.



For more information on configuring the DB2 repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

You can migrate the tables that were created by the SQL script for iBSE using your DB2 database toolset. For more information, consult your database administrator.

Exporting or Importing Targets

After you migrate your repository, you can export or import targets with their connection information and persistent data between repositories.

Procedure: How to Export a Target

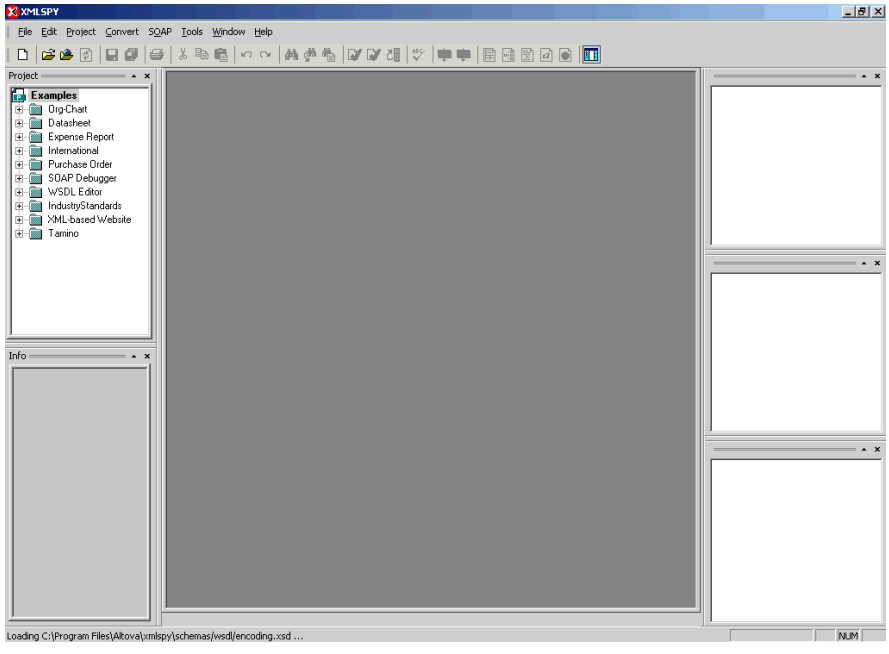
To export a target:

1. Copy the iBSE administrative services for Application Explorer URL, for example:

```
http://localhost:7777/ibse/IBSEServlet/admin/iwae.ibs?wsdl
```

2. Open a third party XML editor, for example, XMLSPY.

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE administrative services for Application Explorer URL.

5. Click *OK*.

   The soap operation name dialog box opens and lists the available control methods.

6. Select the *EXPORTTARGET(EXPORTTARGET parameters)* control method and click *OK*.

   A window opens that shows the structure of the SOAP envelope.

7. Locate the *Text view* icon in the tool bar.

In the following image, the pointer points to the Text view icon.

8. To display the structure of the SOAP envelope as text, click the *Text view* icon.

   The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```
<m:EXPORTTARGET
xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:af">
<m:target>String</m:target>
<m:name>String</m:name>
</m:EXPORTTARGET>
```

   a. For the <m:target> tag, replace the String placeholder with the EIS target system name as it appears in Application Explorer and verify whether this value is case sensitive.

   b. For the <m:name> tag, replace the String placeholder with the name of the target you want to export.

10. From the SOAP menu, select *Send request to server.*

    A response is returned that contains the <m: exporttime> and <m: contents> elements. You must use these elements when importing your target.

Procedure: How to Import a Target

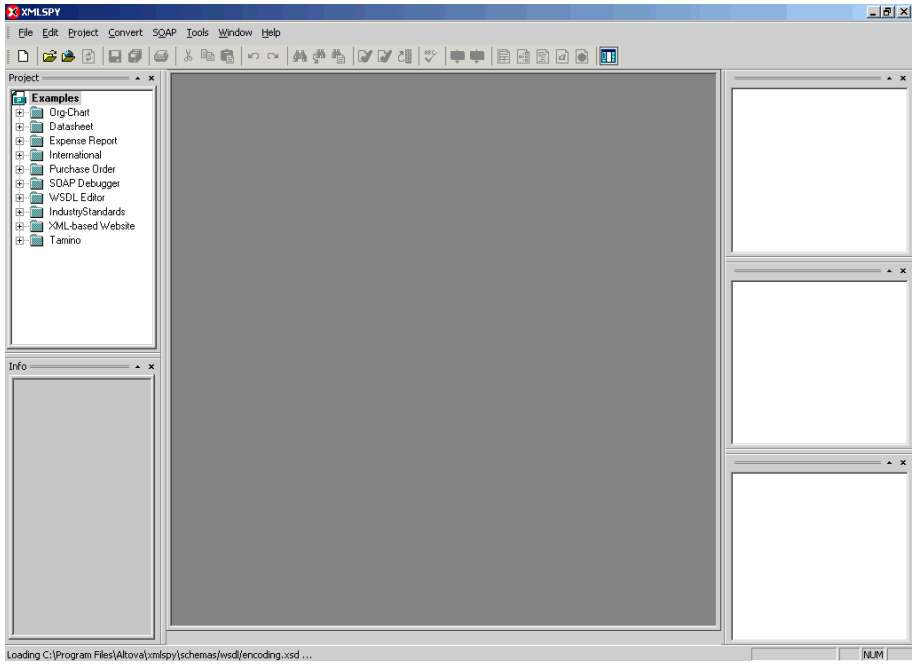
To import a target:

1. Copy the iBSE administrative services for Application Explorer URL, for example:

```
http://localhost:7777/ibse/IBSEServlet/admin/iwae.ibs?wsdl
```

2. Open a third party XML editor, for example, XMLSPY.

The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE administrative services for Application Explorer URL and click *OK*.

The soap operation name dialog box opens and lists the available control methods.

5. Select the *IMPORTTARGET(IMPORTTARGET parameters)* control method and click *OK*.

A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the toolbar.

In the following image, the pointer points to the Text view icon.

7. To display the structure of the SOAP envelope as text, click the *Text view* icon.

The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

8. Locate the following section:

```
<m:IMPORTTARGET
xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:af">
<m:targetinstance>
<m:target>String</m:target>
<m:name>String</m:name>
<m:description>String</m:description>
<m:repositoryid>String</m:repositoryid>
<m:exporttime>2001-12-17T09:30:47-05:00</m:exporttime>
<m:contents>R0lGODlhcgGSALMAAAQCAEMmCZtuMFQxDS8b</m:contents>
</m:targetinstance>
</m:IMPORTTARGET>
```

   a. For the <m:target> tag, replace the String placeholder with the EIS target system name.

   b. For the <m:name> tag, replace the String placeholder with the new name of the target you want to import.

   c. For the <m:description> tag, replace the String placeholder with a description of the target.

   d. For the <m:repositoryid> tag, copy and paste the contents of the <m:repositoryid> tag that was returned when you exported your target.

   e. For the <m: exporttime> tag, copy and paste the contents of the <m: exporttime> tag that was returned when you exported your target.

   f. For the <m: contents> tag, copy and paste the contents of the <m: contents> tag that was returned when you exported your target.

9. From the SOAP menu, select *Send request to server*.

Retrieving or Updating Web Service Method Connection Information

After you migrate your repository, you can retrieve or update connection information for your Web service methods.

Procedure: How to Retrieve Web Service Method Connection Information

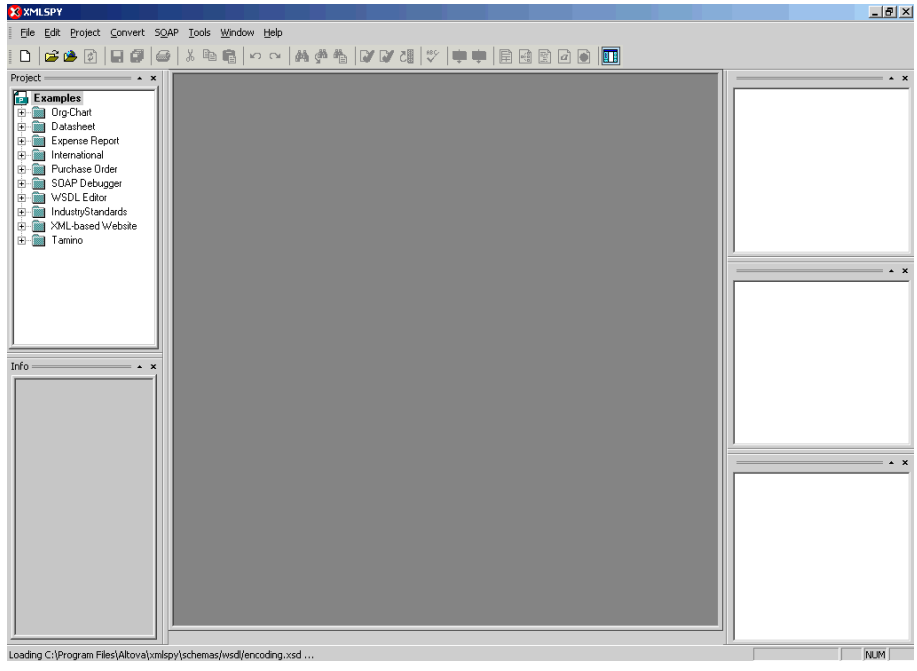
To retrieve Web service method connection information:

1. Copy the iBSE configuration service URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl
   ```

2. Open a third party XML editor, for example, XMLSPY.

The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE configuration service URL, and click *OK*.

The soap operation name dialog box opens and lists the available control methods.

5. Select the *GETMTHCONNECTION(GETMTHCONNECTION parameters)* control method and click *OK*.

A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the toolbar.

In the following image, the pointer points to the Text view icon.

7. To display the structure of the SOAP envelope as text, click the *Text view* icon.

The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

8. Locate the following section:

```
<m:GETMTHCONNECTION
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config">
<m:servicename>String</m:servicename>
<m:methodname>String</m:methodname>
</m:GETMTHCONNECTION>
```

   a. For the <m:servicename> tag, replace the String placeholder with the name of the Web service.

   b. For the <m:methodname> tag, replace the String placeholder with name of the Web service method.

9. From the SOAP menu, select *Send request to server.*

   A response is returned that contains the <m: descriptor> element. You must use this element when updating your Web service method.

Procedure: How to Update Web Service Method Connection Information

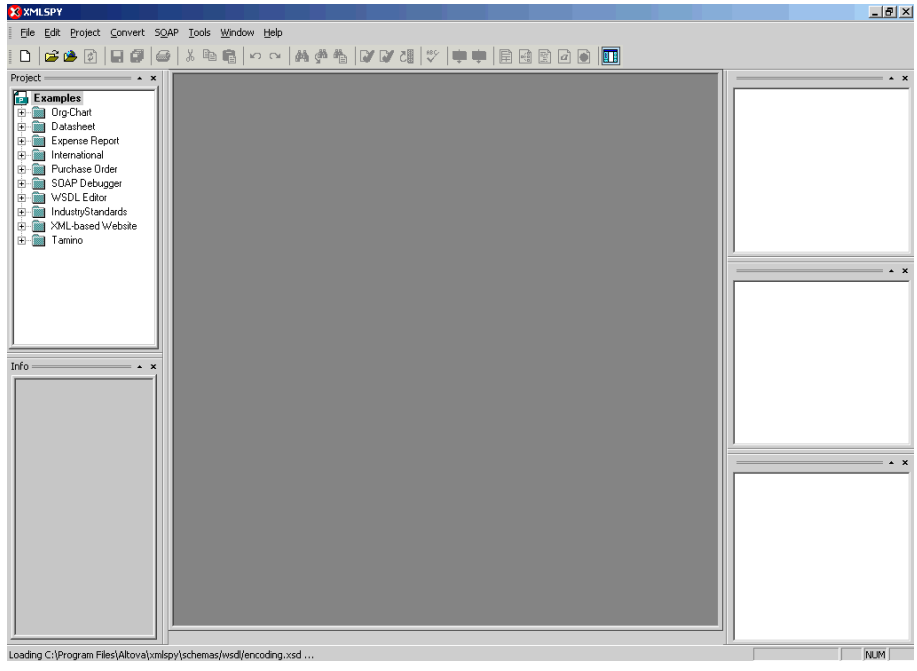
To update Web service method connection information:

1. Copy the iBSE configuration service URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl
   ```

2. Open a third party XML editor, for example, XMLSPY.

The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE configuration service URL, and click *OK*.

The soap operation name dialog box opens and lists the available control methods.

5. Select the *SETMTHCONNECTION(SETMTHCONNECTION parameters)* control method and click *OK*.

A window opens that shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the toolbar.

In the following image, the pointer points to the Text view icon.

7. To display the structure of the SOAP envelope as text, click the *Text view* icon.

The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

8. Locate the following section:

```
<m:SETMTHCONNECTION
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config">
<m:servicename>String</m:servicename>
<m:methodname>String</m:methodname>
<m:descriptor format="" channel="">
     <m:option title="">
       <m:group title="">
          <m:param/>
       </m:group>
     </m:option>
</m:descriptor>
</m:SETMTHCONNECTION>
```

   a. For the <m:servicename> tag, replace the String placeholder with the name of the Web service.

   b. For the <m:methodname> tag, replace the String placeholder with the name of the Web service method.

   c. For the <m: descriptor> tag, copy and paste the contents of the <m: descriptor> tag that was returned when you retrieved Web Service method connection information.

9. Modify the contents of the <m: descriptor> tag to change the existing Web Service method connection information.

10. From the SOAP menu, select *Send request to server.*

Starting or Stopping a Channel Programmatically

The following topic describes how to start or stop a channel programmatically.

Procedure: How to Start a Channel Programmatically

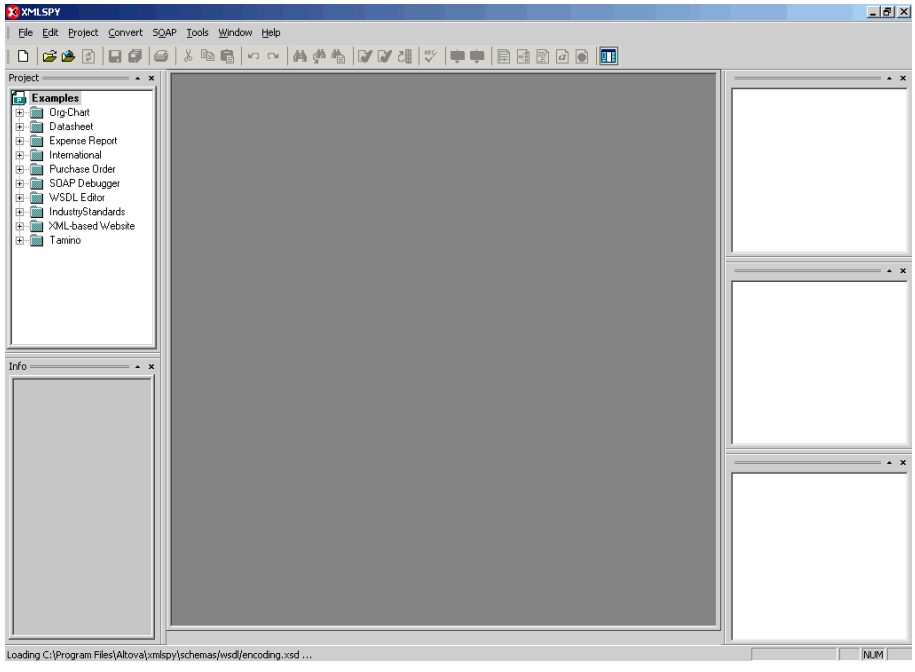
To start a channel programmatically:

1. Copy the iBSE control event URL, for example:

   ```
   http://localhost:7777/ibse/IBSEServlet/admin/iwevent.ibs?wsdl
   ```

2. Open a third party XML editor, for example, XMLSPY.

The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.

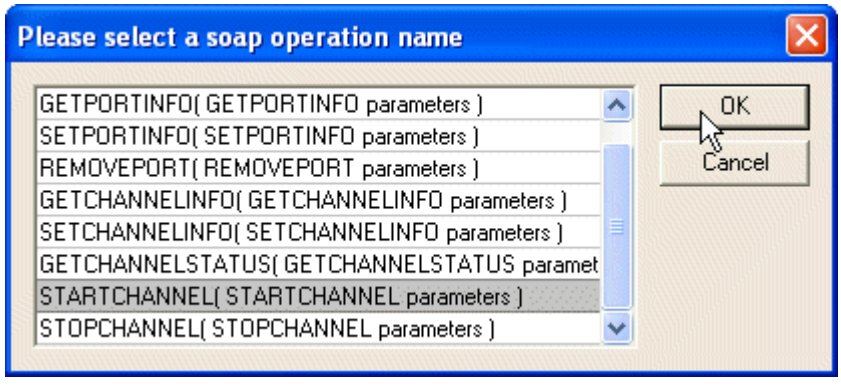


3. From the SOAP menu, select *Create new SOAP request*.

The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE control event URL, and click *OK*.

The following image shows the soap operation name dialog box that opens with a list of available control methods. You can select one and click OK or to escape from the dialog box, you can click Cancel.



5. Select the *STARTCHANNEL(STARTCHANNEL parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the toolbar.

   In the following image, the pointer points to the Text view icon.

7. To display the structure of the SOAP envelope as text, click the *Text view* icon.

   The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

8. Locate the following section:

```
<SOAP-ENV:Body>
  <m:STARTCHANNEL
     xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:event">
           <m:channel>String</m:channel>
  </m:STARTCHANNEL>
</SOAP-ENV:Body>
```

9. For the <m:channel> tag, replace the String placeholder with the name of the Channel you want to start.

10. From the SOAP menu, select *Send request to server.*

Procedure: How to Stop a Channel Programmatically

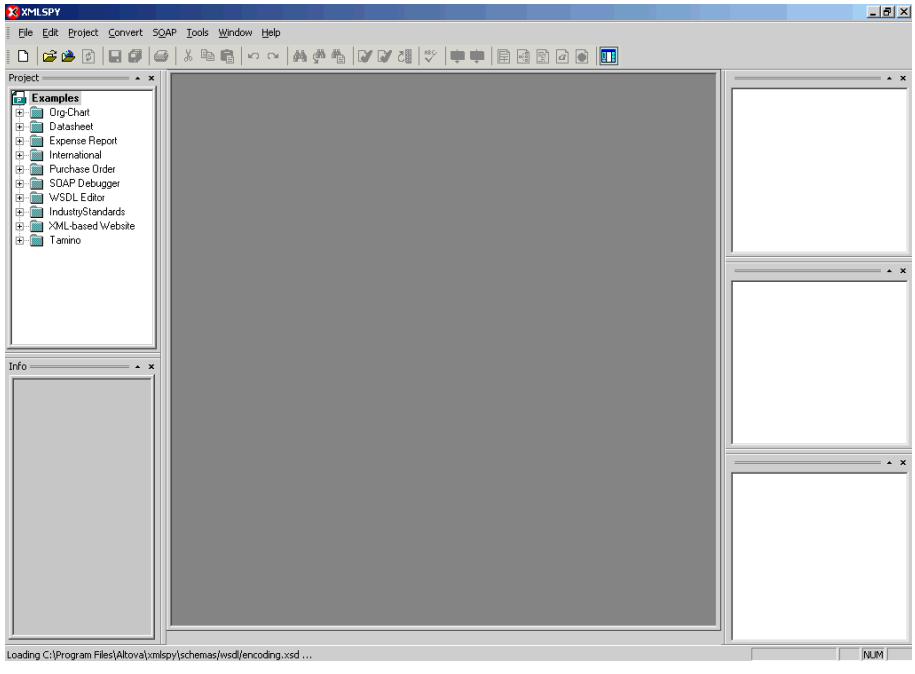
To stop a channel programmatically:

1. Copy the iBSE control event URL, for example:

   `http://localhost:7777/ibse/IBSEServlet/admin/iwevent.ibs?wsdl`

2. Open a third party XML editor, for example, XMLSPY.

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.

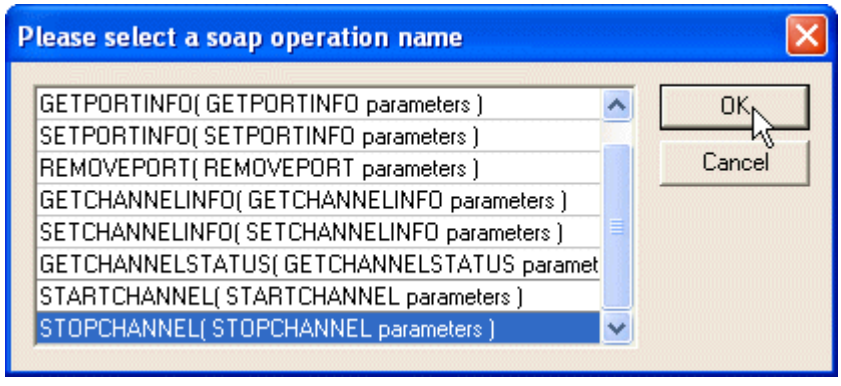


3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE control event URL, and click *OK*.

The following image shows the soap operation name dialog box that opens with a list of available control methods. You can select one and click OK or to escape from the dialog box, you can click Cancel.



5. Select the *STOPCHANNEL(STOPCHANNEL parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the toolbar.

   In the following image, the pointer points to the Text view icon.

7. To display the structure of the SOAP envelope as text, click the *Text view* icon.

   The <SOAP-ENV:Header> tag is not required and can be deleted from the SOAP envelope.

8. Locate the following section:

```
<SOAP-ENV:Body>
  <m:STOPCHANNEL
     xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:event">
           <m:channel>String</m:channel>
  </m:STOPCHANNEL>
</SOAP-ENV:Body>
```

9. For the <m:channel> tag, replace the String placeholder with the name of the Channel you want to stop.

10. From the SOAP menu, select *Send request to server.*

CHAPTER 6

Troubleshooting and Error Messages

- Troubleshooting
- iBSE Error Messages

The following section explains limitations and workarounds when connecting to SWIFT. The adapter-specific errors listed in this chapter can arise whether using the adapter with a JCA, or with an iBSE configuration.

Troubleshooting

This topic provides troubleshooting information for SWIFT, separated into four categories:

- Application Explorer
- JCA
- iBSE

Note: Log file information that can be relevant in troubleshooting can be found in the following locations:

- The JCA trace information can be found under the `C:\Program Files\iWay55\config\base\log` directory.
- iBSE trace information can be found under the `C:\Program Files\iWay55\bea\ibse\ibselogs` directory.
- The log file for Application Explorer can be found under the `C:\Program File\iWay55\tools\iwae\bin` directory.

Application Explorer

Error	Solution
Cannot find your SWIFT dictionary for services. The following error message appears: `<?xml version="1.0" encoding="UTF-8" ?><eda><error timestamp="2004-10-14T19:51:40Z">XD[FAIL] cause: 0 subcause: 0 message: com.iwaysoftware.transform.processor.kernel.KernelException: Could not load Structure for SWIFT. e:\Program Files\iWay55\config\base\swift\2003\dictionaries\MT103.dic: The system cannot find the file specified</error></eda>`	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the dictionary file.

Error	Solution
Cannot find your SWIFT template for a service. The following error message appears: `<?xml version="1.0" encoding="UTF-8" ?><eda><error timestamp="2004-10-14T19:58:18Z">XD[FAIL] cause: 0 subcause: 0 message: Transform template file :e:\Program Files\iWay55\config\base\swift\2003\templates\XMLtoMT103.xch, does not exist.</error></eda>`	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the template file.
Your SWIFT schema is missing. The following error message appears when you create a Web service using Application Explorer: `schema is missing for the component`	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the schema file.
Your SWIFT XML document is invalid for a service. The following error message appears: `<?xml version="1.0" encoding="UTF-8" ?><eda><error timestamp="2004-10-14T20:03:35Z">XD[FAIL] cause: 0 subcause: 0 message: com.iwaysoftware.transform.processor.output.OutputFormatException: Dictionary and output don not match for SWIFT output element INFO.</error></eda>`	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the XML document.

Error	Solution
Cannot find your SWIFT template for an event. The following error message appears: <?xml version="1.0" encoding="ISO-8859-1" ?><eda><error code="2" timestamp="2004-10-14T20:27:34Z" source="com.ibi.preparsers.XDSWIFTpreParser" stage="PREPARSE">XD[FAIL] cause: 0 subcause: 0 message: Transform template file :e:\Program Files\iWay55\config\base\swift\2003\templates\MT103toXML.xch, does not exist.</error></eda>	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the template file.
Cannot find your SWIFT dictionary for events. The following error message appears: <?xml version="1.0" encoding="ISO-8859-1" ?><eda><error code="2" timestamp="2004-10-14T20:29:27Z" source="com.ibi.preparsers.XDSWIFTpreParser" stage="PREPARSE">XD[FAIL] cause: 0 subcause: 0 message: com.iwaysoftware.transform.processor.kernel.KernelException: Could not load Structure for SWIFT. e:\Program Files\iWay55\config\base\swift\2003\dictionaries\MT103.dic: The system cannot find the path specified.</error></eda>	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the dictionary file.

Error	Solution
Your SWIFT XML document is invalid for an event. The following error message appears: <?xml version="1.0" encoding="ISO-8859-1" ?><eda><error code="2" timestamp="2004-10-14T21:01:11Z" source="com.ibi.preparsers.XDSWIFTpreParser" stage="PREPARSE">XD[FAIL] cause: 0 subcause: 0 message: com.iwaysoftware.transform.TransformException: Error occurred when processing SWIFT input 71A:OUR. Detail: Attempted mandatory loop or segment SWIFT/103 while processing input 71A:OUR.The possible reason for this error could be 1. Sequence violation (most cases). 2. Input pattern of 71A:OUR is not defined in the dictionary.</error></eda>	Using Application Explorer, disconnect from your SWIFT target and reconnect to extract the XML document.

JCA

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to a JCA configuration: `Could not initialize JCA`	In the Details tab in the right pane, ensure that the directory specified in the Home field points to the correct directory, for example, ***iway_home***`/lib`

iBSE Error Messages

This topic discusses the different types of errors that can occur when processing Web services through the Integration Business Services Engine (iBSE).

General Error Handling in iBSE

The iBSE serves as both a SOAP gateway into the adapter framework and as the engine for some of the adapters. In both design time and execution time, various conditions can cause errors in iBSE when Web services that use adapters are running. Some of these conditions and resulting errors are exposed the same way, regardless of the specific adapter; others are exposed differently, based on the adapter being used. This topic explains what you can expect when you encounter some of the more common error conditions on an adapter-specific basis.

Usually the SOAP gateway (**agent**) inside iBSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, how it is exposed depends on the adapter and the API or interfaces that the adapter uses. A few scenarios cause the SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside iBSE receives an invalid SOAP request, a SOAP fault element is generated in the SOAP response. The SOAP fault element contains fault string and fault code elements. The fault code contains a description of the SOAP agent error.

The following SOAP response document results when iBSE receives an invalid SOAP request:

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body>
      <SOAP-ENV:Fault>
         <faultcode>SOAP-ENV:Client</faultcode>
         <faultstring>Parameter node is missing</faultstring>
      </SOAP-ENV:Fault>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

In this example, iBSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this Web service.

Adapter-Specific Error Handling

When an adapter raises an exception during execution, the SOAP agent in iBSE produces a SOAP fault element in the generated SOAP response. The SOAP fault element contains fault code and fault string elements. The fault string contains the native error description from the adapter target system. Since adapters use the target system interfaces and APIs, whether or not an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in iBSE, and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

APPENDIX A

Using Application Explorer in BEA WebLogic Workshop to Create XML Schemas and Web Services

Topics:

- Starting Application Explorer in BEA WebLogic Workshop
- Creating a New Configuration
- Connecting to SWIFT
- Creating an XML Schema
- Creating an iWay Business Service
- Adding a Control for an iWay Resource in BEA WebLogic Workshop
- Adding an iWay Extensible CCI Control to a BEA WebLogic Workshop Application

This section describes how to use iWay Java Swing Application Explorer running in BEA WebLogic Workshop to create XML schemas for SWIFT.

Starting Application Explorer in BEA WebLogic Workshop

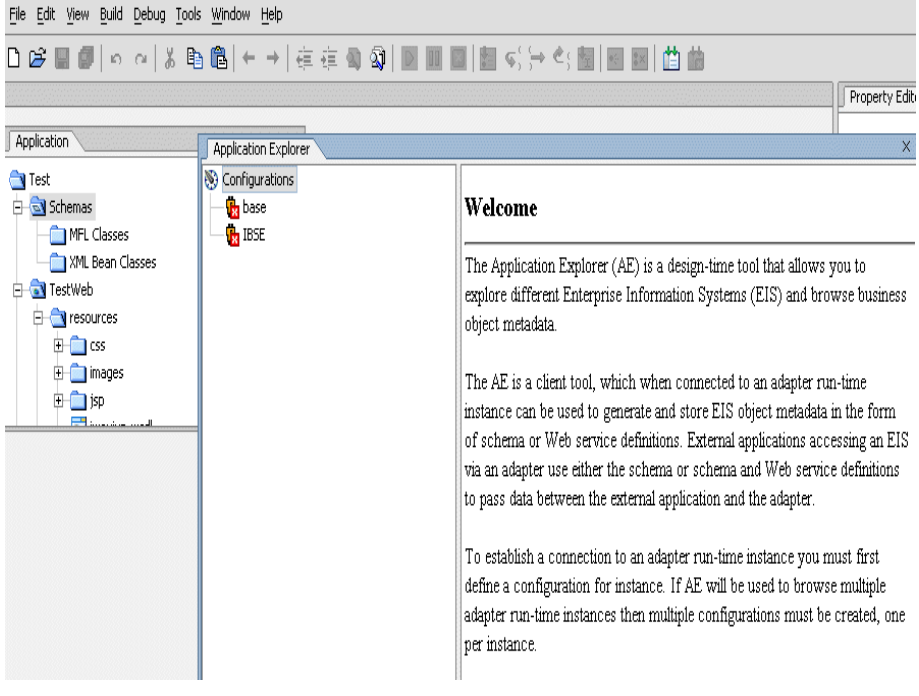
The server must be started where iWay Application Explorer is running. Before you can use Application Explorer, you must start BEA WebLogic server.

Procedure How to Start Application Explorer in BEA WebLogic Workshop

To start Application Explorer running in BEA WebLogic Workshop:

1. Before starting Application Explorer, ensure that BEA WebLogic Server is running.
2. Start BEA WebLogic Workshop.
3. From the BEA WebLogic Workshop View menu, select *Windows* and then, *iWay Application Explorer.*

   Application Explorer opens in BEA WebLogic Workshop.



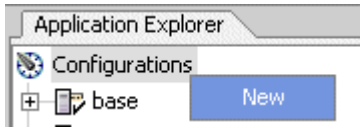
You can resize and drag-and-drop the Application Explorer window within BEA WebLogic Workshop. For example, you can drag it to the upper part of BEA WebLogic Workshop.

Creating a New Configuration

Before you can start using Application Explorer, you must define a new configuration for iBSE or JCA.

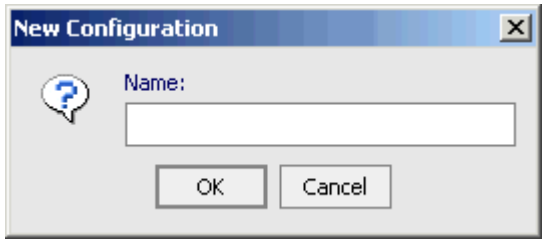
Procedure **How to Create a New Configuration for iBSE or JCA**

To create a new configuration:



1. Right-click *iWay Configurations* and select *New*.

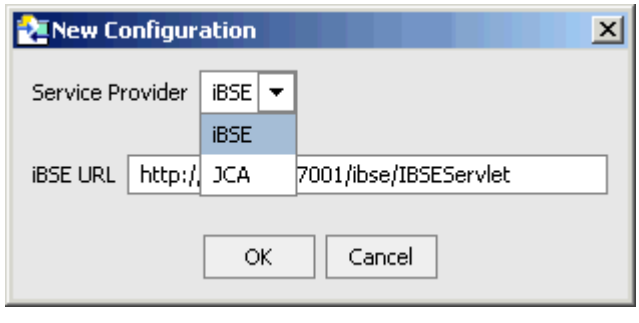
   The New Configuration dialog box opens.



2. Type the name of the new configuration and click *OK*.

   Note: If you are creating a new JCA configuration, type *base* in the name field. You must use this value if you are pointing to the default iWay configuration.

   The following dialog box opens.



3. From the Service Provider drop-down list, select *iBSE* or *JCA*.

- If you select iBSE, type the URL for iBSE, for example,

```
http://localhost:7001/ibse/IBSEServlet
```

where:

localhost

Is where your application server is running.

- If you select JCA, enter the full path to the directory where iWay 5.5 is installed, for example,

```
C:\Program Files\iWay55
```

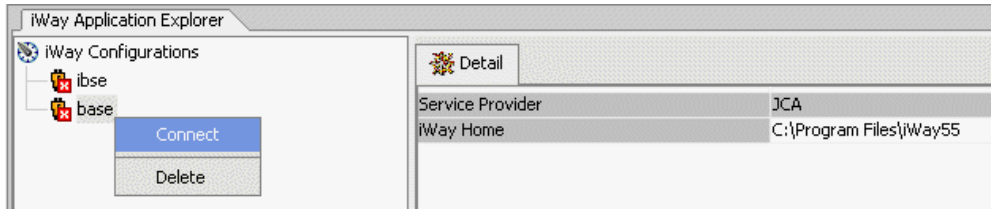
where:

iWay55

Is the full path to your iWay installation.

A node representing the new configuration appears under the iWay Configurations node. The right pane provides details of the configuration you created.

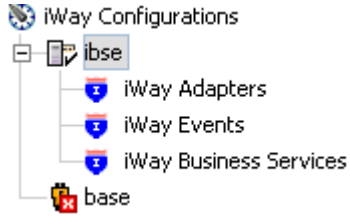
After you add your configuration, you must connect to it.



4. Right-click the configuration to which you want to connect, for example, base, and select *Connect*.

The iWay Adapters and iWay Events nodes appear.

When you connect to iBSE, the iWay Adapters, iWay Events, and iWay Business Services nodes appear.



5. To display the service and event adapters that are installed, expand each node.

   The iWay Adapters list includes a SWIFT node that enables you to connect to SWIFT metadata and create XML request and response schemas to use to listen for events or create Web Services. For more information, see *Creating an iWay Business Service* on page A-15.

   The iWay Events list includes a SWIFT node that enables you to create ports and channels for SWIFT event handling. For more information, see *Understanding iWay Event Functionality* on page B-3.

Connecting to SWIFT

To browse SWIFT, you must create an SWIFT target and connect to it. The target serves as your connection point. You must establish a connection to SWIFT every time you start iWay Application Explorer or after you disconnect from SWIFT.

The left pane displays the application systems supported by Application Explorer. These are based on the iWay adapters you installed and are licensed to use.

Creating and Connecting to a Target

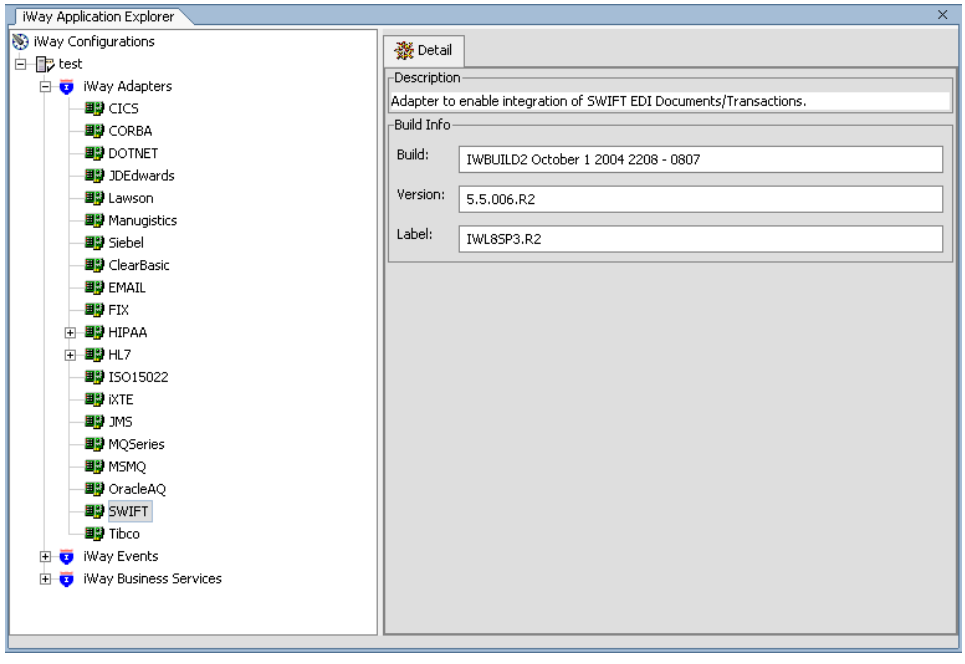
To connect to SWIFT for the first time, you must create a new target. The target is automatically saved after it is created.

Procedure How to Create a New Target

To create a target:

1. In the left pane, expand *iWay Adapters* and click the *SWIFT* node.

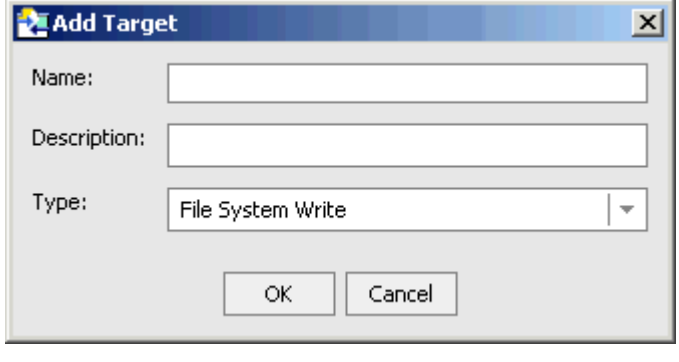
Descriptive information (for example, title and product version) for the iWay Adapter for SWIFT appears in the right pane.



2. To view the options, right-click the *SWIFT* node.

3. Select *Add Target*.

The Add target dialog box opens.



a. In the Name field, type a descriptive name for the target, for example, SWIFTTarget.

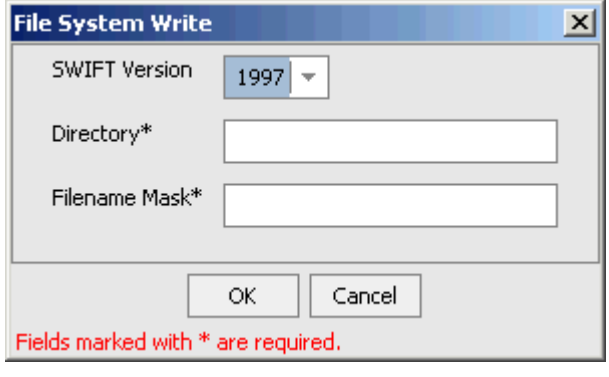
b. In the Description field, type a brief description of the target.

c. From the Target Type drop-down list, select one of the following transports from the drop-down list:

- File System Write. For more information on the properties required, see *File System Write Properties* on page A-8.
- File Transfer Protocol (FTP). For more information on the properties required, see *File Transfer Protocol Properties* on page A-9.
- HyperText Transfer Protocol (HTTP). For more information on the properties required, see *HyperText Transfer Protocol Properties* on page A-10.
- IBM MQSeries (MQ). For more information on the properties required, see *MQSeries Properties* on page A-10.
- TCP Session. For more information on the properties required, see *TCP Properties* on page A-11.

4. Click *OK*.

The File System Write dialog box opens.



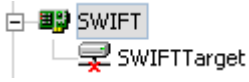
a. From the SWIFT Version drop-down list, select the SWIFT version you are using.

b. In the Directory field, type the location where the output of the service is placed.

c. In the Filename Mask field, type a file pattern, which can contain an asterisk which gets expanded to a fine timestamp.

5. Click *OK*.

In the left pane, the new target (SWIFTTarget) appears below the SWIFT node.



You can now connect to the target you defined.

Reference File System Write Properties

The following table provides definitions for the properties required for the File System Write target type.

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Directory	The directory to which output messages are emitted.
Filename Mask	The output file name (can contain an asterisk), which gets expanded to a timestamp. A pound sign can be used as a mask for a sequence count. Each pound symbol represents a whole number integer value. For example, File## counts up to 99 before restarting at 0, File### counts up to 999 before restarting at 0, and so on.

Reference **File Transfer Protocol Properties**

The following table provides definitions for the properties required for the File Transfer Protocol target type.

Settings tab

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Host	FTP target system.
Port	FTP target system port.
User	User ID to use when connecting to the FTP host.
Password	Password associated with the user ID.
Directory	The directory to which output messages are emitted.
Filename Mask	The output file name (can contain an asterisk), which gets expanded to a timestamp. A pound sign can be used as a mask for a sequence count. Each pound symbol represents a whole number integer value. For example, File## counts up to 99 before restarting at 0, File### counts up to 999 before restarting at 0, and so on.

Advanced tab

Property	Definition
Retry Interval	The maximum wait interval between retries when a connection fails. Retry interval duration in *xx*H:*xx*M:*xx*S format. For example, 1H:2M:3S is 1 hour 2 minutes and 3 seconds.
Maxtries	Maximum number of retry attempts if a write failure occurs.

Reference HyperText Transfer Protocol Properties

The following table provides definitions for the properties required for the File Transfer Protocol target type.

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
HTTP URL	The HTTP URL.
Header	The HTTP header field.

Reference MQSeries Properties

The following table provides definitions for the properties required for the MQSeries target type.

Settings tab

Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Queue Manager	Name of the MQSeries queue manager to be used.
Queue Name	Queue on which request documents are received.
Correlation ID	The correlation ID to set in the MQSeries message header.

MQ Client tab

Property	Definition
Host	Name of the MQSeries queue manager to be used.
Port	Queue on which request documents are received.
Channel	The correlation ID to set in the MQSeries message header.

Reference

TCP Properties

The following table provides definitions for the properties required for the TCP target type.

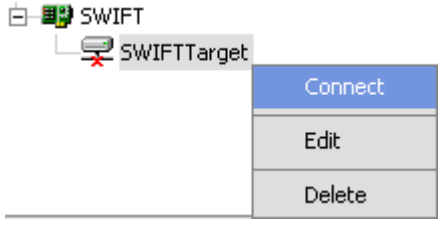
Property	Definition
SWIFT Version	Select the SWIFT version from the drop-down list. The values are: • 1997 • 2001 • 2002 • 2003
Host	Host name or host address.
Port	TCP listening port.
Encoding	Document character set.

Procedure

How to Connect to a Target

To connect to a SWIFT target:

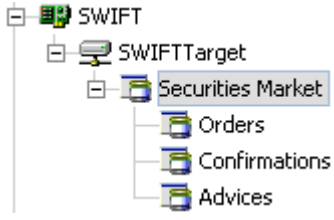
1. In the left pane, expand the *SWIFT* node and select the target to which you want to connect, for example, SWIFTTarget.



2. In the left pane, right-click the target and select *Connect*.

The SWIFTTarget node in the left pane changes to reflect that a connection was made.

3. Expand the target node to reveal the list of SWIFT interfaces.



Managing a Target

Although you can maintain multiple open connections to different application systems, iWay Software recommends that you close connections when they are not in use. After you disconnect, you can modify an existing target.

You can modify the connection parameters when your system properties change. You also can delete a target. The following procedures describe how to disconnect from a target, edit a target, and delete a target.

Procedure How to Disconnect From a Target

To disconnect from a target:

1. Right-click the SWIFT target from which you want to disconnect.
2. Select *Disconnect*.

   Disconnecting from the application system drops the connection, but the node remains. The SWIFTTarget node in the left pane changes to reflect that you disconnected from the target.

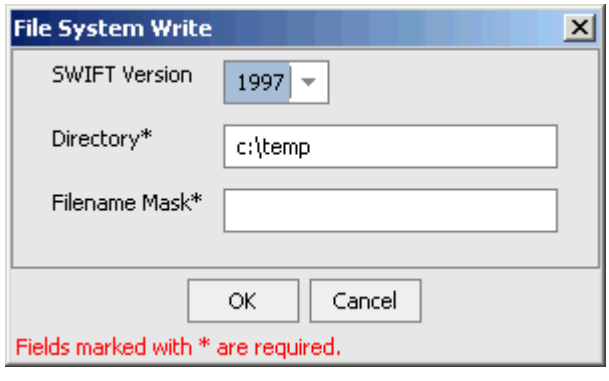
Procedure How to Edit a Target

To edit a target:

1. Ensure that the target you want to edit is disconnected.

2. In the left pane, right-click the target and select *Edit*.

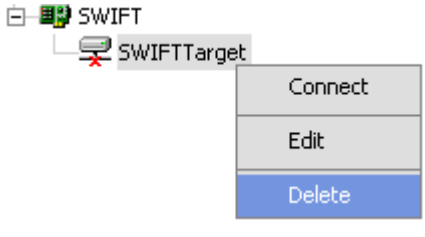
   The following dialog box opens.



3. Change the properties in the dialog box as required and click *OK*.

Procedure How to Delete a Target

To delete a target:



1. In the left pane, right-click the target.
2. Select *Delete*.

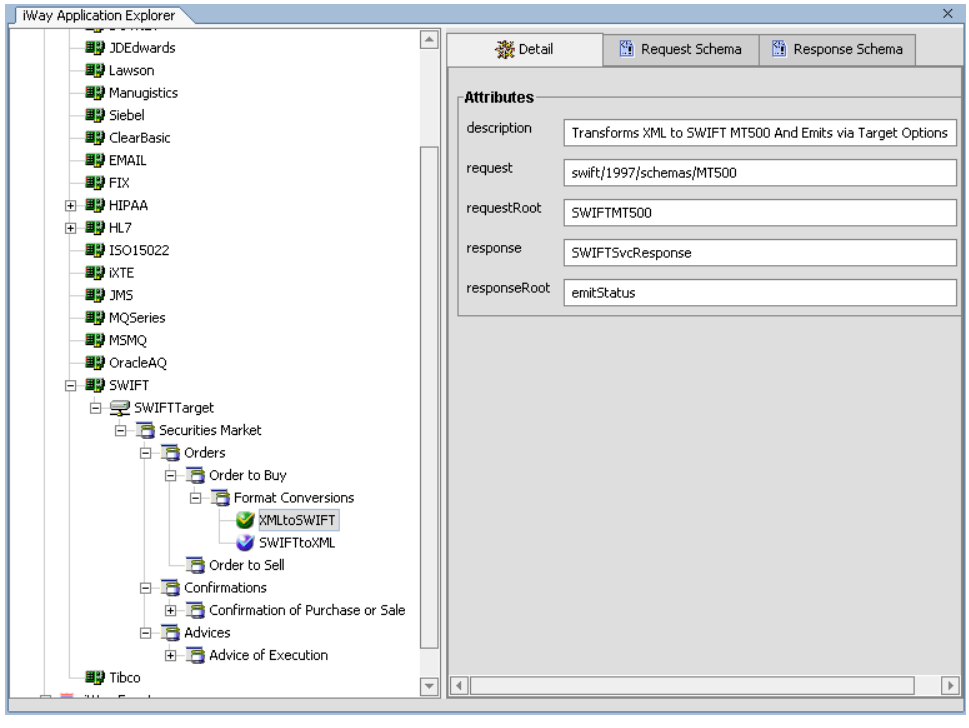
The SWIFTTarget node disappears from the left pane.

Creating an XML Schema

After you create a new configuration and connect to SWIFT, iWay Application Explorer enables you to create a request or response schema.

Procedure How to Create a Request and Response Schema

To create a request and response schema:



1. Expand the SWIFT node and select the node for which you want to create the schema.

   The following XML schemas appear for the interface:

   - Request
   - Response

2. To view the appropriate schema in the right pane, click the *Request Schema* or the *Response Schema* tab.

Reference Schema Location

After you browse the Component Interfaces and make a selection, the request and response XML schemas are automatically created for that Component Interface and stored in the repository you created, for example:

```
drive:\Program Files\iWay55\bea\ibse\wsdl\schemas\service\SWIFT
\SWIFTTarget\SA45280C
```

where:

SWIFTTarget

Is the name of the SWIFT target.

SA45280C

Is a randomly generated folder name indicating where the schemas are stored.

Creating an iWay Business Service

You can create an iWay business service (also known as a Web service) for objects you want to use with your adapter. To generate a business service, you must deploy the iWay Adapter for SWIFT using the Integration Business Services Engine (iBSE). iBSE exposes functionality as Web services and serves as a gateway to heterogeneous back-end applications and databases.

A Web service is a self-contained, modularized function that can be published and accessed across a network using open standards. It is the implementation of an interface by a component and is an executable entity. For the caller or sender, a Web service can be considered as a "black box" that may require input and delivers a result. Web services integrate within an enterprise as well as across enterprises on any communication technology stack, whether asynchronous or synchronous, in any format.

You can make a Web service available to other services within a host server by generating WSDL (Web Services Description Language) from the Web service.

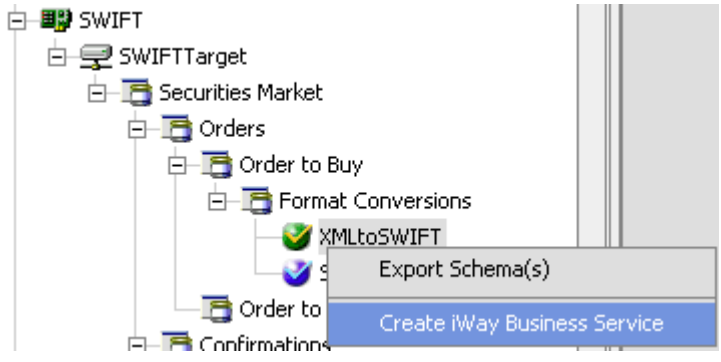
Because Application Explorer runs within BEA WebLogic Workshop, you can easily incorporate iWay Web services into BEA WebLogic Workflows. To enable BEA WebLogic Workshop to use iWay Web services, you export the WSDL to a directory accessible to BEA WebLogic Workshop.

Note: In a J2EE Connector Architecture (JCA) implementation of iWay adapters, Web services are not available. When the adapters are deployed to use the iWay Connector for JCA, the Common Client Interface provides integration services using the iWay adapters. For more information, see the *iWay Installation and Configuration for BEA WebLogic* manual and the i*Way Connector for JCA for BEA WebLogic Server User's Guide*.

Procedure How to Create an iWay Business Service

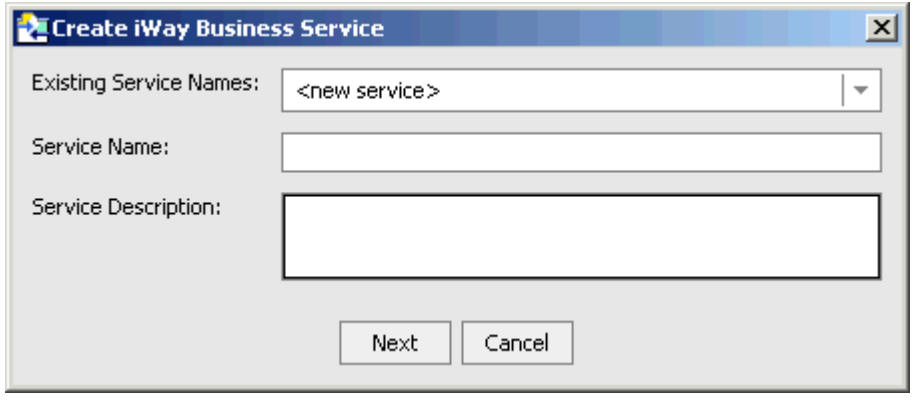
To create an iWay Business service:

1. Expand the SWIFT node and select the interface for which you want to create a business service.



2. Right-click and select *Create iWay Business Service*.

   The Create iWay Business Service dialog box opens.



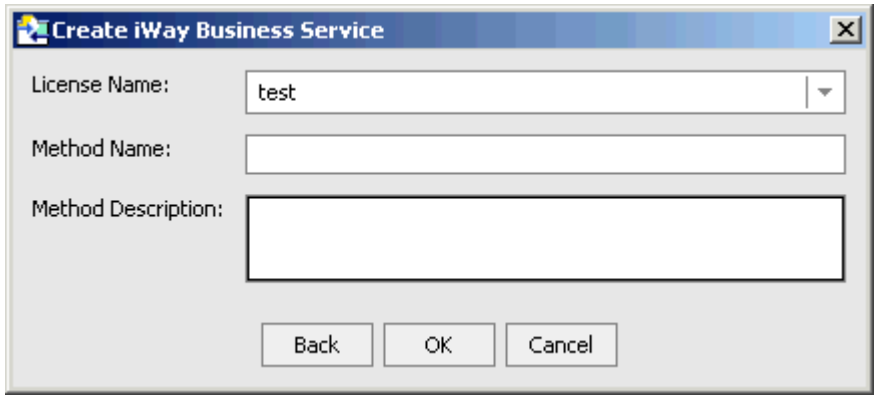
   a. From the Existing Service Names drop-down list, select whether you want to create a new service name or use an existing service name.

   b. In the Service Name field, type a name for the business service, for example, SWIFTService.

   c. In the Service Description field, type a brief description of the business service.

3. Click *Next*.

The Create iWay Business Service dialog box displays additional fields.



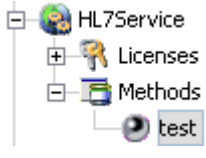
a. From the License Name drop-down list, select a license.

b. In the Method Name field, type a name for the method.

c. In the Method Description field, type a brief description for the method.

4. Click *OK*.

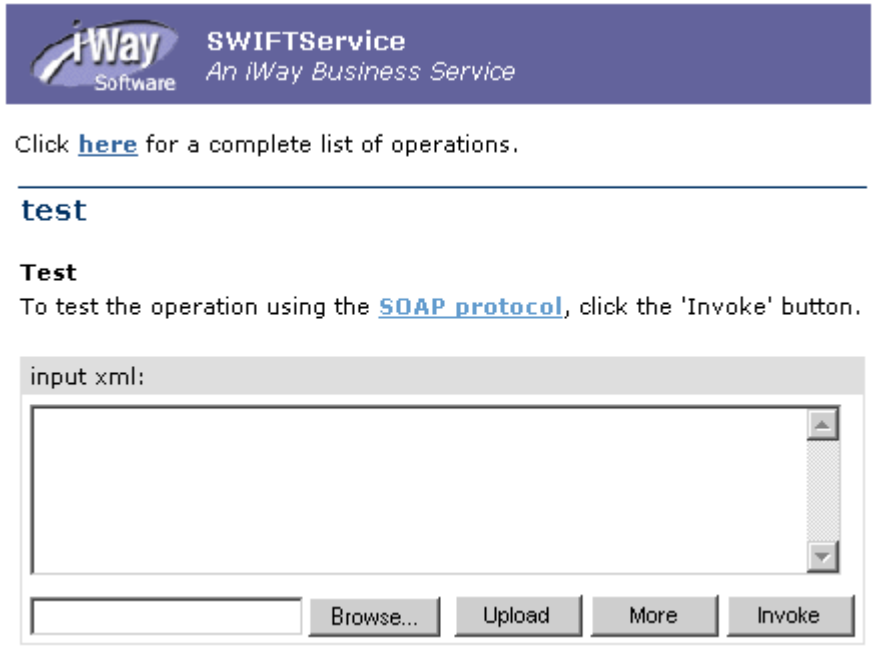
The business service and method appear below the iWay Business Services node.



In the left pane, all the available business services that were created appear.

5. Click the node for which you created the business service in the right pane.

The test pane opens in a new browser window.



6. To invoke the service, enter a sample XML document in the input xml field.
7. Click *Invoke*.

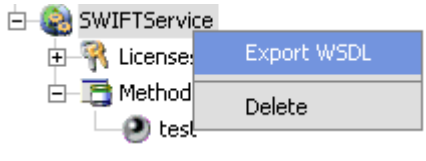
   The result appears in the right pane.

Exporting iWay WSDL for Use in BEA WebLogic Workshop Workflows

Because iWay Application Explorer runs within BEA WebLogic Workshop, you can easily incorporate iWay Web services into BEA WebLogic Workflows. To enable BEA WebLogic Workshop to use iWay Web services, you simply export the WSDL to a directory accessible to BEA WebLogic Workshop.

Procedure How to Export iWay WSDL for Use in BEA WebLogic Workshop Workflows

To export WSDL to a directory accessible to BEA WebLogic Workshop:

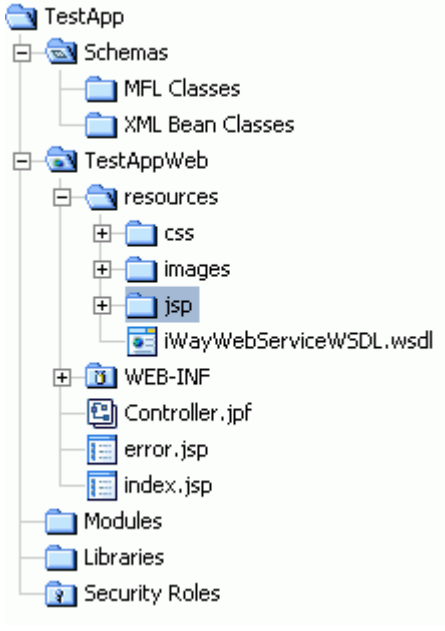


1. After you create a Web service, right-click the Web service name and select *Export WSDL*.

   The Save dialog box appears.

2. Save the WSDL to a directory accessible to BEA WebLogic Workshop, for example, the \resources directory in your BEA WebLogic Workshop Web application directory structure.

   The WSDL file appears under the resources folder of your Web application:



Adding a Control for an iWay Resource in BEA WebLogic Workshop

Java controls provide a convenient way to incorporate access to iWay resources. You can add controls in BEA WebLogic Workshop to use Web services created by the Java Swing version of iWay Application Explorer, or you can add controls that enable you to take advantage of the JCA resources of Application Explorer.

Adding a Web Service Control to a BEA WebLogic Workshop Application

After you create an iWay Web service using Application Explorer and export the WSDL file, you can create a control for the Web service.

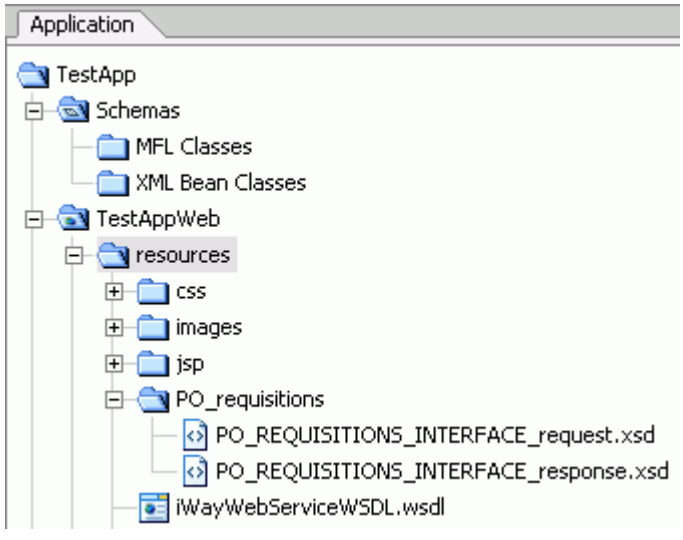
For more information on exporting a WSDL file, see *How to Export iWay WSDL for Use in BEA WebLogic Workshop Workflows* on page A-18.

Procedure How to Add a Web Service Control

To add a Web service control:

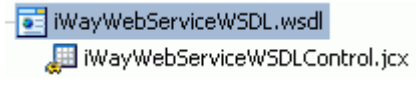
1. After exporting the WSDL file from Application Explorer, locate the file in the Application tab of your BEA WebLogic Workshop application.

   For example, a WSDL file saved to the \resources directory in your BEA WebLogic Workshop Web application directory structure appears as follows.



2. Right-click the *WSDL* file and select *Generate Service Control*.

   The control for the WSDL appears below the WSDL file in the resources tree.



Adding an iWay Extensible CCI Control to a BEA WebLogic Workshop Application

An iWay control enables access to resources provided by Application Explorer when it is used in conjunction with a JCA deployment. You must add an iWay control before using it in a BEA WebLogic Workshop application workflow.

The following topic describes the enhanced CCI control, which is extensible and provides JCX with typed inputs and outputs for JCA in BEA WebLogic Workshop.

Overview

The extensible iWay CCI control provides:

- **Method and tag validation**. BEA WebLogic Workshop provides warnings regarding invalid methods and tags.
- **Improved error handling**.

You can define new methods that rely on the generic *service* and *authService* methods. For example, you can define a JCX with a new method without writing casting code or explicit transformations such as the following:

```
public ResponseDataType MethodName(RequestDataType VariableName) throws
Exception;
```

where:

ResponseDataType

Is the XML Bean Class value that is generated from the response schema.

MethodName

Is the method name used by the extensible CCI control.

RequestDataType

Is the XML Bean Class value that is generated from the request schema.

VariableName

Is the request variable that stores the request document, which is used as input by the extensible CCI control.

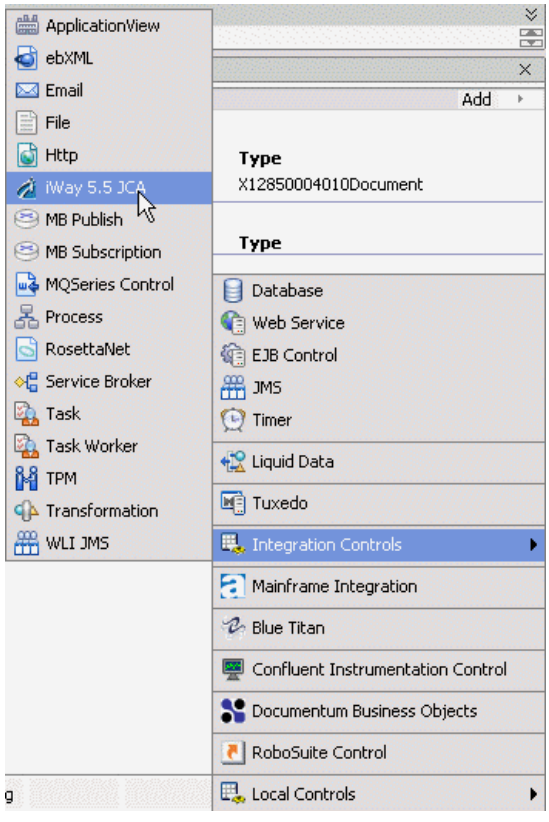
In addition, the extensible CCI control now generates a JCX file to which you can add your own methods. For more information, see *Defining a Control Using the Extensible CCI Control* on page A-21.

You can also use dynamic class casting to specify schema-based input or output XmlObjects to be casted into a pure XmlObject as a service method, which is expected by the CCI control. For more information, see *Using Dynamic Class Casting* on page A-28.

Example Defining a Control Using the Extensible CCI Control

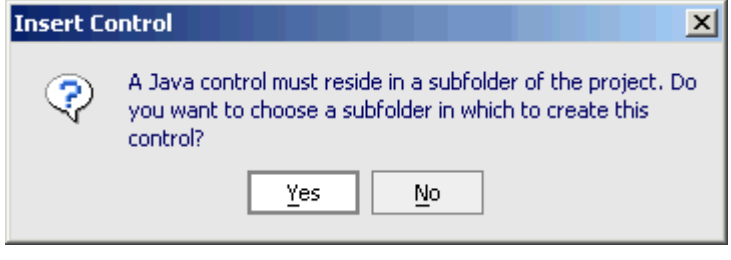
The following sample JCX demonstrates how to define a control for SWIFT using the extensible CCI control in BEA WebLogic Workshop.

1. Start BEA WebLogic Workshop and create a new project.



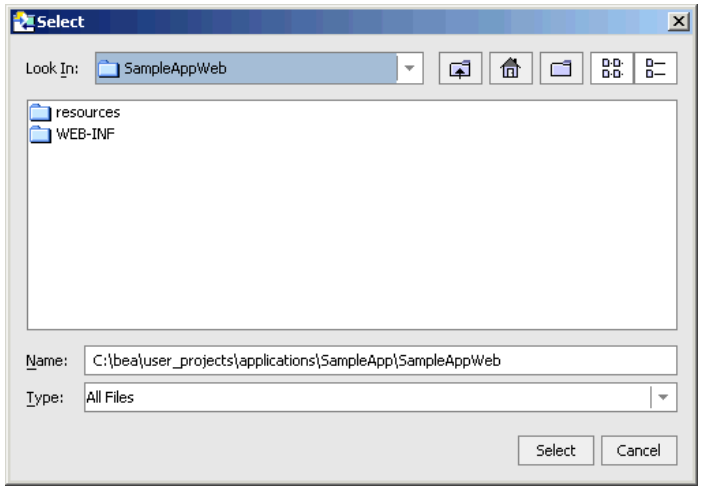
2. Click *Add* from the Controls section in the Data Palette tab, select *Integration Controls*, and click *iWay 5.5 JCA*.

The Insert Control message box opens.



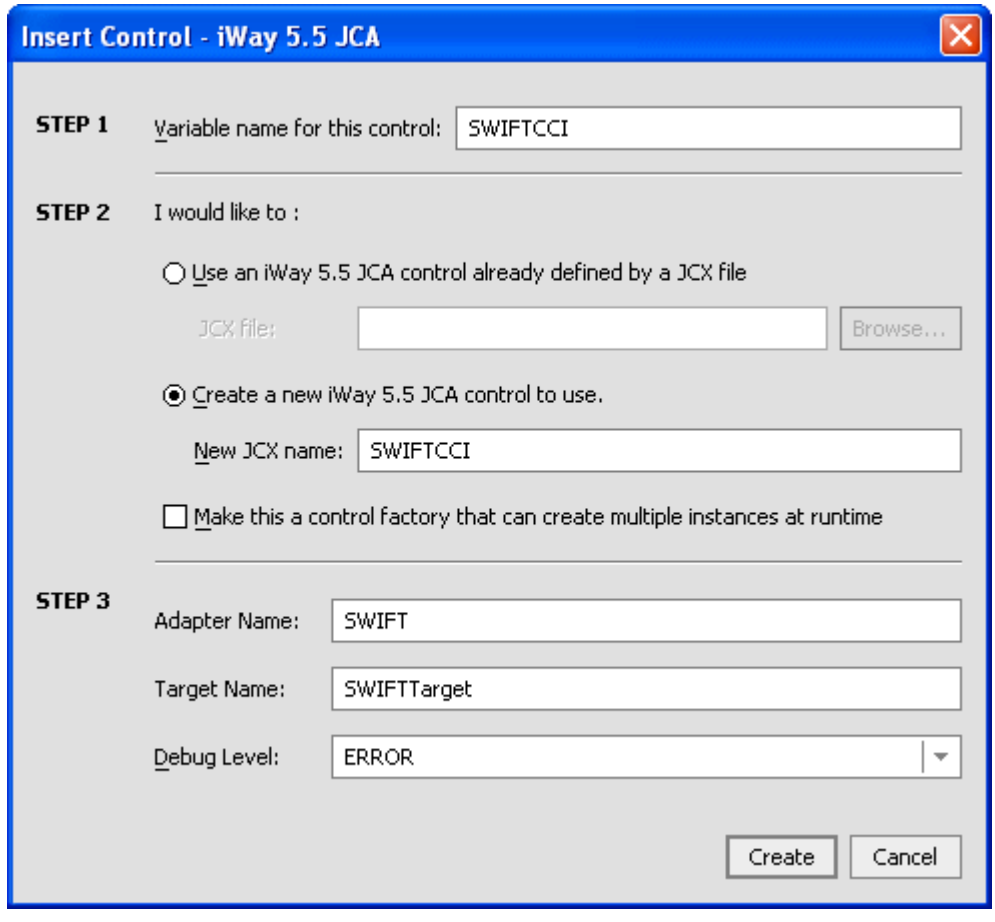
3. Click *Yes*.

The Select dialog box opens.



4. Choose a subfolder for the CCI control and click *Select*.

The Insert Control - iWay 5.5 JCA dialog box opens.



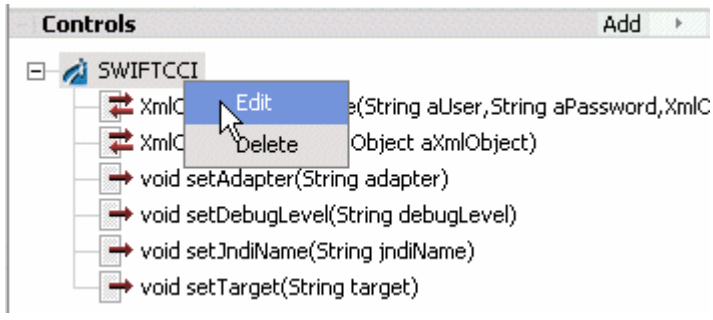
a. Provide a variable name for the control.

b. Click *Create a new iWay 5.5 JCA control to use* and provide a new JCX name.

c. Enter the adapter name, target name, and select a debug level from the drop-down list.

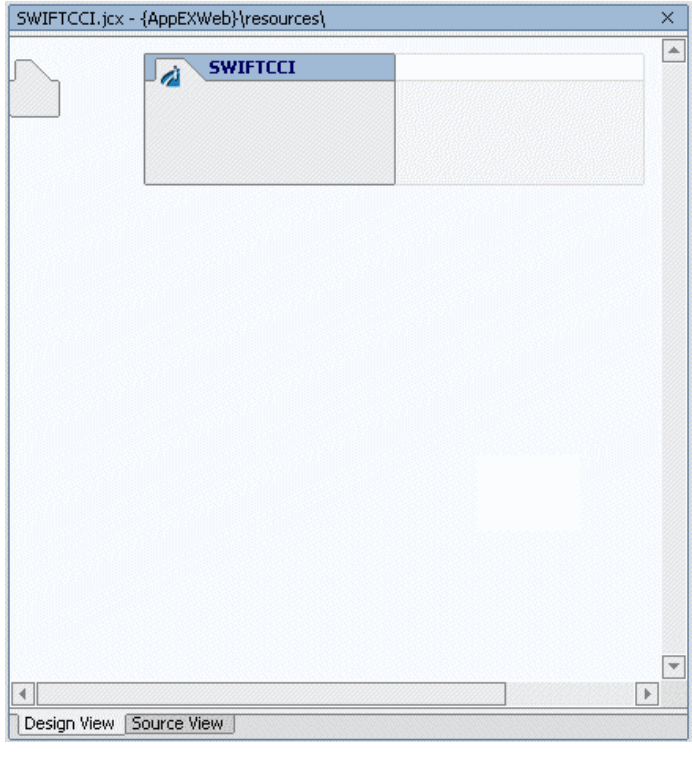
5. Click *Create*.

A new JCX file is created.



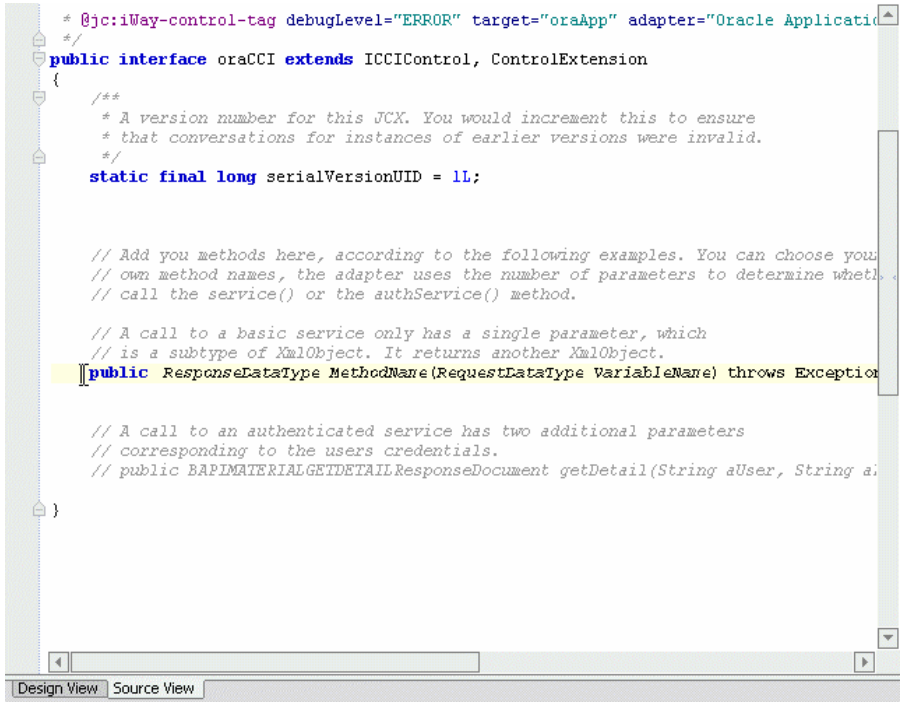
6. Right-click the control, for example, SWIFTCCI, and select *Edit*.

The Design View for the control opens.



7. Click the *Source View* tab.

The Source View for the control opens.



Perform the following steps:

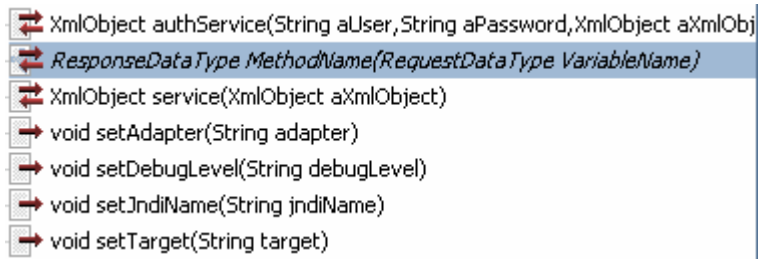
a. Uncomment the public class definition.

b. Change the existing response data type to match your response data type that is generated from your SWIFT response schema.

c. Change the existing method name to match your method.

d. Change the existing request data type to match your request data type that is generated from your SWIFT request schema.

The following control is now available in BEA WebLogic Workshop and can be added to a workflow:



Note: You can view available data types under the *XML Bean Classes* folder in the *Application* tab, which are added once you import your XML request or response schemas from Application Explorer.

These data types are case sensitive and must be entered exactly as shown.

Using the Extensible CCI Control

The extensible CCI control functions much like a database control since it generates JCX files to which you can add your own methods.

Your own methods can use the correct input and output types rather than the generic XmlObject types that the JCA control uses. Since the control is just a proxy that uses a reflection to call the relevant method, it handles the casting for you. You are no longer required to write custom code that does the cast or transformations that are cast between an XmlObject.

For example, instead of the generic XmlObject:

```
XmlObject service(XmlObject input) throws java.lang.Exception;
```

you call:

```
public ResponseDataType MethodName(RequestDataType VariableName) throws
Exception;
```

where:

ResponseDataType

Is the XML Bean Class value that is generated from the response schema.

MethodName

Is the method name used by the extensible CCI control.

RequestDataType

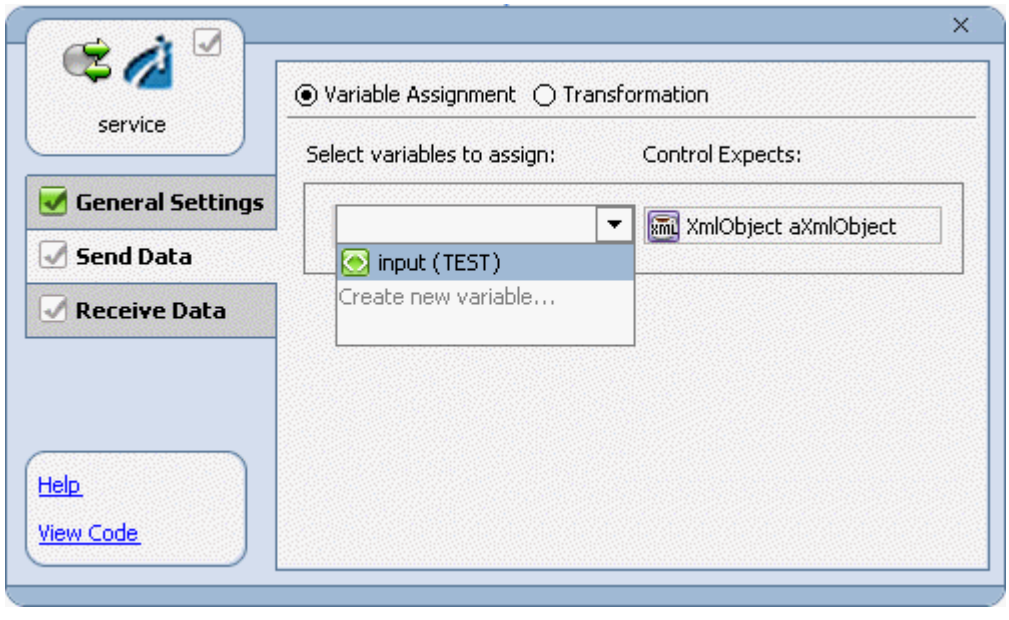
Is the XML Bean Class value that is generated from the request schema.

VariableName

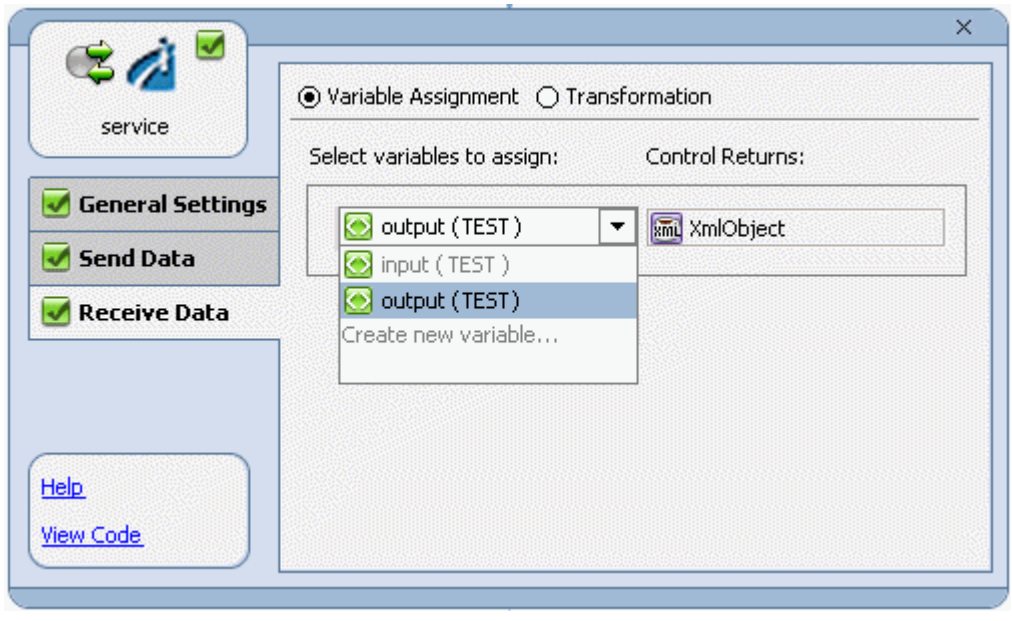
Is the request variable that stores the request document, which is used as input by the extensible CCI control.

Example Using Dynamic Class Casting

The following example uses dynamic class casting to specify a schema-based input XmlObject to be casted into a pure XmlObject as a service method, which is expected by the CCI control.



The following example uses dynamic class casting where the CCI control returns a pure XmlObject, which is casted dynamically into a schema-based output XmlObject.



APPENDIX B

Using Application Explorer in BEA WebLogic Workshop for Event Handling

Topics:

- Starting Application Explorer in BEA WebLogic Workshop
- Understanding iWay Event Functionality
- Creating an Event Port
- Modifying an Event Port
- Creating a Channel
- Modifying a Channel
- Deploying iWay Components in a Clustered BEA WebLogic Environment

This section describes how to use iWay Java Swing Application Explorer running in BEA WebLogic Workshop to create events for SWIFT. In addition, this section provides information on deploying components in a clustered BEA WebLogic environment.

Starting Application Explorer in BEA WebLogic Workshop

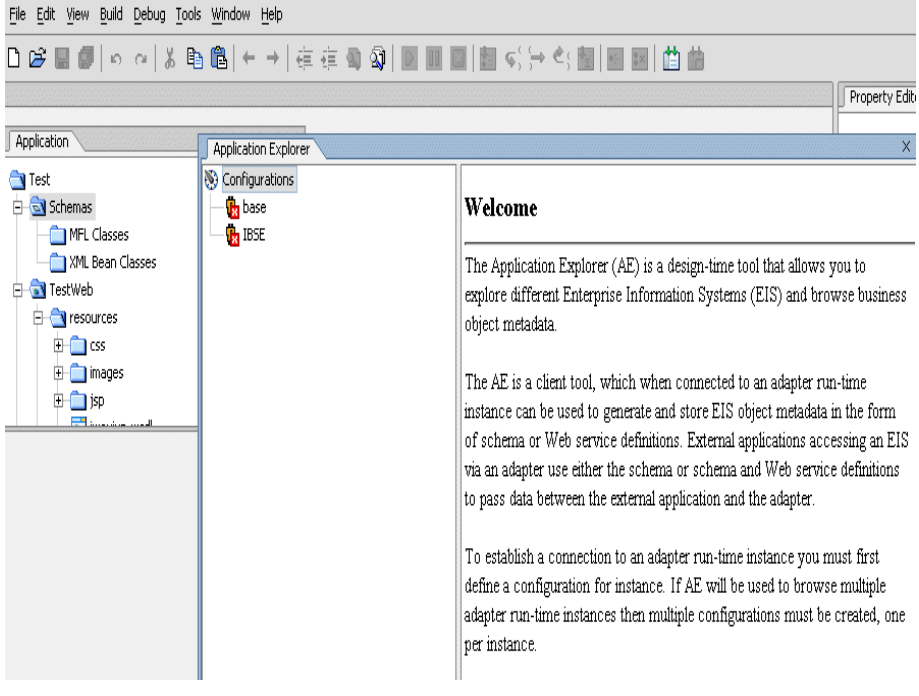
The server must be started where iWay Application Explorer is running. Before you can use Application Explorer, you must start BEA WebLogic server.

Procedure How to Start Application Explorer in BEA WebLogic Workshop

To start Application Explorer running in BEA WebLogic Workshop:

1. Before starting Application Explorer, ensure that BEA WebLogic Server is running.
2. Start BEA WebLogic Workshop.
3. From the BEA WebLogic Workshop View menu, select *Windows* and then, *iWay Application Explorer.*

   Application Explorer opens in BEA WebLogic Workshop.



You can resize and drag-and-drop the Application Explorer window within BEA WebLogic Workshop. For example, you can drag it to the upper part of BEA WebLogic Workshop.

Understanding iWay Event Functionality

Events are generated as a result of activity in an application system. You can use events to trigger an action in your application. For example, SWIFT may generate an event when customer information is updated. If your application must perform in response to activity, your application is a consumer of this event.

After you create a connection to your application system, you can add events using Application Explorer. To define an iWay event, you must create a port and a channel.

- Port

  A port associates a particular business object exposed by the adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption. For more information, see *Creating an Event Port*.

- Channel

  A channel represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by the adapter. For more information, see *Creating a Channel* on page B-19.

Creating an Event Port

The following procedures describe how to create an event port using iWay Application Explorer. The following port dispositions are available when using iBSE:

- File
- iBSE
- MSMQ
- JMSQ
- SOAP
- HTTP
- MQ Series
- Mail

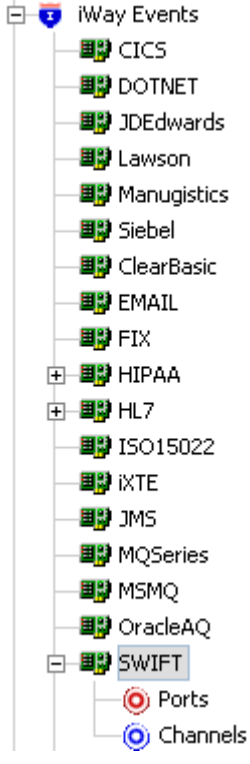
Note: The MAIL disposition option will be supported in a future release.

With a JCA implementation, the following port dispositions are available:

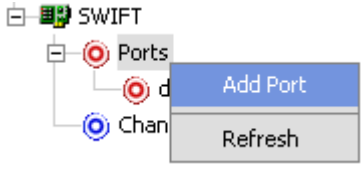
- File
- JMSQ
- MQ Series
- HTTP

Procedure How to Create an Event Port for File

To create an event port for File:

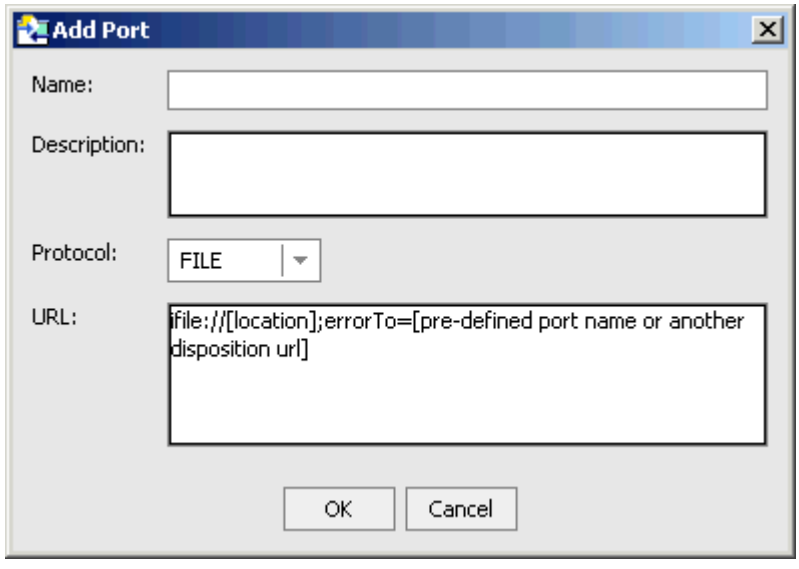


1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.



2. Right-click and select *Add Port*.

The Add Port dialog box opens.



a. In the Name field, type a name for the event port, for example, SWIFTFile.

b. In the Description field, type a brief description.

c. From the Protocol drop-down list, select *FILE*.

d. In the URL field, type a destination file to which the event data is written, using the following format:

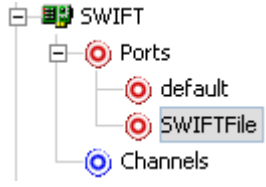
```
file://location[;errorTo=errorDest]
```

The following table describes the URL parameters.

Parameter	Description
location	The full directory path and file name to which the data is written.
errorTo	Location where error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

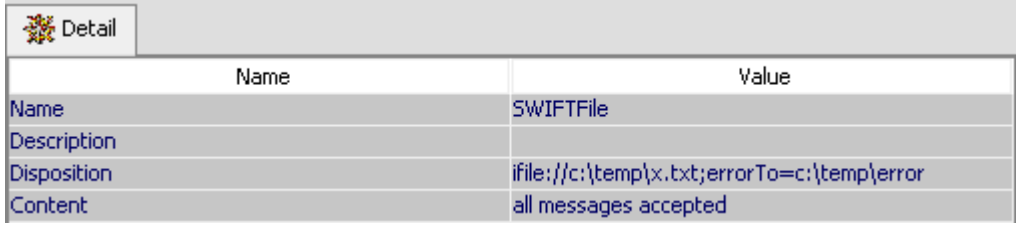
3. Click *OK*.

In the left pane, the event port appears below the Ports node.



4. To review the port settings, select the port name.

   In the right pane, a table appears that summarizes the information associated with the event port you created.



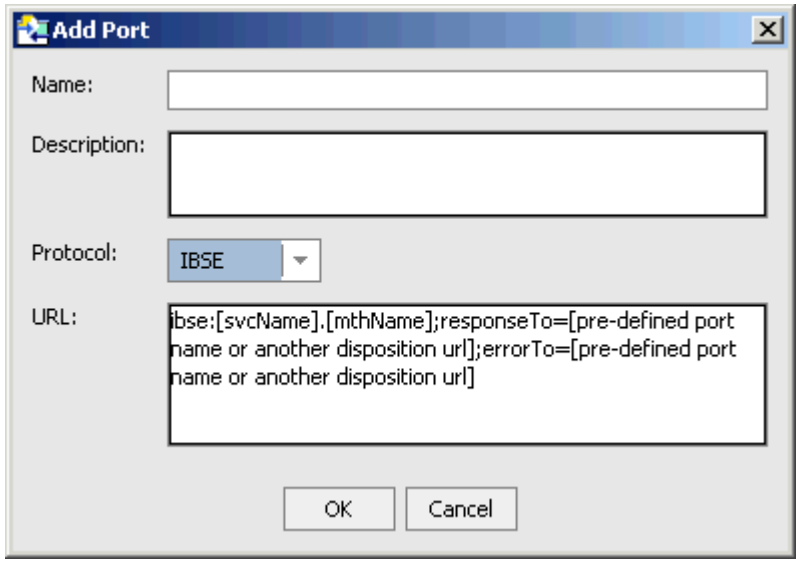
   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

Procedure How to Create an Event Port for iBSE

To create an event port for iBSE:

1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.
2. Right-click and select *Add Port*.

The Add Port dialog box opens.



a. In the Name field, type a name for the event port, for example, SWIFTiBSE.

b. In the Description field, type a brief description.

c. From the Protocol drop-down list, select *IBSE*.

d. In the URL field, enter an iBSE destination using the following format:

```
ibse:/svcName.methName[;responseTo=respDest][;errorTo=errorDest]
```

The following table describes the disposition parameters.

Parameter	Description
svcName	Name of the service created with iBSE.
methName	Name of the method created for the Web service.
respDest	Location where responses to the Web service are posted. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.
errorDest	Location where error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

3. Click *OK*.

   In the left pane, the event port appears below the Ports node.

4. To review the port settings, select the port name.

   In the right pane, a table appears that summarizes the information associated with the event port you created.

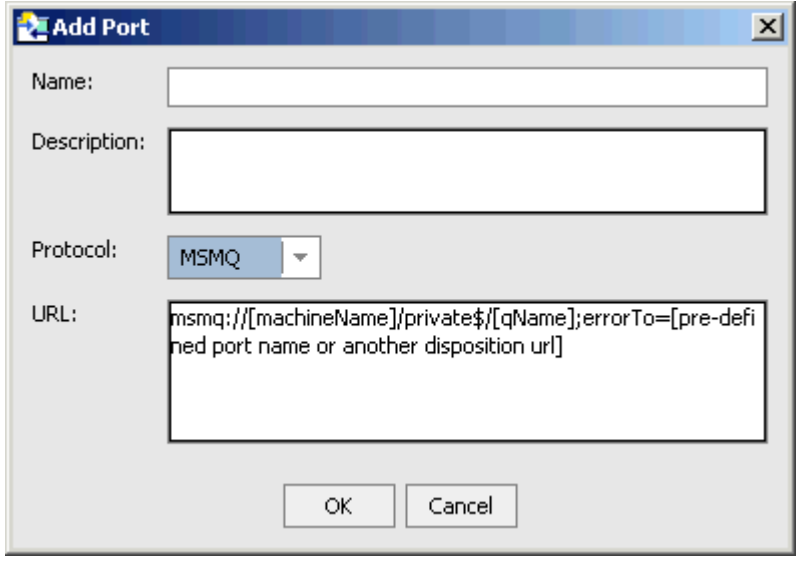
   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

Procedure How to Create an Event Port for MSMQ

To create an event port for a Microsoft Message Queuing (MSMQ) queue:

1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.
2. Right-click and select *Add Port*.

   The Add Port dialog box opens.



   a. In the Name field, type a name for the connection, for example, SWIFTMSMQ.
   b. In the Description field, type a description for the target name you just created.
   c. From the Protocol drop-down list, select *MSMQ*.
   d. In the URL field, enter an MSMQ destination in the following format:

      `msmq:/host/queueType/queueName[;errorTo=errorDest]`

The following table defines the disposition parameters.

Parameter	Description
host	Name of the host on which the Microsoft Queuing system runs.
queueType	The type of queue. For private queues, enter *Private$*. Private queues are queues that are not published in Active Directory. They appear only on the local computer that contains them. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue.
queueName	Name of the queue where messages are placed.
errorDest	Location where error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

3. Click *OK*.

   In the left pane, the event port appears below the Ports node.

4. To review the port settings, select the port name.

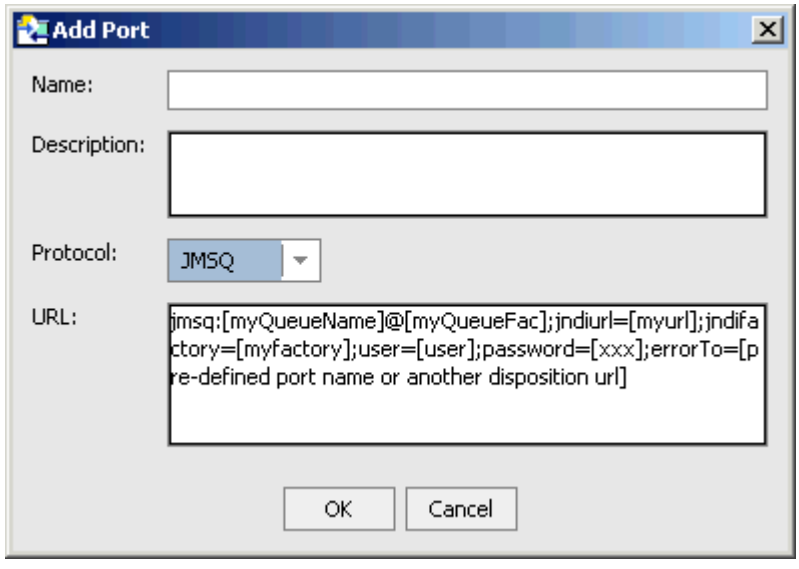
   In the right pane, a table appears that summarizes the information associated with the port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

Procedure How to Create a Port for JMS

To create a port for a JMS queue:

1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.
2. Right-click and select *Add Port*.

The Add Port dialog box opens.



a. In the Name field, type a name for the event port, for example, SWIFTJMSQ.

b. In the Description field, type a brief description.

c. From the Protocol drop-down list, select *JMSQ*.

d. In the URL field, enter a JMSQ destination using the following format:

```
jmsq:queue@conn_factory;jndiurl=jndi_url;jndifactory=jndi_factory;
user=userID;password=pass[;errorTo=errorDest]
```

The following table describes the URL parameters.

Parameter	Description
queue	Name of a queue to which events are emitted.
conn_factory	The connection factory, a resource that contains information about the JMS Server. The WebLogic connection factory is: `javax.jms.QueueConnectionFactory`

Parameter	Description
jndi_url	The URL of the application server. For BEA WebLogic Server, the URL is `t3://host:port` where: *host* Is the machine name where BEA WebLogic Server resides. *port* Is the port on which BEA WebLogic Server is listening. The default port if not changed at installation, is 7001.
jndi_factory	Is JNDI context.INITIAL_CONTEXT_FACTORY and is provided by the JNDI service provider. For BEA WebLogic Server, the WebLogic factory is weblogic.jndi.WLInitialContextFactory.
userID	User ID associated with this queue.
pass	Password associated with this user ID.
errorDest	Location where error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

3. Click *OK*.

   The event port appears below the Ports node in the left pane.

4. To review the port settings, select the port name.

   In the right pane, a table appears that summarizes the information associated with the event port you created.

   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

Procedure How to Create a Port for the SOAP Disposition

This topic describes how to configure the SOAP disposition for synchronous event processing.

The SOAP disposition allows an event response to launch a Web service specified by a WSDL file. A soapaction is optional, the default is "".

To create a port for a SOAP disposition using Application Explorer:

1. Click the *iWay Events* tab.

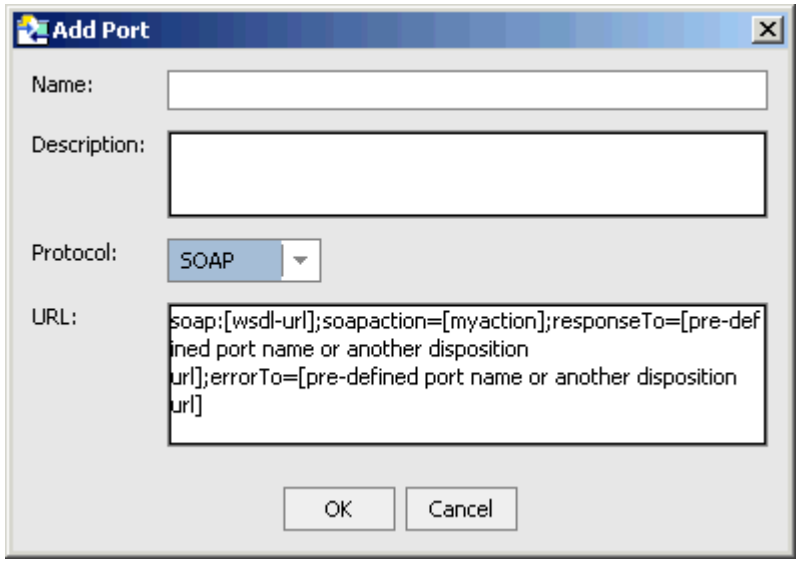
   The iWay Event Adapters window opens.

2. In the left pane, expand the SWIFT adapter node.

3. Select the *ports* node.

4. Move the pointer over *Operations* and select *Add a new port*.

   The Add Port dialog box opens.



   a. Type a name for the event port and provide a brief description.

   b. From the Disposition Protocol drop-down list, select *SOAP*.

   c. In the Disposition field, enter a SOAP destination using the following format:

```
soap:wsdl-url;soapaction=action[;responseTo=respDest]
   [;errorTo=errorDest]
```

The following table lists and describes the disposition parameters for SOAP.

Parameter	Description
wsdl-url	The URL to the WSDL file that is required to create the SOAP message. For example: `http://localhost:7001/ibse/IBSEServlet/test/sw2xml2003MQ.ibs?wsdl` This value can be found by navigating to the iWay Business Services tab and opening the *Service Description* link in a new window. The WSDL URL appears in the Address field. You can also open the WSDL file in a third party XML editor (for example, XMLSPY) and view the SOAP request settings to find this value.

action	The method that will be called by the disposition. For example:
	`SWIFTMT200.mt200Request@test@@`
	where:
	`SWIFTMT200`
	Is the name of the Web service you created using Application Explorer.
	`mt200`
	Is the method being used.
	`test`
	Is the license that is being used by the Web service.
	This value can be found by navigating to the iWay Business Services tab and opening the *Service Description* link in a new window. Perform a search for *soapAction*.
	You can also open the WSDL file in a third party XML editor (for example, XMLSPY) and view the SOAP request settings to find this value.

respDest	The location to which responses are posted. A predefined port name or another full URL. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

Parameter	Description
errorDest	The location to which error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

5. Click *OK*.

   The port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the port you created.

   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

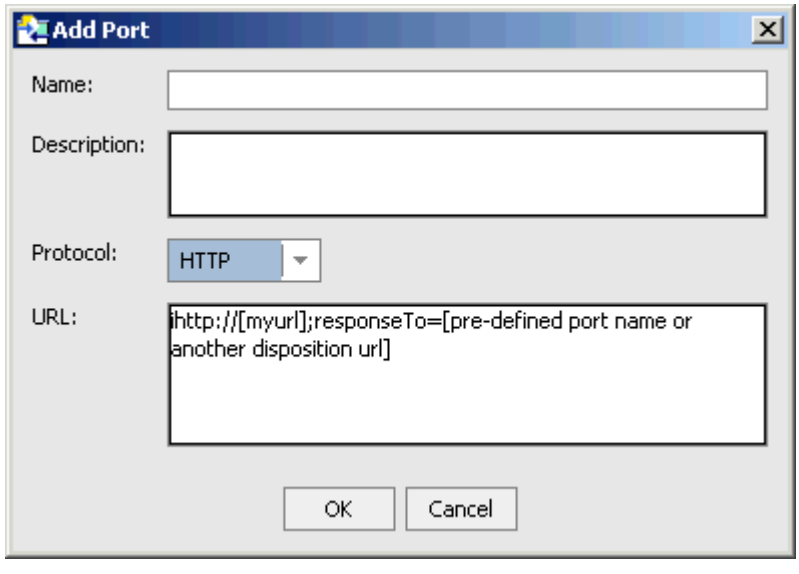
Procedure How to Create an Event Port for HTTP

The HTTP disposition uses an HTTP URL to specify an HTTP end point to which the event document is posted.

To create an event port for HTTP disposition:

1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.
2. Right-click and select *Add Port*.

   The Add Port dialog box opens.



a. In the Name field, type a name for the event port, for example, SWIFTHTTP.

b. In the Description field, type a brief description.

c. From the Protocol drop-down list, select *HTTP*.

d. In the URL field, enter an HTTP destination using the following format:

```
ihttp://url;responseTo=respDest
```

The following table describes the URL parameters.

Parameter	Description
url	The URL target for the post operation.
respDest	Location where responses are posted. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.
host	Name of the host on which the Web server resides.
port	Port number on which the Web server is listening.

3. Click *OK*.

The event port appears below the Ports node in the left pane.

4. To review the port settings, select the port name.

In the right pane, a table appears that summarizes the information associated with the event port you created.

You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

Procedure How to Create an Event Port for MQ Series

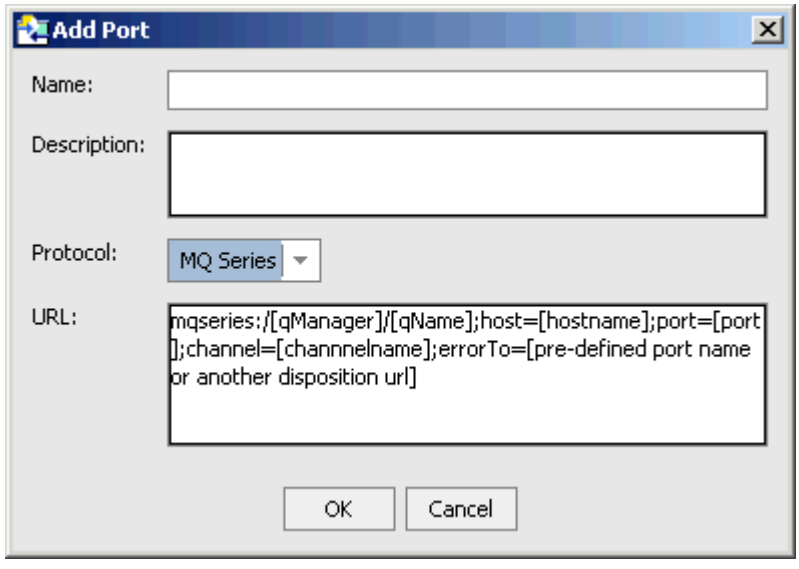
The MQ Series disposition allows an event to be enqueued to an MQ Series queue. Both queue manager and queue name may be specified.

To create a port for an MQ Series queue:

1. In the left pane of Application Explorer, expand the SWIFT node under iWay Events, and then select *Ports*.

2. Right-click and select *Add Port*.

The Add Port dialog box opens.



a. In the Name field, type a name for the event port, for example, SWIFTMQSeries.

b. In the Description field, type a brief description.

c. From the Protocol drop-down list, select *MQ Series*.

d. In the URL field, enter an MQ Series destination using the following format:

```
mqseries:/qManager/qName;host=hostName;port=portNum;
channel=chanName[;errorTo=errorDest]
```

The following table describes the URL parameters.

Parameter	Description
qManager	Name of queue manager to which the server must connect.
qName	Name of the queue where messages are placed.
hostName	Name of the host on which MQ Series resides (MQ client only).
portNum	Port number for connecting to an MQ Server queue manager (MQ client only).
chanName	Case-sensitive name of the channel that connects with the remote MQ Server queue manager (MQ client only). The default MQ Series channel name is SYSTEM.DEF.SVRCONN.

Parameter	Description
errorDest	Location where error logs are sent. Optional. A predefined port name or another disposition URL. The URL must be complete, including the protocol.

3. Click *OK*.

   The event port appears below the Ports node in the left pane.

4. To review the port settings, select the port name.

   In the right pane, a table appears that summarizes the information associated with the event port you created.

   You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

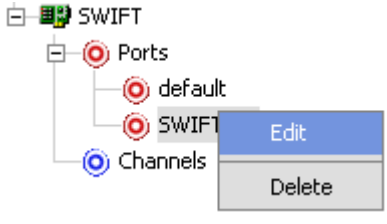
Modifying an Event Port

The following procedures describe how to edit and delete an event port using iWay Application Explorer. To review the port settings, select the port name. In the right pane, a table appears that summarizes the information associated with the event port you created.

Procedure How to Edit an Event Port

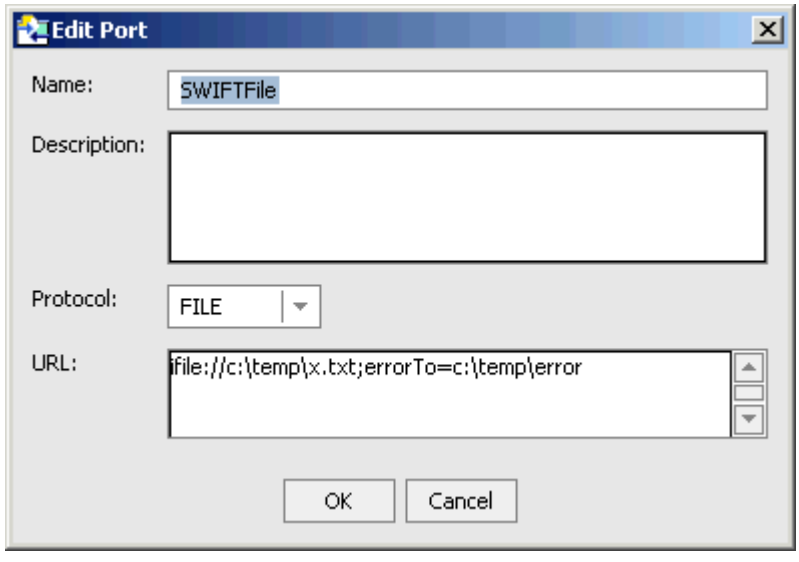
To edit an event port:

1. To view the available ports, click the *Ports* node in the left pane.



2. Right-click the port you want to edit, and select *Edit*.

The Edit Port dialog box opens.

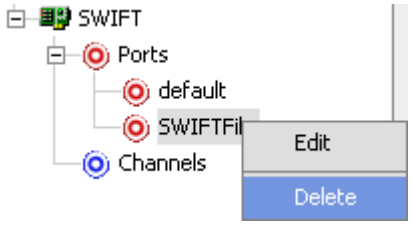


3. Make the required changes and click *OK*.

Procedure How to Delete an Event Port

To delete an existing event port:

1. To view the available ports, click the *Ports* node in the left pane.



2. Right-click the port you want to remove, and select *Delete*.

   The event port node disappears from the ports list in the left pane.

Creating a Channel

The following procedure describes how to create a channel for a SWIFT event. All defined event ports must be associated with a channel.

You can create the following types of channels using Application Explorer:

- File System Listener (File)
- Hypertext Transfer Protocol (HTTP)
- TCP Listener (TCP)
- IBM MQSeries (MQ)
- File Transfer Protocol (FTP)

Procedure How to Create a Channel for a File System Listener

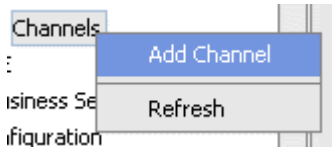
To create a channel for a File System Listener (FILE):

1. In the left pane, below the configuration you created, expand the *iWay Events* node.

   The list of adapters appears.

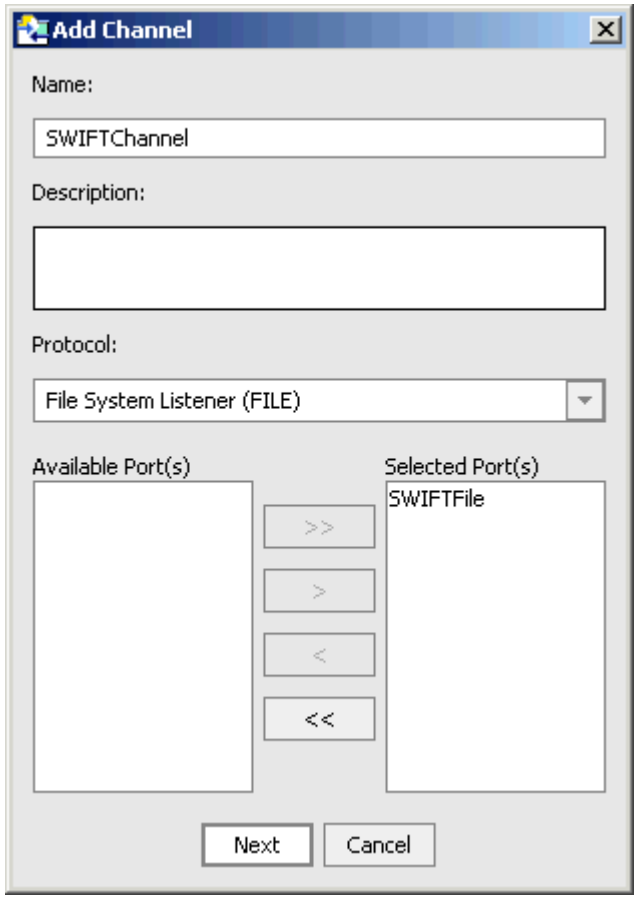
2. Click the adapter node, for example, SWIFT.

   The node expands and displays the Ports and Channels nodes.



3. Right-click the *Channels* node and select *Add Channel*.

The Add Channel dialog box opens.



a. In the Name field, type a name for the channel, for example, SWIFTChannel.

b. In the Description field, type a brief description.

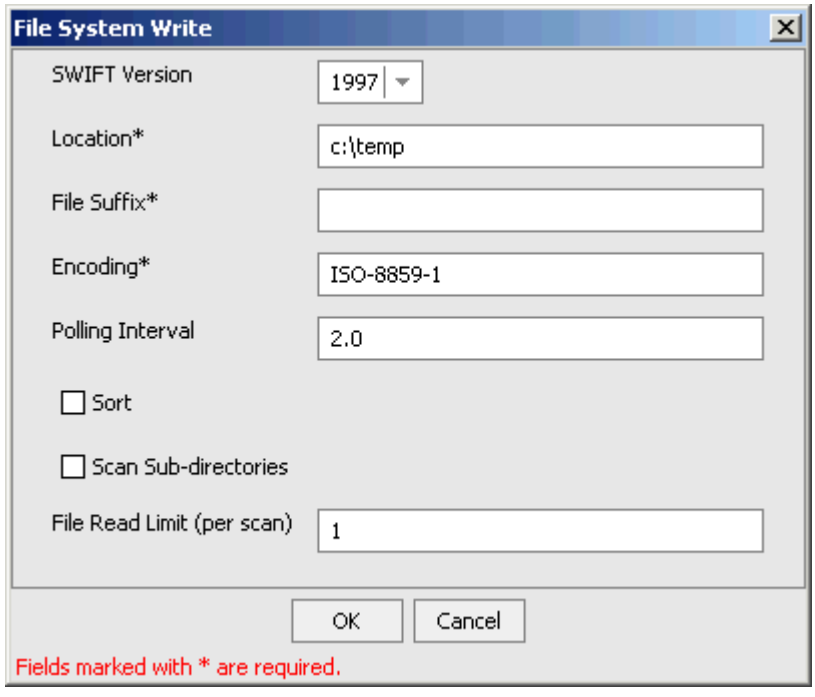
c. From the Protocol drop-down list, select a type of listener:

- File System Listener (FILE)
- HyperText Transfer Protocol
- TCP Listener (TCP)
- IBM MQ Series (MQ)
- File Transfer Protocol (FTP)

d. To associate one or more available ports with this channel, select the port in the Available box and click the double right arrow (>>) button to move it to the Selected box.

4. Click *Next*.

A dialog box opens that is specific to the protocol you selected.



5. Enter values for the parameters that are listed.

For information on the parameters for a File System Listener (FILE) listener, see *File System Listener (FILE) listener Configuration Parameters* on page B-23.

For information on the parameters for a HyperText Transfer Protocol listener, see *HyperText Transfer Protocol Listener Configuration Parameters* on page B-24.

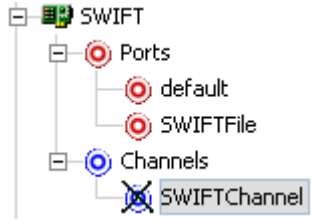
For information on the parameters for a TCP Listener, see *TCP Listener Configuration Parameters* on page B-25.

For information on the parameters for an IBM MQ Series (MQ) listener, see *IBM MQ Series (MQ) Listener Configuration Parameters* on page B-25.

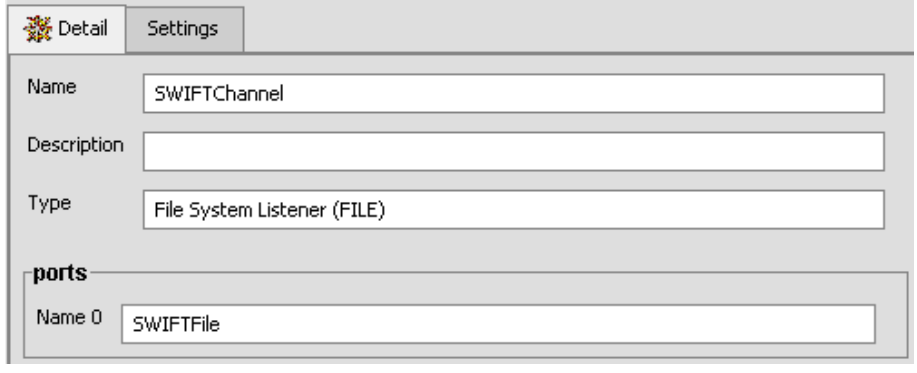
For information on the parameters for a File Transfer Protocol (FTP) listener, see *File Transfer Protocol (FTP) Listener Configuration Parameters* on page B-26.

6. Click *OK*.

The channel appears below the Channels node in the left pane.

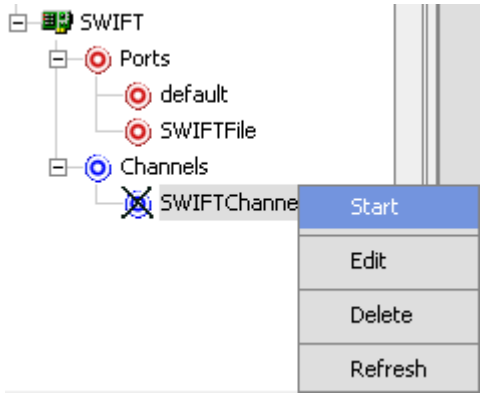


When you select the event port, the channel information appears in the right pane.



A Ports area appears on the Details tab that displays the name of the event port you assigned to this channel.

You are ready to start your channel to listen for events.



7. To activate your event configuration, right-click the channel node, for example, SWIFTChannel.

a. Select *Start*.

b. To stop the channel at any time, right-click the channel and select *Stop*.

Reference File System Listener (FILE) listener Configuration Parameters

On the Settings tab:

Parameter	Description
Location	The directory where messages are received. DOS-style file patterns are valid for this parameter. You can specify a file pattern as well as a directory. For example, c:\xyz\ab*cd (without a file suffix) takes the file suffix from that parameter. If you use a pattern, files are selected based on the suffix and then the pattern. AB?CD selects ABxCD. AB*CD selects ABxxxCD.
File Suffix	File extension for the file event. This limits input files to those with the specified extensions. The "." is not required. The minus sign ("-") indicated that there is no extension. If the file extension is zip, the unzipped files must conform to the event schema, or they will fail. This function also works with transform configured.
Encoding	The host on which the MQ Server is located (for the MQ Client only).
Polling Interval	This is a time, expressed as xxH:xxM:xxS For example 1 hour, 2 minutes, and 3 seconds is: 1H:2M:3S The maximum interval between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. The side-effect of a high value is that a worker thread cannot respond to a stop command. If this value is set to 0, the listener runs once and terminates. The default value is 2 seconds.
Sort	The case-sensitive name of the channel that connects with the remote MQ Server queue manager (for the MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.
Scan Sub-directories	Location where error documents are sent. This can be a predefined port name or another full URL. Optional.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference **HyperText Transfer Protocol Listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Port	The port where the adapter listens for the HTTP transfer.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference TCP Listener Configuration Parameters

On the Settings tab:

Parameter	Description
Port	The port where the adapter listens for the TCP transfer.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.
Allowable Client Host	The name or address of the client restricted to accessing this adapter.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference IBM MQ Series (MQ) Listener Configuration Parameters

On the Settings tab:

Parameter	Description
Queue Manager	The name of the MQ queue manager to be used.
Queue Name	The name of the MQ Series or WebSphere MQ queue that the SWIFT system polls.

Parameter	Description
Polling Interval	The maximum wait interval (in the format *nn*H:*nn*M:*nn*S) between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. However, with a high value, the worker thread cannot respond to a stop command. If timeout is set to 0, the listener runs once and terminates. The default is 2 seconds.

On the MQ Client tab:

Parameter	Description
Host	The host where the MQ Server is located.
Port	The port number used to connect to an MQ Server.
Channel	The channel between an MQ Client and an MQ Server.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

Reference **File Transfer Protocol (FTP) Listener Configuration Parameters**

On the Settings tab:

Parameter	Description
Host	The name of the FTP host.
Port	The port where the adapter listens on the FTP transfer.

Parameter	Description
User	The user name to log onto the FTP Server.
Password	The password for the FTP user.
Location	The directory where messages are received. DOS-style file patterns are available for this parameter. You can specify a file pattern as well as a directory. For example, c:\xyz\ab*cd (without a file suffix) takes the file suffix from that parameter. If you use a pattern, files are selected based on the suffix and then the pattern. AB?CD selects ABxCD. AB*CD selects ABxxxCD.
Encoding	The character set encoding for inbound documents. For example, UTF-8. The default is ISO-8859-1 US and Western Europe.
Polling Interval	The maximum wait interval (in the format *nn*H:*nn*M:*nn*S) between checks for new documents. The higher this value, the longer the interval, and the fewer system resources that are used. However, with a high value, the worker thread cannot respond to a stop command. If timeout is set to 0, the listener runs once and terminates. The default is 2 seconds.

On the Advanced tab:

Parameter	Description
Transform Type	Select the pre-built transform template from the drop-down list. To enable batch processing, select *BatchSplitter* from the drop-down list. The batch splitter preparses an entire EDI document and splits the document into individual transactions. Each transaction retains its Interchange Header/Trailer information. Once the batch splitter is finished splitting the EDI document, the transactions are ready to be transformed into XML.
Location for ack copies	The directory in which the acknowledgement document is placed.

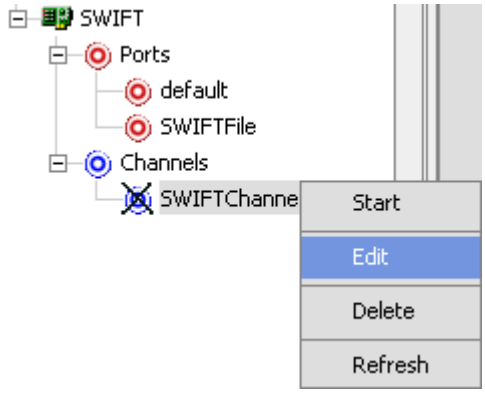
Modifying a Channel

The following procedures describe how to edit and delete a channel using Application Explorer. To review the channel settings, you select the channel name. In the right pane, a table appears that summarizes the information associated with the channel you created.

Procedure How to Edit a Channel

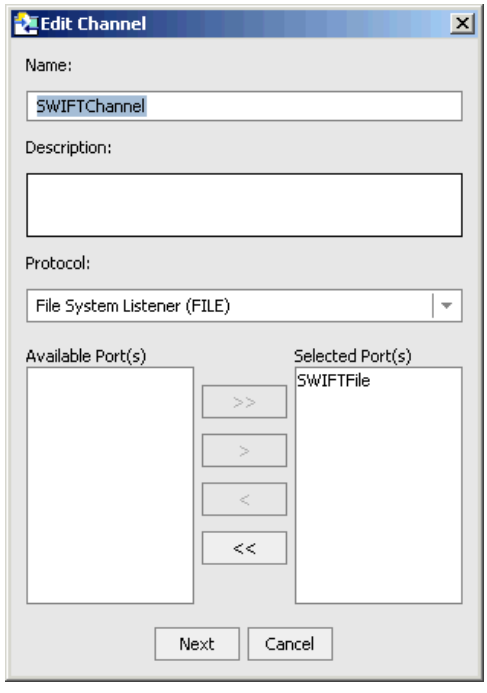
To edit a channel:

1. To view the available channels, click the *Channels* node in the left pane.



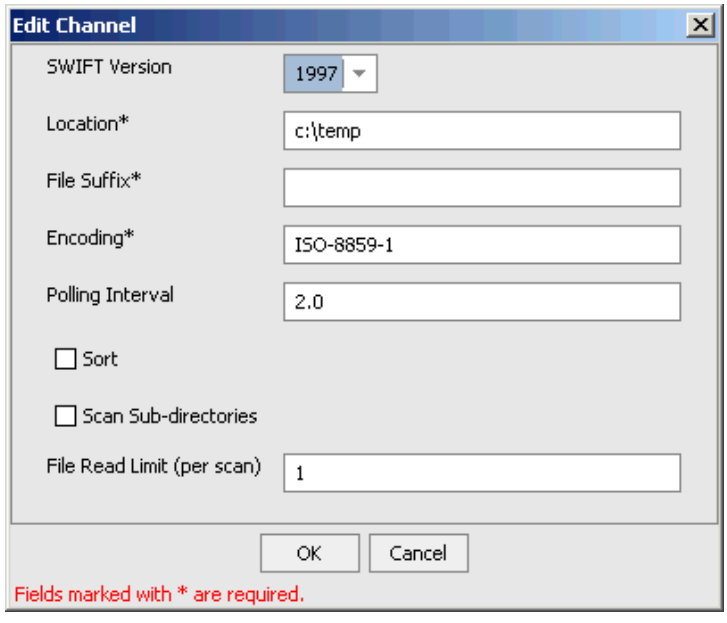
2. Right-click the channel you want to edit, for example, SWIFTChannel, and select *Edit*.

The Edit Channel dialog box opens.



3. Make the required changes to the channel configuration.
4. Click *Next*.

The following dialog box opens.

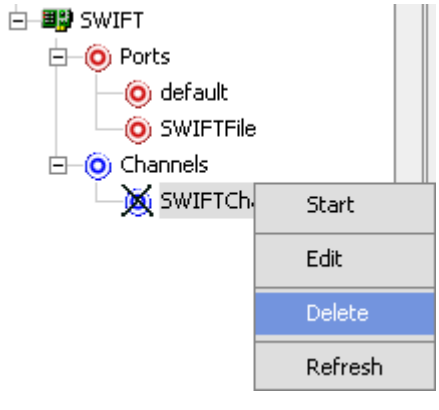


5. Make the required changes and click *OK*.

Procedure How to Delete a Channel

To delete an existing channel:

1. In the left pane, right-click the channel, for example, SWIFTChannel.



2. Select *Delete*.

   The channel disappears from the Channels list.

Deploying iWay Components in a Clustered BEA WebLogic Environment

Events can be configured in a clustered BEA WebLogic environment. You can deploy iBSE or JCA to this environment. This topic uses iBSE as an example, but you can follow the same procedures when deploying JCA. The only difference is that you need to deploy the JCA connector .RAR file to the clustered environment.

A cluster consists of multiple server instances running simultaneously, yet appears to clients to be a single server instance. The server instances that contain a cluster can be run on one machine, but are usually run on multiple machines.

Clustering provides the following benefits:

- Load balancing
- High availability

Service requests are processed through the HTTP router and routed to an available managed server.

Events are server-specific and are not processed through the HTTP router. You must configure each server separately.

Procedure How to Deploy iWay Components in a Clustered Environment

To deploy iWay components in a clustered environment:

1. Using the BEA Configuration Wizard:
   a. Configure an administrative server to manage the managed servers.
   b. Add and configure as many managed servers as required.
   c. Add and configure an HTTP router. This does not have to be a part of WebLogic and can be an outside component.
   d. If you configure the HTTP router within WebLogic, start it by entering the following command:

```
StartManagedWebLogic HTTPROUTER http://localhost:7001
```

   where:

   HTTPROUTER
   Is the name of the server on which the HTTP router is running.

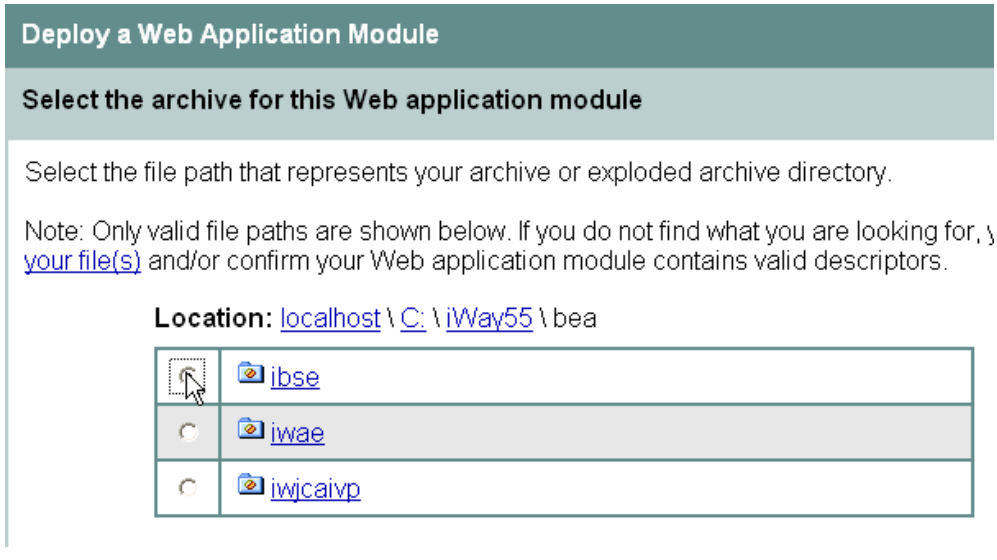
   `http://localhost:7001`
   Is the location of the admin console.

   e. Add the managed servers to your cluster/clusters.

For more information on configuring WebLogic Integration for deployment in a clustered environment, see *Deploying WebLogic Integration Solutions*.

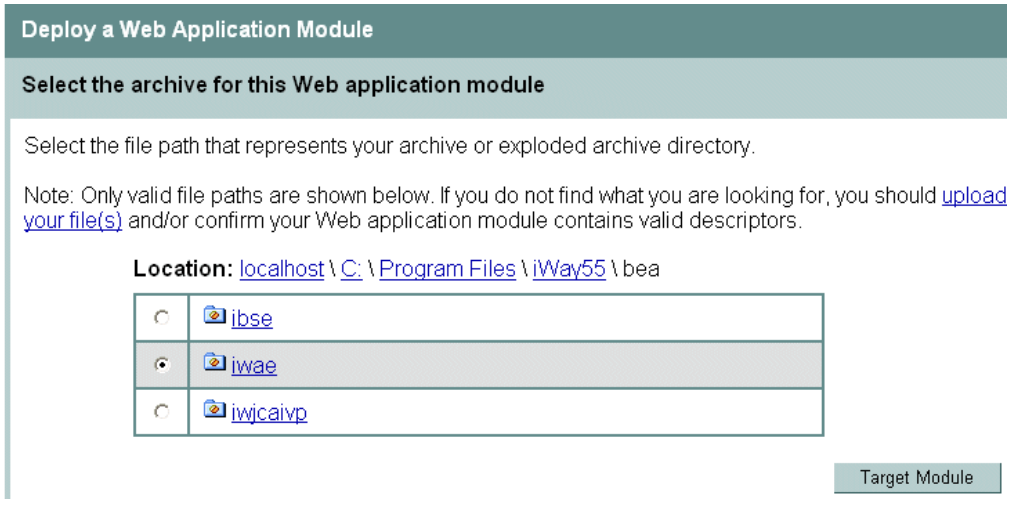
2. Start the WebLogic Server and open WebLogic Server Console.
3. Deploy iBSE to the cluster by selecting *Web Application Modules* from the Domain Configurations section, and clicking *Deploy a new Web Application Module*.

   A page appears for you to specify where the Web application is located.
4. To deploy iBSE, select the option button next to the ibse directory and then click *Target Module*.

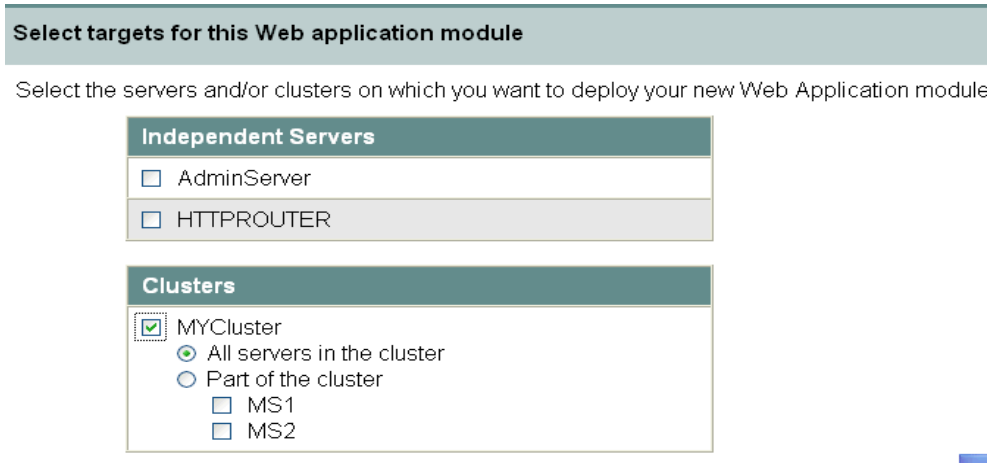


5. To deploy servlet Application Explorer, select the option button next to the iwae directory and then click *Target Module*.

   If you are using servlet Application Explorer, deploy it only on the admin server or one of the managed servers.

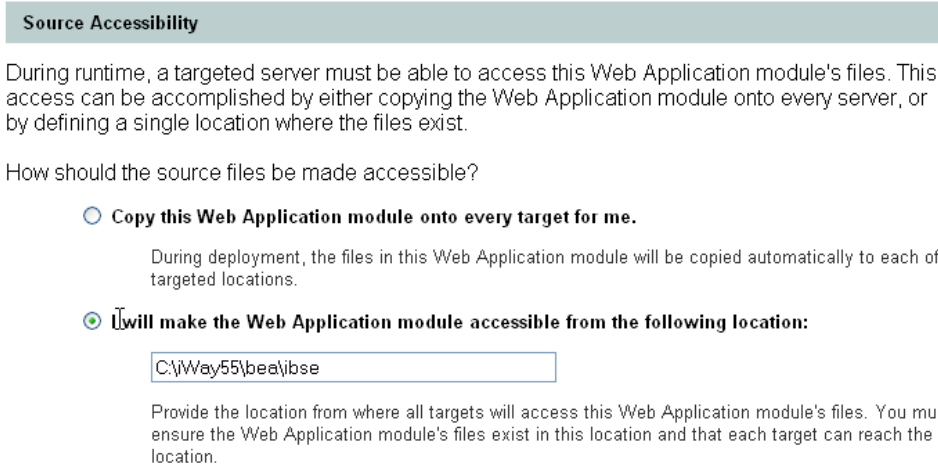


The following window opens.



6. Select the servers and/or clusters on which you want to deploy the application and click *Continue*.

The following window opens.



7. Select the *I will make the Web Application module accessible from the following location* option button and provide the location from which all targets will access iBSE.

iWay Software recommends that you use a single instance of iBSE, rather than copying iBSE onto every target.

Note: iBSE must use a database repository (SQL or Oracle). Do not use a file repository. You can select this in the Repository Type drop-down list in the iBSE monitoring page. After configuring a database repository, you must restart all of the managed servers.

`http://hostname:port/ibse/IBSEConfig/`

where:

hostname

Is where your application server is running. Use the IP address or machine name in the URL; do not use localhost.

port

Is the port specific to each server, since you deploy iBSE to an entire cluster. For example, 8001, 8002, or any other port that is specified for each managed node.

8. Click *Deploy*.

Procedure How to Configure Ports and Channels in a Clustered Environment

You can use Swing Application Explorer deployed in BEA WebLogic Workshop or Servlet Application Explorer to configure ports and channels in a clustered environment.

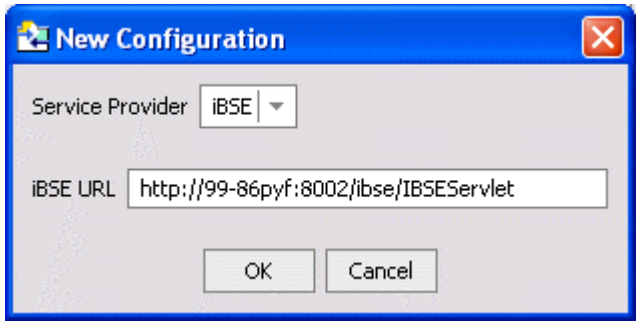
Note: Before using Servlet Application Explorer in a clustered environment, you must edit the web.xml file and specify the correct URL to your iBSE deployment. The default location on Windows is:

```
C:\Program Files\iWay55\bea\iwae\WEB-INF\web.xml
```

For more information on configuring the web.xml file for the Servlet Application Explorer, see *iWay Installation and Configuration for BEA WebLogic*.

To configure ports and channels in a clustered environment:

1. Open Swing Application Explorer in BEA WebLogic Workshop.
2. Create a new connection to the iBSE instance. For information on creating a new configuration, see *Creating a New Configuration* on page A-3.



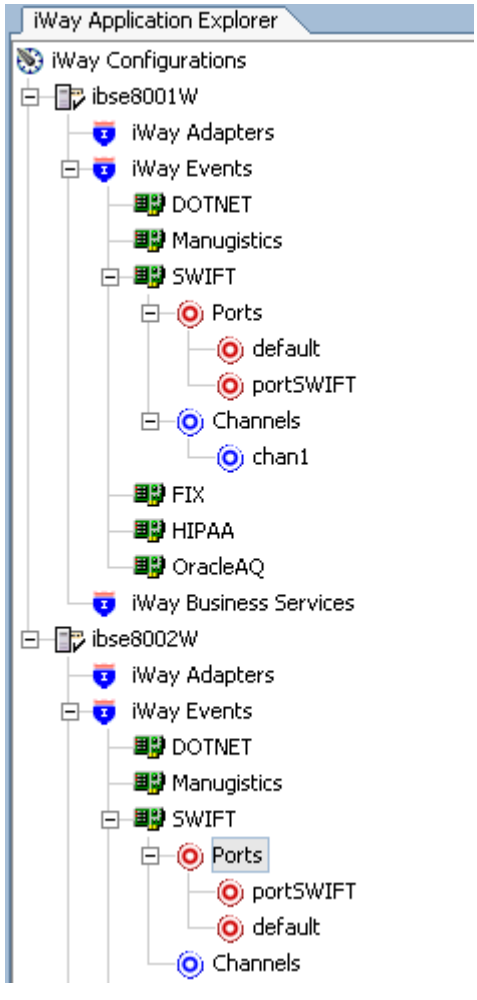
Note: Use the IP address or machine name in the URL; do not use localhost.

3. Connect to the new configuration and select the iWay Events node in the left pane of Application Explorer.

4. Add a new port for the SWIFT adapter. For more information, see *Creating an Event Port* on page B-3.
5. Create a channel and add the port you created. For more information, see *Creating a Channel* on page B-19.
6. Click *Next* and enter the application server parameters.
7. Start the channel.
8. Create a new configuration and connect to the second iBSE instance.

   The connection to iBSE must be configured to each instance of the managed server.

   The following graphic shows two configurations.



The following operations performed on one managed server will be replicated on all other managed servers:

- Create port and channel: Creates the channel and port under all available servers.
- Delete port and channel. Deletes the port and channel under all available servers.

The following operations must be performed on each server:

- Start channel. Starts the channel for the specific server.
- Stop channel. Stops the channel for the specific server.

Reader Comments

In an ongoing effort to produce effective documentation, the Documentation Services staff at Information Builders welcomes any opinion you can offer regarding this manual.

Please use this form to relay suggestions for improving this publication or to alert us to corrections. Identify specific pages where applicable. You can contact us through the following methods:

Mail: Documentation Services - Customer Support
Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898

Fax: (212) 967-0460

E-mail: books_info@ibi.com

Web form: http://www.informationbuilders.com/bookstore/derf.html

Name:_____

Company:_____

Address:_____

Telephone:_____Date:_____

E-mail:_____

Comments:

Reader Comments

Information Builders, Two Penn Plaza, New York, NY 10121-2898 **(212) 736-4433**