

iWay

iWay Application System Adapter for SAP R/3
User's Guide
Version 5 Release 5

Updated for J2EE CA 1.5

EDA, EDA/SQL, FIDEL, FOCCALC, FOCUS, FOCUS Fusion, FOCUS Vision, Hospital-Trac, Information Builders, the Information Builders logo, Parlay, PC/FOCUS, SmartMart, SmartMode, SNAPPack, TableTalk, WALDO, Web390, WebFOCUS and WorldMART are registered trademarks, and iWay and iWay Software are trademarks of Information Builders, Inc.

Sun and Java are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2005, by Information Builders, Inc and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

Preface

This document is written for system integrators who develop client interfaces between SAP and other applications. It describes how to use the iWay Application System Adapter for SAP R/3 to integrate SAP IDocs, RFCs, and BAPIs with your Sun Java System Application Server. It is assumed that readers understand Web technologies and have a general understanding of Microsoft Windows and UNIX systems.

How This Manual Is Organized

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

Chapter/Appendix		Contents
1	Introducing the iWay Application System Adapter for SAP R/3	Provides an overview of the iWay Application System Adapter for SAP R/3. Discusses key features and functionality of the adapter.
2	Configuring SAP Inbound Processing	Describes how to configure your SAP system for inbound (client) processing.
3	Creating XML Schemas for SAP	Describes how to create XML schemas for SAP business objects using Servlet Application Explorer.
4	Creating and Publishing iWay Business Services	Describes how to create and publish iWay Business Services using Servlet Application Explorer.
5	Configuring the Event Adapter for SAP	Describes how to create ports and channels to listen for SAP events using Servlet Application Explorer.
6	Using Web Services Policy-Based Security	Describes how to configure Web services policy-based security using Servlet Application Explorer.
7	Management and Monitoring	Describes how to use the management and monitoring tools provided by the iWay Business Services Engine and the JCA Test Tool.
8	Understanding SAP Events	Describes how to configure and test your SAP system for event processing.

9	Troubleshooting and Error Messages	Describes limitations and workarounds when connecting to SAP. The adapter-specific errors listed in this chapter can arise whether using the adapter with a JCA, or with an iBSE configuration.
A	Sample Files and Coding Techniques	Provides sample request and response documents sent between SAP and the iWay Application System Adapter for SAP R/3 as well as a sample RFC module.

Documentation Conventions

The following table lists and describes the conventions that apply throughout this manual.

Convention	Description
<code>THIS TYPEFACE</code> or <code>this typeface</code>	Denotes syntax that you must enter exactly as shown.
<code><i>this typeface</i></code>	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
<code><u>underscore</u></code>	Indicates a default setting.
<code><i>this typeface</i></code>	Represents a placeholder (or variable) in a text paragraph, a cross-reference, or an important term.
<code>this typeface</code>	Highlights a file name or command in a text paragraph that must be lowercase.
<code><i>this typeface</i></code>	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.

Customer Support

Do you have questions about the iWay Application System Adapter for SAP R/3?

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 Application System Adapter for SAP R/3 questions. iWay Software 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.iwaysoftware.com>. It connects you to the tracking system and known-problem database at the iWay Software 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.iwaysoftware.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 iWay Software 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, or iWay Business Services Engine.
Container Version	

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

Component	Version
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 <i>not</i> 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
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

1. Introducing the iWay Application System Adapter for SAP R/3	1-1
Features of the iWay Application System Adapter for SAP R/3	1-2
SAP Certification	1-2
Supported Platforms	1-3
SAP Business Components	1-4
Integrating With SAP	1-5
Understanding Web Services and Java Connector Architecture Functionality	1-6
Web Services	1-6
Java Connector Architecture	1-6
Component Information for the iWay Application System Adapter for SAP R/3	1-7
Component Information Roadmap	1-7
Application Explorer	1-7
The iWay Business Services Engine (iBSE)	1-7
The Enterprise Connector for J2EE Connector Architecture (JCA)	1-8
2. Configuring SAP Inbound Processing	2-1
Overview	2-2
Configuring a Logical System	2-3
Configuring a Distribution Model	2-8
Defining a Partner Profile	2-11
3. Creating XML Schemas for SAP	3-1
Overview	3-2
Starting Servlet Application Explorer	3-2
Establishing a Target for SAP	3-3
Creating a New Target	3-4
Connecting to a Target	3-10
Disconnecting From a Target	3-12
Modifying a Target	3-13
Deleting a Target	3-14
Viewing Application System Objects	3-15
Creating an XML Schema	3-18
4. Creating and Publishing iWay Business Services	4-1
Understanding iWay Business Services	4-2
Creating iWay Business Services	4-2
Creating Business Services With Application Explorer	4-2
Identity Propagation	4-11
5. Configuring the Event Adapter for SAP	5-1
Understanding Event Functionality	5-2

Adding, Modifying, or Deleting a Port	5-2
Editing an Event Port	5-16
Deleting an Event Port	5-17
Adding, Modifying, or Deleting a Channel	5-18
Creating a Channel	5-18
Modifying a Channel	5-25
Deleting a Channel	5-26
Synchronous Event Processing	5-26
6. Using Web Services Policy-Based Security	6-1
iWay Business Services Policy-Based Security	6-2
Configuring iWay Business Services Policy-Based Security	6-3
7. Management and Monitoring	7-1
Managing and Monitoring Services and Events Using iBSE	7-2
Managing and Monitoring Services and Events Using the JCA Test Tool	7-16
Setting Engine Log Levels	7-20
Configuring Connection Pool Sizes	7-22
Migrating Repositories	7-22
File Repositories	7-23
iBSE Repositories	7-23
JCA Repositories	7-27
Migrating Event Handling Configurations	7-27
Exporting or Importing Targets	7-31
Retrieving or Updating Web Service Method Connection Information	7-35
Starting or Stopping a Channel Programmatically	7-39
8. Understanding SAP Events	8-1
Overview	8-2
Related Concepts and Terminology	8-2
Client and Server Programs	8-2
SAP Gateway	8-2
Program IDs and Load Balancing	8-3
Registering Your Program ID in SAPGUI	8-3
Testing the SAP Event Adapter	8-6
Application Link Enabling Configuration for the Event Adapter	8-8
Testing the SAP ALE Configuration	8-17
9. Troubleshooting and Error Messages	9-1
Troubleshooting	9-2
Error Messages in Application Explorer	9-2
Error Messages in SAP	9-5
Error Messages in JCA	9-6

- Error Messages in iBSE9-6
 - General Error Handling in iBSE9-6
 - Adapter-Specific Error Handling9-7
- A. Sample Files and Coding Techniques A-1**
 - Sample RFC Request Document A-2
 - Sample RFC Response Document A-2
 - Sample IDOC XML for Message Type DEBMAS A-4
 - Collected IDocs A-9
 - Sample RFC Module A-9
 - Sample Wrapper Module to Call Functions on Remote Destinations (Service) A-10
 - Using Staging BAPIs to Retrieve SAP BW Metadata A-12

Contents

CHAPTER 1

Introducing the iWay Application System Adapter for SAP R/3

Topics:

- Features of the iWay Application System Adapter for SAP R/3
- SAP Certification
- SAP Business Components
- Integrating With SAP
- Understanding Web Services and Java Connector Architecture Functionality
- Component Information for the iWay Application System Adapter for SAP R/3

The following section provides an overview of the iWay Application System Adapter for SAP R/3.

Features of the iWay Application System Adapter for SAP R/3

The iWay Application System Adapter for SAP R/3 provides a means to exchange real-time business data between SAP systems and other application, database, or external business partner systems. The adapter enables external applications for inbound and outbound processing with SAP.

The adapter uses XML messages to enable non-SAP applications to communicate and exchange transactions with SAP using one of the following two methods.

- **Event Adapter.** Applications use this capability if they require access to SAP data only when an SAP business event occurs.
- **Request/response.** Applications use this capability when they must initiate an SAP business event.

If the request is for retrieving data from SAP, then the adapter sends the application a response message in the form of an XML document with the data embedded.

The iWay Application System Adapter for SAP R/3 provides:

- Support for bidirectional message interactions.
- The Servlet Application Explorer, a GUI tool which uses SAP object repository metadata to build XML schemas and Web services to handle adapter requests or event data.
- Support for Remote Function Calls (RFC), Business Application Programming Interfaces (BAPI), and Intermediate Documents (IDoc) interfaces to SAP.

SAP Certification

SAP has certified the iWay Application System Adapter for SAP R/3 for use with all versions of SAP, including mySAP.com solutions.

The adapter provides state-of-the-art middleware solutions for SAP Basis and SAP Web application server-based systems. This adapter has achieved three interface certifications that promote cost-effective and low-risk solutions:

- **CA-ALE certification.** Enhances electronic data interchange (EDI) subsystem interface with SAP Basis and SAP Web Application Server. Using direct program-to-program remote communication and transformation from non-SAP systems to SAP solution-based systems, the adapter expedites the conversion, import, and export of critical intermediate documents (IDocs).

- **CA-AMS certification.** Rapidly bridges SAP Basis and SAP Web Application Server data exchange with other applications through pure message delivery. As an ALE (Application Link Enabling) Message Handler, the adapter sends IDoc messages without a requirement for conversion from one or more SAP solution-based systems.
- **CA-XML certification.** Eases the communication between external middleware with SAP Basis and SAP Web Application Server over the Internet using XML, HTTP, or HTTPS. The adapter immediately transfers SAP solution specifications into XML for straight transfer into application subsystem repositories.

Supported Platforms

The following SAP platforms are supported by the iWay Application System Adapter for SAP R/3:

- SAP Web Application Server Version 6.1 and higher.
- SAP R/3 4.0 and higher.

Note: SAP R/3 3.1 is “off SAP maintenance.” This release is supported under special circumstances, but not all functions can be supported.

If you have questions regarding this release, contact your SAP representative. This version also requires an SAP support pack to support the functions of the Business Object Repository.

- SAP Enterprise R/3 4.7.
- All or portions of: MySAP.com technology solutions (SAP BW, SAP APO, SAP CRM, SAP SRM, SAP EBP, SAP SEM, SAP WP, SAP KW).

Note: Release versions may vary by product component. In addition, SAP functions may vary by SAP product version and support package.

The following operating systems are supported:

- Windows NT/2000/XP
- UNIX

SAP Business Components

The iWay Application System Adapter for SAP R/3 is designed to provide standard access to SAP business components such as Remote Function Call (RFC) modules, BAPIs (Business Application Programming Interfaces), and IDocs (Intermediate Documents), that are used to support existing business processes. These business components and methods are available to the adapter as requests of SAP and to the event adapter when SAP invokes its remote requests and work in the following ways:

- **Business Application Programming Interfaces (BAPIs)** are interfaces within the business framework that are used to link SAP components to one another or to third-party components. BAPIs are called synchronously and return information.
- **Remote Function Call (RFC) Modules** are SAP application interfaces that enable clients to invoke SAP technologies and receive responses.

Note: Depending on the release or service pack installed, certain RFCs, for example, RFC_CUSTOMER_GET, may not exist in your particular SAP system. Therefore, the examples included in this documentation may not be relevant to your system. If this is the case, you should use the examples as a general reference for adapter functionality and choose an RFC that exists within your SAP application environment.

As described in SAP Release Note 109533, SAP Function Modules (RFCs) can be delivered with different release statuses. SAP supports only RFCs that are awarded with the Released for Customer status. There is no claim to the release independencies of the interfaces and the continued existence/functionality of the modules. For more information on the status of a specific function module, consult your SAP Service Marketplace.

- **Intermediate Documents (IDocs)** are the “logical messages” that correspond to different business processes. They allow different application systems to be linked by a message-based interface. The IDoc type indicates the SAP format to use to transfer the data for a business transaction. An IDoc is a real business process in the form of an IDoc type that can transfer several message types. An IDoc type is described by the following components:
 - **Control records.** A control record contains data that identifies the sender, the receiver, and the IDoc structure. An IDoc contains one control record.
 - **Data records.** A data record consists of a fixed administration part and a data part (segment). The number and format of the segments can be different for each IDoc type.

- **Status records.** A status record describes the processing stages through which an IDoc passes.

The following scenario is an example of IDoc functionality and its components:

Purchase order number 4711 was sent to a vendor as IDoc number 0815. IDoc number 0815 is formatted in IDoc type ORDERS01 and has the status records “created” and “sent.” The purchase order corresponds to the “logical” message ORDERS.

Integrating With SAP

You can use the iWay Application System Adapter for SAP R/3 to invoke an SAP business process, such as add/update account, or you can use the adapter as part of an integration effort to connect SAP and non-SAP systems.

BAPIs and RFCs are called synchronously by the adapter and always return data (either technical error information or a well-formed response document). IDocs are processed asynchronously.

The adapter is bidirectional and can process an event in SAP by receiving RFCs and IDocs directly from SAP. The SAP system can be configured to send an IDoc or RFC out to a logical system when a certain event occurs, in this case to the adapter. The output sent by SAP can be in any of the following forms:

- An RFC request, for example, RFC_CUSTOMER_GET.
- A BAPI request, for example, BAPI_COMPANYCODE_GETLIST.
- An IDoc.

For request processing, the iWay Application System Adapter for SAP R/3 can send requests to SAP using the BAPI, RFC, or IDoc interfaces.

The adapter quickly and easily integrates your SAP IDocs, RFCs, and BAPIs with mission critical SAP system applications and other enterprise applications. Adapter benefits include:

- Eliminating the requirement for custom coding.
- Consistent data representation—a standard XML representation of event data and request/response documents for SAP. The developer is freed from the specific details of the SAP interface (BAPI, RFC, IDoc,) and the specific configuration details of the target SAP system.
- Adherence to SAP ABAP serialization rules and SAP Interface Repository standards published by SAP AG.

Understanding Web Services and Java Connector Architecture Functionality

The following section describes how the iWay Application System Adapter for SAP R/3 can incorporate Web services and Java Connector Architecture technology.

Web Services

Web services allow SAP calls to be made across the Internet or an intranet, using specialized versions of the XML language that allow a developer to specify the parameters, connections methods, and remote calls and store them for reference in a repository. At runtime, a person, an interface, or another function, can read this repository and automatically invoke the service. Web services currently do not have industry standards for transactional behavior. Web services are useful when your function calls must be made across firewall boundaries. Using Web services, you can use functions provided by external providers, as long as you know the function interface.

Web Services Example

A Web service exposes the “cup” interface, which provides a teacup. The Acme Company exposes the “tea” Web service, which provides a brown liquid when the correct parameter “money” is provided. A cup of tea can be received by invoking the “tea” Web service and passing the “money” parameter. Additional components are not required to receive tea via the “tea” Web service.

Java Connector Architecture

Java Connector Architecture (JCA) provides a reusable component model to build and deploy multi-tier applications that are platform and vendor-independent. JCA acts as a type of envelope or “container” that will allow the adapter to run inside the Sun Java System Application Server and connect to SAP and immediately return the results. JCA is useful when your SAP system resides within a local intranet or is accessed directly. JCA implements JAVA Connection and Transaction models. JCA requires a resource adapter to be physically deployed on the host application server to access the remote EIS system.

Using combinations of JCA and Web services is possible. For example, a JCA application can be invoked via a Web service or a Web service may be implemented inside a JCA container. The standards and protocols are still evolving.

JCA Example

An application server, “Table” implements the “tea” container, which contains “tea, gas heat, water, and cup”. A JCA container can be deployed that contains tea in the “Table” application server. If any of the elements of “tea” container are missing, for example, “gas heat”, then no tea is produced. The container must physically possess the “tea, gas heat, water, and cup” components to receive the tea.

Component Information for the iWay Application System Adapter for SAP R/3

The iWay Application System Adapter for SAP R/3 works with Application Explorer in conjunction with the following components:

- iWay Business Services Engine (iBSE)
- Enterprise Connector for J2EE™ Connector Architecture (JCA)

Component Information Roadmap

The following table lists the location of component information for the iWay Application System Adapter for SAP R/3. A description of Application Explorer, the iWay Business Services Engine (iBSE), and the Enterprise Connector for J2EE Connector Architecture (JCA) follows the table.

Deployed Component	For more information, see
Application Explorer	<ul style="list-style-type: none"> • Chapters 3, 4, and 5 of this guide • <i>iWay Installation and Configuration</i>
iWay Business Services Engine (iBSE)	<ul style="list-style-type: none"> • <i>iWay Installation and Configuration</i>
Enterprise Connector for J2EE Connector Architecture (JCA)	<ul style="list-style-type: none"> • <i>iWay Connector for JCA User's Guide</i> • <i>iWay Installation and Configuration</i>

Application Explorer

Application Explorer uses an explorer metaphor to browse the SAP system for BAPIs, RFCs, and IDocs. The explorer enables you to create XML schemas and Web services for the associated object. In addition, you can create ports and channels to listen for events in SAP. External applications that access SAP through the iWay Application System Adapter for SAP R/3 use either XML schemas or Web services to pass data between the external application and the adapter.

The iWay Business Services Engine (iBSE)

The iWay 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 solves Enterprise Application Integration (EAI) hurdles that other programming models cannot. It enables programs to communicate with one another using a text-based 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 Enterprise Connector for J2EE Connector Architecture (JCA)

The Enterprise Connector for J2EE Connector Architecture (JCA) enables developers of JCA-compliant applications to deploy adapters as JCA resources.

The 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.

Two connectors are distributed in the iWay installation package. One conforms to the JCA 1.0 specification, with extensions that allow for the consumption of events. The other conforms to the JCA 1.5 specification. The JCA 1.0 connector provides for event functionality through the configuration of ports and channels. When using the adapter in conjunction with a JCA 1.5 connector, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities. For more information on event capabilities of the iWay JCA connectors, see Chapter 5, *Configuring the Event Adapter for SAP*.

For more information on installing and deploying both connectors, see *iWay Installation and Configuration*

CHAPTER 2

Configuring SAP Inbound Processing

Topics:

- Overview
- Configuring a Logical System
- Configuring a Distribution Model
- Defining a Partner Profile

The following section describes how to configure your SAP system for inbound (client) processing.

Overview

SAP Remote Function calls require no system setup other than the Connection Target Parameters. If you do not intend to send IDocs to the SAP system, you may skip this chapter.

In inbound IDoc processing, the SAP adapter reads an XML document and creates an SAP Standard format IDoc.

The adapter can optionally transform incoming data into the standard format via XML transformation or SAP ALE transformation.

Once assembled as an IDoc, the file is then sent to SAP for inbound processing. Multiple IDocs can also be assembled into a larger document for efficient processing. For more information, see the SAP documentation.

ALE IDocs used for transmission of Electronic Data Interchange (EDI) message do require information about the intended target and method of transmission stored on the SAP application server.

An IDoc consists of a Header, which contains sender, processing type, receiver and other information, and multiple data segments, which contain the information to be processed.

SAP requires a "logical system" entry for each transmission system, to store the parameters for processing of the IDoc messages. The Logical System will store type information about the Partner and the kind of messages expected from the partner. The "Partner Profile" defines the kind of message and the type of SAP function called to process that particular IDOC in an application. SAP also requires a filtering model, whether or not it is used for a particular message, called a "Distribution Model" that defines a message type and applies optional segment filters. Once these three things have been established, Inbound ALE/IDOC processing can begin.

The Control Section or reference structure file EDI_DC40 (defined in SAP R/3), must be completed and contains all the identifying information about the IDoc. The assembled header and data records are sent to SAP by the adapter. SAP does not need a incoming port specified, it will take the incoming RFC stream and assign a port designation. In the Partner Profile, a function module must be identified to process the IDoc in the R/3 system.

Usually, IDocs are written directly to the database and slowly read by the application (for example, Purchasing for Purchase Orders). This can take quite awhile depending on the type of data and the application. So the adapter can "post to the database and return" or "post and wait" and this is defined in the Partner Profile on the host system.

In any case, you may send a status IDoc message to get the status of your IDoc, or use appropriate transaction codes in SAP to view the IDocs online.

You must perform the following steps to configure SAP for inbound IDoc processing:

1. Configure a logical system.
2. Configure a distribution model.
3. Define an inbound partner profile.

Configuring a Logical System

In any distributed environment, each participating system must have a unique ID to avoid confusion. In SAP, the name of the logical system is used as the unique ID. This name is assigned explicitly to one client in an SAP system.

Procedure: How to Configure a Logical System

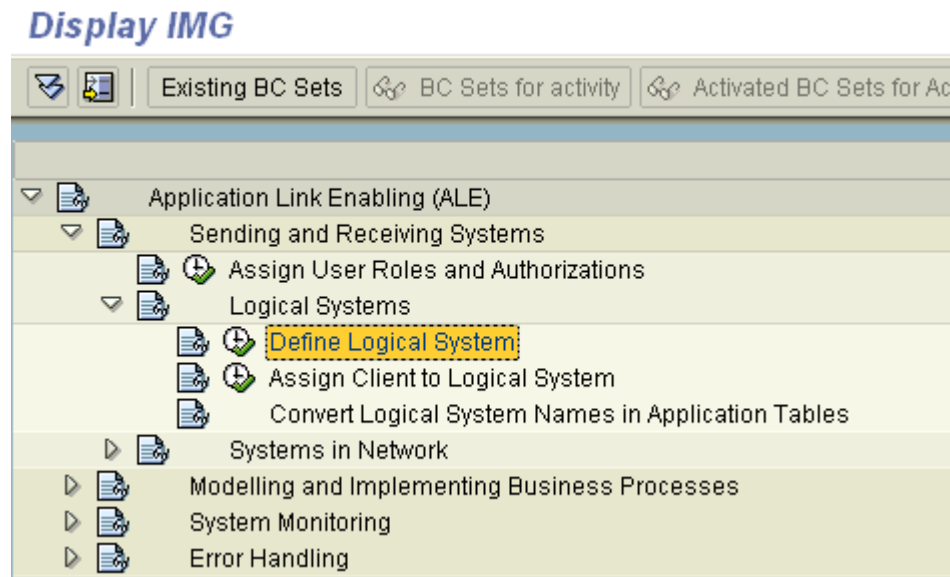
The following image shows the /nsale transaction in the field under the menu bar.



To configure a logical system:

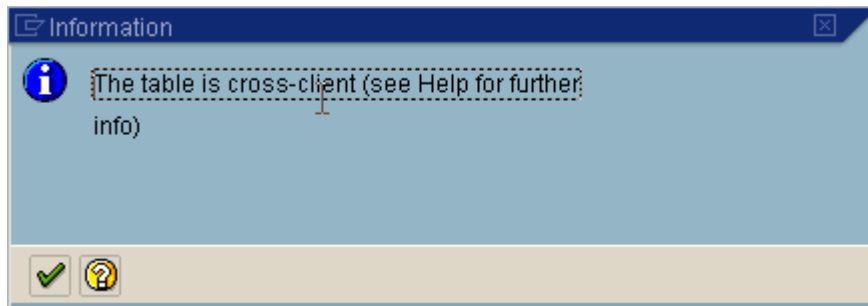
1. Execute the *sale* transaction.

The Display IMG window opens.



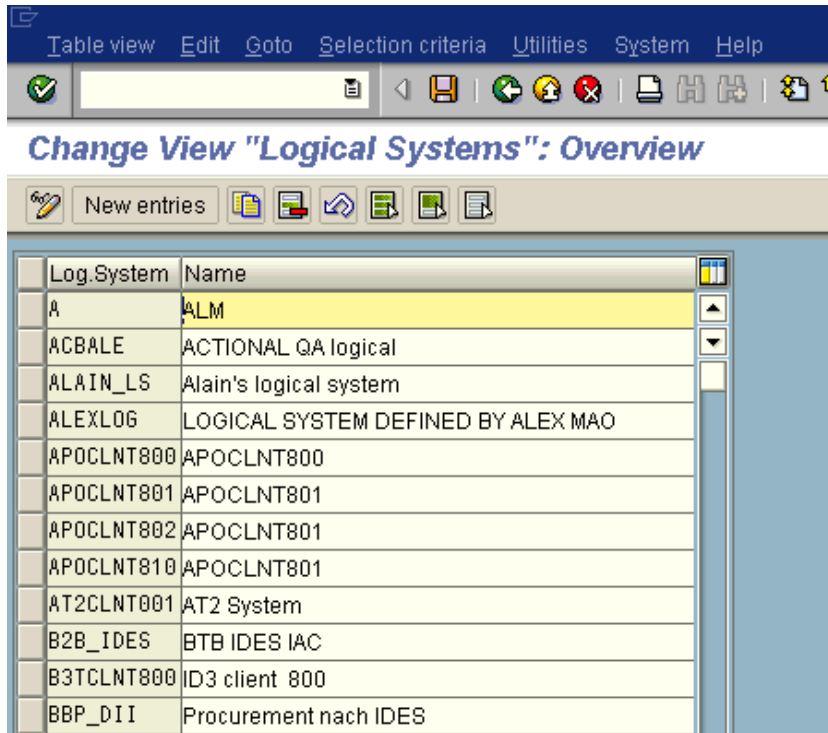
2. Expand *Sending and Receiving Systems*, then *Logical Systems*, and then, select *Define Logical System*.
3. Click the *IMG - Activity* icon.

An information window appears that informs you that the table is cross-client as shown in the following image.



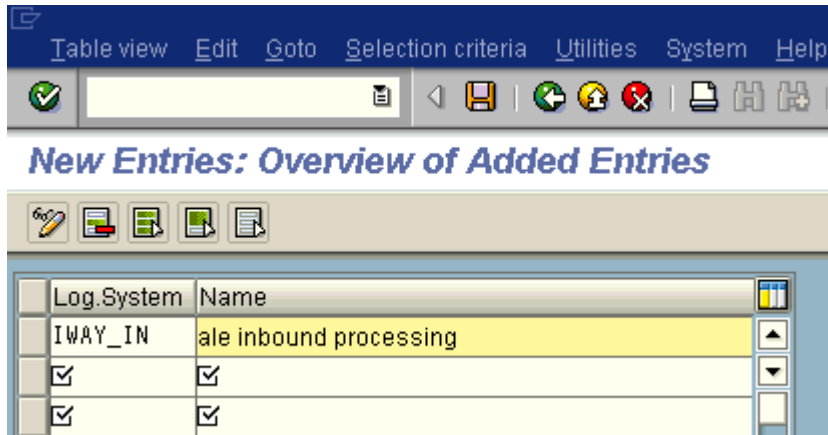
4. Click the checkmark icon to continue.

The Change View "Logical Systems" window opens with a list of logical systems and their names as shown in the following image.



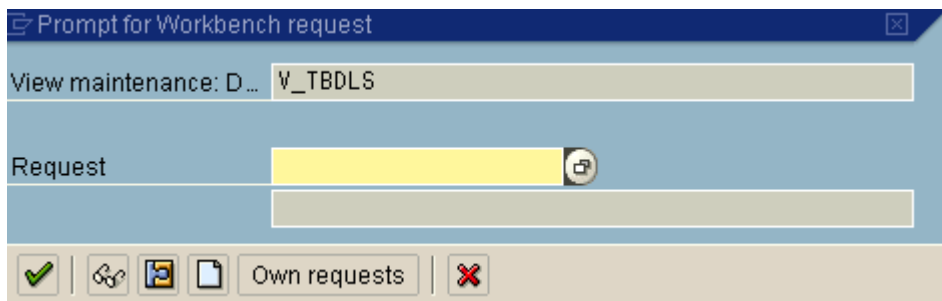
5. Click the *New entries* button.

The New Entries window opens where you can type information for the logical system and its corresponding name as shown in the following image.



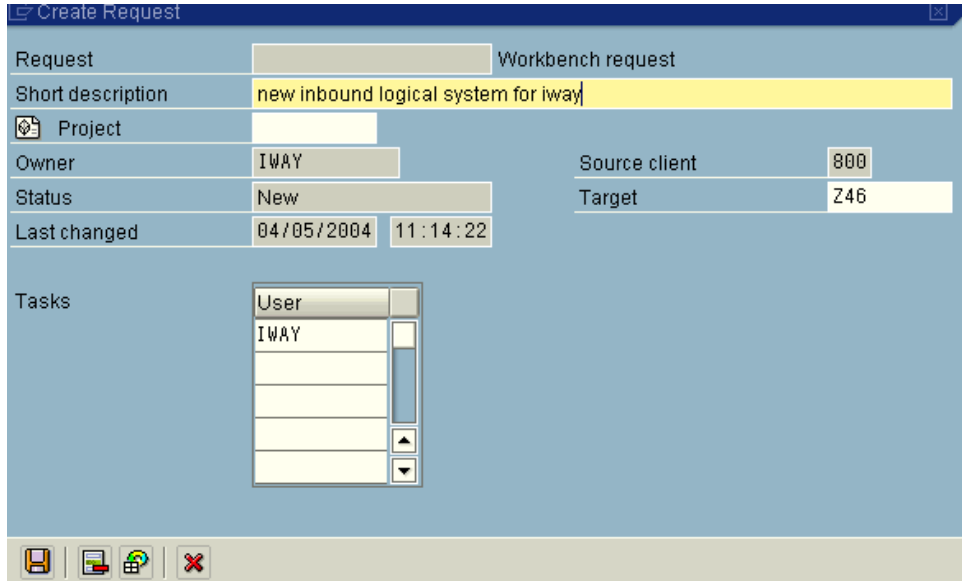
- a. In the Log.System column, type the Logical System, for example, IWAY_IN.
 - b. In the Name column, type a corresponding description.
6. Click Save.

The Prompt for Workbench request window opens as shown in the following image.



7. Click the Create Request icon.

The Create Request window opens and displays fields that are already populated or empty fields in which to specify information about your request as shown in the following image.



- a. In the Request field, type a name.
 - b. In the Short description field type a brief description of your request.
8. Click Save.

The logical system you configured, for example, IWAY_IN, appears in the list as shown in the following image.

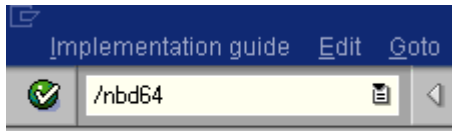
IWAYMKT	IWAY marketing logical system
IWAY_IN	ale inbound processing
JRB46LS	jr logical

Configuring a Distribution Model

A distribution model is used to describe the ALE message flow between logical systems. Business objects are distributed to connected recipients according to a unique distribution model that can contain rules of varying complexity depending on the type of business objects involved.

Procedure: How to Configure a Distribution Model

The following image shows the /nbd64 transaction in the field under the menu bar.

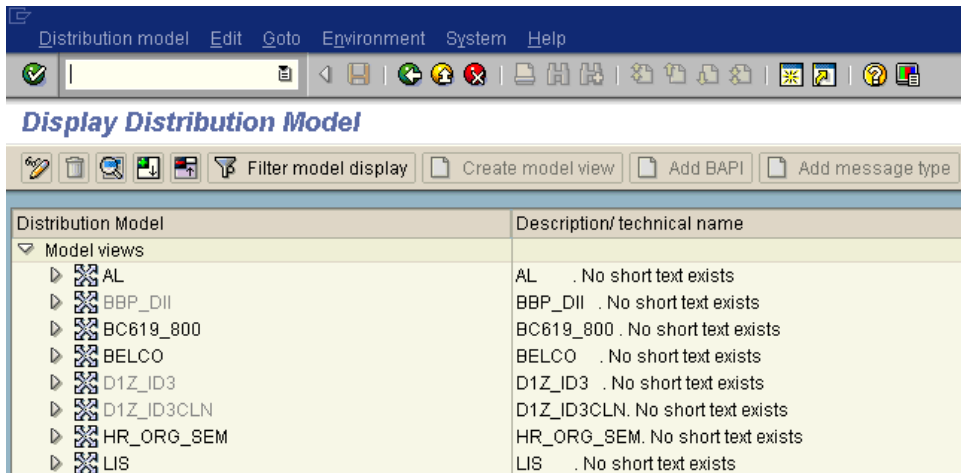


Display IMG

To configure a distribution model:

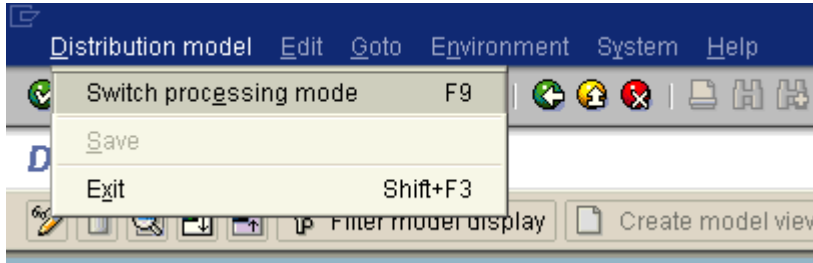
1. Execute the /bd64 transaction.

The Display Distribution Model window opens and displays a list of available distribution models and their descriptions as shown in the following image.



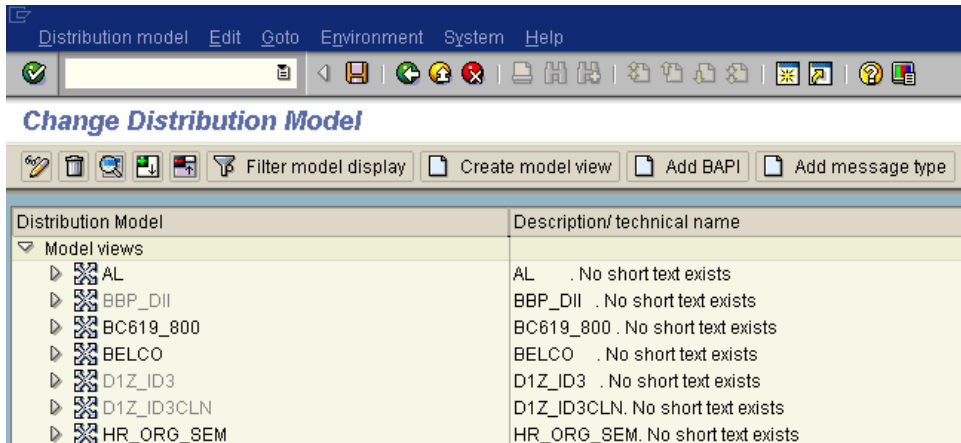
2. In the menu bar, click *Distribution model*.

The Distribution model menu opens as shown in the following image.



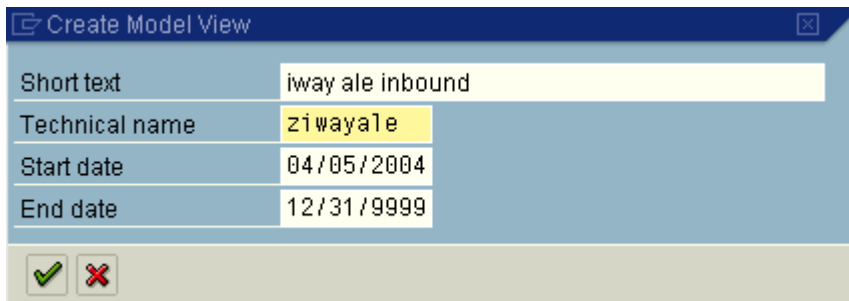
3. Select *Switch processing mode*.

The Display Distribution Model window switches to the Change Distribution Model window as shown in the following image.



4. Click the *Create model view* button.

The Create Model View window opens and includes fields for Start and End dates and fields for the name for your distribution model as shown in the following image.



Configuring a Distribution Model

- a. Type a model view name, for example, iway ale inbound in the Short text field.
 - b. Type a technical name, for example, ziwayale, which also serves as a description as shown in the previous image.
5. To enter the information, click the checkmark icon.

You are returned to the main Change Distribution Model window. The distribution model you configured is now added to the list as shown in the following image.

▶	detlef	DETLEF
▶	iway Distribution Model for alpha class	IWAYMOD09
▶	iway ale inbound	ZIWAYALE
▶	iway marketing distribution model	IWAYMKT

6. Click the *Add message type* button.

The Add Message Type window opens and includes fields to name your message type as shown in the following image.

Model view	ZIWAYALE
Sender	IWAY_IN
Receiver	IWAY_IN
Message type	MATMAS

- a. In the Sender and Receiver fields, enter the logical system you configured, for example, IWAY_IN.
To browse from a list of logical systems, you can click the icon to the right of each field.
 - b. In the Message type field, type the message type you want to use, for example, MATMAS.
To browse from a list of available message types, you can click the icon to the right of the field.
 - c. To enter the information, click the checkmark.
You are returned to the main Change Distribution Model window.
7. Click *Save*.

Defining a Partner Profile

Partner profiles are a prerequisite for data exchange. You define who can exchange messages with the SAP system using which port.

Procedure: How to Define a Partner Profile

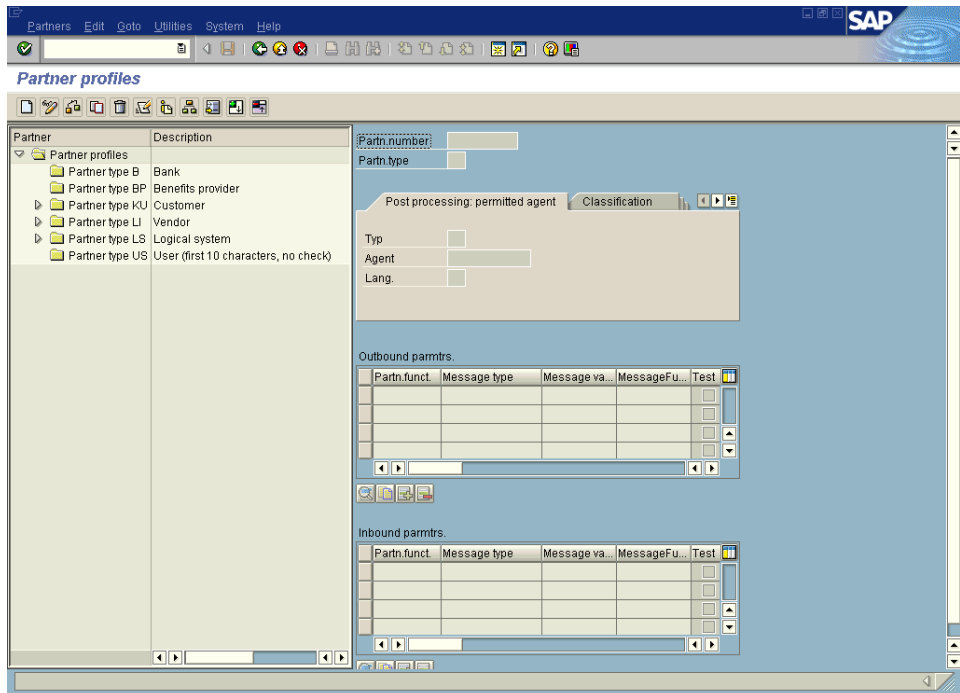
The following image shows the /nwe20 transaction in the field under the menu bar.



To define a partner profile for a specific IDoc:

1. Execute the *we20* transaction.

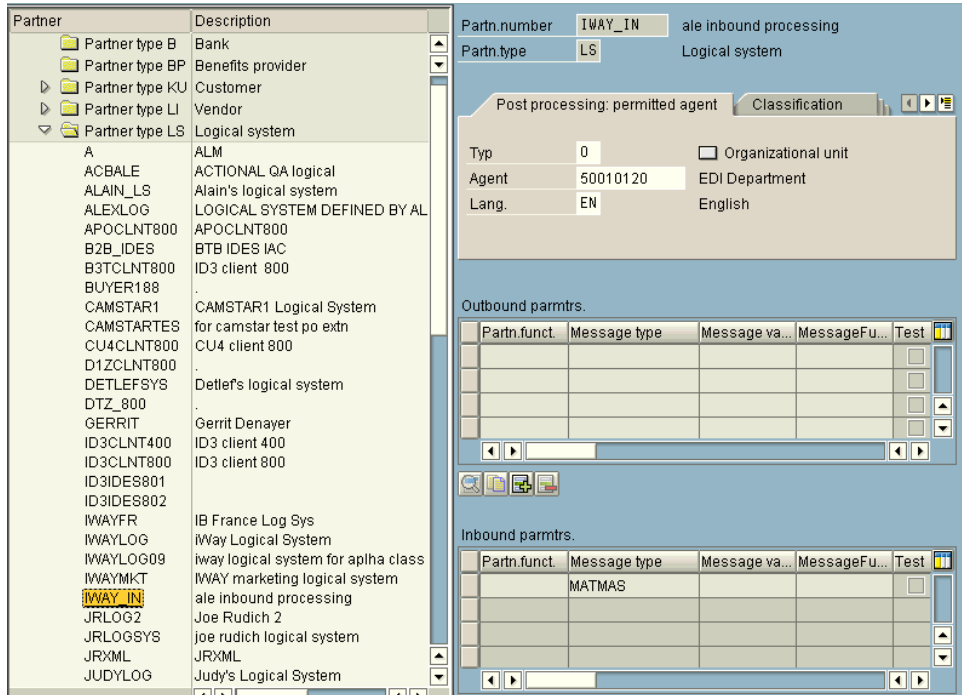
The Partner profiles window opens and displays two panes with information about the logical system as shown in the following image.



2. In the left pane, expand Partner type LS and select the logical system you configured from the list, for example, IWAY_IN.

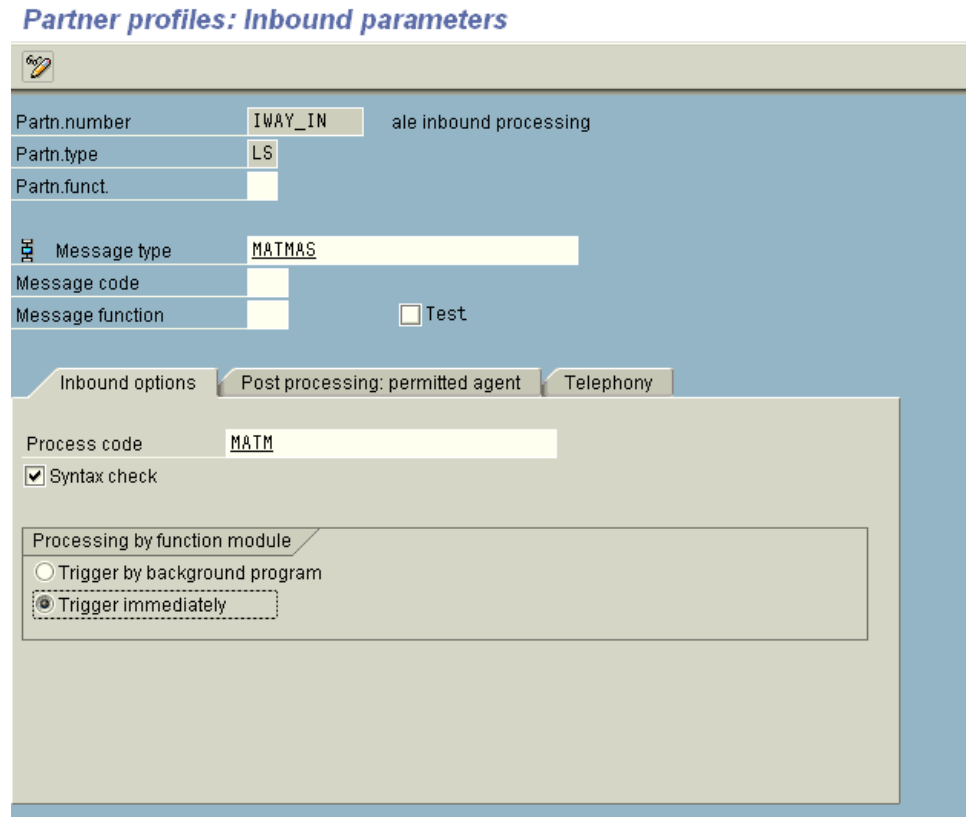
The following image shows the folder expanded in the left pane. The right pane displays the details of the expanded folder including the logical system and type, language, and so forth as shown in the following image.

Note: In the right pane, the Partn.number field refers to the name of the logical system.



3. Click Save.
4. From the Inbound parameters table, click the *Create inbound parameter* icon.

The Partner profiles: Inbound parameters window opens as shown in the following image.



- a. In the Message type field, enter the message type you want to use, for example, MATMAS.

To browse from a list of available message types, you can click the icon to the right of the field.

The Inbound options tab is selected by default.

- b. In the Process code field, enter the process code you want to use, for example, MATM.

To browse from a list of available process codes, you can click the icon to the right of the field.

- c. In the Processing by function module area, select one of the following options:

Trigger by background program. In this case, the iWay Application System Adapter for SAP R/3 writes IDocs to the SAP database, which is processed immediately.

Trigger immediately. In thXML_Prep_Cleanup.fis case, the iWay Application System Adapter for SAP R/3 waits for the SAP system to process IDocs. This can take anywhere from one to fifteen minutes.

- 5. Click *Save*.

CHAPTER 3

Creating XML Schemas for SAP

Topics:

- Overview
- Starting Servlet Application Explorer
- Establishing a Target for SAP
- Viewing Application System Objects
- Creating an XML Schema

The following section describes how to create XML schemas for SAP business objects using Application Explorer as deployed to a Sun Java System Application Server.

The functionality of the Application Explorer is standard despite the deployment type. This section uses the Java™ servlet implementation of Application Explorer to provide graphic examples.

Overview

The iWay Application System Adapter for SAP R/3 enables the processing of SAP BAPIs, RFCs, and IDocs.

External applications that access SAP through the adapter use either XML schemas or Web services to pass data between the external application and the adapter. You can use 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 Servlet Application Explorer, see the *iWay Installation and Configuration* documentation.

SAP must be installed, configured, and available for client access. Application Explorer need not reside on the same system as the application system being accessed, but network access is required.

Starting Servlet Application Explorer

Before you can use iWay Servlet Application Explorer, you must start the Sun Java Application Server. Then, you can open Application Explorer.

Procedure: How to Open Servlet Application Explorer

To open Application Explorer:

1. Enter the following URL in your browser window:

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

where:

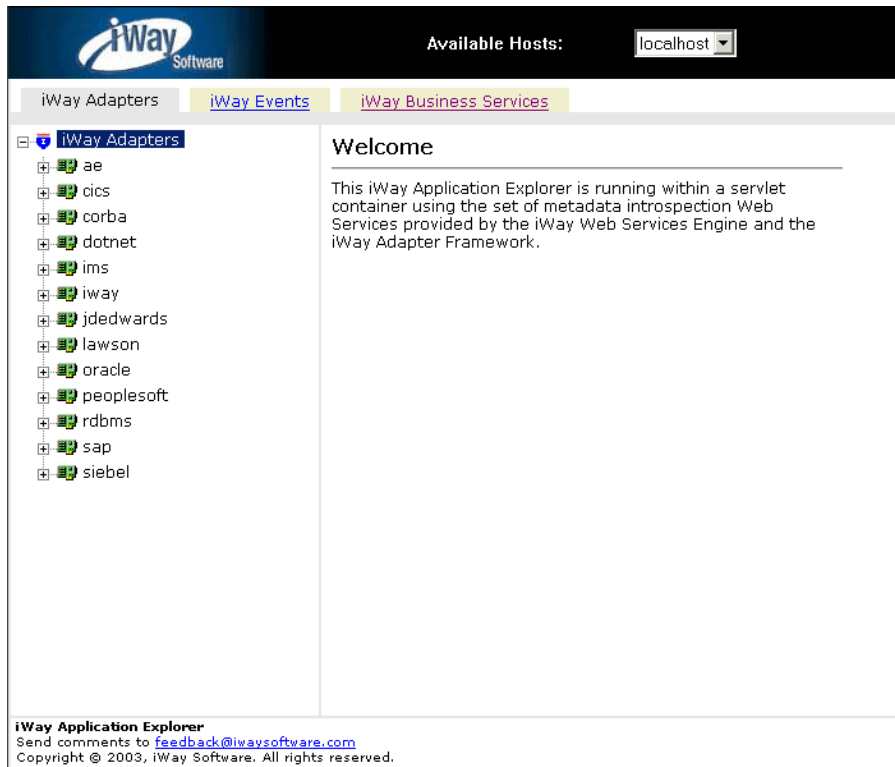
`hostname`

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

`port`

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

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



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

For more information on adding instances, see *iWay Installation and Configuration*.

You are now ready to create new targets for SAP.

Establishing a Target for SAP

To browse SAP business objects, you must create a target for the system you intend to use. The target serves as your connection point and is automatically saved after you create it. You must establish a connection to this system every time you start Application Explorer or after you disconnect from the system.

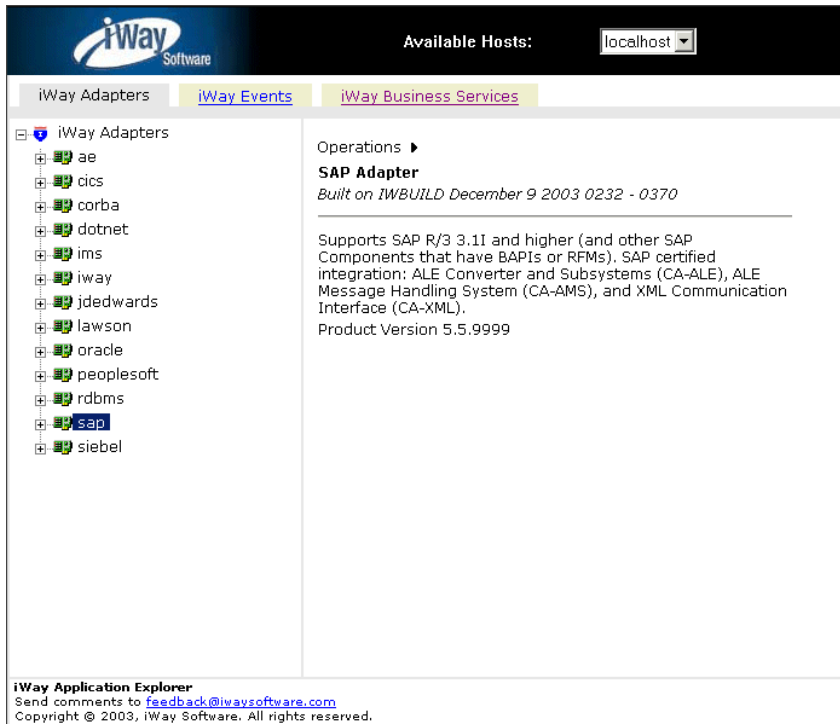
A list of supported application systems appears in the left pane of Application Explorer. The list is based on the adapters that you installed and have licenses to use.

Creating a New Target

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

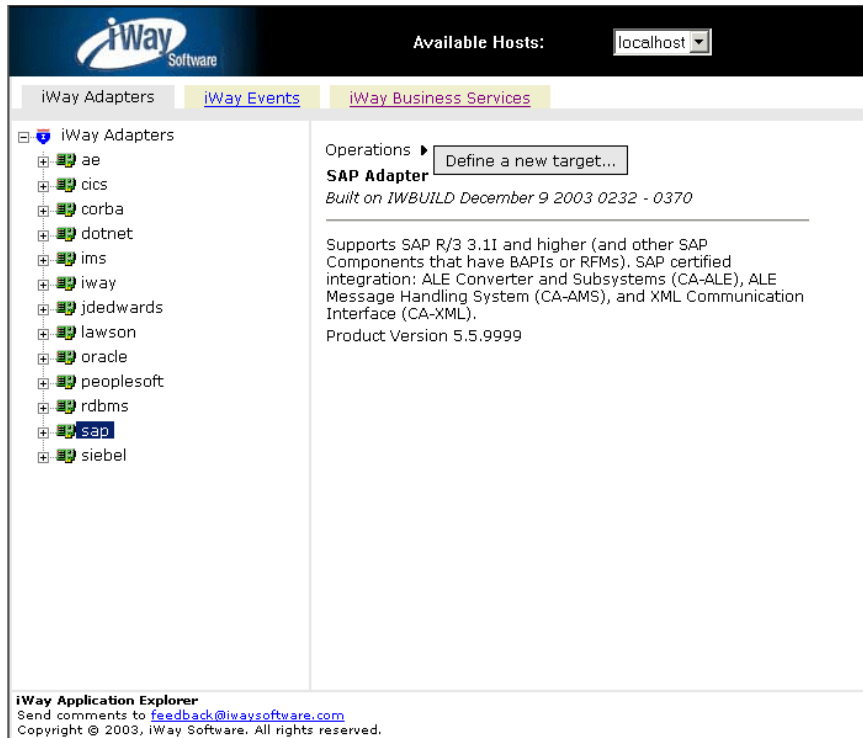
Procedure: How to Create a New Target

The following image shows a window with a navigation pane that lists supported adapters on the left and information about a selected adapter in the right pane.



1. To create a new target: In the left pane, click the *SAP* node. Descriptive information (for example, title and product version) regarding the iWay Application System Adapter for SAP R/3 appears in the right pane.
2. In the right pane, move the pointer over *Operations* and select *Define a new target*.

The following image shows the Define a new target menu option that appears in the right pane, as well as title and product version information for the adapter.



The Add a new SAP target pane opens on the right as shown in the following image.

Add a new SAP target

Targets represent configured connections to instances of backend systems. Choose a name and description for the new target that you wish to create.

Target Name:

Description:

Target Type:

- a. In the Target Name field, type a name for the target, for example, SAPTarget.
 - b. In the Target Description field, type a brief description for the connection.
 - c. From the Target Type drop-down list, select the type of target you are connecting to. The default value is Application Server.
3. Click *Next*.

The Set connection info pane opens on the right. The following tabs are available:

- System (Required)
- User (Required)
- Advanced
- Security

Note: The SAP connection parameters are consistent with those found in your SAP system. For more information on parameter values that are specific to your SAP configuration, consult your SAP system administrator.

The System tab is active as shown in the following image.

Set connection info

The screenshot shows a dialog box titled "Set connection info" with four tabs: "System", "User", "Advanced", and "Security". The "System" tab is selected. Below the tabs are three input fields: "Application Server" with the value "isdsrv2", "System number" with the value "00", and "Edi version" with a dropdown menu showing "3". At the bottom of the dialog are four buttons: "Help", "< Back", "Finish", and "Cancel".

- a. In the Application Server field, type the host name or IP address for the computer that is hosting the SAP application.
- b. In the System Number field, type the system number defined to SAP for client communications.
- c. From the EDI Version drop-down list, select the Electronic Data Interchange (EDI) document version that you are using with the iWay Application System Adapter for SAP R/3. Version 3 is the default value.

4. Click the *User* tab.

The User tab becomes available as shown in the following image.

Set connection info

System **User** Advanced Security

Client:

User:

Password:

Language:

Codepage:

SAP trace:

- a. In the Client field, type the client number defined for the SAP application for client communications.
- b. In the User field, type a valid user ID for the SAP application.
- c. In the Password field, type a valid password for the SAP application.
- d. In the Language field type a language key. EN (English) is the default.
- e. In the Codepage field, type a character code page value.
- f. To enable traces, select the SAP Trace check box.

After you provide information for the System and User tabs, you have completed the basic SAP target configuration. However, you can specify additional parameters in the Advanced and Security tabs.

5. To view the Advanced tab, click *Advanced*.

The Advanced tab contains the following fields:

Connection pool size. This field is used to specify the number of client connections in a pool you want to make available to SAP for Web service calls.

A default connection pool size of 1 is available by default. If you want to use a connection pool size that is greater than 1, enter the value in this field.

Connections to an R3 server take up valuable resources on both the client and the remote server. The section of SAP documentation “Memory Management (BC-CST-MM)” explains in detail the resources required on the SAP system. A user logon also consumes processing time. You can create a pool of connections that will be continually recycled to minimize the resource and time constraints. The size of the connection pool will be determined by the size of your SAP server, available memory, maximum logons and the size of your client Java Virtual Machine. In estimating the size of the pool, you may calculate pool size by the amount of server resources to be consumed, the number and size of the documents to be received, and the size of your Java Virtual Machine.

Connection pool name. Enter a name for your connection pool only if you specified a connection pool size that is greater than 1.

BAPI Exception Handling. If your application is Java centric, you can select *Throws Exception* so that code components may catch the exception and react accordingly. If your application is document based, you can select *Creates Document* to have an XML document created that contains the Java exception. It is up to your application to read the XML document and obtain the error.

Commit with Wait. All SAP Business Objects that change data must commit work to the database. Some BAPI’s developed in version 3.1 of the R/3 system, use an internal commit behavior and their commit behavior cannot be changed by the adapter. As soon as they are called, they commit the work they have done. BAPI’s developed since release 3.1 use the external commit method. The adapter issues a commit command, and the commit is put in the database que. If there is some application error in the first part of the commit, the error message “Posting could not be carried out” will be returned, and the adapter will roll back the transaction. If in writing to the database a database error occurs, a short dump will be issued in the database records of SAP, but no message will be returned to the adapter about the failure. If a high degree of accuracy is needed in your application, select “With Wait”, and the adapter will wait until all records have been physically written to the database before returning from the function call. The “Commit With Wait” has a definite performance impact on your adapter performance, use it with deliberation. The commit behavior of BAPI’s is found in SAP documentation under “Bapi Programming Guide and Reference (CA-BFA)”

This option is disabled by default.

6. To view the Security tab, click *Security*.

The Security tab contains the following fields:

Logon ticket (SSO2). If you are using a Secure Network Communications (SNC) adapter with SAP, enter the name of the SSO2 logon ticket you are using.

Logon ticket (X509). If you are using an SNC adapter with SAP, enter the name of the X.509 logon ticket you are using.

Note: Depending on the SAP system release, logins using Single-Sign-On (SSO) or X.509 certificates are being supported.

For SSO specify the user to be \$MYSAPSSO2\$ and pass the base64 encoded ticket as the passwd parameter.

For X509 specify the user to be \$X509CERT\$ and pass the base64 encoded certificate as the passwd parameter

For more information regarding SSO or X.509 configuration, see your SAP system documentation.

SNC mode. By default, SNC is disabled. To enable SNC, select 1 from the drop-down list.

SNC partner. Enter the name of the RFC server or message server (load balancing) that provides the SNC services.

SNC level. Select the version of the SNC library from the drop-down list.

SNC name. Enter the name of the SNC library you are using.

SNC library path. Enter the path to the SNC library.

SNC provides protection for the communication links between the distributed components of an R/3 System. Using SNC, SAP R/3 can support products which adhere to the GSS-API Version 2 standard. SNC supports application level (end-to-end security), Smartcard authentication, and single sign-on.

7. After you provide all the required information for your target, click *Finish*.

The SAP target (SAPTarget) appears below the sap node in the left pane as shown in the following image.



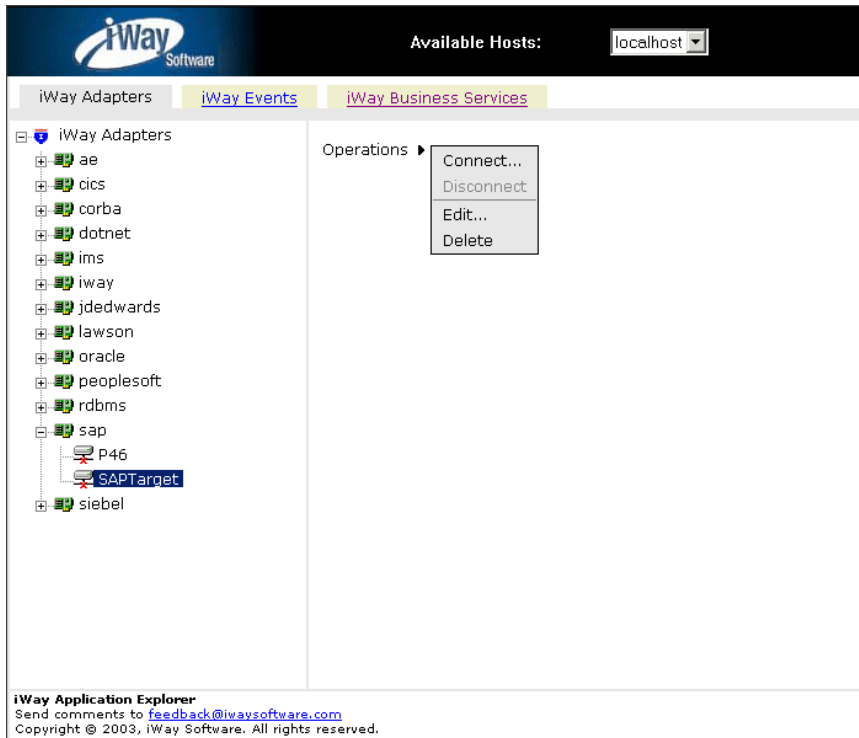
You are now ready to connect to your SAP target.

Connecting to a Target

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

Procedure: How to Connect to a Target

The following image shows the SAP target selected in the left pane. In the right pane, the Operations menu appears in its expanded form.



To connect to a target:

1. In the left pane, expand the sap node and select the target you defined, for example, SAPTarget.
2. In the right pane, move the pointer over *Operations* and select *Connect* from the menu that pops up.

The following image shows the Connect to SAPTarget pane that opens on the right, with fields for entering a client, a user, a password, a language, and a code page. Additionally, you can select to enable an SAP trace.

Connect to SAPTarget

System User Advanced Security

Client:

User:

Password:

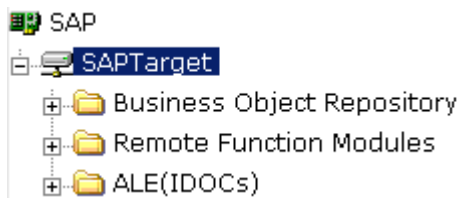
Language:

Codepage:

SAP trace:

3. Enter a valid password and click *OK*.

The SAPTarget node in the left pane changes to reflect that a connection was made as shown in the following image.



4. Expand the *SAPTarget* node.

The following SAP business objects appear.

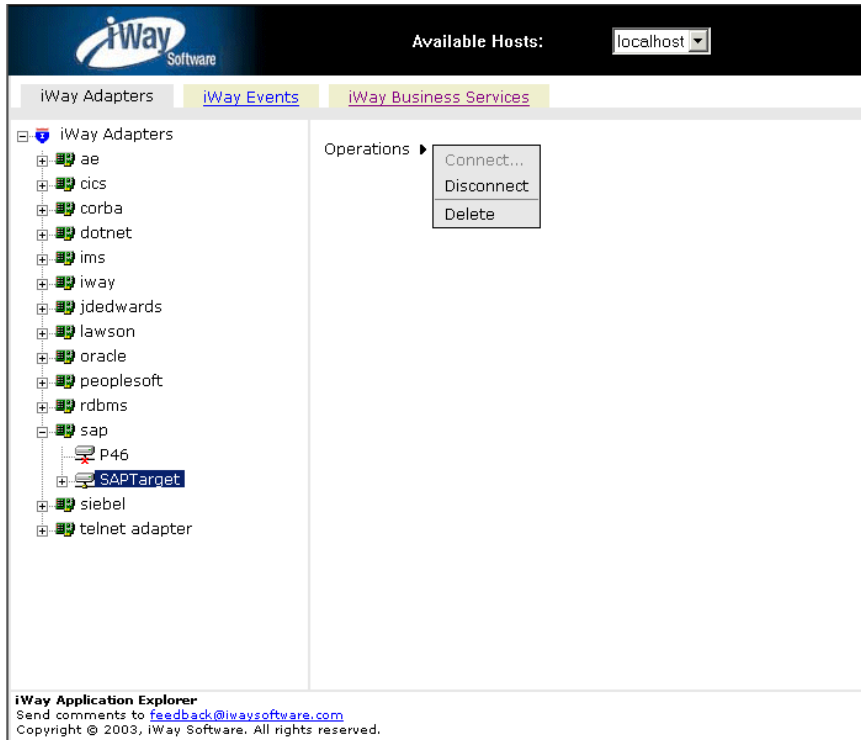
- Business Object Repository
- Remote Function Modules
- ALE (IDocs)

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

The following image shows the SAP target selected in the left pane. In the right pane, the Operations menu appears in its expanded form.



To disconnect from a target:

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

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

In the left pane, the SAPTarget node changes to reflect that a connection was closed as shown in the following image.

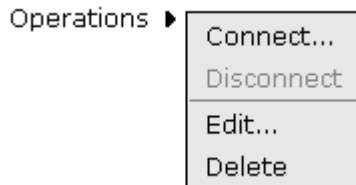


Modifying a Target

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

Procedure: How to Edit a Target

The following image shows the Operations menu with options to Connect to, Disconnect from, Edit, or Delete a target.



To edit a target:

1. In the left pane, click the target, for example, SAPTarget.
2. In the right pane, move the pointer over *Operations* and select *Edit*.

The Edit pane opens on the right with fields for a target name, a description, and a target type.

Edit SAP target SAPTarget

Targets represent configured connections to instances of backend systems. Choose a name and description for the new target that you wish to create.

Target Name:

Description:

Target Type:



3. Modify the connection information and then, click *Next* to continue modifying additional information, if required.
4. When you are finished making 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

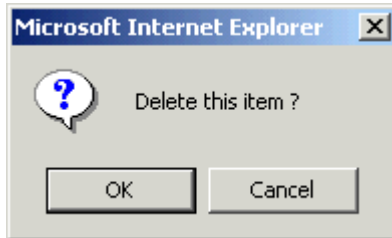
The following image shows the Operations menu with options to Connect to, Disconnect from, Edit, or Delete a target.



To delete a target:

1. In the left pane, click the target, for example, SAPTarget.
2. In the right pane, move the pointer over *Operations* and select *Delete*.

A confirmation dialog box opens, asking if you want to delete the target as shown in the following image.



3. To delete the target you selected, click *OK*.

The SAPTarget node disappears from the left pane.

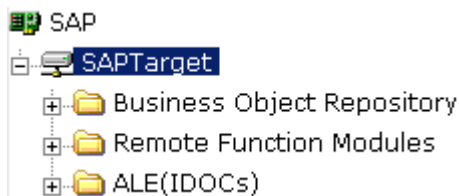
Viewing Application System Objects

After you are connected to SAP, Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view SAP BAPI, RFC, and iDOC metadata stored in the SAP Business Object repository.

Note: Depending on the release or service pack installed, certain RFCs, for example, RFC_CUSTOMER_GET, may not exist in your particular SAP system. Therefore, the examples included in this documentation may not be relevant to your system. If this is the case, you should use the examples as a general reference for adapter functionality and choose an RFC that exists within your SAP application environment.

Procedure: How to View Application System Objects

The following image shows the SAP target selected and expanded to display the available application system objects.

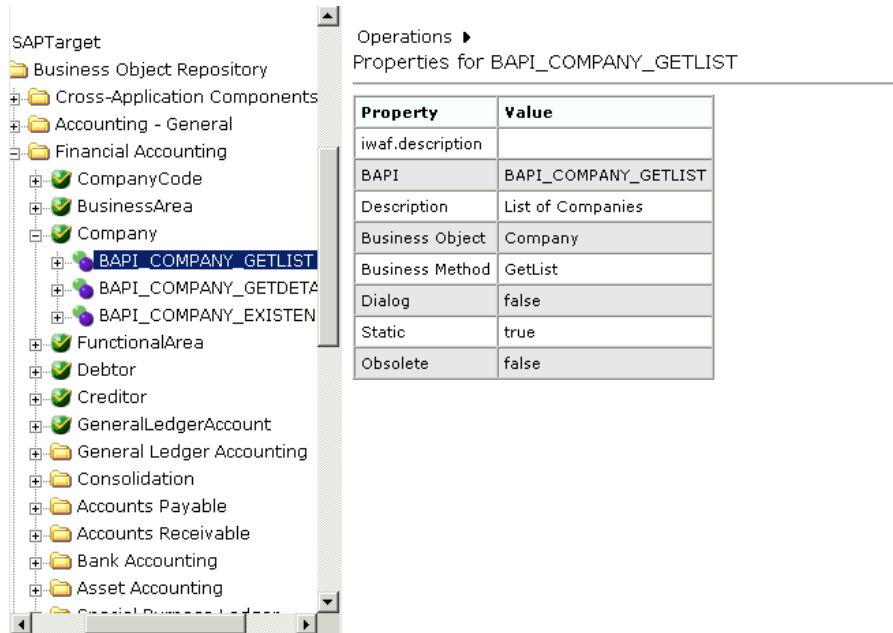


To view application system objects:

1. Click the icon to the left of the target name, for example, SAPTarget.
This expands the target to expose the available system objects.
2. To expand the desired SAP repository node, click the icon to the left of the repository name, for example, Business Object Repository.

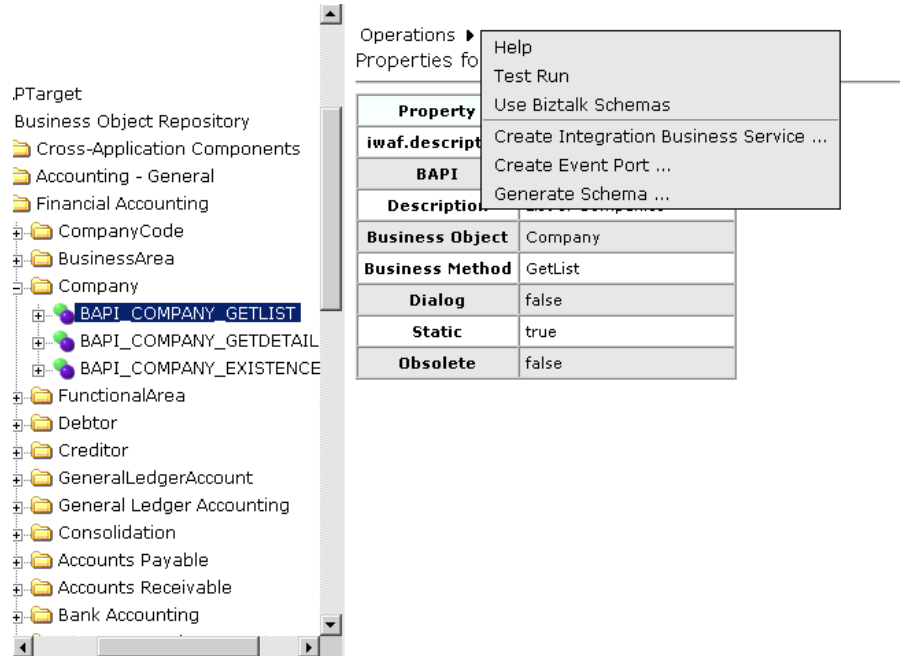
3. Click the icon to the left of the *Financial Accounting* group.

A list of business objects related to Financial Accounting appears in the left pane as shown in the following image.



- a. Scroll down and click the icon to the left of the *Company* business object.
- b. Click the icon to the left of the BAPI method named *BAPI_COMPANY_GETLIST*.

The following image shows the Operations menu expanded in the right pane over a table listing properties and values for the BAPI method named BAPI_COMPANY_GETLIST.



4. In the right pane, move the pointer over *Operations* to view the context menu.

The following options are available from the context menu:

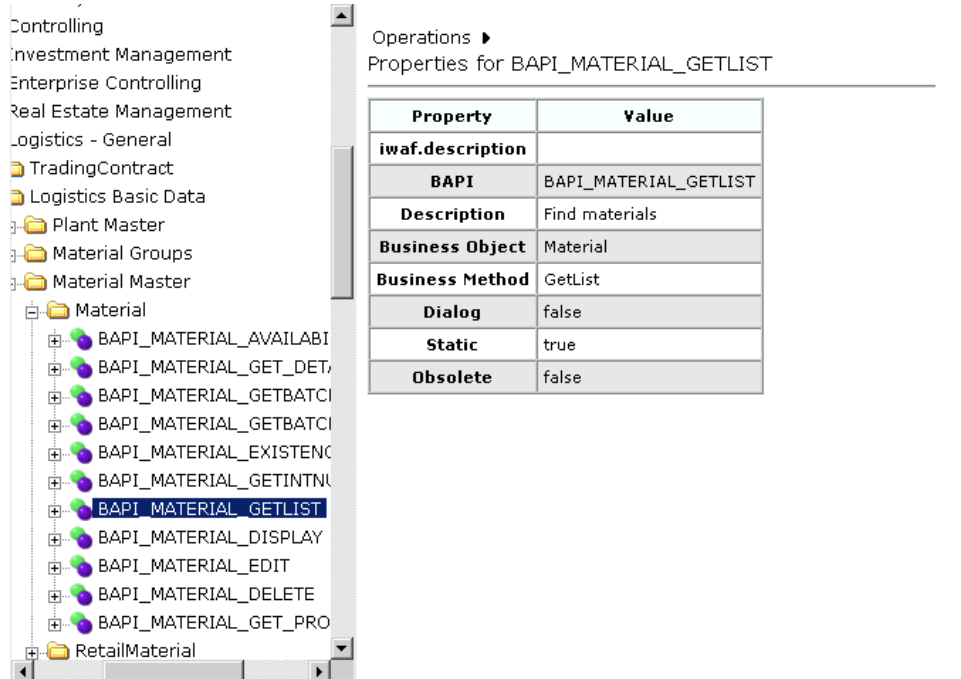
- Help
- Test Run
- Use Biztalk Schemas
- Create iWay Business Services
- Create Event Port
- Generate Schema

Creating an XML Schema

After you browse the SAP business object repository, you can generate XML request and response schemas for the object you wish to use with your adapter.

Procedure: How to Create XML Schemas

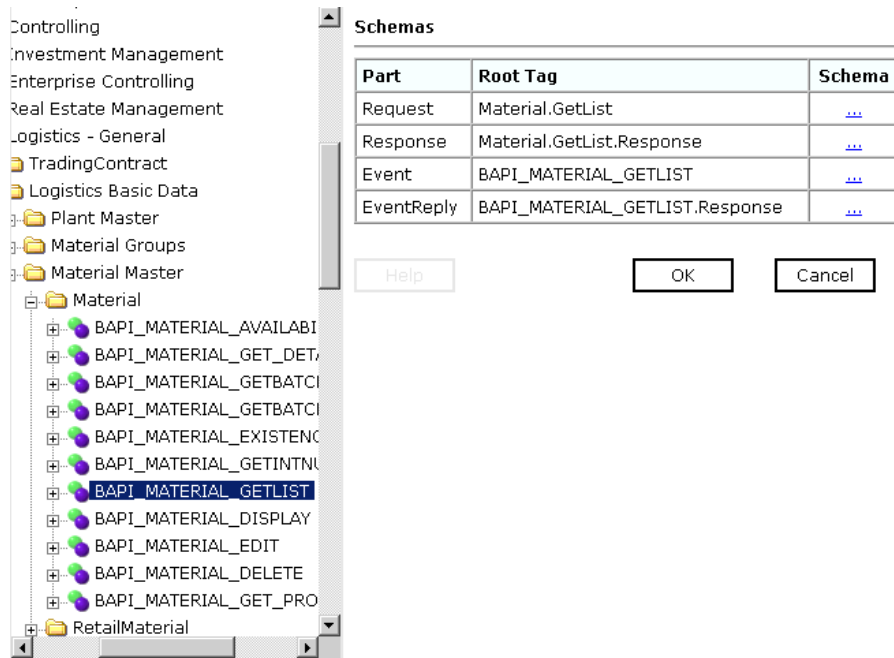
The following image shows the Operations menu expanded in the right pane over a table listing properties and values for the BAPI method named BAPI_MATERIAL_GETLIST.



To create XML request and response schemas for the SAP BAPI method called BAPI_MATERIAL_GETLIST:

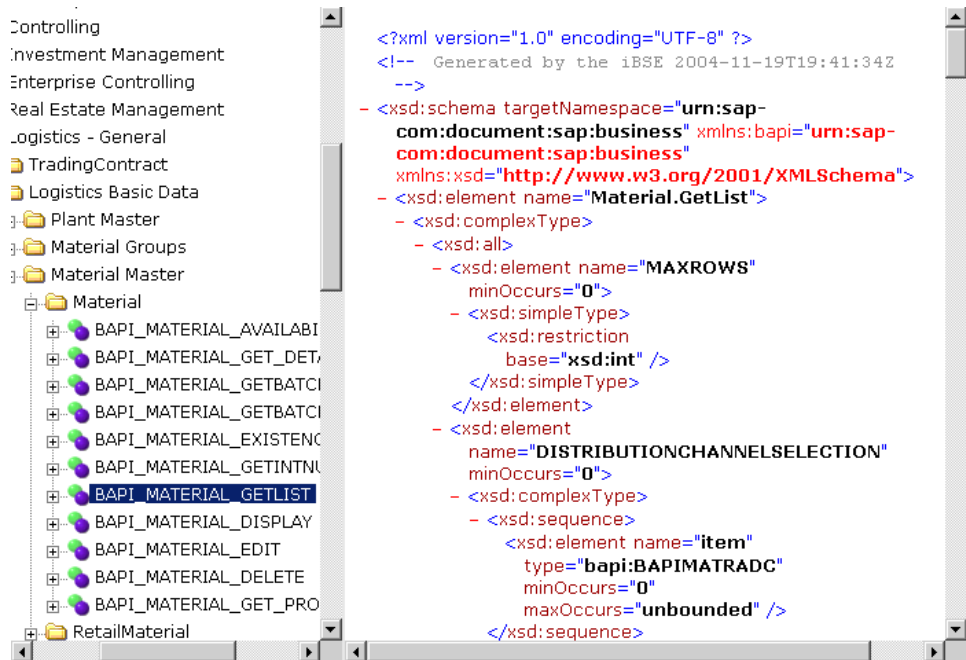
1. Select the *BAPI_MATERIAL_GETLIST* method in the Business Object Repository.
2. In the right pane, move the pointer over *Operations* and select *Generate Schema*. Request, response, and event schemas are created for your business object.

The following image shows the Schemas pane that opens on the right. A table defines the root tag for each schema and provides hyperlinks.



3. Click the hyperlink associated with the type of schema you want to view.

For example, if you click the Request schema, the schema appears in the right pane as shown in the following image.



4. Click the *Back* button on your Web browser to return to the previous window.

After the schemas are created, you can create iWay Business Services. For more information, see Chapter 4, *Creating and Publishing iWay Business Services*.

You also can create events after the schemas are created. For more information, see Chapter 5, *Configuring the Event Adapter for SAP*.

CHAPTER 4

Creating and Publishing iWay Business Services

Topics:

- Understanding iWay Business Services
- Creating iWay Business Services

This section describes how to create and publish iWay Business Services using iWay Application Explorer.

The functionality of Application Explorer is standard despite the deployment type. This section uses the Java™ servlet implementation of Application Explorer to provide graphic examples.

Understanding iWay Business Services

iWay Servlet Application Explorer provides Web developers with a simple, consistent mechanism for extending the capabilities of the iWay Application System Adapter for SAP R/3. The iWay Business Services Engine (iBSE) exposes functionality as Web services. It serves as a gateway to heterogeneous back-end applications and databases.

A Web service is a self-contained, modularized function that you can publish and access 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.

After you browse the SAP business object repository and create an XML schema for the object, you can generate an iWay Business Services for the object you wish to use with your adapter.

Creating iWay Business Services

The following topics describe how to create iWay Business Services, including how to generate WSDL (Web Services Description Language) from a Web service.

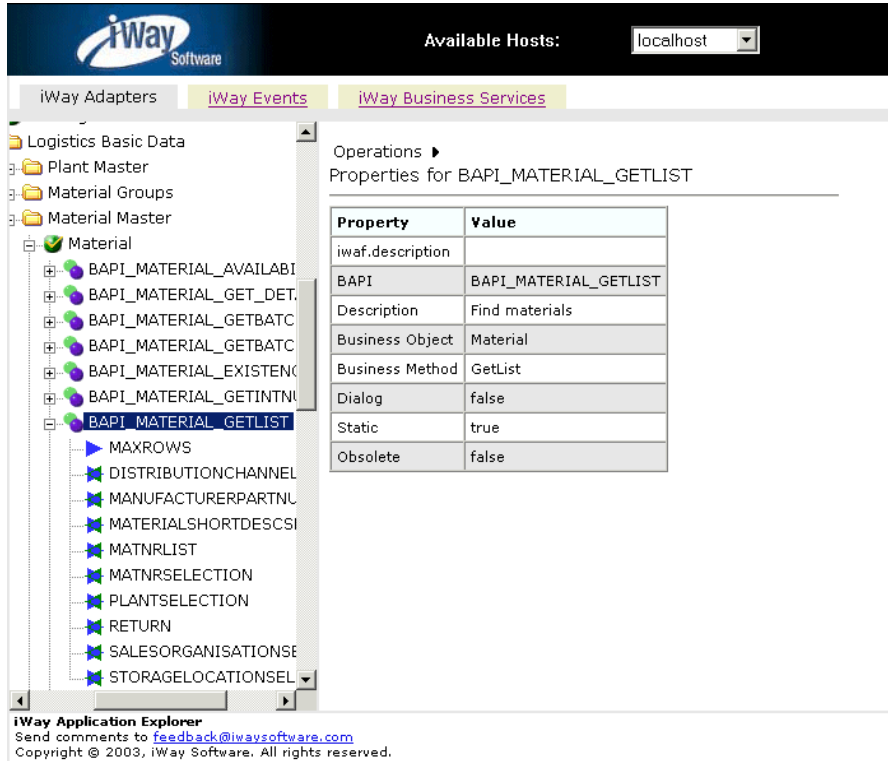
Creating Business Services With Application Explorer

The following procedure describes how to create iWay Business Services using iWay Servlet Application Explorer. The procedure uses the SAP BAPI method called BAPI_MATERIAL_GETLIST as an example and returns a list of materials.

Note: If you want your Web service to use connection pooling, you must specify connection pooling information when connecting or reconnecting to your SAP target.

Procedure: How to Create an iWay Business Services

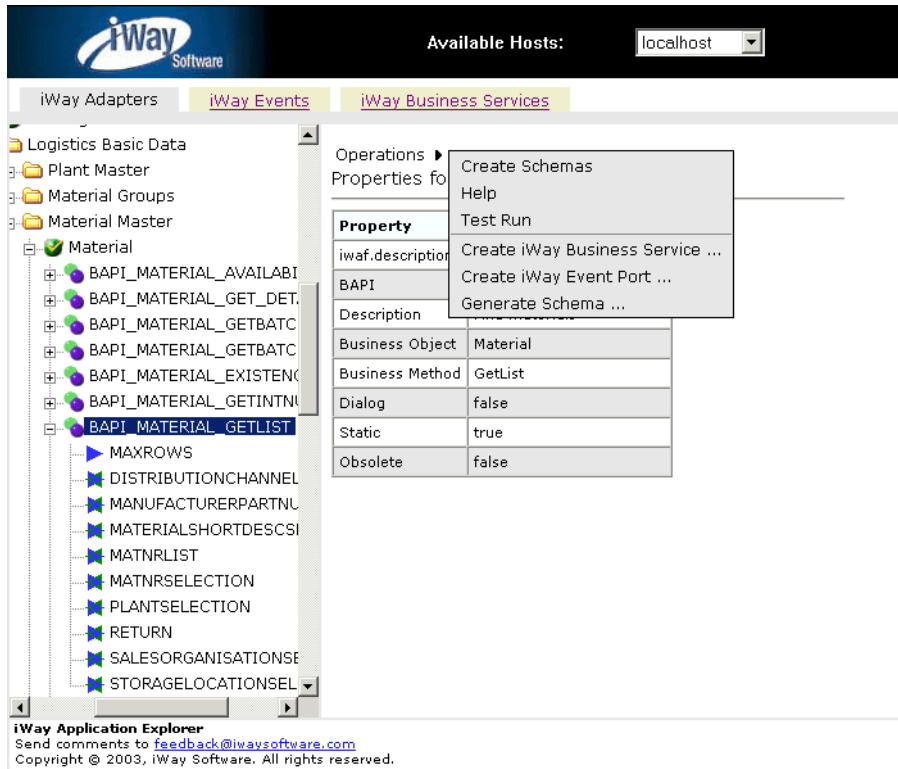
The following image shows the closed Operations menu in the right pane and a table listing properties and values for the BAPI method named BAPI_MATERIAL_GETLIST.



To create an iWay Business Services:

1. From the Business Object Repository, select the *BAPI_MATERIAL_GETLIST* method.
2. In the right pane, move the pointer over *Operations*.

The Operations menu expands as shown in the following image.



3. Select *Create iWay Business Services*.

The Create Web Service for BAPI_MATERIAL_GETLIST pane opens on the right with options to create a new service or use an existing service.

Create Web Service for BAPI_MATERIAL_GETLIST

- Create a new service
- Use an existing service

Help < Back Next > Cancel

4. Select *Create a new service* and click *Next*.

The pane opens where you provide the specific information for the iWay Business Service you are defining as shown in the following image.

Create Web Service for BAPI_MATERIAL_GETLIST

Service Name:

Description:

License:

- a. In the Service Name field, type a name for the iWay Business Service.
 - b. In the Description field, type a brief description.
 - c. In the License field, select the license definition you want to use.
5. Click *Next*.

A second Create Web Service pane opens on the right that includes fields for the method name and a description as shown in the following image.

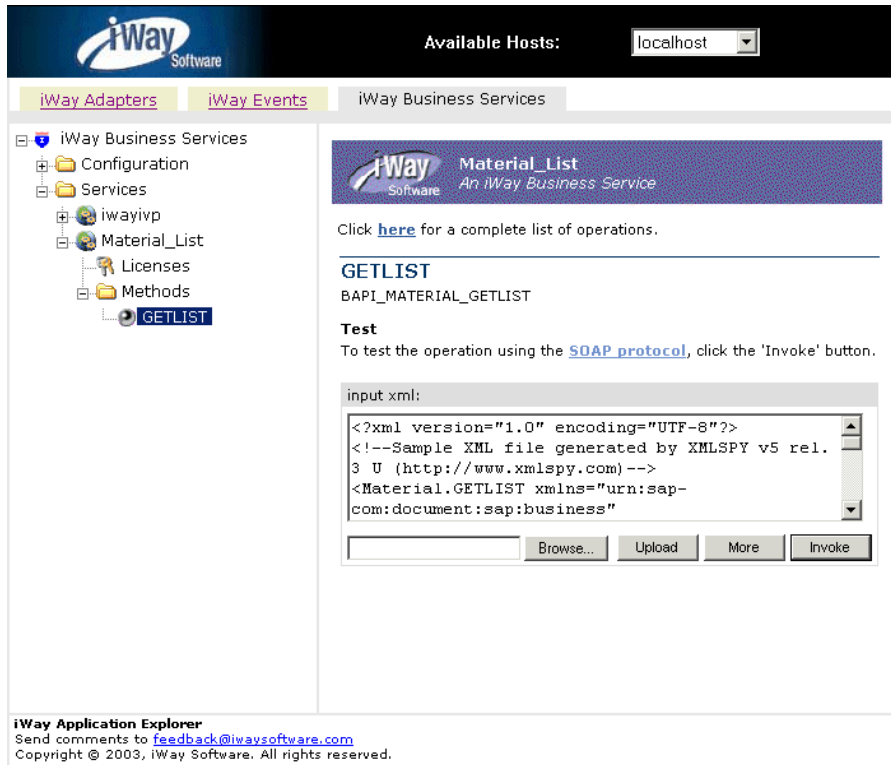
Create Web Service for BAPI_MATERIAL_GETLIST

Method Name:

Description:

- a. In the Method Name field, type a descriptive name for the method.
 - b. In the Description field, type a brief description for the method.
6. Click *Finish*.

The iWay Business Services Engine tab becomes available.



All of the available services that were created appear in the left pane. The Material_List service node is expanded, and the GETLIST method is automatically selected.

The test window for the GETLIST method opens in the right pane.

7. Enter an XML instance of the schema you generated previously for the SAP business component.

The document queries the service in the input xml field.

To use the identical sample input XML illustrated in this example, see *Sample iWay Business Services Input XML* on page 4-9.

8. Click *Invoke*.

The result appears in the right pane, as shown in the following image.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <SOAP-ENV:Envelope xmlns:xsd="http://www.w3.org/2001/
  ENV="http://schemas.xmlsoap.org/soap/envelope/" xmr
- <SOAP-ENV:Body>
  - <GETLISTResponse xmlns="urn:iwaysoftware:ibse:jul200:
    - <Material.GETLIST.Response xmlns:SOAP-ENV="http://s
      com:document:sap:business" schemaLocation="urn:
        \BEAAPP~1\sessions\default\SAP\beasap46\ser
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-i
        <DISTRIBUTIONCHANNELSELECTION xmlns="" />
        <MANUFACTURERPARTNUMB xmlns="" />
      - <MATERIALSHORTDESCSEL xmlns="">
        - <item>
          <SIGN>E</SIGN>
          <OPTION>CP</OPTION>
          <DESCR_LOW>*</DESCR_LOW>
          <DESCR_HIGH />
        </item>
      </MATERIALSHORTDESCSEL>
    - <MATNRLIST xmlns="">
      - <item>
        <MATERIAL>0000000000000000038</MATERIAL>
        <MATL_DESC>Classification test</MATL_DESC>
        <MATERIAL_EXTERNAL />
        <MATERIAL_GUID />
        <MATERIAL_VERSION />
      </item>
    </MATNRLIST>
  </GETLISTResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Example: Sample iWay Business Services Input XML

The following input XML retrieves a list of materials using the SAP BAPI_MATERIAL_GETLIST method.

```

    <?xml version="1.0" encoding="UTF-8" ?>
- <!-- Sample XML file generated by XMLSPY v5 rel. 3 U
(http://www.xmlspy.com)
-->
- <Material.GETLIST xmlns="urn:sap-com:document:sap:business"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:sap-com:document:sap:business
C:\temp\service_BAPI_MATERIAL_GETLIST.xsd">
    <MAXROWS>1000</MAXROWS>
- <DISTRIBUTIONCHANNELSELECTION>
- <item>
    <SIGN />
    <OPTION />
    <DISTR_CHAN_LOW />
    <DISTR_CHAN_HIGH />
    </item>
</DISTRIBUTIONCHANNELSELECTION>
- <MANUFACTURERPARTNUMB>
- <item>
    <MANU_MAT />
    <MFR_NO />
    </item>
</MANUFACTURERPARTNUMB>
- <MATERIALSHORTDESCSEL>
- <item>
    <SIGN />
    <OPTION />
    <DESCR_LOW />
    <DESCR_HIGH />
    </item>
</MATERIALSHORTDESCSEL>
- <MATNRLIST>
- <item>
    <MATERIAL />
    <MATL_DESC />
    <MATERIAL_EXTERNAL />
    <MATERIAL_GUID />
    <MATERIAL_VERSION />
    </item>
</MATNRLIST>
- <MATNRSELECTION>
- <item>
    <SIGN>E</SIGN>
    <OPTION>BT</OPTION>

```

```
<MATNR_LOW>1000</MATNR_LOW>
<MATNR_HIGH>1010</MATNR_HIGH>
</item>
</MATNRSELECTION>
- <PLANTSELECTION>
- <item>
  <SIGN />
  <OPTION />
  <PLANT_LOW />
  <PLANT_HIGH />
</item>
</PLANTSELECTION>
- <RETURN>
- <item>
  <TYPE />
  <ID />
  <NUMBER />
  <MESSAGE />
  <LOG_NO />
  <LOG_MSG_NO />
  <MESSAGE_V1 />
  <MESSAGE_V2 />
  <MESSAGE_V3 />
  <MESSAGE_V4 />
  <PARAMETER />
  <ROW>0</ROW>
  <FIELD />
  <SYSTEM />
</item>
</RETURN>
- <SALESORGANISATIONSELECTION>
- <item>
  <SIGN />
  <OPTION />
  <SALESORG_LOW />
  <SALESORG_HIGH />
</item>
</SALESORGANISATIONSELECTION>
- <STORAGELOCATIONSELECT>
- <item>
  <SIGN />
  <OPTION />
  <STLOC_LOW />
  <STLOC_HIGH />
</item>
</STORAGELOCATIONSELECT>
</Material.GETLIST>
```


Identity Propagation

If you test or execute a Web service using a third-party XML editor, for example, XMLSPY, the user name and password values that you specify in the SOAP header must be valid and are used to connect to SAP. The user name and password values that you provided for SAP when you created a target using Application Explorer are overwritten for this Web service request.

The following is a sample SOAP header that is included in the WSDL file for a Web service:

```
<SOAP-ENV:Header>
  <m:ibsinfo xmlns:m="urn:schemas-iwaysoftware-com:iwse">
    <m:service>String</m:service>
    <m:method>String</m:method>
    <m:license>String</m:license>
    <m:disposition>String</m:disposition>
    <m:Username>String</m:Username>
    <m>Password>String</m>Password>
    <m:language>String</m:language>
  </m:ibsinfo>
</SOAP-ENV:Header>
```

Note: You can remove the following tags from the SOAP header, since they are not required:

```
<m:disposition>String</m:disposition>
<m:language>String</m:language>
```

CHAPTER 5

Configuring the Event Adapter for SAP

Topics:

- Understanding Event Functionality
- Adding, Modifying, or Deleting a Port
- Adding, Modifying, or Deleting a Channel
- Synchronous Event Processing

This section describes how to use Application Explorer to connect to SAP and listen for events as deployed to a Sun Java System Application Server.

The functionality of Application Explorer is standard despite the deployment type. This section uses the Java™ servlet implementation of Application Explorer to provide graphic examples.

Understanding Event Functionality

Events are generated as a result of activity on an application system. You can use events to trigger an action in your application.

Applications or functions within SAP may broadcast processing information at predefined points, called events. You must configure an event listener if you are to receive events from SAP. For example, the SAP Business Object "Material" may raise the event status "Material.Assigned" when a material has been created. If you wish to consume this event, you must configure an event listener to capture this event within SAP and transmit the event notification to your system.

To create an event, you must create a port and a channel using Application Explorer.

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

The following is a description of how ports and channels work:

- 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 *Adding, Modifying, or Deleting a Port on page 5-2*.

- 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 *Adding, Modifying, or Deleting a Channel on page 5-18*.

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

Adding, Modifying, or Deleting a Port

The following procedures describe how to create an event port using Servlet Application Explorer. You can create a port for for an SAP business function from the iWay Adapters tab or from the iWay Events tab.

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

When you use the Application Explorer with an iWay Business Services Engine (iBSE) implementation, the following port dispositions are available:

- File
- iBSE
- MSMQ
- JMS queue
- SOAP
- HTTP
- MQSeries
- MAIL

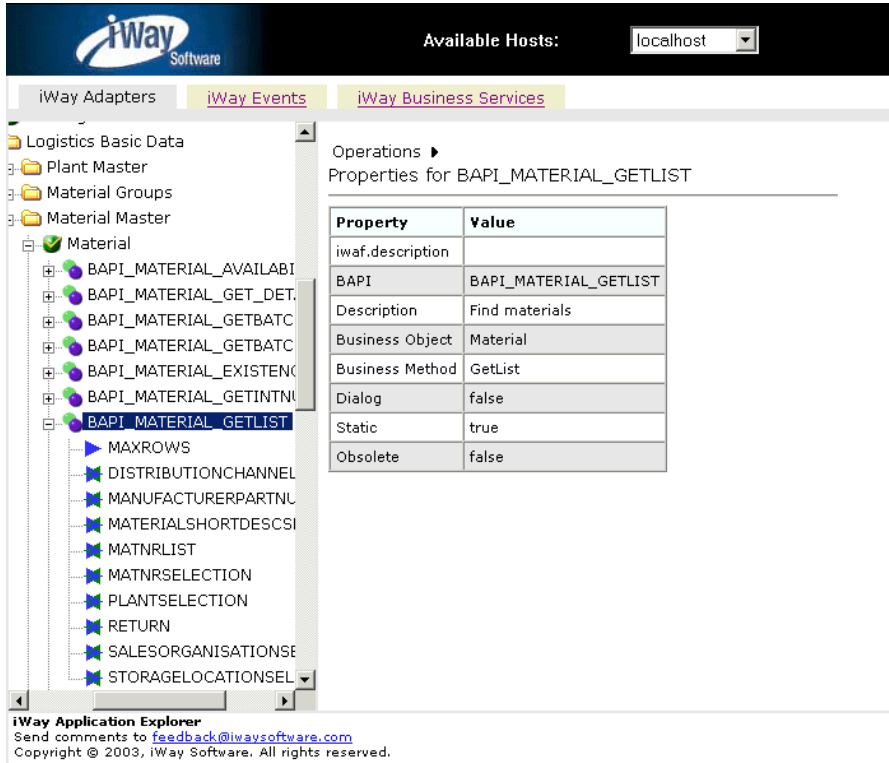
Note: The MAIL disposition option will be supported in a future release.

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

- File
- HTTP
- JMS queue
- MQSeries

Procedure: How to Create a Port for the File Disposition

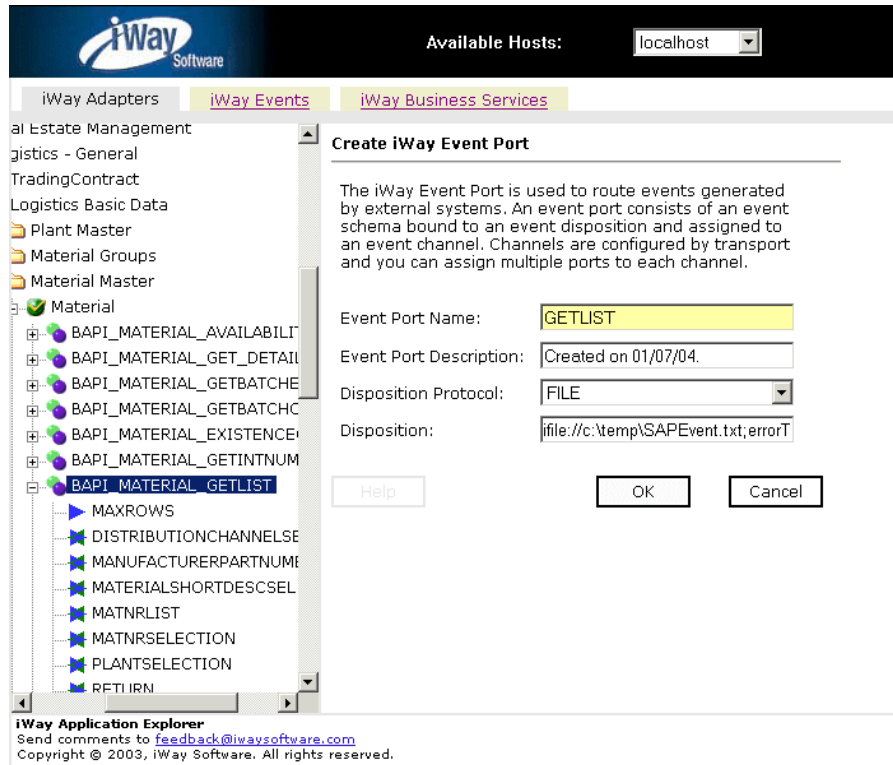
The following image shows the closed Operations menu in the right pane and a table listing properties and values for the BAPI method named BAPI_MATERIAL_GETLIST.



To create a port for the File disposition using Application Explorer:

1. Click the *iWayAdapters* tab.
2. Select the *BAPI_MATERIAL_GETLIST* method from the Business Object Repository.
3. In the right pane, move the pointer over *Operations* and select *Create iWay Event Port*.

The Create iWay Event Port pane opens on the right as shown in the following image.



- a. In the Event Port Name field, type a name.
 - b. In the Event Port Description field, type a brief description.
 - c. From the Disposition Protocol drop-down list, select *FILE*.
4. Click Next.

The Specify FILE Disposition pane opens on the right as shown in the following image and includes information about the File disposition.

Specify FILE Disposition

Disposition type File uses an iWay file url to specify the destination filename or directory in which the event document is stored. During run-time, the destination file name may need to be indexed to avoid overwriting. It supports an optional errorTo port or url.

Disposition Url:



5. In the Disposition Url field, type a destination where the event data is written.

When pointing Application Explorer to an **IBSE** deployment, specify the destination file using the following format:

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

When pointing Application Explorer to a **JCA** deployment, specify the destination file using the following format:

```
location
```

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

The following table lists and describes the disposition parameters for File.

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

For example:

```
ifile://c:\temp\SAPEvent.txt;errorTo=ifile://c:\temp\error
```


6. Click *Finish*.

The iWay Events tab becomes available. The event port appears under the ports node in the left pane. In the right pane, a table summarizes all the information associated with the port you created.

The screenshot shows the iWay Application Explorer interface. At the top, there is a header with the iWay Software logo and an 'Available Hosts' dropdown menu set to 'localhost'. Below the header are three tabs: 'iWay Adapters', 'iWay Events', and 'iWay Business Services'. The 'iWay Events' tab is active.

The left pane displays a tree view under 'iWay Event Adapters'. The tree structure is as follows:

- iWay Event Adapters
 - cics
 - dotnet
 - lawson
 - oracle
 - peoplesoft
 - rdbms
 - sap
 - channels
 - ports
 - default
 - GETLIST (highlighted)
 - TEST_EVENT
 - siebel

The right pane shows the details for the selected 'GETLIST' port under the 'Operations' section:

Port Name	GETLIST
Description	Created on 01/07/04.
Disposition	ifile://c:\temp\SAPEvent.txt,errorTo=ifile://c:\temp\error
Target	sap
SchemaLink	...

At the bottom of the interface, there is a footer for 'iWay Application Explorer' with contact information and a copyright notice: 'Send comments to feedback@iwaysoftware.com. Copyright © 2003, iWay Software. All rights reserved.'

7. To view the event schema that was created for the event port, click *SchemaLink*.

You are now ready to associate the event port for File with a channel. For more information, see *Adding, Modifying, or Deleting a Channel* on page 5-18.

Procedure: How to Create a Port for the iBSE Disposition

The iBSE disposition allows an event to launch an iWay Business Services method. To create a port for an iBSE disposition using Application Explorer:

1. Click the *iWay Events* tab.
2. In the left pane, expand the *SAP* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 *IBSE*.
- c. In the Disposition field, enter an iBSE destination using the following format:

`ibse:svcName.mthName; [responseTo=responseTo]; [errorTo=errorDest]`

The following table lists and describes the disposition parameters for iBSE.

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

Procedure: How to Create a Port for the MSMQ Disposition

The MSMQ disposition supports public and private queues. To create a port for an MSMQ disposition using Application Explorer:

1. Click the *iWay Events* tab.
2. In the left pane, expand the *SAP* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 *MSMQ*.
- c. In the Disposition field, enter an MSMQ destination in the format:

`msmq://host/private$/qName; [errorTo=errorDest]`

The following table lists and defines the disposition parameters for MSMQ.

Parameter	Description
host	Name of the host on which the Microsoft Queuing system runs.
queueType	Type of queue. For private queues, enter <i>Private\$</i> . 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.
qName	Name of the queue in which messages are placed.
errorTo	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.

Procedure: How to Create a Port for the JMS Queue Disposition

The JMS queue disposition allows an event to be enqueued to a JMS queue. To create a port for a JMS queue disposition using Application Explorer:

1. Click the *iWay Events* tab.
2. In the left pane, expand the *SAP* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 *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=;
```

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

The following table lists and 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. You must create the connection factory, for example: <code>sampleQCF</code>
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, <code>java.naming.provider.url</code> The URL of the Sun Java System Application Server is <code>iiop://localhost:3700</code> where: 3700 Is a default port.
jndifactory	Is JNDI context.INITIAL_CONTEXT_FACTORY and is provided by the JNDI service provider. For Sun Java System Application Server, this is <code>com.sun.jndi.cosnaming.CNCTXFactory</code>
user	Valid user name required to access a JMS server.
password	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 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.

Procedure: How to Create a Port for the SOAP Disposition

To create a port for a SOAP disposition using Application Explorer:

- 1.** Click the *iWay Events* tab.
- 2.** In the left pane, expand the *SAP* node.
- 3.** Select the *ports* node.
- 4.** Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 a 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 lists and defines the parameters for the disposition.

Parameter	Description
wSDL-url	<p>The URL to the WSDL file that is required to create the SOAP message, for example:</p> <p>http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl</p> <p>where:</p> <p>webservice</p> <p>Is the name of the Web service you created using Application Explorer.</p> <p>To find this value, you can navigate to the iWay Business Services tab and open the Service Description link in a new window. The WSDL URL appears in the Address field.</p> <p>Alternatively, you can open the WSDL file in a third-party XML editor (for example, XMLSPY) and view the SOAP request settings.</p>
soapaction	Method that will be called by the SOAP disposition.
method	Web service method you are using. Value is found in the WSDL file.
namespace	XML namespace you are using. Value can be found in the WSDL file.
responseTo	<p>Location to which responses are posted. Can be a predefined port name or another URL. Optional.</p> <p>The URL must be complete, including the protocol.</p>
errorTo	<p>Location to which error logs are sent. Optional.</p> <p>Predefined port name or another disposition URL. The URL must be complete, including the protocol.</p>

Note: To use the SOAP disposition with a synchronous event, use Remote Function Modules to generate the schema and WSDL file instead of using the Business Object Repository for any RFC or BAPI.

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 a Port for the HTTP Disposition

The HTTP disposition uses an HTTP URL to specify a HTTP end point to which the event document is posted. To create a port for an HTTP disposition using Application Explorer:

1. Click the *iWay Events* tab.
2. In the left pane, expand the *SAP* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 *HTTP*.
- c. In the Disposition field, enter an HTTP destination.

When pointing Application Explorer to an **ibSE** deployment, specify the destination file using the following format:

`ihttp://url;responseTo=respDest`

When pointing Application Explorer to a **JCA** deployment, specify the destination file using the following format:

`http://host:port/uri`

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

The following table lists and describes the disposition parameters for HTTP.

Parameter	Description
url	The URL target for the post operation.
respDest	Location to which responses are posted. A predefined port name or another full URL. Optional. Predefined port name or another 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.
uri	Universal resource identifier that completes the URL specification.

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 a Port for the MQSeries Disposition

The MQSeries 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 MQSeries disposition using Application Explorer:

1. Click the *iWay Events* tab.
2. In the left pane, expand the *SAP* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

The Create New 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 *MQSeries*.
- c. In the Disposition field, enter an MQSeries destination.

When pointing Application Explorer to an **IBSE** deployment, specify the destination file using the following format:

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

When pointing Application Explorer to a **JCA** deployment, specify the destination file using the following format:

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

Important: When using the adapter in conjunction with the iWay Connector for JCA 1.5, there is no need to create event ports to dispose of event data. However, you must create a channel to enable event listening capabilities.

The following table lists and describes the disposition parameters for MQSeries.

Parameter	Description
qManager	Name of the queue manager to which the server must connect.
qName or respqueue	Name of the queue where messages are placed.
host	Host on which the MQ Server is located (MQ Client only).
port	Number to connect to an MQ Server queue manager (MQ client only).
channel	Case-sensitive name of the channel that connects with the remote MQ Server queue manager (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 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.

Editing an Event Port

You can edit an existing event port.

Procedure: How to Edit an Event Port

To edit an event port:

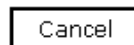
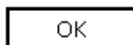
1. Select the event port you want to edit.
2. In the right pane, move the pointer over *Operations* and select *Edit*.

The Edit Port pane opens on the right, with fields to edit the description, disposition protocol, and disposition of the port as shown in the following image. You cannot change the name of the port name.

Edit Port

Choose parameters of the port that you wish to edit.

Port Name:	<input type="text" value="GETLIST"/>
Description:	<input type="text" value="Created on 01/07/04."/>
Disposition Protocol:	<input type="text" value="FILE"/>
Disposition:	<input type="text" value="file://c:\temp\x.txt"/>



3. Make the required changes to the event port configuration fields and click *OK*.

Deleting an Event Port

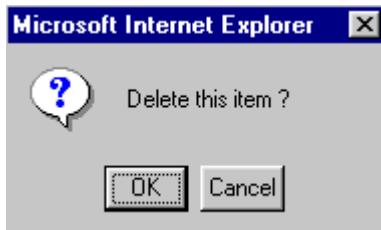
You can delete an existing event port.

Procedure: How to Delete an Event Port

To delete an 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 as shown in the following image, asking whether to delete the event port.



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

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

Adding, Modifying, or Deleting a Channel

The following topics describe how to create, modify, or remove a channel for your event adapter.

Creating a Channel

All defined event ports must be associated with a channel. You can create a channel using Servlet Application Explorer. The following procedure also describes how to create a channel as well as how to start or stop a channel.

Procedure: How to Create a Channel

To create a channel using Application Explorer:

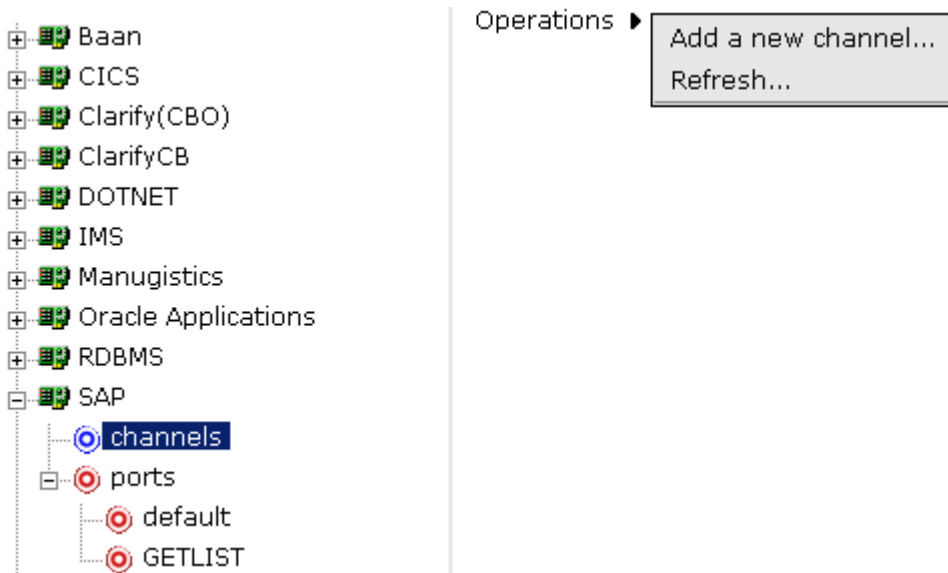
1. Click the *iWay Events* tab.

The list of adapters that support events appears in the left pane.



2. Expand the *iWay Events* node, for example, SAP.

The ports and channels nodes appear in the left pane as shown in the following image.



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 SAP channel* pane opens on the right as shown in the following image.

Add a new SAP channel

Choose a name and description for the new channel that you wish to create.

Channel Name:

Description:

Channel Type:



- a. In the Channel Name field, type a name, for example, TEST_CHANNEL.
- b. In the Description field, type a brief description.
- c. From the Channel Type drop-down list, select *SAP Channel -- App Server*.

5. Click *Next*.

The Edit channels pane opens on the right as shown in the following image, with fields to enter a gateway host, gateway service, program ID of the server, application server, and system number for the channel.

Edit channels

The screenshot shows the 'Edit channels' dialog box with the following fields and values:

Field	Value
Gateway host:	isdsrv2
Gateway service:	sapgw00
Program ID of the server:	JRDEST
Application Server:	isdsrv2
System number:	00

Navigation buttons: Help, < Back, Next >, Cancel

The System tab is active.

- a. On the System tab, enter the information that is specific to your SAP system.

Note: The program ID of the server is case sensitive.

- b. Click the *User* tab.

The User tab becomes active as shown in the following image, with fields to enter a client, user, password, language, and code page.

Edit channels

The screenshot shows a dialog box titled "Edit channels" with four tabs: "System", "User", "Advanced", and "preemitter". The "System" tab is highlighted in yellow. The "User" tab is active and contains the following fields:

- Client: 800
- User: ibi
- Password: *****
- Language: EN
- Codepage: (empty)

At the bottom of the dialog, there are four buttons: "Help", "< Back", "Next >", and "Cancel".

- c. Enter the user information that is specific to your SAP system.
- d. Click the *Advanced* tab.

The *Advanced* tab becomes active as shown in the following image, with fields to enter or select an IDoc format, user defined function modules, and processing mode options, as well as check boxes to enable SAP traces or Unicode encoding of the event data.

Edit channels

[System](#) [User](#) **Advanced** [preemitter](#)

IDOC Format:

User Defined Function Modules:

SAP trace:

Unicode:

Processing Mode:

- e. Specify any additional information or criteria for the channel you are creating.
- f. Click the *preemitter* tab.

The *preemitter* tab becomes active as shown in the following image, with a checkbox that enables you to strip the SAP payload of an event document.

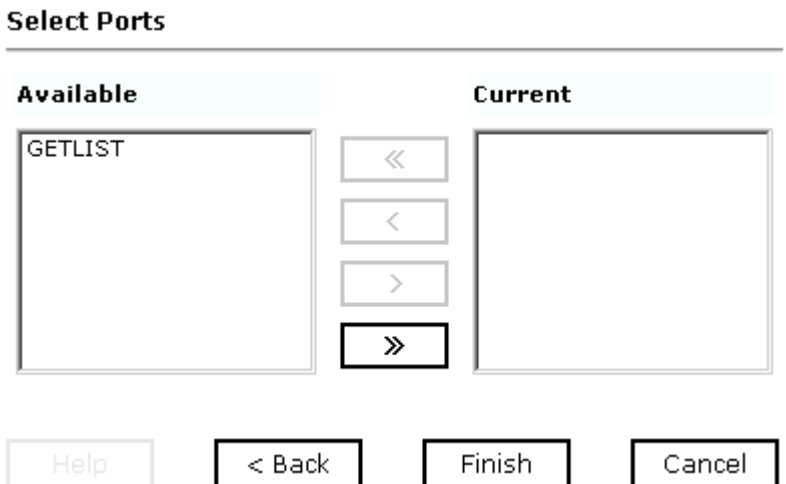
Edit channels

[System](#) [User](#) [Advanced](#) **preemitter**

Strip the Sap Payload:

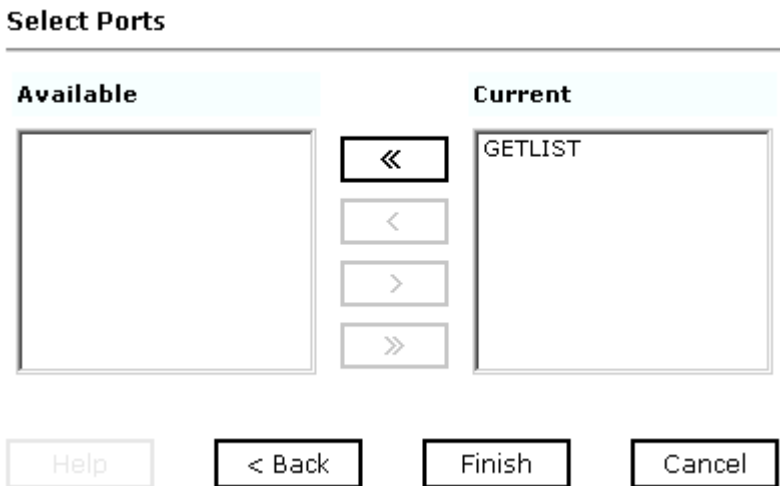
- 6. Click *Next*.

The Select Ports pane opens as shown in the following image where you can move ports between the Current and Available list.



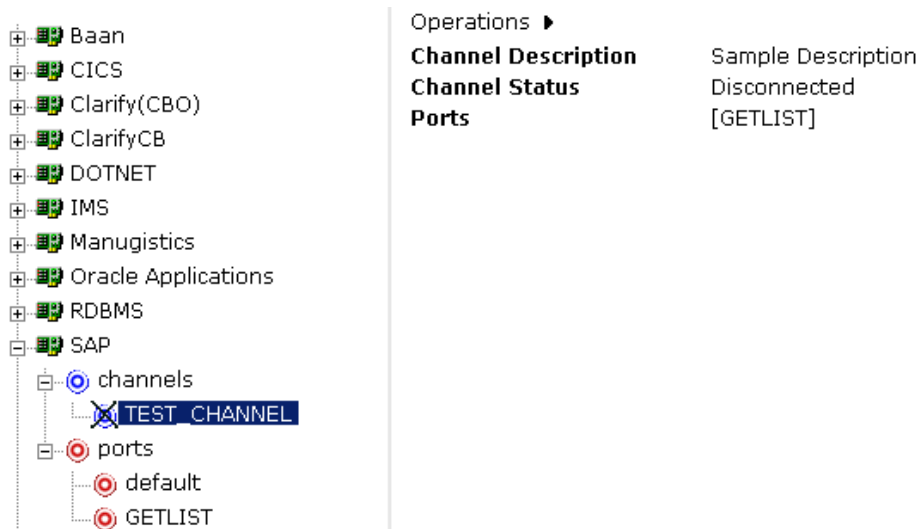
- a. Select an event port from the list of current ports.
- b. To transfer the port to the list of available ports, click the single right arrow button or to associate all event ports, click the double right arrow button.

The port appears in the list of available ports as shown in the following image.

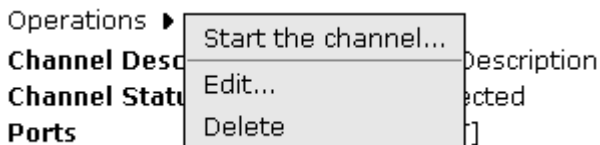


7. Click *Finish*.

The summary pane opens on the right with the channel description, channel status, and available ports as shown in the following image.

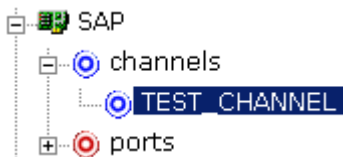


All the information is associated with the channel you created. The channel also appears under the channels node in the left pane. An X through the icon indicates that the channel is currently disconnected. You must start the channel using the Operations menu to activate your event configuration as shown in the following image.



8. Move the pointer over *Operations* and select *Start the channel*.

The channel you created is now active, and the X through the icon in the left pane disappears as shown in the following image.



9. To stop the channel at any time, move the pointer over *Operations* and select *Stop the channel*.

Modifying a Channel

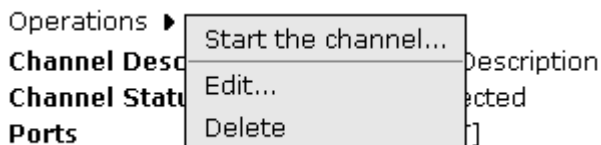
You can edit an existing channel.

Procedure: How to Edit a Channel

To edit an existing channel:

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

The Operations menu opens in the right pane over the description of the channel, its status, and ports as shown in the following image.



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

The Edit channels pane opens on the right as shown in the following image, with fields to edit the gateway host, gateway service, program ID of the server, application server, and system number of the channel.

Edit channels

System User Advanced preemitter

Gateway host:

Gateway service:

Program ID of the server:

Application Server:

System number:

3. Make the required changes to the channel configuration fields, click *Next* to continue and when you complete your edits, click *Finish*.

Deleting a Channel

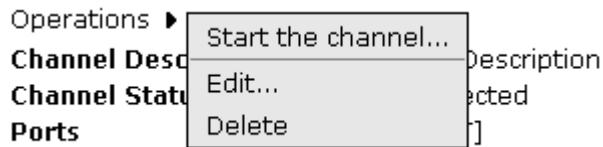
You can remove an existing channel.

Procedure: How to Delete a Channel

To delete an existing channel:

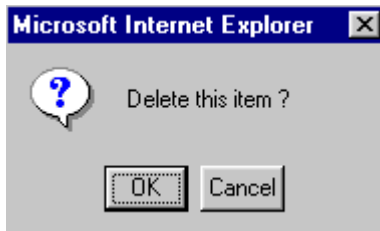
1. In the left pane, select the channel you want to delete.

The Operations menu opens in the right pane over the description of the channel, its status, and ports as shown in the following image.



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

A confirmation dialog box opens asking if you want to delete this item as shown in the following image.



3. To delete the channel you selected, click *OK*.

The channel disappears from the list in the left pane.

Synchronous Event Processing

You can configure synchronous event processing using Application Explorer to trigger a Web service once an event occurs in the SAP system. The event response that is received can then be routed to another disposition for further processing.

Procedure: How to Configure Synchronous Event Processing Using Application Explorer

To configure synchronous event processing:

1. Create a Web service for an SAP Remote Function Module, for example, BAPI_MATERIAL_GETLIST.
2. View the WSDL file.
3. Create a port using the SOAP disposition.

The Create New 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 a 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 lists and defines the parameters for the disposition.

Parameter	Description
wSDL-url	<p>The URL to the WSDL file that is required to create the SOAP message, for example:</p> <p>http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl</p> <p>where:</p> <p>webservice</p> <p>Is the name of the Web service you created using Application Explorer.</p> <p>To find this value, you can navigate to the iWay Business Services tab and open the Service Description link in a new window. The WSDL URL appears in the Address field.</p> <p>Alternatively, you can open the WSDL file in a third-party XML editor (for example, XMLSPY) and view the SOAP request settings.</p>
soapaction	Method that will be called by the SOAP disposition
method	Web service method you are using. You can find this value in the WSDL file.
namespace	The XML namespace you are using. You can find this value in the WSDL file.
responseTo	<p>Location to which responses are posted. Can be a predefined port name or another URL. Optional.</p> <p>The URL must be complete, including the protocol.</p>
errorTo	<p>Location to which error logs are sent. Optional.</p> <p>Predefined port name or another disposition URL. The URL must be complete, including the protocol.</p>

Note: To use the SOAP disposition with a synchronous event, use Remote Function Modules to generate the schema and WSDL file instead of using the Business Object Repository for any RFC or BAPI.

The following is an example of a completed SOAP disposition:

```
soap:http://localhost:7001/ibse/IBSEServlet/test/soapWS.ibs?wsdl;soa  
paction=soapWS.GETLISTRequest@test@@;method=GETLIST;namespace=ur  
n:iwaysoftware:ibse:jul2003:GETLIST;responseTo=ifile://c:\output\s  
ap\soapOut.xml
```

4. Create a channel.

The Edit channels window opens in the right pane.

5. Provide the required information to connect to SAP in the System and User tabs.

6. In the Advanced tab, select *REQUEST_RESPONSE* from the Processing Mode drop-down list.

7. In the premitter tab, select the *Strip the SAP Payload* check box.

8. Associate the port you created earlier with the new channel.

9. Start the channel.

A Web service for an SAP Remote Function Module, for example, *BAPI_MATERIAL_GETLIST*, is triggered after an event occurs in the SAP system. The response document is returned and routed to a file location.

CHAPTER 6

Using Web Services Policy-Based Security

Topics:

- iWay Business Services Policy-Based Security
- Configuring iWay Business Services Policy-Based Security

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

iWay Business Services Policy-Based Security

iWay 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 iWay 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 iWay Business Services. Instead, you can use one policy for many iWay 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 iWay 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 6-3 or *How to Create a Group to Associate With a Policy* on page 6-5.

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 6-7.

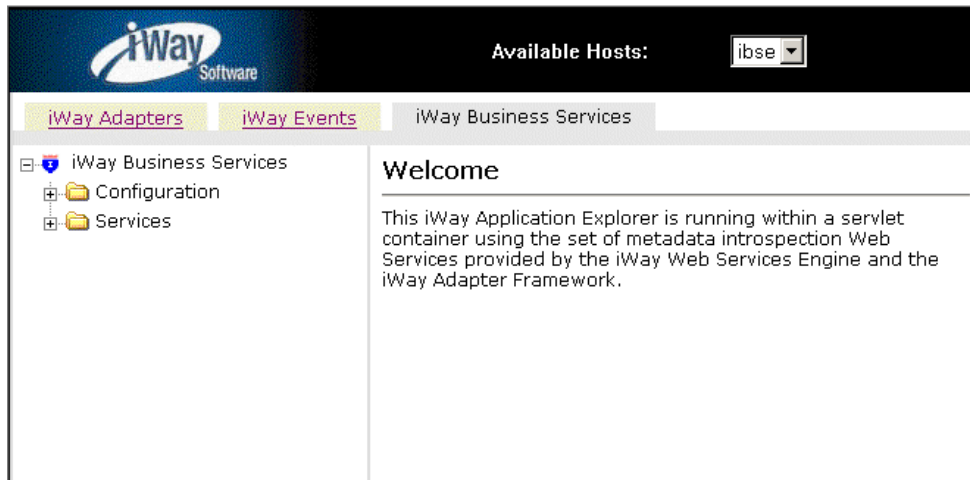
You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to iBSE and therefore, need not be applied to an individual business service. You need not create a policy, however, you must enable the Security Policy option in Servlet Application Explorer. For more information, see *How to Configure IP and Domain Restrictions* on page 6-10.

Procedure: How to 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 iWay Business Services. The iWay Business Services tab is active and displays a Welcome screen on the right. The image shows the iWay Business Services node expanded in the left pane.



- a. Click the *iWay 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.

Add a new user

Name:

Password:

Description:

- 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.

Operations ▶



Users

A user is an object that can be granted or denied permissions to run iWay Business Services. A user can belong to one or more groups. Policies that specify particular rights can be associated with user.

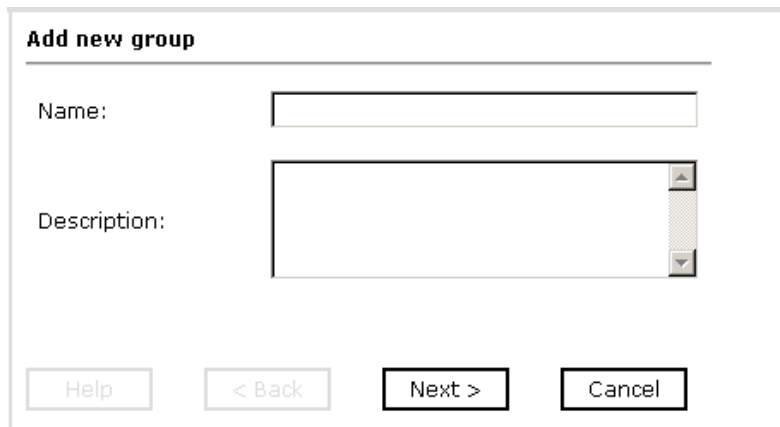
User Id	Description
<input type="checkbox"/> bse1	

Procedure: How to 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 *iWay 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.



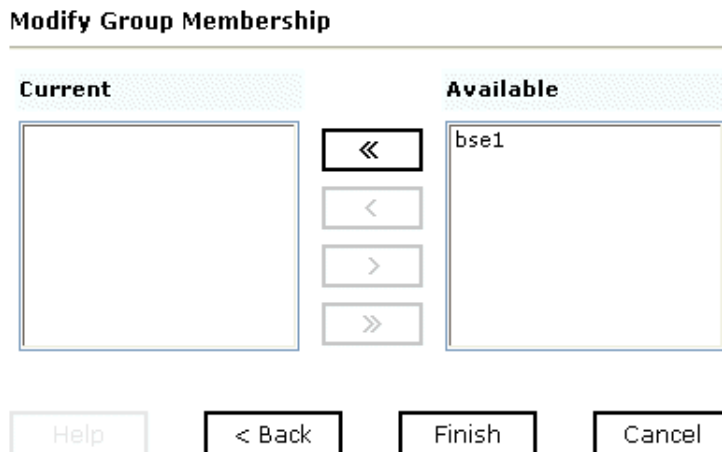
Add new group

Name:

Description:

- a. In the Name field, type 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.



Modify Group Membership

Current **Available**

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.

Operations ▶



Groups

A group is an object that can be granted or denied permissions to run iWay Business Services. A group is used as a container for one or more users. Policies that specify particular rights can be associated with a group.

Group name	Description
<input type="checkbox"/> newgroup	

Procedure: How to How to Create an Execution Policy

To create an execution policy:

1. Open *Servlet Application Explorer*.
 - a. Click the *iWay 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.

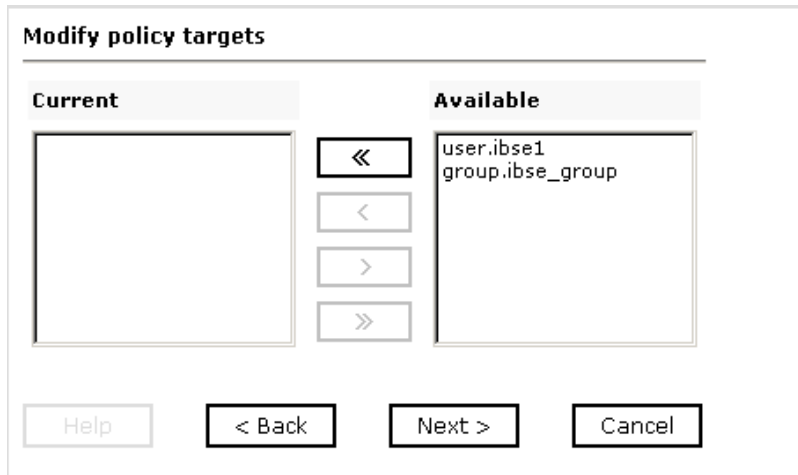
The dialog box is titled 'Add a new policy' and contains the following fields and buttons:

- Name:
- Type:
- Description:
- Buttons:

- a. In the Name field, type 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.


Member Id	Permission
user.ibse1	Deny
group.ibse_group	Deny

Buttons: Help, < Back, Finish, Cancel

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.

Operations ▶

 Policies

You can configure policies for the iWay Business Services Engine to manage resource execution, service routing, data restrictions and failover/recovery actions.

Name	Type	Description
<input type="checkbox"/> ibse_policy	Execution	

Procedure: How to How to Configure IP and Domain Restrictions

To configure IP and domain restrictions:

1. Open *Servlet Application Explorer*.

- a. Select the *iWay 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.

The image shows a dialog box titled "Add a new IP/Domain". It contains the following fields and controls:

- IP(Mask)/Domain:** A text input field.
- Type:** A dropdown menu with "Single" selected.
- Access Control:** A dropdown menu with "Deny" selected.
- Description:** A text area.
- Buttons:** "Help", "OK", and "Cancel" buttons are located at the bottom of the dialog.

- 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).

Operations ▶



IP and Domain

You can configure the iWay Business Services Engine to use policies that control access from a single IP address, a group of IP addresses, or all addresses within a particular domain.

IP(Mask) / Domain	Access	Description
<input type="checkbox"/> test	Deny	

CHAPTER 7

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 iWay 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

iWay 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 application 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.

iBSE Settings:		Save
Property Name	Property Value	
System		
Language	English	
Adapter Lib Directory	C:\Program Files\iWay55\lib	
Encoding	UTF-8	
Debug Level	NONE	
Number of Async. Processors	0	
Security		
Admin User	iway	
Admin Password	****	
Policy	<input type="checkbox"/>	
Repository		
Repository Type	File System	
Repository Url	file://C:\Program Files\iWay55\bea\ibe	
Repository Driver		
Repository User		
Repository Password		
Repository Pooling	<input type="checkbox"/>	
More configuration...		
		Save

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.

Property Name	Property Value
Monitoring	
Repository Type	File System
Repository Url	file://C:\Program Files\Way55\bea
Repository Driver	
Repository User	
Repository Password	
Repository Pooling	<input type="checkbox"/>
Auditing	
Store Message	<input type="radio"/> yes <input checked="" type="radio"/> no
Max Message Stored	10,000
Save Configuration Save History View Events View Services	
Start Monitoring	

- 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 application 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 screenshot shows a window titled "Service Statistics". It is divided into two main sections: "Web Service Methods" and "Statistics".

Web Service Methods

Service	Method
all	

Statistics

Total Time	55 min
Total Request Count	1
Total Success Count	1
Total Error Count	0
Average Request Size	409.0 bytes
Average Response Size	665.0 bytes
Average Execution Time	656 ms
Last Execution Time	828 ms
Average Back End Time	530 ms
Last Back End Time	765 ms
Successful Invocations	select a correlation id
Failed Invocations	select a correlation id

At the bottom right of the window is a button labeled "< home".

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).

The screenshot shows a window titled "Service Statistics". It is divided into two main sections: "Web Service Methods" and "Statistics".

Web Service Methods: This section contains two drop-down menus. The first is labeled "Service" and has the value "E0100033" selected. The second is labeled "Method" and has the value "all methods" selected.

Statistics: This section contains a table with the following data:

Total Time	1 hrs
Total Request Count	1
Total Success Count	1
Total Error Count	0
Average Request Size	409.0 bytes
Average Response Size	665.0 bytes
Average Execution Time	656 ms
Last Execution Time	656 ms
Average Back End Time	530 ms
Last Back End Time	530 ms
Successful Invocations	select a correlation id
Failed Invocations	select a correlation id

At the bottom right of the window, there are two buttons: "Suspend Service" and "< home".

- 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).

Service Statistics

Web Service Methods

Service	Method
<input type="text" value="B0100033"/>	<input type="text" value="GetEffectiveAddress"/>

Statistics

Total Time	1 hrs
Total Request Count	1
Total Success Count	1
Total Error Count	0
Average Request Size	409.0 bytes
Average Response Size	665.0 bytes
Average Execution Time	656 ms
Last Execution Time	656 ms
Average Back End Time	530 ms
Last Back End Time	530 ms
Successful Invocations	<input type="text" value="select a correlation id"/>
Failed Invocations	<input type="text" value="select a correlation id"/>

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.

The screenshot shows a window titled "Invocation Statistics" with three main sections: "Message Information", "Client Information", and "Detail".

Message Information	
Received	2004-09-14 12:04:16.312
Sent to adapter	2004-09-14 12:04:16.406
Received from adapter	2004-09-14 12:04:16.936
Responded	2004-09-14 12:04:16.968
Status	SUCCESS

Client Information	
Client IP	127.0.0.1
Client Host Name	127.0.0.1
User Name	

Detail	
Message	Size
Request Message	409 bytes
Response Message	665 bytes

< home

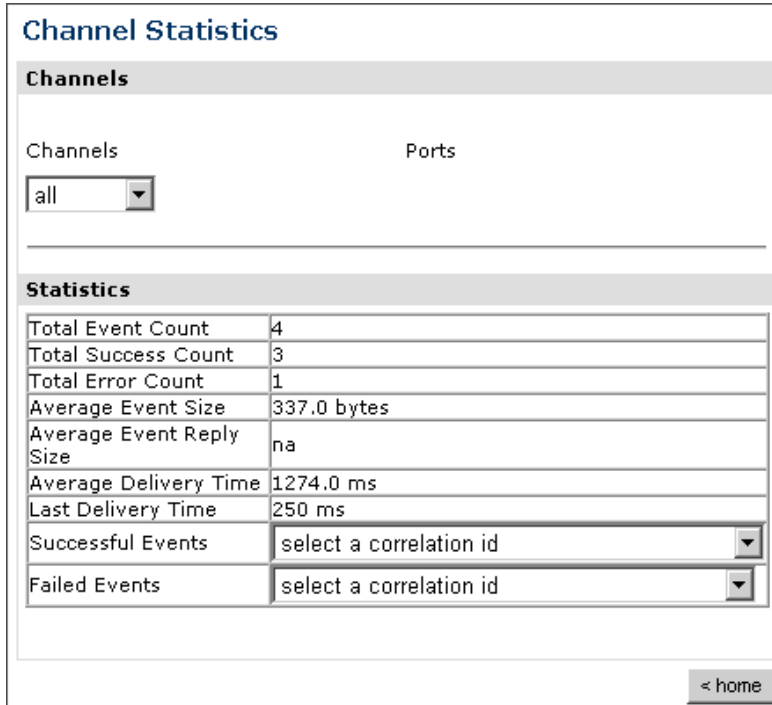
7. To view the XML request document in your Web browser, click *Request Message*.
You can also view the XML response document for the service.
8. To return to the iBSE Monitoring Settings window, click *home*.

Procedure: How to Monitor Events

To monitor events:

1. Ensure that your application 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.

Channel Statistics

Channels

Channels: TestChan ▾ Ports: all ▾

Statistics

Total Event Count	3
Total Success Count	2
Total Error Count	1
Average Event Size	401.0 bytes
Average Event Reply Size	na
Average Delivery Time	1542.0 ms
Last Delivery Time	250 ms
Successful Events	select a correlation id ▾
Failed Events	select a correlation id ▾

Suspend Channel
Start Channel
< home

- 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.

The screenshot shows a window titled "Channel Statistics". It is divided into two main sections: "Channels" and "Statistics".

Channels Section:

- Label: Channels
- Drop-down menu: TestChan
- Label: Ports
- Drop-down menu: TestPort

Statistics Section:

Total Event Count	2
Total Success Count	2
Total Error Count	0
Average Event Size	446.0 bytes
Average Event Reply Size	na
Average Delivery Time	2189.0 ms
Last Delivery Time	na
Successful Events	select a correlation id
Failed Events	select a correlation id

Buttons:

- Suspend Channel
- Start Channel
- < home

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.

The screenshot shows a window titled "Message Statistics". It contains two main sections: "Message Information" and "Messages".

Message Information

Received At	2004-09-14 12:18:20.842
Disposed At	● TestPort
Delivered At	2004-09-14 12:18:23.562

Messages

Detail	size
Event Message	446 bytes
Reply Message	na

In the bottom right corner of the window, there is a button labeled "< home".

- 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](#)

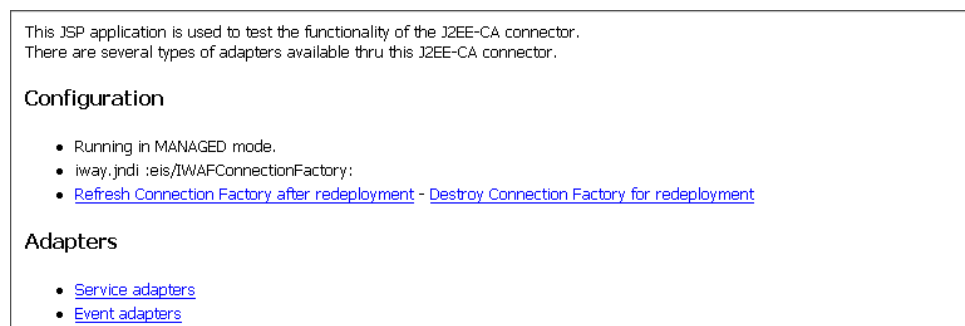
Is the name of the machine where your application server is running.

[port](#)

Is the HTTP port for the application server, 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.



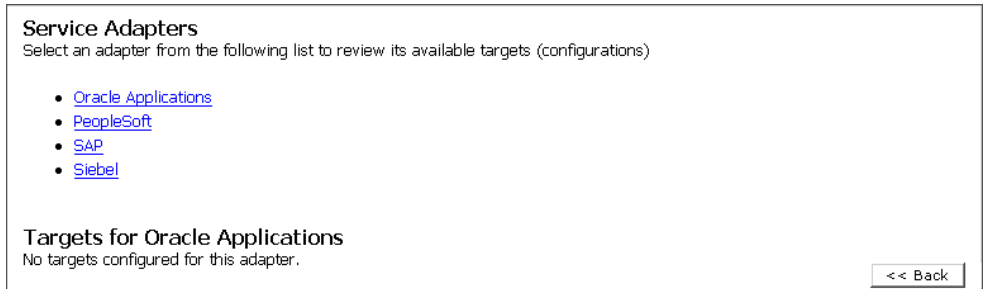
The JCA Test Tool runs in managed mode by default.

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.
 - 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.

The screenshot displays a web-based interface for testing services. It is divided into several sections:

- Service Adapters:** A section titled "Service Adapters" with the instruction "Select an adapter from the following list to review its available targets (configurations)". It contains a bulleted list of links: [Oracle Applications](#), [PeopleSoft](#), [SAP](#), and [Siebel](#).
- Targets for Siebel:** A section titled "Targets for Siebel" with a bulleted list containing [TestService](#).
- Statistics for Siebel target TestService:** A section showing performance metrics: TotalRequestCount : 0, TotalSuccessCount : 0, TotalErrorCount : 0, AverageExecutionTime : 0 msec, and LastExecutionTime : 0 msec.
- Request for Siebel target TestService:** A section with the instruction "Enter the data for this interaction. The configured user/password will be used if the User name is not provided." It includes three input fields: "User:" (a text box), "Password:" (a text box), and "Input Doc:" (a large text area with a vertical scrollbar).
- Buttons:** At the bottom of the form are three buttons: "Send", "Reset", and "<< Back".

- 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.

TotalRequestCount	: 0
TotalSuccessCount	: 0
TotalErrorCount	: 0
AverageExcecutionTime	: 0 msec.
LastExcecutionTime	: 0 msec.

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 HTTP port for the application server, 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.

This JSP application is used to test the functionality of the J2EE-CA connector. There are several types of adapters available thru this J2EE-CA connector.

Configuration

- Running in MANAGED mode.
- `iway.jndi :eis/IWAFConnectionFactory`:
- [Refresh Connection Factory after redeployment](#) - [Destroy Connection Factory for redeployment](#)

Adapters

- [Service adapters](#)
- [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.
 - 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.

Current channel Statistics	
Commands: start refresh	
Active: false	
TotalRequestCount	: 0
TotalSuccessCount	: 0
TotalErrorCount	: 0
AverageExcecutionTime	: 0 msec.
LastExcecutionTime	: 0 msec.
Statistics for port 'fileIN'	
TotalRequestCount	: 0
TotalSuccessCount	: 0
TotalErrorCount	: 0
AverageExcecutionTime	: 0 msec.
LastExcecutionTime	: 0 msec.

Setting Engine Log Levels

The following section describes how to set engine log levels for Servlet iBSE and JCA. For more information, see the 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 HTTP port for the application server, 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*.

Tracing information is written to the `ibselogs` directory where your application server accesses or has expanded Servlet iBSE.

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 connection-factory-dd (View Source for full doctype...)>
- <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 />
</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 5.5 Installation and Configuration* 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.

File Repositories

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

```
drive:\Program Files\iWay55\ibse\ibserrepo.xml
```

where:

```
drive
```

Is the location of your iWay 5.5 installation.

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

```
drive:\ProductionConfig\ibse\ibserrepo.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 5.5 Installation and Configuration* 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 7-27.

Procedure: How to Migrate an iBSE Repository Configured for Oracle

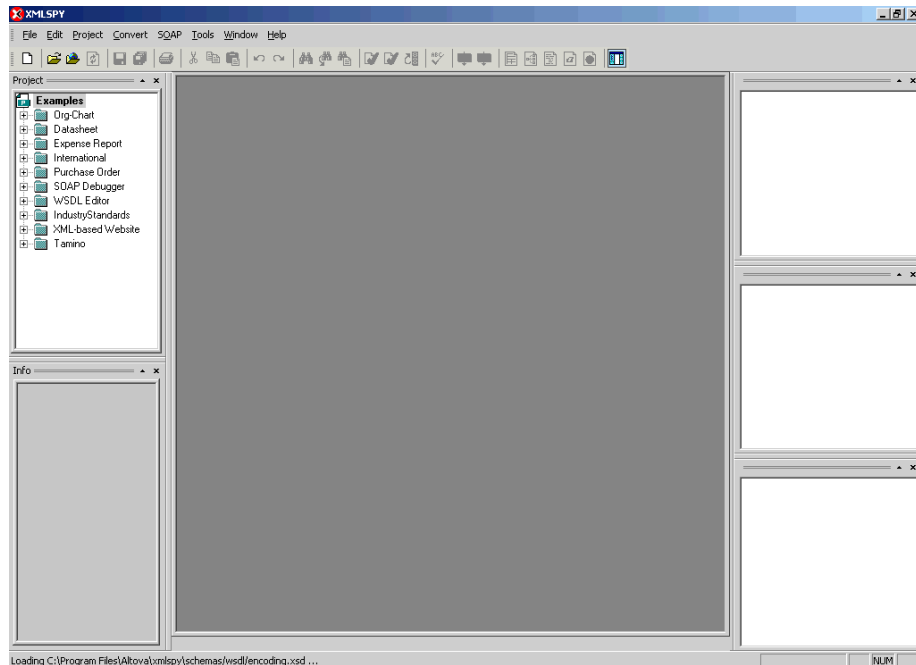
To migrate an iBSE repository that is configured for Oracle:

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

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

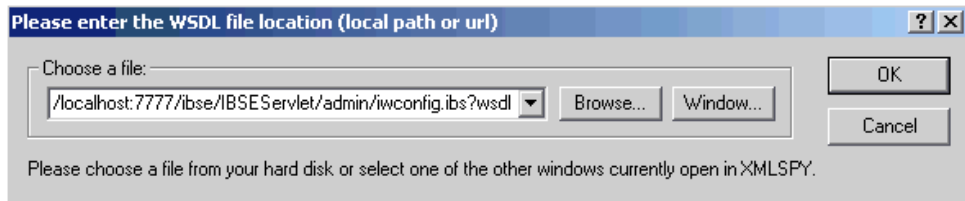
2. Open a third party XML editor, for example, XMLSPY.

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.



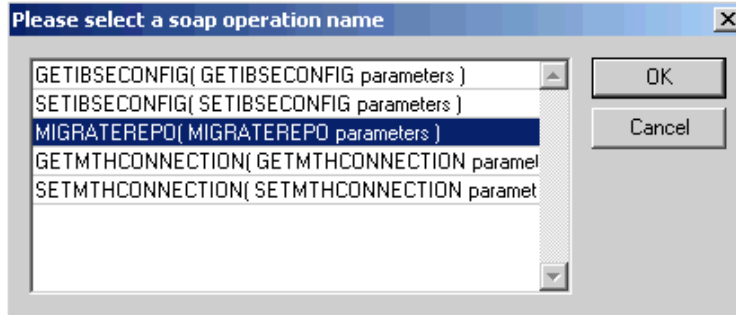
3. From the SOAP menu, select *Create new SOAP request*.

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.



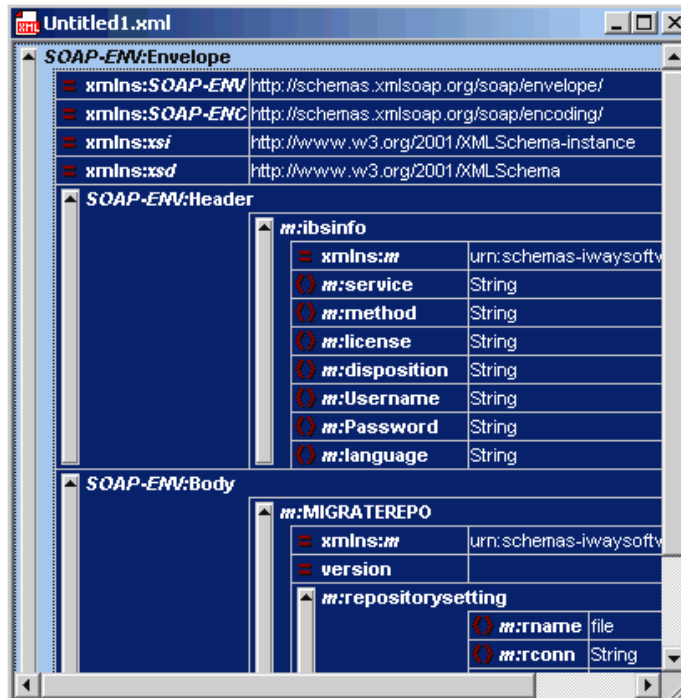
4. In the Choose a file field, paste the iBSE configuration service URL.
5. Click OK.

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 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:

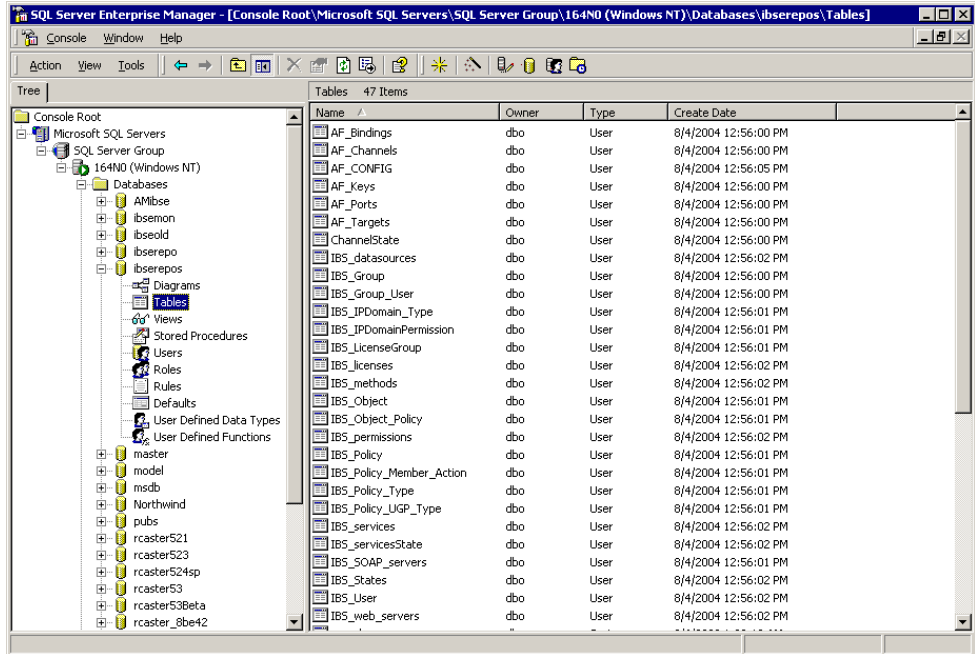
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:


```
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 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.

AF_Bindings
 AF_Channels
 AF_CONFIG
 AF_Keys
 AF_Ports
 AF_Targets
 ChannelState
 IBS_datasources
 IBS_Group
 IBS_Group_User
 IBS_IPDomain_Type
 IBS_IPDomainPermission
 IBS_LicenseGroup
 IBS_licenses
 IBS_methods
 IBS_Object
 IBS_Object_Policy
 IBS_permissions
 IBS_Policy
 IBS_Policy_Member_Action
 IBS_Policy_Type
 IBS_Policy_LUGP_Type
 IBS_services
 IBS_servicesState
 IBS_SOAP_servers
 IBS_States
 IBS_User
 IBS_web_servers

For more information on configuring the Oracle repository, see the 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:

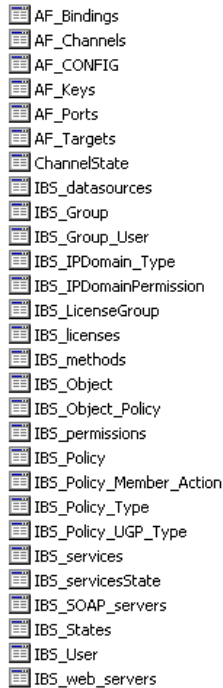
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

`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.



AF_Bindings
AF_Channels
AF_CONFIG
AF_Keys
AF_Ports
AF_Targets
ChannelState
IBS_datasources
IBS_Group
IBS_Group_User
IBS_IPDomain_Type
IBS_IPDomainPermission
IBS_LicenseGroup
IBS_licenses
IBS_methods
IBS_Object
IBS_Object_Policy
IBS_permissions
IBS_Policy
IBS_Policy_Member_Action
IBS_Policy_Type
IBS_Policy_UGP_Type
IBS_services
IBS_servicesState
IBS_SOAP_servers
IBS_States
IBS_User
IBS_web_servers

For more information on configuring the Sybase repository, see the 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.

A screenshot of a database table list showing various tables used by iBSE. The tables are listed in a vertical column, each with a small icon to its left. The tables are:

- AF_Bindings
- AF_Channels
- AF_CONFIG
- AF_Keys
- AF_Ports
- AF_Targets
- ChannelState
- IBS_datasources
- IBS_Group
- IBS_Group_User
- IBS_IPDomain_Type
- IBS_IPDomainPermission
- IBS_LicenseGroup
- IBS_licenses
- IBS_methods
- IBS_Object
- IBS_Object_Policy
- IBS_permissions
- IBS_Policy
- IBS_Policy_Member_Action
- IBS_Policy_Type
- IBS_Policy_UGP_Type
- IBS_services
- IBS_servicesState
- IBS_SOAP_servers
- IBS_States
- IBS_User
- IBS_web_servers

For more information on configuring the DB2 repository, see the 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:

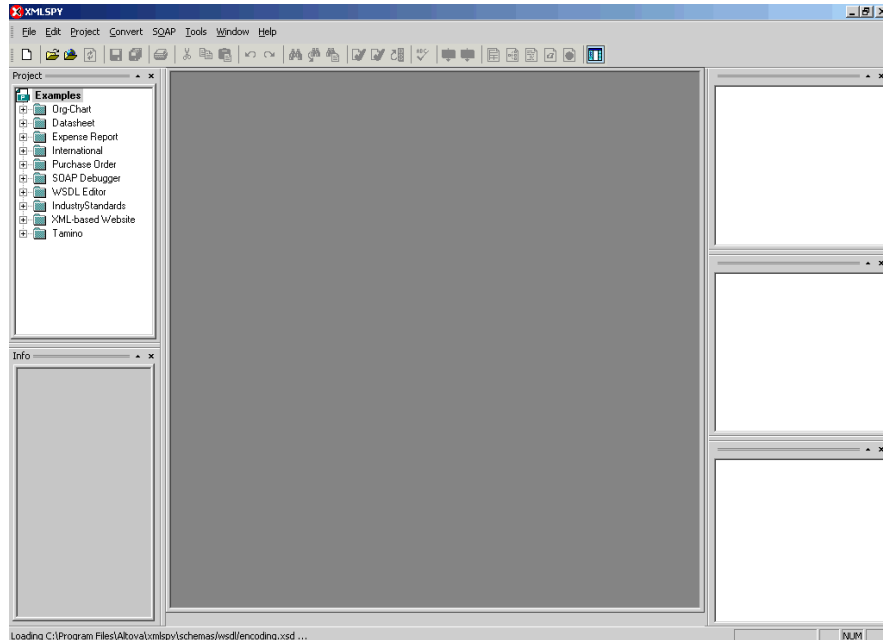
1. Copy the iBSE administrative services for Application Explorer URL, for example:

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

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

Exporting or Importing Targets

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

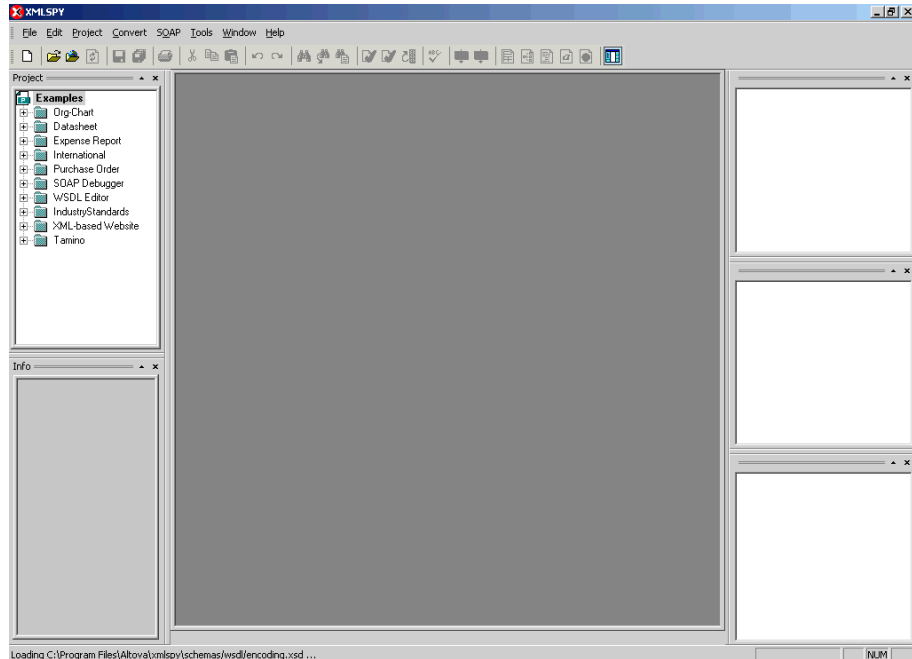
To import a target:

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.

Exporting or Importing Targets

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>R01GODlhcgGSALMAAAQCAEMmCZtuMFQxDS8b</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:

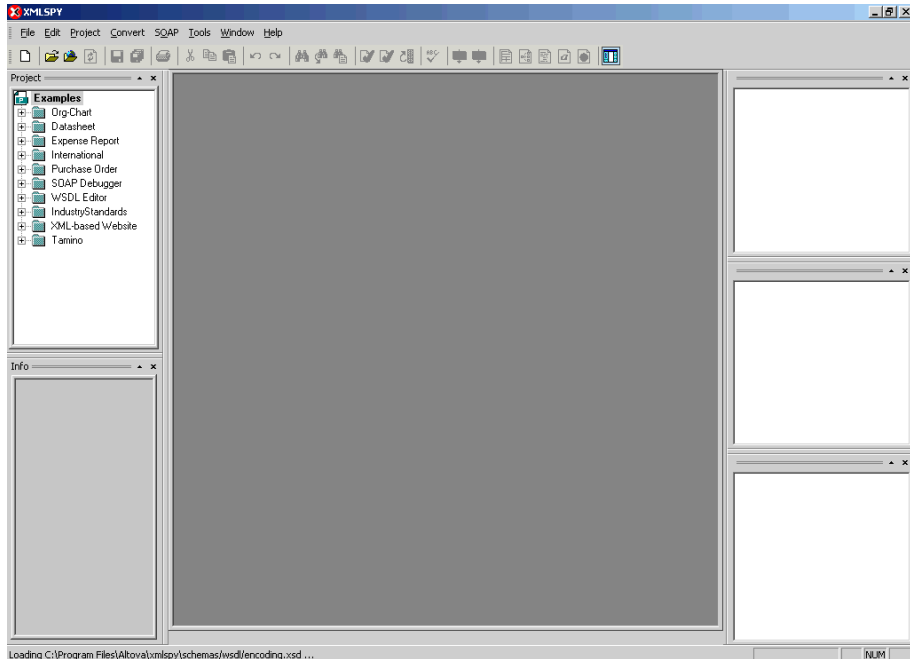
1. Copy the iBSE configuration service URL, for example:

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

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

Retrieving or Updating Web Service Method Connection Information

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:

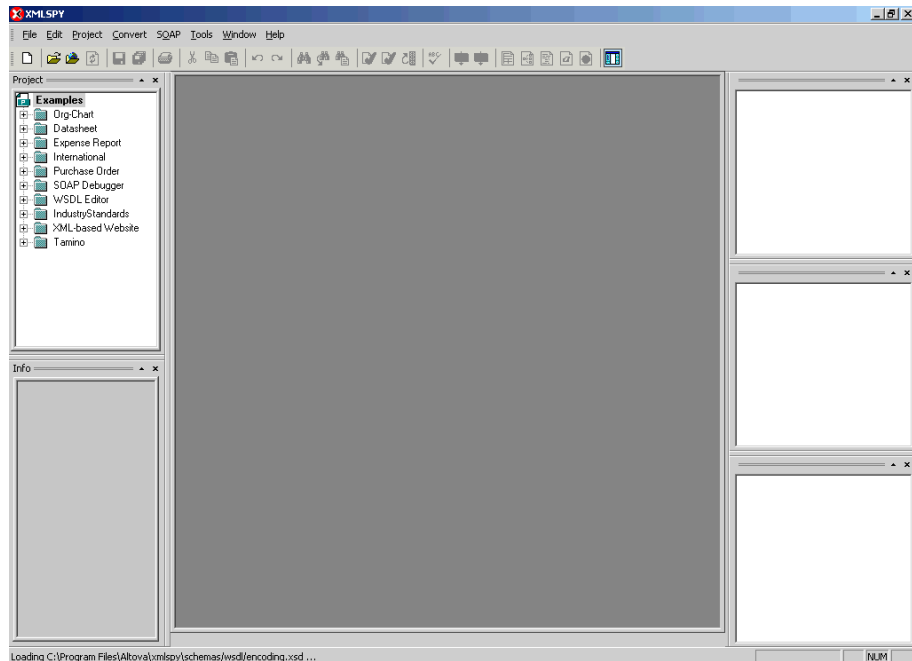
1. Copy the iBSE configuration service URL, for example:

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

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

Retrieving or Updating Web Service Method Connection Information

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

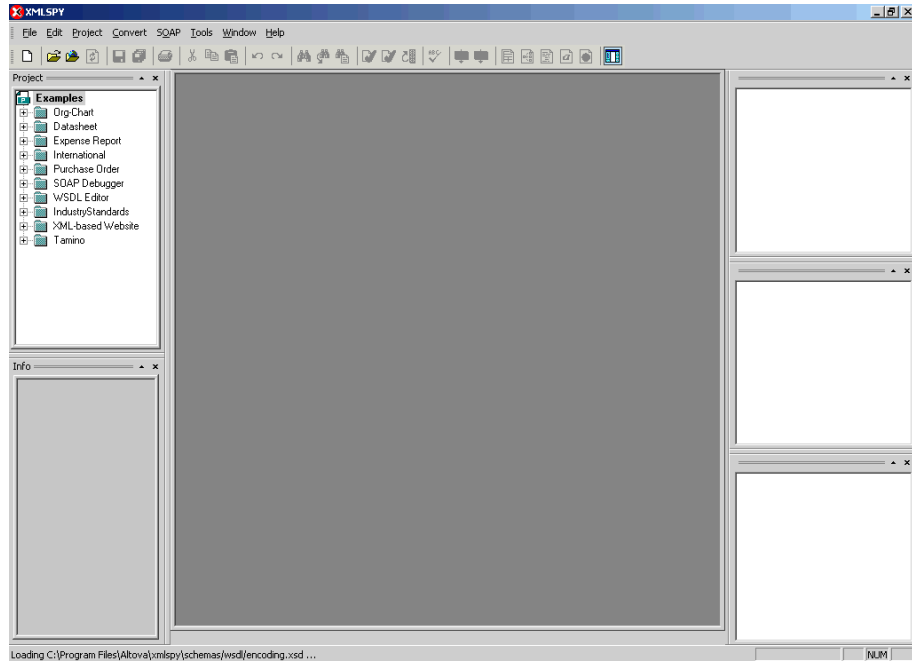
To start a channel programmatically:

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.

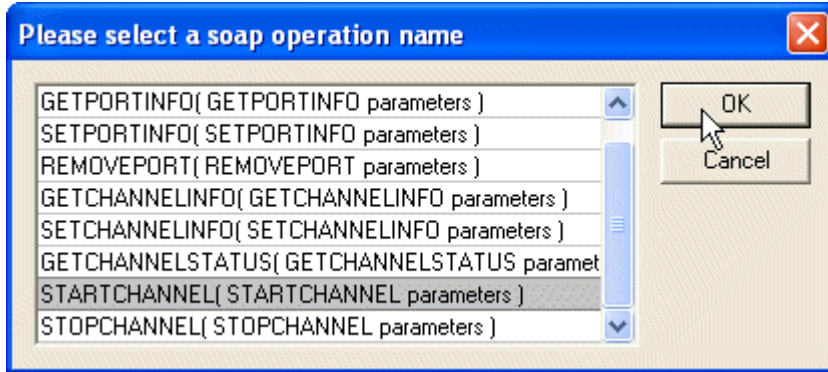
Starting or Stopping a Channel Programmatically

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

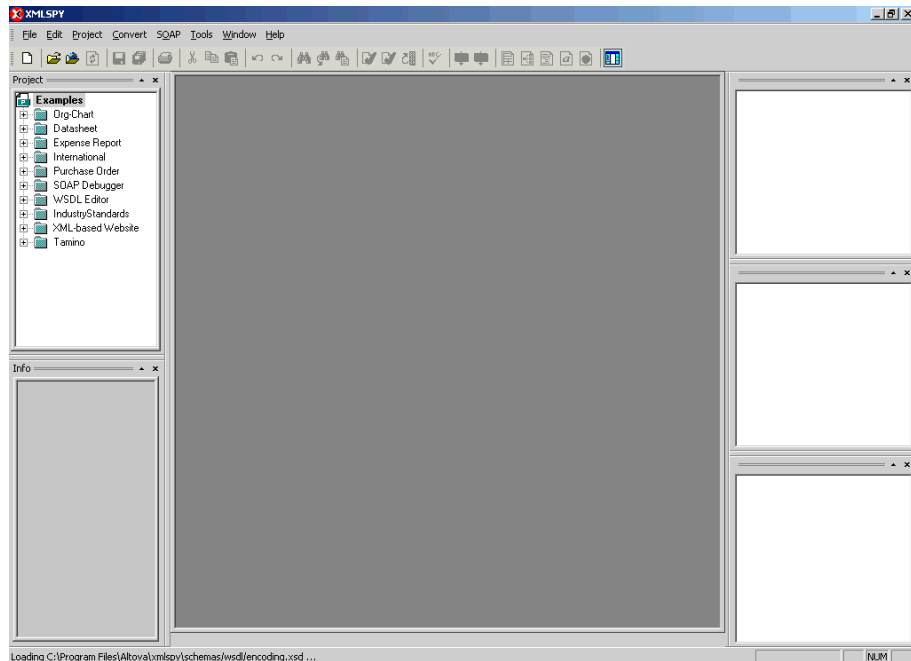
To stop a channel programmatically:

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.

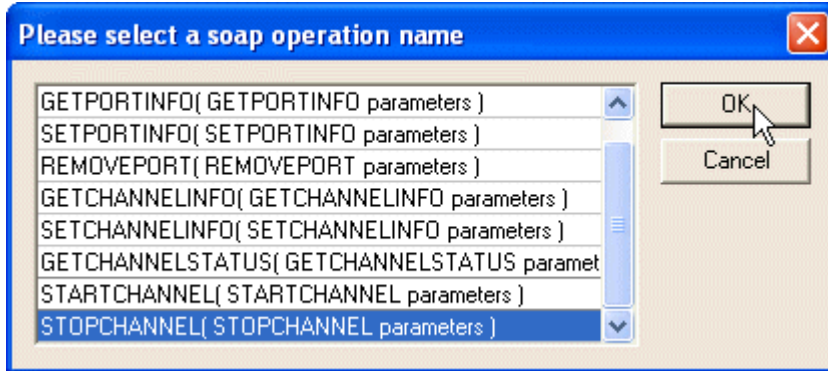


3. From the SOAP menu, select *Create new SOAP request*.

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*.

Starting or Stopping a Channel Programmatically

CHAPTER 8

Understanding SAP Events

Topics:

- Overview
- Related Concepts and Terminology
- Registering Your Program ID in SAPGUI
- Testing the SAP Event Adapter
- Application Link Enabling Configuration for the Event Adapter
- Testing the SAP ALE Configuration

The following topics provide an overview of event functionality in SAP and describe how to configure and test your SAP system for event processing.

Overview

An event in SAP is defined as an occurrence of a status change in an object. Events are created when the relevant status change occurs. The event creation must be implemented by SAP or yourself. An event is created from specific application programs (the event creator) and then “published” system-wide. Any number of receivers can respond to the event with their own “response mechanisms”. An event is usually defined as a component of an object type.

SAP “pseudo events” are events that are not processed by the SAP Event manager, but are called from an ABAP program or Remote Function call (using the “Destination” parameter).

Related Concepts and Terminology

The following topics list and define specific terminology related to SAP and SAP event handling.

Client and Server Programs

RFC programs for non-SAP systems can function as either the caller or the called program in an RFC communication. There are two types of RFC programs:

- RFC Client
- RFC Server

The RFC client is the instance that calls the RFC to execute the function that is provided by an RFC server. The functions that can be executed remotely are called RFC functions, and the functions provided by the RFC API are called RFC calls.

SAP Gateway

The SAP Gateway is a secure Application Server. No connections are accepted unless they have been pre-registered previously from the SAP presentation Client. A server connection presents itself to the Gateway and exposes a Program Identifier. If the Program Identifier is found in the list of registered Program Id's, the Gateway server then offers a connection to the server, which “Accepts” a connection. This Program id then is linked with an “RFC Destination” within SAP, which allows SAP Function Modules and ALE documents (IDocs or BAPI IDocs) to be routed to the destination. The RFC Destination functions as a tag to mask the Program Id to SAP users.

An RFC server program can be registered with the SAP gateway and wait for incoming RFC call requests. An RFC server program registers itself under a Program ID at an SAP gateway and not for a specific SAP system.

In SAPGUI, the destination must be defined with transaction SM59, using connection type T and Register Mode. Moreover, this entry must contain information on the SAP gateway at which the RFC server program is registered.

Program IDs and Load Balancing

If the Gateway Server has a connection to a particular server instance and another server instance presents itself to the gateway, the gateway offers the connection and then begins functioning in “Load Balancing” mode. Using a proprietary algorithm, the Gateway will send different messages to each server depending on demand and total processing time. This may cause unpredictable results in the WLI 8.1 scenario where messages are validated by schema and application.

When configuring multiple events in WLI using a single SAP program ID, SAP will load balance the event data. For example, if multiple remote function calls or BAPIs use the same program ID (for example, IWAYID) and multiple SAP listeners are configured with this program ID, then SAP will send one request to one listener and the next to another listener, and so on.

There is a load balancing algorithm present in the SAP Gateway Server. This mechanism is proprietary to SAP application development, and may work by comparing total throughput of the connection, the number of times in wait state, and so on. This means connection one may receive 9 messages and connection two 1. If 5 of 9 messages are rejected for schema validation, and the 1 on the other id for schema validation, the customer can very easily make a case of missing messages.

Registering Your Program ID in SAPGUI

To enable your SAP system to issue the following calls or interfaces to the SAP event adapter, you must register your program ID under an RFC destination.

- Remote Function Calls (RFC)
- Business Application Programming Interfaces (BAPI)
- Intermediate Documents (IDoc)

The RFC destination is a symbolic name (for example, IWAYDEST) that is used to direct events to a target system, masking the program ID. The Program ID is configured in both SAPGUI and the event adapter.

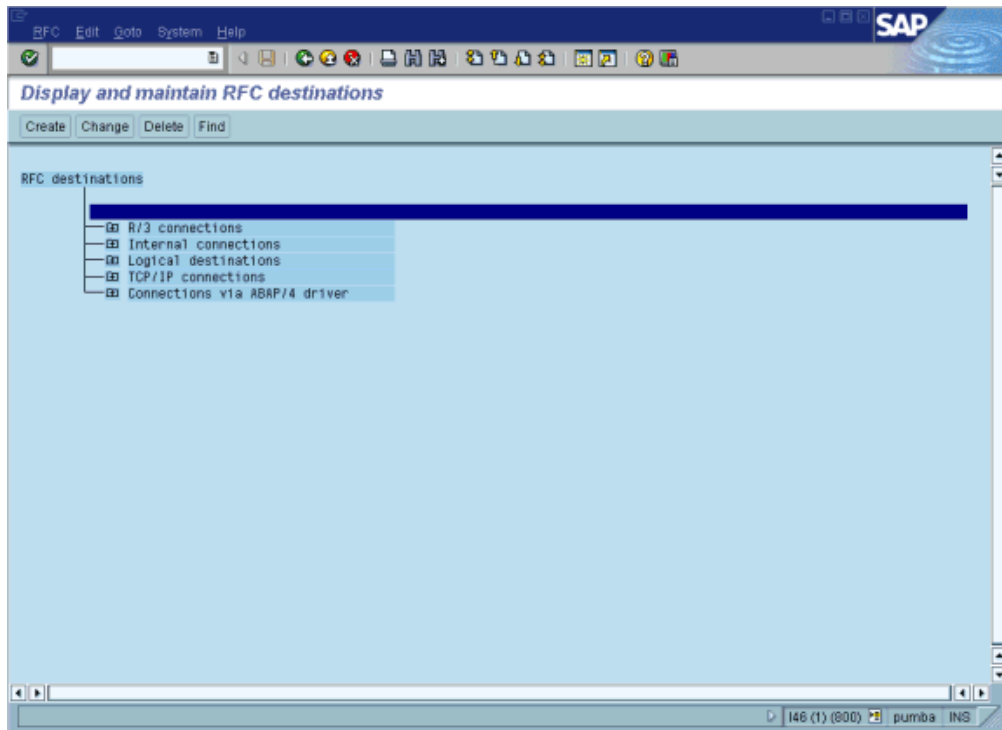
Procedure How to Register Your Program ID

To register your program ID:

1. Launch the *SAP Workbench* and logon to the SAP system.
2. Select *Tools, Administration, Network*, and then, *RFC destination*.

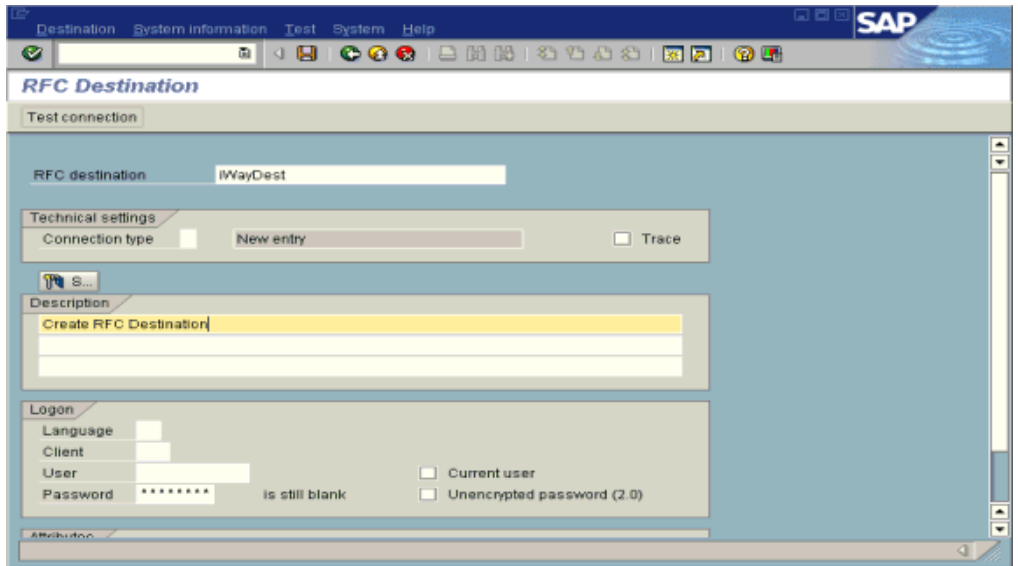
3. Execute the *SM59* transaction.

The Display and maintain RFC destinations window opens and displays a list of connections and drivers you can manage as shown in the following image.



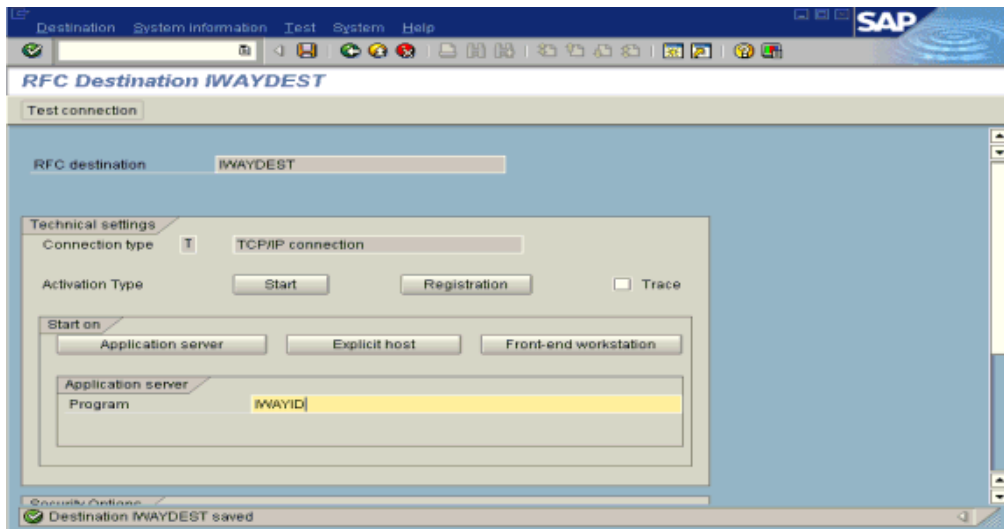
4. Select *TCP/IP connections*.
5. Click *Create*.

The RFC Destination window opens and displays fields where you provide information about the RFC destination as shown in the following image.



- a. In the RFC destination field, type a name, for example, IWAYDEST.
The value you type in this field is case-sensitive.
 - b. In the Connection type field, type *T* (for destination type, TCP/IP).
 - c. In the Description field, type a brief description.
6. Click *Save* from the tool bar or select *Save* from the Destination menu.

The RFC Destination IWAYDEST window opens.



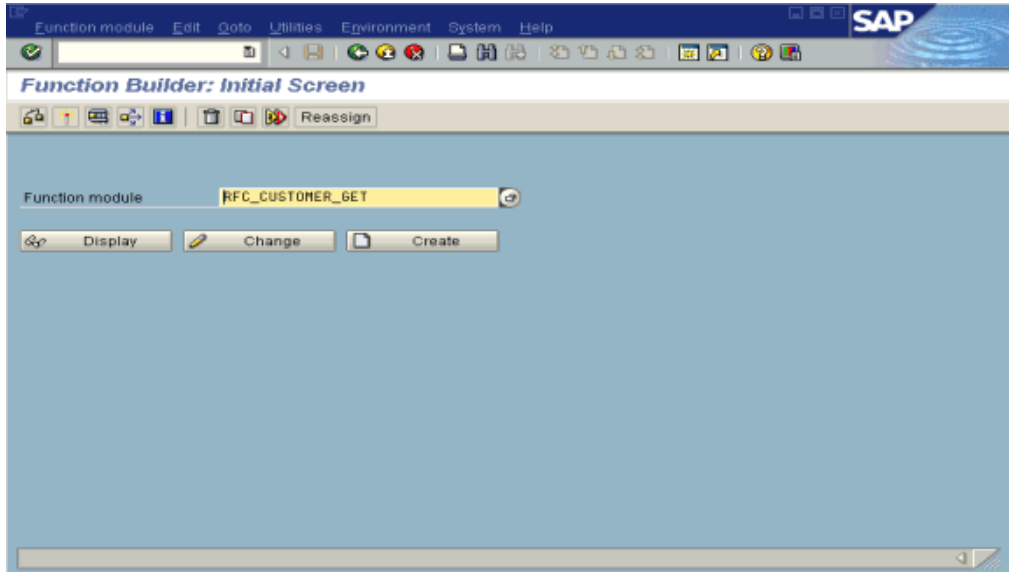
- a. For the Activation Type, click the *Registration* button.
- b. In the Program field, type *IWAYID*.
7. Click *Save* from the tool bar or select *Save* from the Destination menu.
8. Ensure your event adapter is running.
9. To verify that the SAP system and the iWay Application System Adapter for SAP R/3 are communicating, click *Test connection*.

Testing the SAP Event Adapter

In the SAP Server, the SE37 transaction enables you to send RFCs (Remote Function Calls) or BAPIs (Business Application Programming Interfaces) to any RFC destination. For more information on RFC destinations, see *Registering Your Program ID in SAPGUI* on page 8-3.

Procedure How to Test the SAP Event Adapter by Sending RFCs or BAPIs Manually

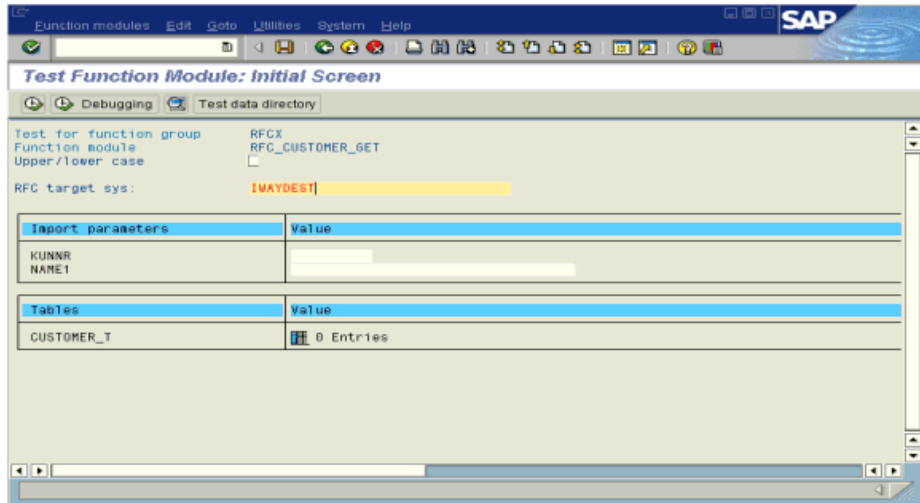
The following image shows the Function Builder: Initial Screen where you can select to display, change, or create a function module.



To test the SAP event adapter:

1. In the Function Builder: Initial Screen, select a function module, for example, RFC_CUSTOMER_GET.
2. To choose single test, press *F8* and click the *Single Test* icon or select *Function module, Test*, and then, *Single Test*.
3. Enter an RFC target system, for example, IWAYDEST.
4. Enter input data for the particular RFC module, for example, AB*.
5. To execute, press *F8*.

The Test Function Module: Initial Screen opens as shown in the following image. It includes information about the test, the function module, and the target system. You can select a check box for Upper/lower case. One pane lists the import parameters, and another has fields for the values. A third pane lists tables, and the fourth lists the number of entries.



6. Enter data into the SAP GUI and click the *Execute* button.

The function name and input data are transferred by RFC to create an XML document on the Sun Java System Application Server with the parameters input in SAPGUI.

Application Link Enabling Configuration for the Event Adapter

The SAP event adapter receives IDocs (Intermediate Documents) from SAP.

When using collected IDocs on any platform during inbound processing (service mode), if the DOCNUM field does not have a unique document number for each IDoc, the system creates an IDoc for each header record in the collected IDoc file and duplicates the data for each IDoc.

Ensure the DOCNUM field is included in the EDI_DC40 structure and that each IDoc has a unique sequence number within the collected IDoc file.

To configure an SAP system to send IDocs to the SAP event adapter, use the ALE (Application Link Enabling) configuration to:

1. Register your program ID in SAPGUI. For more information, see *Registering Your Program ID in SAPGUI* on page 8-3.
2. Define a port.

A port identifies where to send messages. This port can be used only if an RFC destination was previously created. For more information on creating an RFC destination, see *Overview* on page 8-2. For more information, see *How to Define a Port* on page 8-9.

3. Create a logical system.

One type of partner is a logical system. A logical system manages one or more RFC destinations. For more information, see *How to Create a Logical System* on page 8-10.

4. Create a partner profile.

A partner profile is a definition of parameters for the electronic interchange of data with a trading partner using the IDoc interface. To communicate with a partner using the IDoc interface, you must create a partner profile. For more information, see *How to Create a Partner Profile* on page 8-12.

5. Create a distribution model for the partner and message type.

You create a distribution model for the partner and message type you designated. For more information, see *How to Create a Distribution Model for the Partner and Message Type* on page 8-14.

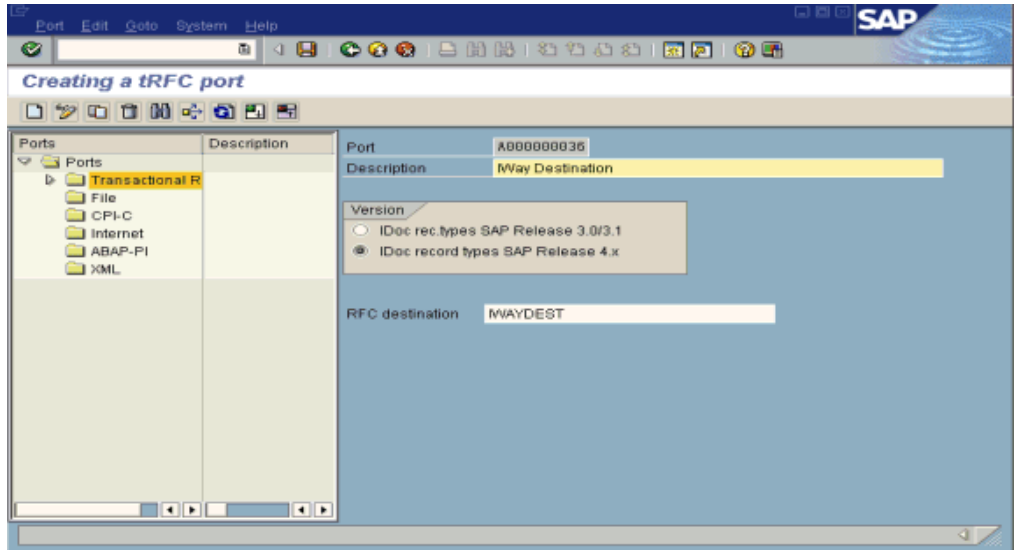
6. Test the SAP event adapter. For more information, see *Testing the SAP ALE Configuration* on page 8-17.

Procedure **How to Define a Port**

To define a port:

- 1.** In the ALE configuration, choose *Tools, Business Communications, IDocs Basis, IDoc*, and then, *Port Definition* or execute the *WE21* transaction.

The Creating a tRFC port window opens as shown in the following image. The window is divided into a Ports pane and a Description pane on the left. On the right is a pane for displaying information about the port.



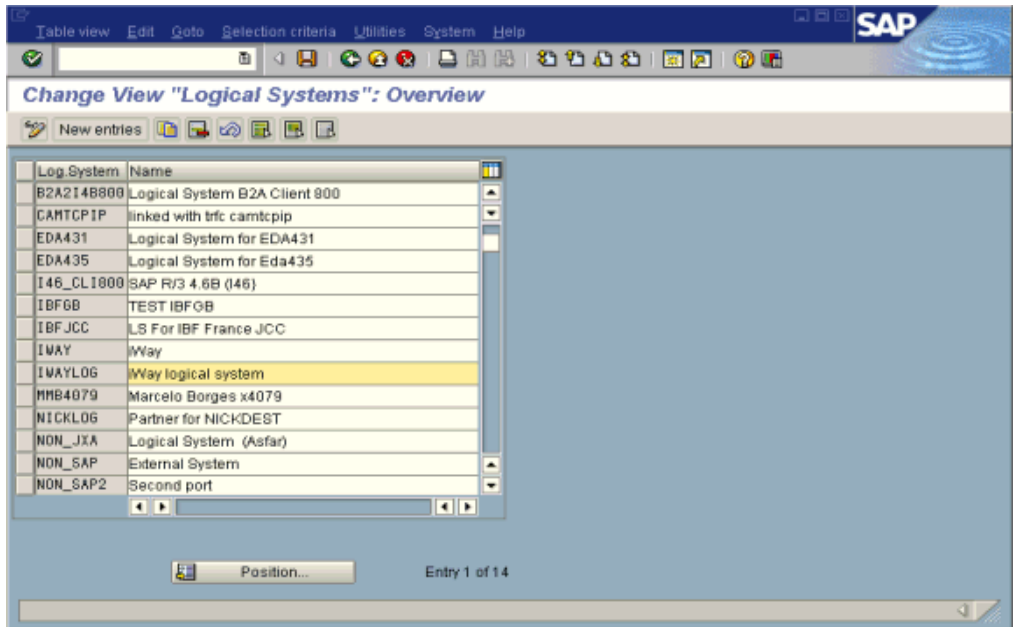
2. In the left pane under ports, select *Transactional RFC* and click *Create*.
3. Select *Generate port name*.
The system generates the port name.
4. In the right pane, select the IDoc version you want to send through this port.
5. Click the destination you created, for example, *IWAYDEST*.
6. Save the session, making note of the system-generated RFC port.

Procedure How to Create a Logical System

To create a logical system called *IWAYLOG*:

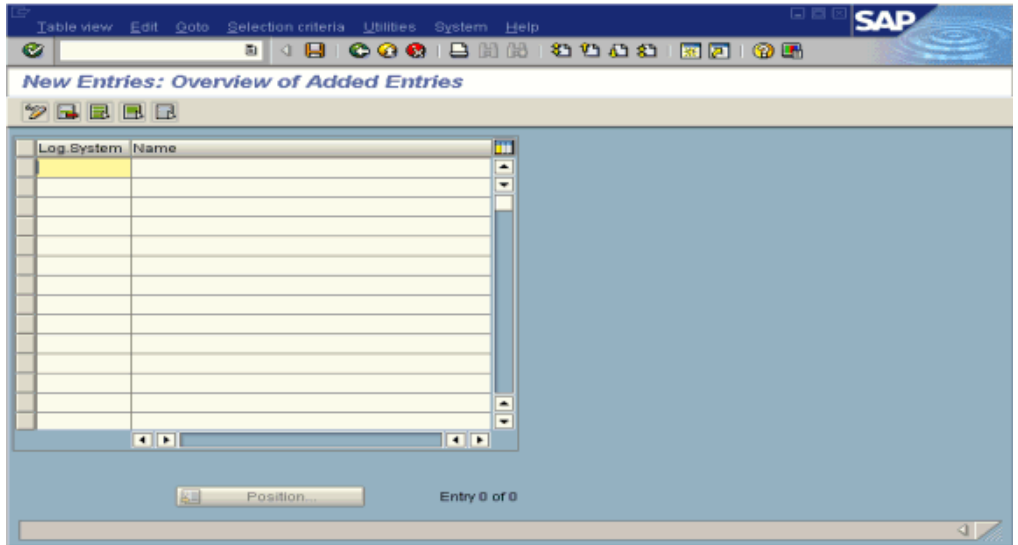
1. In the ALE Configuration, enter the area menu selection *SALE* transaction.
2. Select *SAP Reference IMG*.
3. Expand the following nodes: *Basis Components*, *Application Link Enabling (ALE)*, *Sending and Receiving Systems*, *Logical Systems*, and *Define Logical System*.
4. Click the green check mark beside *Define Logical System*.

The Change View “Logical Systems”: Overview window opens and displays a list of logical systems and their names as shown in the following image.



5. Click *New entries*.

The New Entries: Overview of Added Entries window opens with columns labelled Log.System and Name where you can type new log systems.



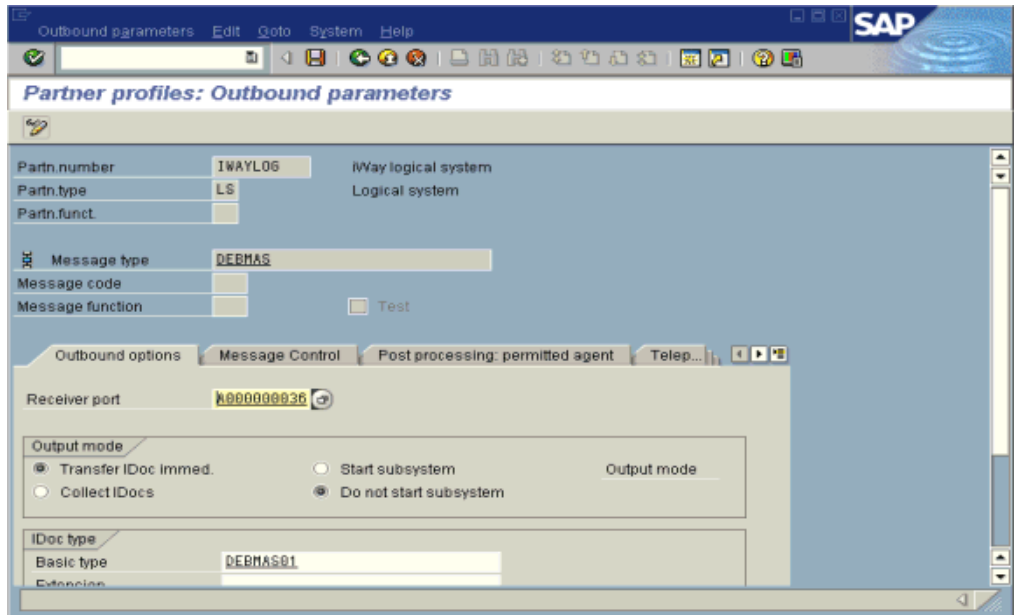
- a. Type an entry for Log System, for example, IWAYLOG.
 - b. In the Name column, type a name (description) for the partner profile.
6. Save the session.

Procedure How to Create a Partner Profile

To create a partner profile:

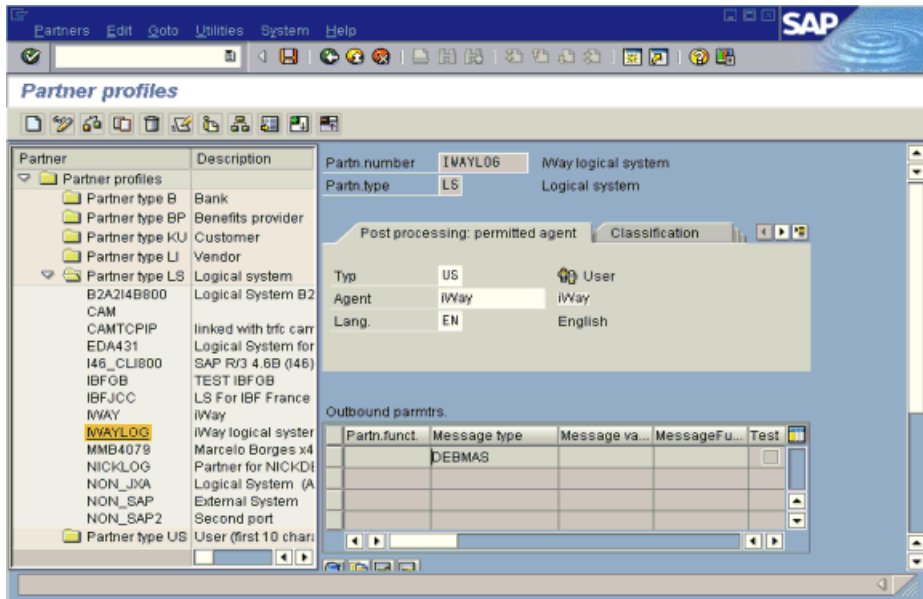
1. In the SAP Workbench, choose *Tools, Business Communication, IDoc Basis, IDoc*, and then, *Partner profile* or execute the *WE20* transaction.

The Partner profiles: Outbound parameters window opens and displays fields for specifying details for the partner profile.



- a. Select Partner type *LS* (Logical system).
- b. Press *F5* (Create).
2. For Type, enter *USER*.
3. For Agent, enter the current user ID, or you may select another agent type.
4. Under the outbound parameter table control, select *Create outbound parameter*.
Partn.type is *LS*.
Message type is *DEBMA* (the IDoc document type).
5. Leave *Partn.funct* blank.
6. Click the *Outbound options* tab.
 - a. Depending on your performance requirements, click *Transfer IDoc Immed* or *Collect IDocs*.
 - b. For the IDoc, type a message type, for example, *DEBMA*.
 - c. Type a receiver port, for example, *A000000036*.
7. Save the session and exit.

The Partner profiles summary window opens and displays information for the logical system that you created as shown in the following image. In the left pane are partners and descriptions. The right pane displays information depending on which tab is active.

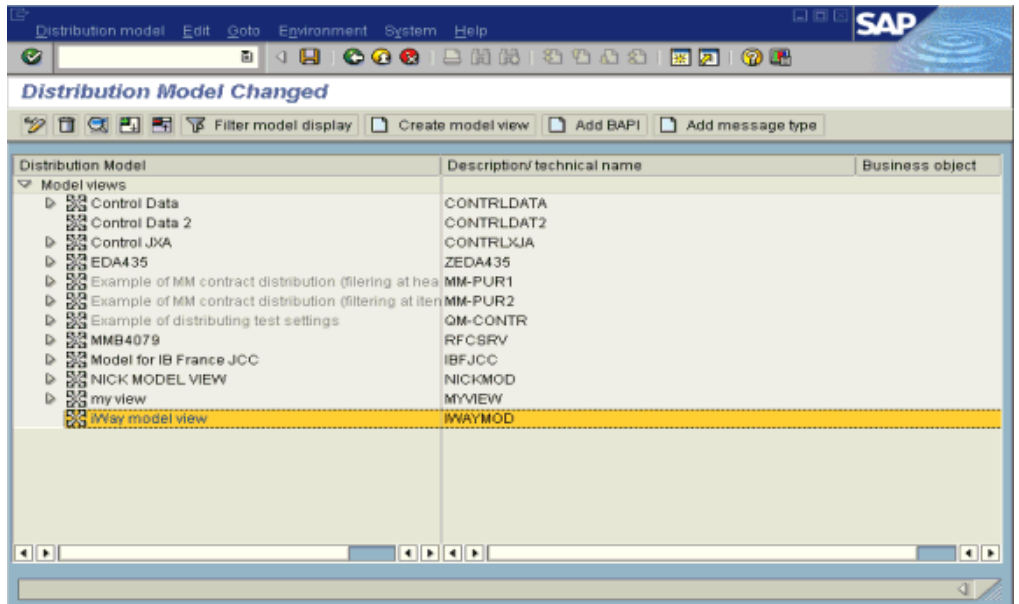


Procedure How to Create a Distribution Model for the Partner and Message Type

To create a distribution model called IWAYMOD:

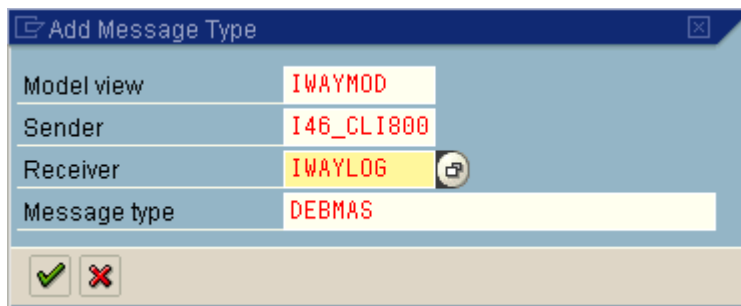
1. In the SAP Workbench, choose *Tools, AcceleratedSAP, Customizing*, and then, *Project Management* or execute the *BD64* transaction.
The Display Distribution Model window opens.
2. Select *Create model view*. (If required, switch processing mode to edit within Distribution Model/Switch Processing Mode.)
3. Type a short text string and a technical name for your new model view.
4. Click the *Save* button.

The Distribution Model Changed window opens with a tree structure of the distribution model as shown in the following image.



5. In the Distribution Model tree, select a new model view.
6. On the right, select *Add message type*.

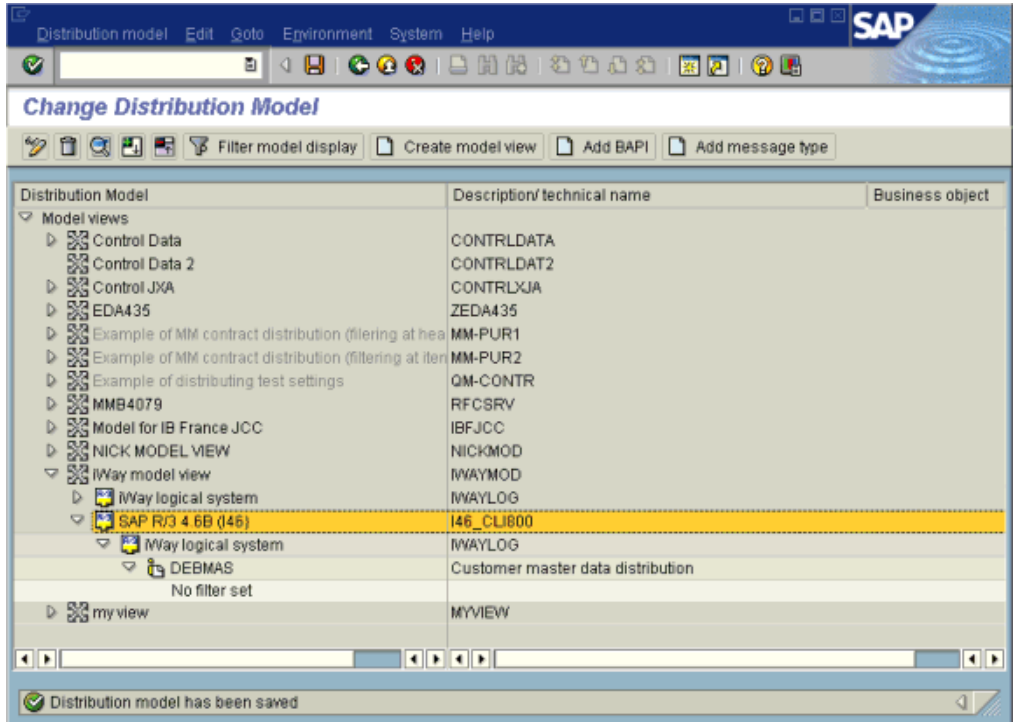
The Add Message Type pane opens and displays fields for specifying the sender and receiver of the message, as well as the message type as shown in the following image.



- a. In the Sender field, provide the sender that points to the SAP system that sends the IDoc, for example, I46_CLI800.
In this case, the sender is an SAP 4.6B system
- b. In the Receiver field, provide the logical system, for example, IWAYLOG.

- c. In the Message type field, provide the type of IDoc, for example, DEBMAS.
7. Click the check mark icon.
8. Click the Save button.

The Change Distribution Model window opens and displays the new model view to use to send message type, DEBMAS, from the I46_CLI800 SAP system to the IWAYLOG logical system as shown in the following image.



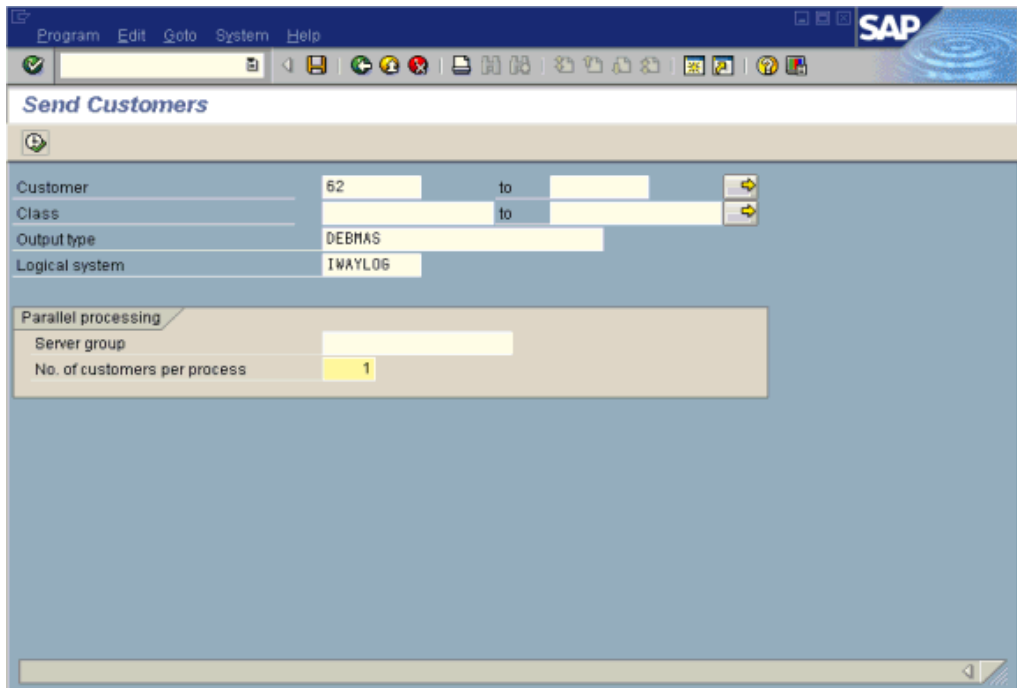
You are now ready to test the connection to the logical system.

Testing the SAP ALE Configuration

In the SAP Server, the BD12 transaction enables you to send IDocs to any logical system, for example, to an event adapter.

Procedure How to Test the SAP ALE Configuration

The following image shows the Send Customers window where you test the message type. It includes fields for Customer, Class, Output type, and Logical system. The Parallel processing pane includes a field for Server group and a field for the number of customers per process.

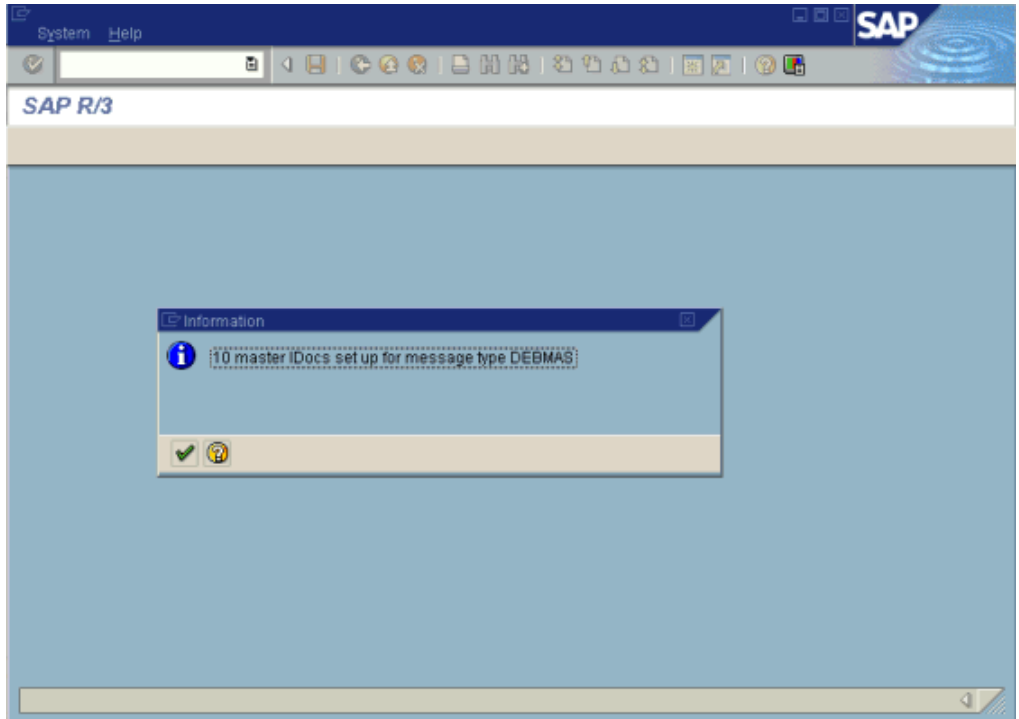


To test the SAP Application Link Enabling (ALE) configuration:

1. In the Send Customers window, type the IDoc message type *DEBMAS* in the Output type field.
2. In the Logical system field, type the logical system, for example, *IWAYLOG*.
3. To transfer data, click *Run*.

Testing the SAP ALE Configuration

The SAP event adapter receives the IDoc in XML format. No response is expected from the event adapter. A window opens and confirms the message entered in previous screens as shown in the following image.



CHAPTER 9

Troubleshooting and Error Messages

Topics:

- Troubleshooting
- Error Messages in Application Explorer
- Error Messages in SAP
- Error Messages in JCA
- Error Messages in iBSE

The following topics explain limitations and workarounds when connecting to SAP. The adapter-specific errors described in this section can arise whether you are using the adapter with a JCA or with an iBSE configuration.

Troubleshooting

An event in SAP is defined as an occurrence of a status change in an object. Events are created when the relevant status change occurs. Event creation must be implemented by SAP or yourself. An event is created from specific application programs (the event creator) and then “published” system-wide. Any number of receivers can respond to the event with their own “response mechanisms”. An event is usually defined as a component of an object type.

This topic provides troubleshooting information for SAP for the following four categories:

- Application Explorer
- SAP
- JCA
- iBSE

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

- The JCA trace information can be found under the following directory:

`C:\Program Files\iWay55\config\base\log`

- iBSE trace information can be found under the following directory:

`C:\Program Files\iWay55\ibse\ibselogs`

- The log file for Application Explorer can be found under the following directory:

`C:\Program File\iWay55\tools\iwae\bin`

Error Messages in Application Explorer

The following table lists errors and solutions when using Application Explorer with the adapter.

Error	Solution
Cannot connect to the adapter from Application Explorer.	Ensure that: <ul style="list-style-type: none">• SAP is running.• The Application Server name, System Number, and Client Number are correct.• The SAP user ID and password are correct.

Error	Solution
<p>Cannot connect to the SAP target through Application Explorer. The following error message appears:</p> <pre data-bbox="267 378 669 458">Error getting target [SAP] - java.lang.Exception: Error Logon to SAP System</pre>	<p>Ensure that you enter the correct connection parameters when connecting to the SAP target.</p>
<p>SAP does not appear in the Application Explorer Adapter node list.</p>	<p>Ensure that the sapjco.jar and sapjcorfc.dll files are added to the lib directory. Ensure the librfc32.dll file is added to the Windows system32 folder.</p>
<p>Cannot connect to your SAP system through Application Explorer. The following error message appears:</p> <pre data-bbox="267 758 701 1394">Problem activating adapter. (com.ibi.sapr3.SapAdapterException: com.sap.mw.jco.JCO\$Exception: (102) RFC_ERROR_COMMUNICATION: Connect to SAP gateway failed Connect_PM GWHOST=isdsrv8, GWSERV=sapgw00, ASHOST=isdsrv8, SYSNR=00 LOCATION CPIC (TCP/IP) on local host ERROR partner not reached (host isdsrv8, service 3300) TIME Fri Aug 27 11:49:14 2004 RELEASE 620 COMPONENT NI (network interface) VERSION 36 RC -10 MODULE ninti.c LINE 979 DETAIL NipConnect2 SYSTEM CALL SO_ERROR ERRNO 10061 ERRNO TEXT WSAECONNREFUSED: Connection refused COUNTER 1). Check logs for more information.</pre>	<p>Ensure that SAP is running and that the parameter values to connect to your Application Server are correct.</p>

Error	Solution
<p>Cannot connect to your SAP system through Application Explorer even though SAP is running. The following error message appears:</p> <pre> Problem activating adapter. (com.ibi.sapr3.SapAdapterException: java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer 'com.sap.mw.jco.rfc.MiddlewareRFC' JCO.nativeInit(): Could not initialize dynamic link library sapjcorfc [no sapjcorfc in java.library.path]. java.library.path </pre>	<p>Ensure that the sapjcorfc.dll file is added to the lib directory, and the librfc32.dll file is added to the Windows system32 folder.</p>
<p>The DLL is loaded in another class loader (iBSE and JCA are installed on the same server). The following error message appears:</p> <pre> com.ibi.sapr3.SapAdapterException: java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer 'com.sap.mw.jco.rfc.MiddlewareRFC' JCO.nativeInit(): Could not initialize dynamic link library sapjcorfc [Native Library F:\iWay55.008.0628\lib\sapjcorfc.dll already loaded in another classloader]. java.library.path </pre>	<p>Ensure that the sapjco.jar file is added to the server class path.</p>

Error Messages in SAP

The following table lists errors and solutions when using the adapter.

Error	Solution
<p>When executing a request, the following error message appears:</p> <pre>AdapterException: java.lang.Exception: Function module CUSTOMER_GETDETAIL2 does NOT exist.</pre>	<p>Check the syntax of your input XML document and ensure the name of the Remote Function module is correct and is available in SAP.</p>
<p>When executing a request, the following error message appears:</p> <pre>AdapterException: java.lang.Exception: Object type unknown for business object: CUST</pre>	<p>Check the syntax of your input XML document and ensure the Object type is correct.</p>
<p>When executing a request, the following error message appears:</p> <pre>AdapterException: java.lang.Exception: Unable to retrieve BAPI name for: CUSTOMER.DETAIL2</pre>	<p>Check the syntax of your input XML document and ensure the name of the BAPI is correct and is available in SAP.</p>
<p>When executing a request, the following error message appears:</p> <pre>java.lang.RuntimeException: com.sap.mw.jco.JCO\$AbapExcept ion: (126) OBJECT_UNKNOWN: Basic type or extension does not exist.</pre>	<p>Check the syntax of your input XML document and ensure the IDoc extension is correct and is available in SAP.</p>
<p>When executing a request, the following error message appears:</p> <pre>AdapterException: java.lang.Exception: BapiError/BapiAbort: You are not authorized to display customers.</pre>	<p>Make sure your user ID has the correct permissions configured in SAP. For more information, consult your SAP administrator.</p>

Error Messages in JCA

The following table lists an error and solution when using JCA.

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

Error Messages in iBSE

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

General Error Handling in iBSE

The iWay Business Services Engine (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, the way 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, when 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 the previous 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. Because adapters use the target system interfaces and APIs, whether 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.

Although it is almost impossible to anticipate every error condition that an adapter may encounter, the following examples describe how adapters handle common error conditions and how they are then exposed to the Web services consumer application.

Example: iWay Application System Adapter for SAP R/3 Invalid SOAP Request

When the iWay Application System Adapter for SAP R/3 receives a SOAP request message that does not conform to the WSDL for the Web service being executed, the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
<SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>Error processing agent [XDSapIfrAgent] - XD[FAIL]
SapIFRException: java.sql.SQLException:
com.ibi.sapjco.SapCallableStatement: execute() j
java.util.NoSuchElementException</faultstring>
</SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: Empty Result From SOAP Request

When the iWay Application System Adapter for SAP R/3 executes an SAP object as a Web service using input parameters passed in the SOAP request message that do not match records in SAP, the following SOAP response is generated:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
<SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>Error processing agent [XDSapIfrAgent] - XD[FAIL]
SapIFRException: java.sql.SQLException:
com.ibi.sapjco.SapCallableStatement: execute() java.sql.SQLException: JCO
Error Key: NO_RECORD_FOUND Short Description:
com.sap.mw.jco.JCO$AbapException: (126) NO_RECORD_FOUND:
NO_RECORD_FOUND</faultstring>
</SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: Failure to Connect to SAP

When the iWay Application System Adapter for SAP R/3 cannot connect to SAP when executing a Web service, the following SOAP response is generated:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>Error processing agent [XDSapIfrAgent] - XD[RETRY] Connect
to SAP gateway failed Connect_PM GWHOST=ESDSUN, GWSERV=sapgw00,
ASHOST=ESDSUN, SYSNR=00 LOCATION CPIC (TCP/IP) on local host ERROR
partner not reached (host ESDSUN, service 3300) TIME Mon Jun 30 16:01:02
2003 RELEASE 620 COMPONENT NI (network interface) VERSION 36 RC -10
MODULE ninti.c LINE 976 DETAIL NiPConnect2 SYSTEM CALL SO_ERROR ERRNO
10061 ERRNO TEXT WSAECONNREFUSED: Connection refused COUNTER
1</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: Invalid SOAP Request

When the iWay Application System Adapter for SAP R/3 receives a SOAP request message that does not conform to the WSDL for the Web services being executed, the following SOAP response is generated:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>RPC server connection failed: Connection refused:
connect</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: Empty Result From a Request

Note: The condition for this adapter does not yield a SOAP fault.

When the iWay Application System Adapter for SAP R/3 executes a SOAP request using input parameters passed that do not match records in the target system, the following SOAP response is generated:

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
    <m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
xmlns="urn:schemas-iwaysoftware-com:iwse"
cid="2A3CB42703EB20203F91951B89F3C5AF">
      <RunDBQueryResult run="1" />
    </m:RunDBQueryResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

APPENDIX A

Sample Files and Coding Techniques

Topics:

- Sample RFC Request Document
- Sample RFC Response Document
- Sample IDOC XML for Message Type DEBMAS
- Sample RFC Module
- Sample Wrapper Module to Call Functions on Remote Destinations (Service)
- Using Staging BAPIs to Retrieve SAP BW Metadata

This section provides sample request and response documents sent between SAP and Sun Java System Application Server. It also includes a sample RFC module and a sample wrapper module to call functions on remote destinations.

Sample RFC Request Document

The following sample code shows a sample RFC request document.

```
<?xml version="1.0" ?>
<doc:RFC_WALK_THRU_TEST xmlns:doc="urn:sapcom:document:sap:business:rfc">
  <TEST_IN>
    <RFCFLOAT>0.0</RFCFLOAT>
    <RFCCHAR1></RFCCHAR1>
    <RFCINT2>0</RFCINT2>
    <RFCINT1>0</RFCINT1>
    <RFCCHAR4></RFCCHAR4>
    <RFCINT4>10</RFCINT4>
    <RFCHEX3>000000</RFCHEX3>
    <RFCCHAR2></RFCCHAR2>
    <RFCTIME>10:09:32</RFCTIME>
    <RFCDATE>2001-09-05</RFCDATE>
    <RFCDATA1>Hello World</RFCDATA1>
    <RFCDATA2></RFCDATA2>
  </TEST_IN>
  <DESTINATIONS>
</DESTINATIONS>
  <LOG>
</LOG>
</doc:RFC_WALK_THRU_TEST>
```

Sample RFC Response Document

The following sample code shows a sample RFC response document.

```
<?xml version="1.0" ?>
<doc:RFC_WALK_THRU_TEST.Response
xmlns:doc="urn:sapcom:document:sap:business:rfc">
```

```
<TEST_OUT>
  <RFCFLOAT>0.0</RFCFLOAT>
  <RFCCHAR1></RFCCHAR1>
  <RFCINT2>0</RFCINT2>
  <RFCINT1>0</RFCINT1>
  <RFCCHAR4></RFCCHAR4>
  <RFCINT4>10</RFCINT4>
  <RFCHEX3>000000</RFCHEX3>
  <RFCCHAR2></RFCCHAR2>
  <RFCTIME>10:09:32</RFCTIME>
  <RFCDATE>2001-09-05</RFCDATE>
  <RFCDATA1>Hello World</RFCDATA1>
  <RFCDATA2></RFCDATA2>
</TEST_OUT>
<DESTINATIONS>
</DESTINATIONS>
<LOG>
</LOG>
</doc:RFC_WALK_THRU_TEST.Response>
```

Sample IDOC XML for Message Type DEBMAS

The following sample code shows a sample IDoc XML document for the DEBMAS message type.

```
<?xml version="1.0" ?>
<DEBMAS01>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>800</MANDT>
    <DOCNUM>000000000236015</DOCNUM>
      <DOCREL>46C</DOCREL>
      <STATUS>30</STATUS>
      <DIRECT>1</DIRECT>
      <OUTMOD>2</OUTMOD>
      <EXPRS<>/EXPRS>
      <TEST></TEST>
      <IDOC TYP>DEBMAS01</IDOC TYP>
      <CIM TYP></CIM TYP>
      <MESTYP>DEBMAS</MESTYP>
      <MESCOD></MESCOD>
      <MESFCT></MESFCT>
      <STD></STD>
      <STDVRS></STDVRS>
      <STDMES></STDMES>
      <SNDPOR>SAPI46</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPFC></SNDPFC>
      <SNDP RN>I46_CLI800</SNDP RN>
      <SNDSAD></SNDSAD>
      <SNLAD></SNLAD>
      <RCVPOR>A000000018</RCVPOR>
      <RCVPRT>LS</RCVPRT>
      <RCVPFC></RCVPFC>
      <RCVPRN>SAMP</RCVPRN>
      <RCVSAD></RCVSAD>
      <RCVLAD></RCVLAD>
      <CRE DAT>2001-09-04</CRE DAT>
      <CRE TIM>16:44:52</CRE TIM>
      <REF INT></REF INT>
      <REFGRP></REFGRP>
      <REFMES></REFMES>
      <ARCKEY></ARCKEY>
      <SERIAL>20010904164452</SERIAL>
    </EDI_DC40>
```



```

<E1KNA1M SEGMENT="1">
  <MSGFN>005</MSGFN>
  <KUNNR>000000001</KUNNR>
  <ANRED></ANRED>
  <AUFSO></AUFSO>
  <BAHNE></BAHNE>
  <BAHNS></BAHNS>
  <BBBNR>000000</BBBNR>
  <BBSNR>00000</BBSNR>
  <BEGRU></BEGRU>
  <BRSCHE></BRSCHE>
  <BUBKZ>0</BUBKZ>
  <DATLT></DATLT>
  <FAKSD></FAKSD>
  <FISKN></FISKN>
  <KNRZA></KNRZA>
  <KONZS></KONZS>
  <KTOKD>0001</KTOKD>
  <KUKLA></KUKLA>
  <LAND1>US</LAND1>
  <LIFNR></LIFNR>
  <LIFSD></LIFSD>
  <LOCCO></LOCCO>
  <LOEVM></LOEVM>
  <NAME1>Apple Corp</NAME1>
  <NAME2></NAME2>
  <NAME3></NAME3>
  <NAME4></NAME4>
  <NIELS></NIELS>
  <ORT01>Floral Park</ORT01>
  <ORT02></ORT02>
  <PFACH></PFACH>
  <PSTL2></PSTL2>
  <PSTLZ>10010</PSTLZ>
  <REGIO>NY</REGIO>
  <COUNC></COUNC>
  <CITYC></CITYC>
  <RPMKR></RPMKR>
  <SORTL>APPLE</SORTL>
  <SPERR></SPERR>
  <SPRAS>E</SPRAS>
  <STCD1></STCD1>
  <STCD2></STCD2>
  <STKZA></STKZA>
  <STKZU></STKZU>
  <STRAS>123 Main street</STRAS>

```

Sample IDOC XML for Message Type DEBMA5

```
<TELBX></TELBX>
<TELF1></TELF1>
<TELF2></TELF2>
<TELFX></TELFX>
<TELTX></TELTX>
<TELX1></TELX1>
<LZONE>000000001</LZONE>
<XZEMP></XZEMP>
<VBUND></VBUND>
<STCEG></STCEG>
<GFORM></GFORM>
<BRAN1></BRAN1>
<BRAN2></BRAN2>
<BRAN3></BRAN3>
<BRAN4></BRAN4>
<BRAN5></BRAN5>
<UMJAH>0000</UMJAH>
<UWAER></UWAER>
<JMZAH>000000</JMZAH>
<JMJAH>0000</JMJAH>
<KATR1></KATR1>
<KATR2></KATR2>
<KATR3></KATR3>
<KATR4></KATR4>
<KATR5></KATR5>
<KATR6></KATR6>
<KATR7></KATR7>
<KATR8></KATR8>
<KATR9></KATR9>
<KATR10></KATR10>
<STKZN></STKZN>
<UMSA1>0</UMSA1>
<TXJCD></TXJCD>
<PERIV></PERIV>
<KTOCD></KTOCD>
<PFORT></PFORT>
<DTAMS></DTAMS>
<DTAWS></DTAWS>
<HZUOR>00</HZUOR>
<CIVVE>X</CIVVE>
<MILVE></MILVE>
<SPRAS_ISO>EN</SPRAS_ISO>
<FITYP></FITYP> |
<STCDT></STCDT>
<STCD3></STCD3>
<STCD4></STCD4>
<XICMS></XICMS>
<CFOPC></CFOPC>
```

```
<TXLW1></TXLW1>
<TXLW2></TXLW2>
<CCC01></CCC01>
<CCC02></CCC02>
<CCC03></CCC03>
<CCC04></CCC04>
<CASSD></CASSD>
<KDKG1></KDKG1>
<KDKG2></KDKG2>
<KDKG3></KDKG3>
<KDKG4></KDKG4>
<KDKG5></KDKG5>
<NODEL></NODEL>
<XSUB2></XSUB2>
<WERKS></WERKS>

<E1KNVVM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <VKORG>0001</VKORG>
  <VTWEG>01</VTWEG>
  <SPART>01</SPART>
  <BEGRU></BEGRU>
  <LOEVM></LOEVM>
  <VERSG></VERSG>
  <AUFSD></AUFSD>
  <KALKS>1</KALKS>
  <KDGRP></KDGRP>
  <BZIRK></BZIRK>
  <KONDA></KONDA>
  <PLTYP></PLTYP>
  <AWAHR>100</AWAHR>
  <INCO1></INCO1>
  <INCO2></INCO2>
  <LIFSD></LIFSD>
  <AUTLF></AUTLF>
  <ANTLF>9</ANTLF>
  <KZTLF></KZTLF>
  <KZAZU>X</KZAZU>
  <CHSPL></CHSPL>
  <LPRIO>00</LPRIO>
  <EIKTO></EIKTO>
  <VSBED>01</VSBED>
  <FAKSD></FAKSD>
  <MRNKZ></MRNKZ>
  <PERFK></PERFK>
  <PERRL></PERRL>
  <WAERS>EUR</WAERS>
  <KTGRD></KTGRD>
```

Sample IDOC XML for Message Type DEBMAS

```
<ZTERM></ZTERM>
<VWERK></VWERK>
<VKGRP></VKGRP>
<VKBUR></VKBUR>
<VSORT></VSORT>
<KVGR1></KVGR1>
<KVGR2></KVGR2>
<KVGR3></KVGR3>
<KVGR4></KVGR4>
<KVGR5></KVGR5>
<BOKRE></BOKRE>
<KURST></KURST>
<PRFRE></PRFRE>
<KLABC></KLABC>
<KABSS></KABSS>
<KKBER></KKBER>
<CASSD></CASSD>
<RDOFF></RDOFF>
<AGREL></AGREL>
<MEGRU></MEGRU>
<UEBTO>0.0</UEBTO>
<UNTTO>0.0</UNTTO>
<UEBTK></UEBTK>
<PVKSM></PVKSM>
<PODKZ></PODKZ>
<PODTG>0</PODTG>
<E1KNVPM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <PARVW>AG</PARVW>
  <KUNN2>000000001</KUNN2>
  <DEFPA></DEFPA>
  <KNREF></KNREF>
  <PARZA>000</PARZA>
</E1KNVPM>
<E1KNVPM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <PARVW>RE</PARVW>
  <KUNN2>000000001</KUNN2>
  <DEFPA></DEFPA>
  <KNREF></KNREF>
  <PARZA>000</PARZA>
</E1KNVPM>
```

```

<E1KNVPM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <PARVW>RG</PARVW>
  <KUNN2>000000001</KUNN2>
  <DEFPA></DEFPA>
  <KNREF></KNREF>
  <PARZA>000</PARZA>
</E1KNVPM>

<E1KNVPM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <PARVW>WE</PARVW>
  <KUNN2>000000001</KUNN2>
  <DEFPA></DEFPA>
  <KNREF></KNREF>
  <PARZA>000</PARZA>
</E1KNVPM>

<E1KNVIM SEGMENT="1">
  <MSGFN>005</MSGFN>
  <ALAND>DE</ALAND>
  <TATYP>MWST</TATYP>
  <TAXKD>0</TAXKD>
</E1KNVIM>

</E1KNVVM>
</E1KNA1M>
</IDOC>
</DEBMAS01

```

Collected IDocs

When using collected IDocs on any platform during inbound processing (service mode), if the DOCNUM field does not have a unique document number for each IDoc, the system creates an IDoc for each header record in the collected IDoc file and duplicates the data for each IDoc.

Ensure the DOCNUM field is included in the EDI_DC40 structure and that each IDoc has a unique sequence number within the collected IDoc file.

Sample RFC Module

After you have configured the SAP event adapter and the RFC destination, you can write ABAP code to execute calls at your new destination (the event adapter).

The following sample code uses a user-defined RFC module called Z_EVENT_DISPATCH.

```
FUNCTION Z_01_EVENT_DISPATCH.  
CALL FUNCTION 'Z_EVENT_DISPATCH'  
  DESTINATION 'IWAYDEST'  
  EXPORTING  
    EVENT = EVENT  
    RECTYPE = RECTYPE  
    OBJTYPE = OBJTYPE  
    OBJKEY = OBJKEY  
  TABLES  
    EVENT_CONTAINER = EVENT_CONTAINER.  
ENDFUNCTION.
```

Sample Wrapper Module to Call Functions on Remote Destinations (Service)

This topic describes how to invoke a service that employs SAP remote data. For example, you can use this technique to write a function using C on a UNIX server that queries an Informix database and returns the response to SAP.

The ABAP command, CALL FUNCTION, takes as an argument, DESTINATION. Using RFC (Remote Function Call) destinations, programs can be executed on external systems, and the results can be returned into SAP function module programs. For more information on this functionality, see your SAP documentation, which is available at the following URL:

<http://help.sap.com>

Since DESTINATION is not part of an individual BAPI (Business Application Programming Interface) or Remote Function Module (RFM), but a parameter of the SAP function mechanism, you require a wrapper module to invoke it as a service. In addition, you must invoke the wrapper module in place of the original function.

The wrapper module is written using the SAP ABAP/4 programming language and contains the same input and output parameters as the original function. You can obtain all of the parameters of a remote function in the function editor by selecting Edit and then, Pattern and entering the function name.

The destination inside the wrapper module must be a valid SAP RFC destination with an RFC Server program running on the remote host. For more information, see the SAP RFC Programming manual, which is available at the following URL:

<http://service.sap.com>

The RFC Server program must return the data to SAP in a format that follows the exact structure of the Remote Function interface, or an abnormal ending occurs in SAP.

The following is an example of a wrapper module for the SAP test function named RFC_CUSTOMER_GET.

```

FUNCTION Z_CALL_EXTERNAL.
*-----
*"*Local interface:
*  IMPORTING
*    VALUE(MYKUNNR) LIKE  KNA1-KUNNR DEFAULT SPACE
*    VALUE(MYNAME1) LIKE  KNA1-NAME1 DEFAULT SPACE
*  EXPORTING
*    VALUE(ERRORCODE) LIKE  SY-SUBRC
*  TABLES
*    MYCUSTOMER_T STRUCTURE  BRFCNA1
*-----
ERRORCODE = 0.
CALL FUNCTION 'RFC_CUSTOMER_GET'
  DESTINATION 'JRDEST'
  EXPORTING
    KUNNR           = MYKUNNR
    NAME1           = MYNAME1
  TABLES
    CUSTOMER_T     = MYCUSTOMER_T
EXCEPTIONS
  COMMUNICATION_FAILURE = -1
  SYSTEM_FAILURE       = -2
  NOTHING_SPECIFIED    = -3
  NO_RECORD_FOUND      = -4
  OTHERS               = -5.

CASE SY-SUBRC.
  WHEN 0.
    ERRORCODE = 0.
    EXIT.
  WHEN -1 .
    ERRORCODE = 1.
    EXIT.
  WHEN -2.
    ERRORCODE = 2.
    EXIT.
  WHEN -3.
    ERRORCODE = 3.
    EXIT.
  WHEN -4.
    ERRORCODE = 4.
    EXIT.
  WHEN -5.
    ERRORCODE = 99999.
    EXIT.
ENDCASE.

```

```
* IF SY-SUBRC <> 0.  
*ERRORCODE = SY-SUBRC.  
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
*       WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
*ENDIF.  
ENDFUNCTION.
```

Using Staging BAPIs to Retrieve SAP BW Metadata

The Staging (or Warehouse Management) BAPIs (Business Application Programming Interfaces) include methods to update and retrieve metadata for InfoObjects, InfoCubes, InfoObjectCatalogs, and the definition of InfoPackages, an SAP Business Warehouse (BW), from a third party tool.

By using these BAPIs, you can connect metadata repositories and extraction engines to the SAP Business Information Warehouse.

For complete documentation on the individual data structures and individual BAPI calls, see Business Information Warehouse available at the following URL:

<http://service.sap.com>

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