



iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide

Version 7.0.x and Higher

Active Technologies, EDA, EDA/SQL, FIDEL, FOCUS, Information Builders, the Information Builders logo, iWay, iWay Software, Parlay, PC/FOCUS, RStat, Table Talk, Web390, WebFOCUS, WebFOCUS Active Technologies, and WebFOCUS Magnify are registered trademarks, and DataMigrator and Hyperstage are trademarks of Information Builders, Inc.

Adobe, the Adobe logo, Acrobat, Adobe Reader, Flash, Adobe Flash Builder, Flex, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or 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 © 2018, 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.

Contents

Preface	7
Documentation Conventions	8
Related Publications	9
Customer Support	9
Help Us to Serve You Better	10
User Feedback	12
Information Builders Consulting and Training	12
1. Introducing the iWay Application Adapter for J.D. Edwards EnterpriseOne	. 13
Executing a J.D. Edwards EnterpriseOne Master Business Function	13
Application Adapters	14
J.D. Edwards EnterpriseOne Platforms, Products, and Releases Supported	14
J.D. Edwards EnterpriseOne Versions and Library Files	15
Accessing Data Stored in J.D. Edwards EnterpriseOne	16
Propagating External Listeners Into EnterpriseOne	16
Propagating Internal Listeners Out of EnterpriseOne	17
J.D. Edwards EnterpriseOne Interoperability Framework	17
J.D. Edwards EnterpriseOne Outbound Processing Framework	19
Deployment Information for Your iWay Adapter	20
iWay Service Manager	20
iWay Explorer	20
iWay Business Services Provider (iBSP)	21
Application Adapter for J.D. Edwards EnterpriseOne Information Roadmap	21
2. Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms	
Matrix	23
Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Overview	23
Supported J.D. Edwards EnterpriseOne Versions	24
Application Adapter for J.D. Edwards EnterpriseOne Operating Systems	25
Application Adapter for J.D. Edwards EnterpriseOne Databases	25
Java Development Kit (JDK)	25
Application Adapter for J.D. Edwards EnterpriseOne Communication Modes	25
J.D. Edwards EnterpriseOne Object Types and Interfaces	25

Communication Types	
Application Adapter for J.D. Edwards EnterpriseOne Operations	
Application Adapter for J.D. Edwards EnterpriseOne Data Types	26
Other Application Adapter for J.D. Edwards EnterpriseOne Functions	27
Application Adapter for J.D. Edwards EnterpriseOne Known Limitations	
Related Information for the Application Adapter for J.D. Edwards EnterpriseOne in Specif	ïc
iWay Releases	27
3. Application Adapter for J.D. Edwards EnterpriseOne Quick Start Guide	29
Application Adapter for J.D. Edwards EnterpriseOne Quick Start Overview	
J.D. Edwards EnterpriseOne Quick Start Guide	
4. Configuring Application Adapter for J.D. Edwards EnterpriseOne Targets and	d
Creating XML Schemas	31
Application Adapter for J.D. Edwards EnterpriseOne Target and XML Schema Overview	
Using GenJava to Generate a Schema	31
Sample GenJava Syntax	32
Starting iWay Explorer	33
Adding the J.D. Edwards EnterpriseOne Adapter to iWay Explorer	38
Working With a Target	40
Creating an XML Schema	50
Creating XML Request Documents	59
5. Creating and Publishing iWay Business Services	65
Understanding iWay Business Services	65
Creating iWay Business Services	65
Creating Business Services With iWay Explorer	66
Connecting to the J.D. Edwards EnterpriseOne Client	
6. Listening for Database Events	81
Understanding Event Functionality	81
7. Application Adapter for J.D. Edwards EnterpriseOne Troubleshooting	97
J.D. Edwards EnterpriseOne Troubleshooting	
Error Messages in iWay Explorer	97
Error Messages in J.D. Edwards EnterpriseOne	
Error Messages in iWay Business Services Provider	

General Error Handling in iBSP	
A. Configuring the Application Adapter for J.D. Edwards EnterpriseOne in an	iWay
Environment	103
Configuring and Deploying the iWay Application System Adapter for J.D. Edwards	
EnterpriseOne	
B. Configuring the Application Adapter for J.D. Edwards EnterpriseOne in iW	ay
Integration Tools Designer	
Using the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools	3
Designer	131
C. Configuring EnterpriseOne for Outbound Transaction Processing	153
Specifying Outbound Functionality for a Business Function	
Outbound Transaction Processing.	153
The Data Export Control Table and the Processing Log Table	159
Configuring an Event Listener for the iWay Application Adapter for J.D. Edwards	
EnterpriseOne	
Configuring the iwoevent.cfg File	
Configuring the Event Stub	161
XML List Method Support	
List Retrieval Engine Table Conversion Wrapper	
Modifying the EnterpriseOne jde.ini File	
D. Sample J.D. Edwards EnterpriseOne Files	
Issuing a Single-Function Request	165
Issuing a Multiple-Function Request	
Sample Sales Order Request	172
Sample Sales Order Response	174

Contents

Preface

This document is written for system integrators with programming backgrounds and an understanding of J.D. Edwards EnterpriseOne in an application space. Extensive knowledge of EnterpriseOne is not required but may be helpful in learning about the adapter.

This document describes how to work with the adapter tools to develop online interconnections to EnterpriseOne. For system integrators concerned with the development of a client/server interface between EnterpriseOne and other applications, this guide addresses the EnterpriseOne integration aspects. It does not cover other applications or application wrappers.

Note: This Release 7.0.x content is currently being updated to support iWay Release 8.0.x software. In the meantime, it can serve as a reference for your use of iWay Release 8. If you have any questions, please contact *Customer_Success@ibi.com*.

How This Manual Is Organized

	Chapter/Appendix	Contents
1	Introducing the iWay Application Adapter for J.D. Edwards EnterpriseOne	Introduces the iWay Application Adapter for J.D. Edwards EnterpriseOne.
2	Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Matrix	Specifies version, platform, and database support information for iWay Application Adapter for J.D. Edwards EnterpriseOne.
3	Application Adapter for J.D. Edwards EnterpriseOne Quick Start Guide	Provides a quick start guide for the iWay Application Adapter for J.D. Edwards EnterpriseOne.
4	Configuring Application Adapter for J.D. Edwards EnterpriseOne Targets and Creating XML Schemas	Describes how to create schemas and iWay Business Services for J.D. Edwards EnterpriseOne functions.
5	Creating and Publishing iWay Business Services	Describes how to create and publish iWay Business Services using iWay Explorer for the iWay Application Adapter for J.D. Edwards EnterpriseOne.

This manual includes the following chapters:

_	Chapter/Appendix	Contents
6	Listening for Database Events	This section describes how to use the iWay Application Adapter for J.D. Edwards EnterpriseOne to listen for events.
7	Application Adapter for J.D. Edwards EnterpriseOne Troubleshooting	Provides troubleshooting information for the iWay Application Adapter for J.D. Edwards EnterpriseOne.
A	Configuring the Application Adapter for J.D. Edwards EnterpriseOne in an iWay Environment	Describes how to configure the adapter in the Service Manager console.
В	Configuring the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools Designer	Describes how to configure the adapter in iWay Integration Tools (iIT) Designer.
С	Configuring EnterpriseOne for Outbound Transaction Processing	Describes how to enable outbound transaction processing in EnterpriseOne and how to modify the jde.ini file for XML and XML List support.
D	Sample J.D. Edwards EnterpriseOne Files	Provides examples of the jdeRequest and jdeResponse XML structures for executing business functions within J.D. Edwards EnterpriseOne.

Documentation Conventions

The following table describes the documentation conventions that are used in this manual.

Convention	Description
THIS TYPEFACE or this typeface	Denotes syntax that you must enter exactly as shown.
this typeface	Represents a placeholder (or variable), a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option that you can click or select.
underscore	Indicates a default setting.
Key + Key	Indicates keys that you must press simultaneously.

Convention	Description
{}	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 ().
· ·	Indicates that there are (or could be) intervening or additional commands.

Related Publications

Visit our Technical Documentation Library at *http://documentation.informationbuilders.com*. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have any questions about this product?

Join the Focal Point community. Focal Point is our online developer center and more than a message board. It is an interactive network of more than 3,000 developers from almost every profession and industry, collaborating on solutions and sharing tips and techniques. Access Focal Point at http://forums.informationbuilders.com/eve/forums.

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

Call Information Builders Customer Support Services (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 questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code number (*xxxx.xx*) when you call.

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

Help Us to Serve You Better

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

The following tables list the environment information our consultants require.

Platform	
Operating System	
OS Version	
JVM Vendor	
JVM Version	

The following table lists the deployment information our consultants require.

Adapter Deployment	For example, JCA, Business Services Provider, iWay Service Manager
Container	For example, WebSphere
Version	
Enterprise Information System (EIS) - if any	
EIS Release Level	
EIS Service Pack	
EIS Platform	

The following table lists iWay-related information needed by our consultants.

iWay Adapter	
iWay Release Level	
iWay Patch	

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

Request/Question	Error/Problem Details or Information
Did the problem arise through a service or event?	
Provide usage scenarios or summarize the application that produces the problem.	
When did the problem start?	
Can you reproduce this problem consistently?	
Describe the problem.	
Describe the steps to reproduce the problem.	
Specify the error message(s).	
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?	

The following is a list of error/problem files that might be applicable.

- Input documents (XML instance, XML schema, non-XML documents)
- ❑ Transformation files
- Error screen shots
- Error output files
- Trace files
- Service Manager package to reproduce problem
- Custom functions and agents in use
- Diagnostic Zip
- Transaction log

For information on tracing, see the *iWay* Service Manager User's Guide.

User Feedback

In an effort to produce effective documentation, the Technical Content Management staff welcomes your opinions regarding this document. Please use the Reader Comments form at the end of this document to communicate your feedback to us or to suggest changes that will support improvements to our documentation. You can also contact us through our website, *http://documentation.informationbuilders.com/connections.asp*.

Thank you, in advance, for your comments.

Information Builders Consulting and Training

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

For information on course descriptions, locations, and dates, or to register for classes, visit our website (*http://education.informationbuilders.com*) or call (800) 969-INFO to speak to an Education Representative.

Chapter

Introducing the iWay Application Adapter for J.D. Edwards EnterpriseOne

The iWay Application Adapter for J.D. Edwards EnterpriseOne provides a means to exchange real-time business data between EnterpriseOne systems and other applications, databases, or external business partner systems. The adapter enables inbound and outbound processing with EnterpriseOne.

This section provides information about the iWay Application Adapter for J.D. Edwards EnterpriseOne to help you accomplish your integration projects.

Note: J.D. Edwards EnterpriseOne was formerly called OneWorld.

In this chapter:

- Executing a J.D. Edwards EnterpriseOne Master Business Function
- J.D. Edwards EnterpriseOne Platforms, Products, and Releases Supported
- □ J.D. Edwards EnterpriseOne Versions and Library Files
- Accessing Data Stored in J.D. Edwards EnterpriseOne
- J.D. Edwards EnterpriseOne Interoperability Framework
- Deployment Information for Your iWay Adapter
- □ Application Adapter for J.D. Edwards EnterpriseOne Information Roadmap

Executing a J.D. Edwards EnterpriseOne Master Business Function

You can use the iWay Application Adapter for J.D. Edwards EnterpriseOne to invoke a J.D. Edwards EnterpriseOne Master Business Function (MBF), such as Address Book, Purchase Order, Sales Order, etc. You can also use the adapter as part of the iWay Business Services integration effort to connect EnterpriseOne with other EIS systems.

The adapter can receive an XML document and invoke one or more MBFs by using the J.D. Edwards EnterpriseOne ThinNet API.

Application Adapters

The iWay Application Adapter for J.D. Edwards EnterpriseOne is an application adapter. Application adapters connect one application to another when those applications were not originally designed to communicate with each other. Adapters are bidirectional, that is, they can send requests to an Enterprise Information System (EIS), as well as receive notification of events occurring in an EIS.

J.D. Edwards EnterpriseOne Platforms, Products, and Releases Supported

The following table indicates which combinations of adapter platforms and J.D. Edwards EnterpriseOne platforms are supported, and for each combination, which J.D. Edwards EnterpriseOne products and releases are supported.

J.D. Edwards EnterpriseOne Platform	J.D. Edwards EnterpriseOne Product and Release	
Windows, AS400, HP 9000/B, Sun	XE (B7333) from SP19 to SP23	
or IBM RS/6000Windows	ERP8.0(B7334)	
	□ EnterpriseOneB9(8.9)	
	EnterpriseOne 8.10 (with Tools release 8.93 and 8.94)	
	EnterpriseOne 8.11 (SP1 and Tools Release 8.95)	
	EnterpriseOne 8.12 (Tools Release 8.96 2.0 and 8.97)	
	□ EnterpriseOne 9.0 (Tools Release 8.98)	
	EnterpriseOne 9.10 (up to Tools Release 9.1.4.7)	

J.D. Edwards EnterpriseOne Versions and Library Files

The files are available at $\startent \$ Classes folder either on Enterprise Server or Fat Client.

J.D. Edwards EnterpriseOne Version	Required Library Files
XE (B7333)	Connector.jar and Kernel.jar
ERP8.0(B7334)	Connector.jar and Kernel.jar
EnterpriseOne 8.9 (B9)	Connector.jar, Kernel.jar, jdeutil.jar, and log4j.jar
EnterpriseOne 8.10	Connector.jar, Kernel.jar, jdeutil.jar, and log4j.jar
EnterpriseOne 8.11 (SP1 and Tools Release 8.95)	Connector.jar, Kernel.jar, jdeutil.jar, and log4j.jar
EnterpriseOne8.12 (Tools Release 8.96 2.0)	Connector.jar, log4j.jar, Base_JAR.jar, EventProcesser_EJB.jar, EventProcesser_JAR.jar., JdeNet_JAR.jar, and System_JAR.jar
EnterpriseOne8.12 (Tools Release 8.97)	Connector.jar, log4j.jar, Base_JAR.jar, EventProcesser_EJB.jar, EventProcesser_JAR.jar., JdeNet_JAR.jar, System_JAR.jar, commons-httpclinet-3.0.jar, jmxri.jar and ManagementAgent_JAR.jar
EnterpriseOne9.0 (Tools Release 8.98.1.3)	Connector.jar, log4j.jar, Base_JAR.jar, EventProcesser_EJB.jar, EventProcesser_JAR.jar., JdeNet_JAR.jar, System_JAR.jar, commons-httpclinet-3.0.jar, jmxri.jar and ManagementAgent_JAR.jar
EnterpriseOne 9.10 (up to Tools Release 9.1.4.7)	ApplicationAPIs_JAR.jar, Base_JAR.jar, commons- httpclinet-3.0.jar, commons-logging.jar, Connector.jar, EventProcesser_EJB.jar, EventProcesser_JAR.jar, JdeNet_JAR.jar, System_JAR.jar, jmxri.jar, ManagementAgent_JAR.jar, jmxremote_optional.jar, jmxremote.jar

Accessing Data Stored in J.D. Edwards EnterpriseOne

J.D. Edwards EnterpriseOne supports multiple methods and technologies to provide interoperability. The three supported entry points are:

Flat files

- Database tables
- □ Master Business Function (MBF) interactive calls

You configure the adapter to send requests to EnterpriseOne. The adapter processes requests for EnterpriseOne Master Business Functions (MBFs), embedded in XML documents, and forwards them to a back-end EnterpriseOne system. The resulting response information is then returned and processed for further routing.

The adapter can receive an XML request document from a client and call a specific function in the target Enterprise Information System (EIS). The adapter acts as a consumer of request messages and provides a response.

You can configure a listener, known as a channel, for the adapter to receive messages from EnterpriseOne. The information the listener receives is used to build an XML record and is forwarded to any specified disposition for further processing. Listeners are consumers of EIS-specific messages and may or may not provide a response.

Propagating External Listeners Into EnterpriseOne

When integrating external listeners into EnterpriseOne using flat file input, the files are imported through a batch program and placed on an unedited transaction table. The records on the transaction table are processed by a batch program that makes calls to the appropriate MBF.

The database table method bypasses the first step in the flat file method, and records are written directly to the unedited transaction table. The records on the transaction table are processed by a batch program that makes calls to the appropriate MBF.

The third method, calling the MBF directly, bypasses the batch processing completely and provides synchronous access to EnterpriseOne.

Propagating Internal Listeners Out of EnterpriseOne

Integrating an EnterpriseOne listener with external systems is similar to the inbound process, except in reverse. The Data Export Control table maintains the determination of whether a transaction must be integrated with an external system. When a transaction must be integrated, the MBF handles logging of all additions, changes, and deletions to the unedited transaction table. After the transaction information is written to the table, a key for that record is sent from the MBF to the subsystem data queue.

The subsystem data queue triggers the processing of the new record by launching an outbound subsystem batch process that is generic and handles all outbound transactions. The outbound subsystem then accesses the Data Export Control table to determine the configured external subscriber to run.

J.D. Edwards EnterpriseOne Interoperability Framework

J.D. Edwards EnterpriseOne provides for integration with systems through its interoperability framework. The adapter uses the EnterpriseOne framework and leverages various integration access methods to provide the greatest amount of flexibility and functionality.

The iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following integration access methods:

- □ J.D. Edwards EnterpriseOne ThinNet API
- □ J.D. Edwards EnterpriseOne XML
- J.D. Edwards unedited transaction tables (Z tables)

The following image diagrams the J.D. Edwards EnterpriseOne inbound processing (from the EIS to application server) framework. It shows the EnterpriseOne components and the agent adapter in the inbound processing sequence.



J.D. Edwards EnterpriseOne Inbound Processing Framework

The adapter uses the J.D. Edwards EnterpriseOne ThinNet API to communicate with the EnterpriseOne application. Using the ThinNet API, the adapter can invoke one or more Master Business Functions (MBFs) in a single Unit Of Work (UOW). When any of the MBF fail, the entire UOW fails, preventing partial updates. Because the adapter runs the MBF, validation of data, business rules, and communications to the underlying database are handled by the EnterpriseOne application.

J.D. Edwards EnterpriseOne Outbound Processing Framework

The following image diagrams the J.D. Edwards EnterpriseOne outbound processing framework. It shows the EnterpriseOne components and the listener adapter in the outbound processing sequence.



J.D. Edwards EnterpriseOne Outbound Processing Framework

In the outbound process, the event starts when a specific MBF is executed in the EnterpriseOne environment. The MBF writes the required information for the event into the appropriate interface table and then notifies the subsystem Batch Function (BF) that an event occurred. The subsystem BF then places an entry about the event on the Subsystem Data Queue.

The outbound subsystem retrieves the data queue entry and looks in the Data Export Control table for the external processes to notify. The outbound subsystem then calls the iWay Application System Adapter for J.D. Edwards EnterpriseOne listener with notification. The listener passes the notification to the generator. The generator then uses the J.D. Edwards EnterpriseOne ThinNet API to retrieve the appropriate information from the interface table.

Deployment Information for Your iWay Adapter

Your iWay adapter works in conjunction with one of the following components:

- □ iWay Service Manager
- iWay Business Services Provider (iBSP)

When hosted in an iWay environment, the adapter is configured through iWay Service Manager and iWay Explorer. iWay Explorer is used to configure system connections, create web services, and configure event capabilities. Service Manager can access this configuration information through the iWay7 repository to create a robust integration solution.

When the adapter is hosted in a third-party application server environment, you can configure iWay Explorer to work in a web services environment.

iWay Service Manager

iWay Service Manager is the heart of the Universal Adapter Framework and is an open transport service bus. Service Manager uses graphical tools to create sophisticated integration services without writing custom integration code by:

□ Creating metadata from target applications.

□ Transforming and mapping interfaces.

□ Managing stateless processes.

Its capability to manage complex adapter interactions makes it ideally suited to be the foundation of a service-oriented architecture.

iWay Explorer

iWay Explorer uses a tree metaphor to introspect a system for metadata. 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 a system. External applications that access a system through the adapter use either XML schemas or web services to pass data between the external application and the adapter.

iWay Business Services Provider (iBSP)

The iWay Business Services Provider (iBSP) exposes, as web services, enterprise assets that are accessible from adapters regardless of the programming language or the particular operating system.

iBSP 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.

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.

Application Adapter for J.D. Edwards EnterpriseOne Information Roadmap

The following table lists the location of deployment and user information for components of the iWay Application Adapter for J.D. Edwards EnterpriseOne.

Deployed Component	For more information, see
iWay Service Manager	Appendix A of this guide
	iWay Service Manager User's Guide
iWay Explorer	Chapters 2, 3, and 4 of this guide
	iWay Installation and Configuration
iWay Business Services Provider (iBSP)	iWay Installation and Configuration



Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Matrix

iWay Software is committed to support the diverse environments and varied systems of our users through support for leading enterprise applications, platforms, and databases.

This section specifies version, platform, and database support information for iWay Application Adapter for J.D. Edwards EnterpriseOne. It is designed to provide a consolidated view of J.D. Edwards EnterpriseOne releases and the various operating systems and databases, on which they are supported.

In this chapter:

- Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Overview
- Supported J.D. Edwards EnterpriseOne Versions
- Application Adapter for J.D. Edwards EnterpriseOne Operating Systems
- Application Adapter for J.D. Edwards EnterpriseOne Databases
- Java Development Kit (JDK)
- Application Adapter for J.D. Edwards EnterpriseOne Communication Modes

- □ J.D. Edwards EnterpriseOne Object Types and Interfaces
- Communication Types
- Application Adapter for J.D. Edwards EnterpriseOne Operations
- Application Adapter for J.D. Edwards EnterpriseOne Data Types
- Other Application Adapter for J.D.
 Edwards EnterpriseOne Functions
- Application Adapter for J.D. Edwards EnterpriseOne Known Limitations
- Related Information for the Application Adapter for J.D. Edwards EnterpriseOne in Specific iWay Releases

Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Overview

J.D. Edwards EnterpriseOne systems and applications that are supported by iWay Application Adapter for J.D. Edwards EnterpriseOne are governed by the underlying Tools Release version.

Note: J.D. Edwards EnterpriseOne versions are generally discussed and referenced by the following:

1. Application version

2. Tools Release version

The Tools Release is the development platform for J.D. Edwards EnterpriseOne applications. Oracle provides a support matrix, which identifies the mapping between J.D. Edwards EnterpriseOne application versions and J.D. Edwards EnterpriseOne Tools Release versions.

Supported J.D. Edwards EnterpriseOne Versions

iWay Application Adapter for J.D. Edwards EnterpriseOne supports J.D. Edwards EnterpriseOne versions as listed in the following table.

J.D. Edwards	One World	EnterpriseOne Application Releases					
Application Release	XE/ERP8	8.10	8.11	8.11 SP1	8.12	9.0	9.10
Tools Release	SP23/24	x	x	х	x	х	x
		8.93	x	х	x	x	x
		8.94	8.94	х	x	х	х
		8.95	8.95	8.95	x	x	x
		8.96	8.96	8.96	8.96	x	x
		8.97	8.97	8.97	8.97	x	x
		8.98	8.98	8.98	8.98	8.98	x
		8.98.1	8.98.1	8.98.1	8.98.1	8.98.1	x
		x	8.98.2	8.98.2	8.98.2	8.98.2	х
		x	x	х	x	x	9.10
		x	x	x	x	x	9.1.0.4
		x	x	x	x	x	9.1.4.7

Application Adapter for J.D. Edwards EnterpriseOne Operating Systems

iWay Application Adapter for J.D. Edwards EnterpriseOne supports all of the operating systems that are listed in the *iWay Installation and Configuration Guide* under *Operating System Requirements*.

Application Adapter for J.D. Edwards EnterpriseOne Databases

iWay Application Adapter for J.D. Edwards EnterpriseOne does not function directly with databases and only operates at the API level.

Java Development Kit (JDK)

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the Java Development Kit (JDK) versions that are listed in the *iWay Installation and Configuration Guide* under *Java Requirements*.

Application Adapter for J.D. Edwards EnterpriseOne Communication Modes

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following communication modes:

- □ Services (Outbound). iWay Application Adapter for J.D. Edwards EnterpriseOne can send messages to J.D. Edwards EnterpriseOne.
- **Events (Inbound).** iWay Application Adapter for J.D. Edwards EnterpriseOne can receive messages from J.D. Edwards EnterpriseOne.

J.D. Edwards EnterpriseOne Object Types and Interfaces

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following J.D. Edwards EnterpriseOne Object Types and Interfaces:

- **Business Functions.** Business functions are used for outbound communications from the J.D. Edwards EnterpriseOne Adapter to J.D. Edwards EnterpriseOne.
- **XML List.** XML List are used for inbound communications from J.D. Edwards EnterpriseOne to the J.D. Edwards EnterpriseOne Adapter.
- **UBE (invoke/run).** UBE are used for outbound communications from the J.D. Edwards EnterpriseOne Adapter to J.D. Edwards EnterpriseOne.
- **Z-Events.** Z- Events are used for inbound communications from J.D. Edwards EnterpriseOne to J.D. Edwards EnterpriseOne Adapter.

Communication Types

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following communication types:

- **Business Functions:** Synchronous
- **XML List:** Synchronous
- **UBE:** Asynchronous
- **Z-Events:** Synchronous

Application Adapter for J.D. Edwards EnterpriseOne Operations

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following operations:

Business Functions. All the operations supported by J.D. Edwards EnterpriseOne.

Note: Operations vary based on each of the Business functions.

- **XML List.** Retrieve data in chunks (Query).
- **UBE.** Invoke/Run.

Application Adapter for J.D. Edwards EnterpriseOne Data Types

iWay Application Adapter for J.D. Edwards EnterpriseOne supports the following data types:

- Character
- Date
- Integer
- Character (Blob)
- Binary (Blob)
- Binary
- String
- Variable String
- UTime
- Identifier (ID)

Numeric

Other Application Adapter for J.D. Edwards EnterpriseOne Functions

UOW (Unit Of Work)

iWay Application Adapter for J.D. Edwards EnterpriseOne can invoke two or more business functions in a single XML request call if those functions are to be executed in a particular session. For example, creating a Sales order that has the following business functions:

- BeginDoc
- EditLine
- EditDoc
- EndDo

Application Adapter for J.D. Edwards EnterpriseOne Known Limitations

Understand the features and limitations of each business function to pick a business function that provides the appropriate level of functionality for your application.

Related Information for the Application Adapter for J.D. Edwards EnterpriseOne in Specific iWay Releases

For more information, see the *iWay New Features Bulletin and Release Notes* documentation for a specific release (for example, iWay Version 7.0.2).



Application Adapter for J.D. Edwards EnterpriseOne Quick Start Guide

This chapter provides a quick start guide for the iWay Application Adapter for J.D. Edwards EnterpriseOne.

In this chapter:

- Application Adapter for J.D. Edwards EnterpriseOne Quick Start Overview
- □ J.D. Edwards EnterpriseOne Quick Start Guide

Application Adapter for J.D. Edwards EnterpriseOne Quick Start Overview

This quick start guide summarizes the high-level key steps that are required to install, configure, and use the iWay Application Adapter for J.D. Edwards EnterpriseOne. The quick start guide does not elaborate on any of the steps in detail. Instead, cross-references are provided for the corresponding sections in the *iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide*. Users of the iWay Application Adapter for J.D. Edwards EnterpriseOne are encouraged to follow the sequence of steps in this guide to quickly connect to a J.D. Edwards EnterpriseOne system and begin using the adapter. To gain a complete understanding about the adapter, it is recommended for users to review the entire *iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide*. Start guide section is not a replacement for that level of detail.

J.D. Edwards EnterpriseOne Quick Start Guide

This section lists and describes the key configuration steps for configuring the iWay Application Adapter for J.D. Edwards EnterpriseOne and then integrating with J.D. Edwards EnterpriseOne.

- 1. Ensure that you are using a supported environment, as described in *Application Adapter for J.D. Edwards EnterpriseOne Supported Platforms Matrix* on page 23.
- 2. Copy the library files to the \lib subdirectory where iWay Service Manager (iSM) is installed.

For more information, see *J.D. Edwards EnterpriseOne Versions and Library Files* on page 15.

3. Generate GenJava wrapper files and copy these files to the Repository folder.

For more information, see Using GenJava to Generate a Schema on page 31.

4. Open iWay Integration Tools (iIT) and access the iWay Explorer tab to create a new configuration.

For more information, see Starting iWay Explorer on page 33.

5. Add the iWay Application Adapter for J.D. Edwards EnterpriseOne to iWay Explorer.

For more information, see Adding the J.D. Edwards EnterpriseOne Adapter to iWay Explorer on page 38.

6. Create and configure an adapter target to your J.D. Edwards EnterpriseOne system.

For more information, see *Working With a Target* on page 40.

7. Examine the available metadata for your J.D. Edwards EnterpriseOne business functions that you want to integrate and create corresponding XML request and response documents.

For more information, see Creating an XML Schema on page 50.

- 8. Create an XML request document (payload) based on the generated XML request schema. For more information, see *Creating XML Request Documents* on page 59.
- 9. Create iWay Business Services (web services) based on the generated XML schema documents.

For more information, see Creating iWay Business Services on page 65.

10.Test the iWay Business Service (web service) that has been created using the sample XML request document.

For more information, see Sample J.D. Edwards EnterpriseOne Files on page 165 and How to Test an iWay Business Service on page 74.

11.Create a process flow that consists of an Adapter object based on your J.D. Edwards EnterpriseOne adapter target configured in iWay Explorer.

For more information, see *Configuring the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools Designer* on page 131.

12.Configure a channel using the iWay Service Manager (iSM) Administration Console, which can be deployed to retrieve incoming documents and produce output based on defined logic.

For more information, see *Configuring the Application Adapter for J.D. Edwards EnterpriseOne in an iWay Environment* on page 103.



Configuring Application Adapter for J.D. Edwards EnterpriseOne Targets and Creating XML Schemas

This section describes how to open a connection to J.D. Edwards EnterpriseOne, how to create schemas for EnterpriseOne functions, and how to create business services (web services) using iWay Explorer.

Note: This guide is specifically for J.D. Edwards EnterpriseOne. iWay Software also has an iWay Application Adapter for J.D. Edwards World.

In this chapter:

- Application Adapter for J.D. Edwards EnterpriseOne Target and XML Schema Overview
- Starting iWay Explorer
- Adding the J.D. Edwards EnterpriseOne Adapter to iWay Explorer
- Working With a Target
- Creating an XML Schema
- Creating XML Request Documents

Application Adapter for J.D. Edwards EnterpriseOne Target and XML Schema Overview

The iWay Application Adapter for J.D. Edwards EnterpriseOne enables the processing of EnterpriseOne business functions through the J.D. Edwards EnterpriseOne ThinNet API.

External applications that access EnterpriseOne through the iWay Application Adapter for J.D. Edwards EnterpriseOne use either XML schemas or web services to pass data between the external application and the adapter. The following topics describe how to use iWay Explorer to create XML documents and web services for the J.D. Edwards Master Business Functions (MBFs) used with the adapter.

For more information on creating web services and on iWay Explorer in general, see the *iWay Explorer User's Guide*.

Using GenJava to Generate a Schema

To create schemas for the adapter, you must use GenJava wrappers. You can create the GenJava wrappers using the EnterpriseOne utility called GenJava. You can use iWay Explorer to generate schemas against EnterpriseOne GenJava wrappers.

The J.D. Edwards OneWorld system administrator usually runs GenJava. When you run GenJava, you can specify a library of business functions, for example, CALLBSFN. GenJava creates Java class files for all the business functions and associated data structures. GenJava also compiles the business functions, generates Java docs, and packages them into two JAR files, as shown in the following list.

- Java classes
- Java documents

For example, if the library is CALLBSFN, you see the JDEJAVA_CALLBSFN.xml, JDEJAVA_CALLBSFNInterop.jar, and JDEJAVA_CALLBSFNInteropDoc.jar files in either the \\system\classes directory or any directory redirected by GenJava (JDE Client or Deployment Server).

GenJava can be generated either from Fat client or Deployment sever.

Required library files must be added to the CLASSPATH.

GenJava is supplied as a command line process with several run-time options, and is located in *<install>\system\bin32*.

For more information on GenJava, see the J.D. Edwards Interoperability Guide.

Sample GenJava Syntax

The following is a sample syntax of GenJava.

```
GenJava /Cat 1 /Cat 2 /Cat 3 /Cat - CALLBSFN
```

The example above generates Java wrappers for the following business functions in the CALLBSFN library:

- Category 1 (Master Business Functions)
- Category 2 (Major Business Functions)
- Category 3 (Minor Business Functions)
- Category (Uncategorized Business Functions)

You must use the correct information to log on to OneWorld, including the UserID, Password, and environment.

If the AddressBook.cmd is placed in the C: drive, then you can also use GenJava by running it with a JDEScript file, for example:

GenJava /cmd .\AddressBook.cmd

This prompts a OneWorld sign-on window for you to enter the user ID, password, and environment. The following syntax shows the AddressBook.cmd file.

```
define library CALLBSFN
login
library CALLBSFN
interface AddressBook
import B0100031
import B0100032
import B0100002
import B0100003
build
logout
```

GenJava generates the wrappers (CALLBSFNInterop.jar, CALLBSFNInteropDoc.jar and CALLBSFN.xml) in Java for all business functions imported in the script file.

Note: The generated files should be placed in a repository folder on the computer where iSM running, for example, c:\\Repository.

Starting iWay Explorer

This section describes how to start iWay Explorer.

Procedure: How to Open iWay Integration Tools

- 1. Navigate to your local drive where you have iIT installed, and open the *eclipse* folder.
- 2. Double-click iit.exe.



iWay Integration Tools suite opens, as shown in the following image.

Procedure: How to Create an iWay Explorer Connection to an iSM Server

This procedure assumes that you have opened iWay Integration Tools (iIT) and are in the Workbench.

- 1. Click the iWay Explorer tab to make it active.
- 2. Click the Launch iWay Resource Creator Wizard button on the tool bar.

In the following image, the iWay Explorer tab is active, and the cursor is pointing to the Launch iWay Resource Creator Wizard button.



When you click the button, the Resource Selection Dialog opens and displays the New iWay Connection pane, as shown in the following image.

🤞 Resource Selection Dial	og	
New iWay Connection		
Select a resource type to create	3.	
Туре	Description	Version
iWay Configuration	Create a connection to an adapter run-time instance.	6.1.6
0		
Ø	Next >	Cancel

- 3. Under the Type heading, click *iWay Configuration*, which is the type of resource that you are going to create.
- 4. Click Next.

The Add iWay Configuration dialog box opens and displays the Select Connection Types pane, as shown in the following image.

对 Add iWay Configuration	
Select Connection Types Please choose what connection method should be used to reach this host.	
Configuration Alias SampleConfig	
Connection Type HTTP Connection JCA Connection	
Connect to Host upon Wizard Completion	
Rext > Finish	Cancel

5. In the Configuration Alias field, type a name for the new configuration (for example, *SampleConfig*).

Tip: The name that you supply is used only for display purposes in the tree. It is not a server connection property.

6. For Connection Type, select HTTP Connection.

The connection to iWay Service Manager is made using HTTP.

7. Optionally, select the *Connect to Host upon Wizard Completion* check box if you want iWay Explorer to automatically connect to this instance of iSM after you have created it. If you select this option, all the explorer environments under the new iSM connection are automatically connected to iSM when this procedure is finished.

If you do not select this option, the explorer environments are not automatically connected to iSM. You can connect to an individual explorer environment when you want to access it.

8. Click *Next* to continue the procedure.
If you selected an HTTP Connection, the Enter Connection Information pane opens, as shown in the following image.

d Add iWay Config	uration	
Enter Connection I Provide the server's cor port.	nformation	ort and the iSM console
Connection String:	http://SampleConfig	
User Name:	iway	
Password:	••••	
SOAP Port/Endpoint:	9000	
Console Port/Endpoint:	9999	
		Presets Local Connection Servlet
?	< Back Nex	t > Finish Cancel

- 9. Verify the values in the fields, or type the valid value or values.
 - □ The Connection String field contains the URL that connects to the iSM.
 - The SOAP Port/Endpoint field contains the SOAP port number.
 - □ The Console Port/Endpoint field contains the port number that the iSM Administration Console is listening on.
 - Optionally, under Presets, click Local Connection to insert values for a local default iSM connection in the fields, or click Servlet to insert values for a sample servlet connection.
- 10. Click Finish.

The new iSM connection is added to the tree on the iWay Explorer tab.

In the following image, an iSM connection named SampleConfig was added to iWay Explorer.The tree is expanded to show the five explorer environments that are available.



Adding the J.D. Edwards EnterpriseOne Adapter to iWay Explorer

iWay Explorer supports access to many different application systems. When you connect to and expand the Adapters node, the iWay adapters for the supported application systems are displayed. They are the iWay adapters that you have installed and are licensed to use.

Procedure: How to Add the JDE Edwards OneWorld Adapter to iWay Explorer

In this procedure, you are going to add the iWay Application System Adapter for JDE Edwards OneWorld to the list of adapters displayed in the Adapters node.

1. Right-click the *Adapters* node, and click *Edit* from the menu.

The Edit Adapters dialog opens, prompting you to select the iWay adapter or adapters to add to iWay Explorer.

2. Select the JDE Edwards OneWorld check box, as shown in the following image.

dapter Name	Description	Select Al
CICS	Supports CICS DPL program access via TCP/IP and the CRM Gateway.	Select No.
DOTNET	Supports any Microsoft .NET assembly annotated with custom attribut	Delectivo
Exchange	Supports access to Microsoft Exchange web services	
HL7	1.0	
IMS	Supports IMS access via IMS Connect in IMS V7 and up, and the CRM G	
iWay	Supports iWay v6.0. iWay Adapter Framework v1.0.	
Java	Supports Plain Old Java Objects (POJOs). iWay Adapter Framework v1.0.	
JDEdwards One World	JD Edwards EnterpriseOne supports from B7333(XE) SP23 to EnterpriseO	
JDEWorld	JDE WORLD 8.1.	
LDAP	1	
LogListener	1.0	
MSCRM2011	Supports MS Dynamics CRM 2011 via WCF Routing Service. iWay Adapt	
MUMPS	Supports MUMPS access via TCP/IP.	
RDBMS	Supports JDBC API v. 3.0 compliant drivers. Adapter Framework version	
Salesforce	Salesforce Adapter 1.0	
Talant	Supports NI/T_TN2270 and TN5250 emulations	

3. Click Finish.

The tree is automatically refreshed and displays the new adapter. In the following image, the JDEdwards One World node is displayed in the Adapters node of iWay Explorer, as shown in the following image.



Working With a Target

To browse the business objects of an application system, you must create a target for that system. The target is the means by which you connect to the system. It contains the logon properties used to access the system.

Using the target, you must establish a connection to an application system every time you want to browse the system in iWay Explorer.

Procedure: How to Create a Target

1. Right-click the *Adapters* node and click *Connect* from the menu, as shown in the following image.



2. Once you are connected, expand the Adapters node.

3. Right-click *JDE Edwards OneWorld*, and click *Add Target* from the menu, as shown in the following image.



The Add Target dialog opens and displays the Generic Target Properties pane, as shown in the following image.

🔏 Add Targ	jet l	
Generic Ta Please ent	irget Properties er the generic properties associated with the new target.	;)
Name:	JDEEnterpriseOne_910	
Description	E C	
Type:	JDE One World 👻	
Connect	t to target upon wizard completion.	
?	< Back Next > Finish	Cancel

- 4. Supply the values for the fields on the dialog box as follows:
 - a. In the Name field, type a descriptive name for the target (for example, *JDEEnterpriseOne_910*).
 - b. In the Description field, optionally type a brief description of the target.
- 5. Select the *Connect to target upon wizard completion* check box if you want iWay Explorer to automatically connect to this target after it has been created.

If you clear this option, iWay Explorer will not automatically connect to the target. From the tree, you can connect to an individual target when you want to access the associated application system.

6. Click Next.

The JDE One World Target Properties pane opens, as shown in the following image.

Add Target	
JDE One World Target Properties Please enter the properties associated with the new target.	
Repository Logon Repository directory	
Schema style ATTRIBUTE_STYLE	
(?) < Back Next > Finis	h Cancel

7. Supply the Repository directory path where the GenJava files are placed. For example, C: $\$ \repository.

8. Click the Logon tab as shown in the following image.

Add Target
JDE One World Target Properties Please enter the properties associated with the new target.
Repository Logon
User id
User password
JDE Environment
Server IP address
Server port
User role

9. Provide the parameter values in the corresponding fields, as listed in the following table.

Parameter	Description
User ID	Valid user ID for J.D. Edwards EnterpriseOne.
User password	Password associated with the user ID.

Parameter	Description
JDE Environment	EnterpriseOne environment, for example, DV910. For more information about this parameter, see your EnterpriseOne documentation or ask your EnterpriseOne system administrator.
Application	XMLInterop or the application name in EnterpriseOne. This field is optional.
Server IP Address	Name of the server on which EnterpriseOne is running. This can be the name of the server, for example, JDE9.10, or its IP address, for example, 123.45.67.89.
Server Port	Port number where the server is listening, for example, 6016.
User Role	User profile for the J.D. Edwards user identifying user privileges.

10. Click *Finish* when you are done.

The new target is added to the Adapters node of iWay Explorer, as shown in the following image.



Procedure: How to Connect to a Target

1. Expand the JDEdwards OneWorld node to locate the name of the target that you want to connect to, for example, JDEEnterpriseOne_910.

2. Right-click the target and click Connect from the menu, as shown in the following image.



3. Enter a valid password for the configured JDEEnterpriseOne_910 target and click Finish.

4. The JDEEnterpriseOne_910 node icon changes to green, and folders are displayed based on files in the repository folder, reflecting a successful connection. You can click a folder and then expand it to display its contents.



Procedure: How to Disconnect From a Target

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

- 1. In the tree, expand the *JDEdwards OneWorld* node to locate the name of the target from which you want to disconnect, for example, *JDEEnterpriseOne_910*.
- 2. Right-click the target and click *Disconnect* from Target from the menu.

The connection to the application system is closed.

Procedure: How to Edit a Target

After you create a target, you can edit the information that you provided during the creation procedure.

- 1. In the tree, expand the JDEdwards OneWorld node to locate the name of the target that you want to edit, for example, *JDEEnterpriseOne_910*.
- 2. Right-click the target, and click *Edit Target* from the menu.

The Edit Target dialog opens and displays the JDE EnterpriseOne target properties.

- 3. Modify the connection properties as required.
- 4. Optionally, select the *Reconnect to target upon wizard completion* check box if you want iWay Explorer to automatically connect to this target after it has been edited.

iWay Explorer will use the modified properties to connect.

5. Click *Finish* when you have made your edits.

Procedure: How to Delete a Target

You can delete a target that is no longer needed. You can delete it whether or not it is closed. If open, the target automatically closes before it is deleted.

- 1. In the tree, expand the JDEdwards OneWorld node to locate the name of the target that you want to delete, for example, *JDEEnterpriseOne_910*.
- 2. Right-click the target and click *Delete Target* from the menu.

iWay Explorer displays a prompt, asking you to confirm the deletion of the selected target, as shown in the following image.

🛃 Confirm Delete	
Delete Target: lawson_tgt?	
	OK Cancel

Creating an XML Schema

A request through the JDE EnterpriseOne server begins with the receipt of a request document and, in most cases, ends with the issue of an XML response document that indicates the result of the business function execution. iWay Explorer creates the following:

- □ XML request schemas
- □ XML response schemas

The following procedure describes how to create the request and response schemas for a JDE EnterpriseOne GetEffectiveAddress Business Function. iWay Explorer enables you to create XML schemas for this function.

Procedure: How to Create an XML Request and Response Schema

1. Expand the Services node and connect to an available target for which you want to create XML request and response schemas.



2. Expand the CALLBSFN node, as shown in the following image.



3. Expand the *AddressBook* node, as shown in the following image.



4. Right-click *GetEffectiveAddress* and select *Export Request Schema* from the context menu, as shown in the following image.





The Save As dialog box opens, as shown in the following image.

- 5. Expand the Schemas folder, select the Schemas subfolder, and then click OK.
- 6. Type a name for the exported request schema. By default, the file name extension is .xsd.

7. Navigate to the Integration Explorer tab in iIT and verify that the exported XML request schema is listed, as shown in the following image.



8. Return to the iWay Explorer tab, right-click *GetEffectiveAddress*, and select *Export Response Schema* from the context menu, as shown in the following image.



The Save As dialog box opens, as shown in the following image.



- 9. Expand the Schemas folder, select the Schemas subfolder, and then click OK.
- 10. Type a name for the exported response schema. By default, the file name extension is .xsd.

11. Navigate to the Integration Explorer tab in iIT and verify that the exported XML response schema is listed, as shown in the following image.



Creating XML Request Documents

After you have generated XML request schemas, you can create XML request documents, which can be used as a payload for an iWay Business Service (web service).

Procedure: How to Create XML Request Documents

1. In the Integration Explorer tab, expand your Integration Project node and then expand the *Schemas* folder.

2. Right-click the *GetEffectiveAddress.xsd* schema file, select *Generate*, and then click *XML* from the context menu, as shown in the following image.



New XML File	
XML Create a new XML file.	
Enter or select the parent folder:	
Test/XML	
 Constant Constant<	
File name: GetEffectiveAddress_request.xml Advanced >>]
C Back Next > Finish	Cancel

The New XML File dialog box opens, as shown in the following image.

3. Select the XML subfolder and click Next.

The Select Root Element pane opens, as shown in the following image.

wson			
Content optio	ns		
Create opti	onal <u>a</u> ttributes		
Create opti	onal elements		
Create first	choice of required choic	e	
Fill elements	s and attributes with <u>d</u> ai	ta	
Namespace I	nformation		
Prefix	Namespace Name	Location Hint	Add
<no pre<="" td=""><td><no na<="" namespace="" td=""><td>/Schemas/AP20.1</td><td>Edb</td></no></td></no>	<no na<="" namespace="" td=""><td>/Schemas/AP20.1</td><td>Edb</td></no>	/Schemas/AP20.1	Edb
			EUK
			Delete

4. Select all of the check boxes in the Content options area and click *Finish*.

The XML request document based on the *GetEffectiveAddress.xsd* schema file is created and added to the XML subfolder, as shown in the following image.



You can double-click the XML file (GetEffectiveAddress.xml) to open and view the contents or structure in the right pane, as shown in the following image.

🖌 Integration - Test/XML/GetEffectiveAddress_request.xml - Way Integration Tools				
File Edit Navigate Search Project Run Window Help				
📫 • 🔜 🗟 🔮 😤 🌌 🜌 🔅 • 🕗 • 💁 •	<i>┩</i> ▼ ■ ■ ● ▼ ● ▼ ⇔ ▼ ⇔ ▼ ⇔ ▼			
💰 Integration E 🛛 🦉 iWay Explorer 🛋 Library Mana 🖓 🗖	Image: SetEffectiveAddress_request.xml Image:			
수 수 😡 🗖 🕵 🏹	xml version="1.0" encoding="UTF-8"?			
En Test	<pre>Constallethod app-" name="GetEffectiveAddress" returnNullData="yes" runOnError="yes" trans="></pre>			
Registers SchefflectiveAddress_request.sd GetEfflectiveAddress_request.sd Trainforms XML GetEfflectiveAddress_request.sml				
BE Outline 123 C C C C C C C C C C C C C C C C C C C	v, □ Properties © Error Log © Console 33 5g Problems □ v □ v □ v □ v □ v □ v □ v □ v □ v □ v			
	df Message Console [INF0]14:18:41 Process flow 'ExchangeTestUsingHTTP' compiled successfully			

5. To use the XML request document, provide the required input value(s) and use this request for invocation purposes.

Creating and Publishing iWay Business Services

This section describes how to create and publish iWay Business Services using iWay Explorer for the iWay Application Adapter for J.D. Edwards EnterpriseOne.

In this chapter:

Chapter

- Understanding iWay Business Services
- Creating iWay Business Services
- Connecting to the J.D. Edwards EnterpriseOne Client

Understanding iWay Business Services

iWay Explorer provides web developers with a simple, consistent mechanism for extending the capabilities of the iWay Application Adapter for J.D. Edwards EnterpriseOne. The iWay Business Services Provider (iBSP) 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.

Creating iWay Business Services

After you browse the J.D. Edwards EnterpriseOne Business Functions and create an XML schema for the Business Function, you can generate an iWay Business Service for the Business Function you wish to use with your adapter.

The Web Service Description Language (WSDL) file is an XML file that describes the web service documents and provides access to the service. It specifies the location of the service and the operations (or methods) that the service exposes.

You can delete an iWay Business Service that you no longer need.

Creating Business Services With iWay Explorer

iWay Explorer provides application developers with a simple, consistent mechanism for extending the capabilities of the iWay Application System Adapter for JDE EnterpriseOne. The iWay Business Services Provider (iBSP) 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 JDE EnterpriseOne Business Functions and create an XML schema for the object, you can generate an iWay Business Service for the object you wish to use with your adapter.

The following procedure describes how to create iWay Business Services using iWay Explorer.

Procedure: How to Create an iWay Business Service

After you browse the business object repository for an application system, and generate XML schemas for an object that you want to use with an iWay adapter, you can create an iWay Business Service for that object. The Web Service Description Language (WSDL) file is an XML file that describes the web service documents and provides access to the service. It specifies the location of the service and the operations (or methods) that the service exposes.

You can delete an iWay Business Service that you no longer need.

1. Expand the JDEdwards One World node and connect to an available target for which you want to create an iWay Business Service.

2. Expand the Services node as shown in the following image.



3. Expand the CALLBSFN node, as shown in the following image.



4. Expand the *Addressbook* folder, as shown in the following image.



5. Right-click *GetEffectiveAddress* and select *Create iWay Business Service* from the context menu, as shown in the following image.



The Add Business Service dialog box opens, prompting you for information about the new service.

Add Business Service		
Select or Create a Busi Create a business service	ness Service from the geteffectiveaddress operation in the jdedwards one world adapter.	
Existing Service Names:	<new service=""></new>	
Service Name:	Addressbook	
Service Description:		*
?	< Back Next > Finish	Cancel

- 6. Supply the values for the fields on the dialog box as follows:
 - a. From the Existing Service Names drop-down list, click <new service> if you want to create a new service name or select an existing service name.
 - b. If you are creating a new service name, type the name in the Service Name field, for example, GetEffectiveAddress.
 - c. In the Service Description field, optionally type a brief description of the new business service.
- 7. Click Next.



The Select Business License pane opens, as shown in the following image.

- 8. Supply the values for the fields on the dialog box as follows.
 - a. From the License drop-down list, select the license definition that you want to use with this business service.
 - b. In the Method Name field, accept the default value or type a descriptive name for the method that the service exposes (for example, GetEffectiveAddress).
 - c. In the Method Description field, optionally type a brief description of the method.
- 9. Click Finish.
The new iWay Business Service is added beneath the Services node in the iWay Explorer tab, as shown in the following image.

🥖 II	tegration - Configuration for http://localhost	:9000 - iWay Integration Tools
File	Edit Navigate Search Project Run Window	Help
File	Edit Navigate Search Project Run Window	Help • • • • • • • • • • • • • • • • • • •
	AddressbookderAB AddressbookderAB Configuration Configuration	 predefined iWay Business Service Provider Services. production The production License is installed by default. It is used for production purpose. test The test License is installed by default. It is used to test iWay Business Service Provider Services.

Procedure: How to Test an iWay Business Service

1. To test the new iWay Business Service, click the test link in the right pane, which displays the available licenses, as shown in the following image.



The iWay Business Services that are licensed under the test license are displayed.

2. Expand the *Addressbook* folder.

The operations (methods) that are supported for this service are displayed.

3. Double-click the GetEffectiveAddress node.

The test pane for the GetEffectiveAddress. method opens, as shown in the following image.

🤞 Iı	ntegration - Configuration for http://localhost	:9000 - iWay Integration Tools
File	Edit Navigate Search Project Run Window	Help
] 🛤	🔹 🖯 🕞 🗠 🛛 🏶 兄 🖉 🖉 🖉 🖓	• Q • Q •] A •] 2 • 3 • 5 + + +
_	🔬 Inte 🤴 iWa 🔀 🛋 Libra 🖓 🗖	🙀 SAP 🖻 MATMAS05_request.xsd 🖹 MATMAS05_request.xml 🛛 🚳 SampleConfig 😂
	♥ 🖨 ⇔ 🖻 ♥	Automation back
	E- 🗞 SampleConfig	Software An iWay Business Service
•2	E Adapters	
	i i Way	Click here for a complete list of operations.
	JDEdwards One World	
	DE910	GetEffectiveAddress
		Te test the ecception union the COAD produced which the 'Involve' butter
	AddressBoc	To test the operation using the <u>SOAP protocol</u> , click the Thyoke button.
	- GetABEffec	input xml:
	 GetEffectiv 	
	🛛 💿 GetMailingN	
	 GetParentA 	
	GetPhone	
		×
	LinitsOfWork	
	⊕ IWJDE9152	Browse Upload More Invoke
	-0 RDBMS	
	🗄 🚳 Services	
	Addressbook9152	
	Addressbook	
	E Cicenses	
	Methods	
	GetEffectiveAddress	
	AddressbookGetAb	
	EGO ITEM PUB	
	InboxGetmessage	
	1 InboxListFolders	
	🗉 🍓 iwayivp	
	🗈 🍓 MATMAS01 📃	

- 4. In the input xml field, enter an XML request document that queries the iWay Business Service named *GetEffectiveAddress*.
- 5. Click Invoke.

The result of the test is displayed in the right pane, as shown in the following image.



Connecting to the J.D. Edwards EnterpriseOne Client

This section describes how to connect to the J.D. Edwards EnterpriseOne Client

Procedure: How to Connect to the J.D. Edwards EnterpriseOne Client

1. From the desktop, double-click the JDEdwards Client icon to start the software, as shown in the following image.



The JD Edwards EnterpriseOne Login dialog appears.

2. Enter a valid user ID and password, and then click OK, as shown in the following image.

JD Edwards EnterpriseOne Login						
JD ED	ORACLE WARDS ENTERPRISEONE					
<u>U</u> ser ID: <u>P</u> assword: <u>E</u> nvironment:	JDE ***1 DV910					
Role: OK Copyright © 2003,201 Oracle and Java are r its affiliates. Other na owners.	Cancel Options << Legal Info - 4, Oracle and/or its affiliates. All rights reserved. egistered trademark of Oracle Corporation and/or mes may be trademarks of their respective					

The JD Edwards Solution Explorer window appears, as shown in the following image.



3. In the Fast Path field, enter *P01012* and then press Enter on the keyboard, as shown in the following image.

💡 JD Edwards Solution Explorer	
File Edit View Tools Applications Help	
]]Fast Path P01012	•
Menu Design Menu Filtering	
□-1= EnterpriseOne Life Cycle Tools / All My Roles	
🔁 🧰 Application Development	
🕁 🧰 Report Management	
🖶 🛄 System Administration Tools	
🛓 🧰 Workflow Management	

P01012 - [Work With Addresses] _ 8 × /indow Help _ 8 × × × v R0, + Re T × 10 Add Copy Del... Close Seg... New... Dis... Ago Links ▼ AB R... @ OLE ... @ Internet Tools Row Report Dine Line Alpha Name Display Phone Search Type Display Address Address Number Alpha Name Long Address Industry Class Sch Typ Tax ID Ø.

The P01012 - [Work With Addresses] window appears, as shown in the following image.

4. In the Address Number field, enter the address number from the XML request that was tested from the web service. For example, 4242.

P01012 - [Wor	rk With	Addresse	s]											
🜔 File Edit Pre	eferences	Row P	Report	Window	Help									
Tools	Sele	€ ct F <u>i</u> nd	+ <u>A</u> dd	∎∎ Сор⊻	1 Del	× <u>C</u> lose	1 Se <u>g</u>	6 <u>N</u> ew	Dis	₩ Abo	inks	🕶 A/B R	🗿 OLE	internet 🗿
Row Report														
One Line		Alpha Na	ame											
Vviho's Who		Search 1	īype			*								
Full AB Codes														
D Full AB														
Who'			424	2										
	Ø.	Add Nur	ress nber				Alpi Nari	ha ne				Long Addres	ss	Industry Class

5. Click Find.

Data from the Address Number that was entered appears on the pane, showing information such as Alpha Name, Long Address Number, Tax ID, Search Type, and Business Unit, as shown in the following image.

P01012 - [Ad	dress Book Revision]
File Edit Pr	eferences Form Window Help
Tools	<u>OK</u> <u>Can</u> Dis Abo
Form A/R	
AIP	Address Number 4242
Phones	
VVho's VVho	
Bank Account	Address Book Mailing Additional 1 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 Additional 2 Related Address Cat Code 1 - 10 Additional 2 Related Address Cat Code 1 - 10 Cat Code 11 - 30 Additional 2 Related Address Cat Code 1 - 10 R
SAVIM	Alpha Name Capital System
Formatt Address	Long Address Number
	Tax ID 45-34126801
Regional Info	Search Type C Customers
	Business Unit 1 Financial/Distribution Company
	Work With Addresses

Chapter 6

Listening for Database Events

This section describes how to use the iWay Application Adapter for J.D. Edwards EnterpriseOne to listen for events.

In this chapter:

Understanding Event Functionality

Understanding Event Functionality

Events are generated as a result of activity in an application system. You can use the application to trigger an event. For example, you can use it to trigger an update, insert, or delete an event.

After you create a connection to your application system, you can receive events using iWay Service Manager. To receive an event, you must fist create a channel. The events will be received using a TCP or HTTP listener.

Procedure: How to Configure a Listener

1. Under the Components section on the left pane, click *Listeners*, as shown in the following image.

iWay Service I	ay Service Manager Management base 💽 🖉 📀 xfoc.44054								
Server <u>Registry</u>	Deployments Tools		Restart Licenses About Logout						
Conduits Channels Inlets Outlets Routes Transformers	Adapters iWay Service Manager impl Services Provider to access registry. Adapters Filter By Name When	ements an adapter co configurational meta	Sontainer to configure/invoke iWay Adapters. The adapter container uses the iWay Business sdata on behalf of its adapters. Listed below are references to adapters defined in the Equals						
Processes	□ Name	Target Referen	nces Description						
Components	Lawson9_adapters	Lawson9 🔒	none						
Adapters	Pictures	RDBMS 🧸	The Pictures adapter defines appropriate configuration information to connect to the sample HSQL pictures database. This database is used in the Pictures						
Ebix			sample.						
Emitters	SciFiBooks	RDBMS 🧸	The SciFiBooks adapter defines the appropriate configuration information to connect to the sample HSQL SciFiBooks database. This database is used in						
Listeners			the Sciri Books sample.						
Pree References to pro	tocol based listeners that can be used thro	oughout the system							

The Listeners pane opens, as shown in the following image.

iste	ners Filter By Name W	here Name Eq	uals 💌	
Г	Name	Туре	References	Description
	file1	File	2	A default/sample file listener.
	javadoc	HTTP 1.0 [deprecated]	A	The javadoc listener is used to make the iWay Service Manager API available to a remote browser.
	pictures,loader	File	A	The pictures listener locates files with a variety of common imag file extensions (img, gif, jpg,).
	pictures.viewer	HTTP 1.0 [deprecated]	2	The pictures viewer is used to kickoff the image retrieval process as defined by the pictures sample.
	<u>scifibooks</u>	Schedule Recurring Execution	A	This listener is defined for use by the SciFi Books sample. It wakes up daily and kicks off the update for the channel.
	SOAP2	SOAP	A	This listener is used by the stock SOAP channel.

2. Click Add.

The Select listener type pane opens, as shown in the following image.

iWay Service Ma	nager	Management base	🗸 – 🔕 🥥 😨 7.0.3.2452
Server <u>Registry</u> Depl	oyments Tools		
Conduits Channels	Listeners Listeners are protocol handlers, t	hat receive input for a channel from a configured endpoint. Listed	below are references to the listeners that are
Inlets	Select listener type		
Outlets Routes Transformers Processes Components Adapters Decryptors Ebix Emitters Encryptors Listeners Preemitters Preparsers	Type *	Type of the new listener AS1 AS2 AS2 [nonblocking] Binary-to-IDOC ConnectDirect CS3 Email Envoy Exchange File HIZ-MLLP-Listener HTTP 1.1 (nonblocking] (nhttp) IEI Internal Queue Java Message Service (jmsq) LDAP High Watermark/File LogListener	^
Reviewers Rules Schemas Services Transforms Variables Parameters		MQ MSMQ Oradeted Queue RDB High Watermark (rdbhwm) RDB Seled with Post-Execution SDB Seled with Post-Execution SoaP Sonic TCP Tibry	v
Registers			

3. Select TCP and click Next.

4. Provide the port number specified in iwoevent.cfg, as shown in the following image.

iWay Service N	Manager	Management base 🗸 🖉 🖉 7.0.3.2452						
Conduits Channels	Listeners Listeners Listeners are protocol handlers defined in the registry.	Restart LICenses About Logout						
Inlets	Configuration parameters	or new listener of type TCP						
Outlets	Port *	Port(socket) number on which messages are exchanged						
Routes								
Transformers		U						
Processes	Local Bind Address	Local bind address for multi-homed hosts: usually leave empty						
Components								
Adapters	Allowable Client	If used, only messages from this fully qualified host name are accepted						
Decryptors								
Ebix								
Emitters	Timeout	Timeout interval for socket in milliseconds						
Encryptors		2						
Listeners	Stream Length Encoded	Sats the form of length encoding for the TCP stream						
Preemitters	Stream Length Encoded							
Preparsers		1						
Reviewers		Pick one						
Rules	Persistent connection	If set, connection is maintained until client closes, MUST have length encoding on						
Schemas								
Services		Taise Dick one						
Transforms		Pick vile						
Variables	Set TCP No Delay	If true, disables Nagle's Algorithm on the client socket. This will result in faster line turnaround at the expense of an increased number of packets.						
Parameters		for the second						
Registers		Pick one						
Recovery	Defer Close of Socket	If true, close of client socket is deferred for one second after response is written. This compensates for an issue seen						
Recycle Bin		on some older versions of z/OS.						
		true						
		Pick one 🗸						

5. Click Next and then provide a name for the listener example in the Name field (for example, JDE_910_Event_listener), as shown in the following image.

iWay Service Mar	lager	Management base 🗸 🔕 (🧭 😨 7.0.3.2452
Server <u>Registry</u> Deplo	oyments Tools		
Conduits Channels	Listeners Listeners are protocol handlers, defined in the registry.	that receive input for a channel from a configured endpoint. Listed below are references to the list	steners that are
Inlets	Select listener type		
Outlets	Name *	Name of the new listener	
Routes		JDE_910_Event_Listener ×	
Transformers	Description		
Processes	Description	Description for the new listener	
Components			
Adapters			
Decryptors	<< Back Finish		
Ebix			
Emitters			
Encryptors			
Listeners			
Preemitters			
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

6. Click Finish.

Procedure: How to Configure a Process and Define a Route

1. In the left pane under Conduits, click Processes, as shown in the following image.

Way Service Manager					ent base		• 🔕	0	
server <u>Registry</u> Depl	oyment								
Conduits	Listen	ers							
Channels	Listenei that are	rs are protocol handlers, e defined in the registry.	that receive input for a char	nnel from a cor	figured endpoint.	Listed below	w are referen	ces to the	listeners
Inlets	-Liste	ners							
Outlets	2.010								_
Routes		Iter By Name Where Nam	e 💌 Equais	2					
Transformers	Г	Name	Туре	References	Description				
Processes		file1	File		A default/sample	filo listoner	,		
References to processe	es that can	be used throughout the syst	tem	-		ine insterier			
Adapters		Javaduc	TTTP 1.0 [deprecated]	A	The javadoc lister Manager API avai	ner is used ilable to a r	to make the emote brows	iWay Serv er.	ice
Decryptors		Lawson9_file_listeners	File	3	none				
Ebix	-	nictures leader	File		The nictures liste	nor locator	files with a v	oriety of o	000000
Emitters		pictures.toader	The state		image file extensi	ions (img, g	gif, jpg,).	anery or co	ommon
Encryptors		pictures.viewer	HTTP 1.0 [deprecated]	2	The pictures.view	eris used t	o kickoff the	image retr	ieval
Listeners					process as define	ed by the pi	ctures samp	le.	
Preemitters		scifibooks	Schedule Recurring	2	This listener is de	efined for us	e by the Sci	Fi Books s	ample.
Preparsers			Execution		It wakes up daily	and kicks	off the update	for the ch	annel.
Reviewers		SOAP2	SOAP	2	This listener is us	ed by the	stock SOAP	channel.	
Rules						,			
Schemas	Add	Delete Renam	e Conv						
Services	-100	Kenam	00001						

The Processes pane opens, as shown in the following image.

Proc	ess Definitions Filter By Name Where Name <u>s</u>	Eq	uals 💌	
	Name	View	References	Description
	move		2	The move1 service defines a move agent that moves the input document stream to the output document stream. It represents the basic echo pattern in iSM.
	Samples.PFIVP.1	(0	2	This sample process, delivered with iWay Designer, copies a subtree of the input document as defined by the PFIVP schema to the root of the output document as defined by PFIVPResponse schema.
	Samples.PFIVPWS.1	()	2	This sample process, delivered with iWay Designer, illustrates the invocation of a simple iWay Business Service from a flow.
	Samples.Pictures.Load.1	€	2	The Pictures.Load process is used to insert images into a RDBMS table.
	Samples.Pictures.RetrieveAlburn.1	()	R	The Pictures.RetrieveAlburn process is used to get images from an RDBMS table and generate a photo alburn as an html page.
	Samples.SciFiBooks.1	(0	2	The SciFiBooks process is used to define the business logic implemented by the SciFi Books sample. This sample is built around the concept of tracking new science fiction books as they are published and released.

2. Click Add.

The following pane opens, which allows you to specify a name for the new process definition.

iWay Service Mar	nager	Management base		>	0	7.0.3.2452
Server <u>Registry</u> Deplo						
Conduits Channels Inlets	Processes Processes are stateless, lightwei the system. Processes, typically Service Provider thru the iWay A	ght, short-lived microflows that are executed by the iWay Service Mana authored by the iWay Service Designer, can be bound to channels or ex dapter.	ger on mes kposed as V	sages/docur Veb Services	nents as th by the iWa	ey pass thru iy Business
Outlets	New Process Definition					
Routes	Name *	Name of the new Process object definition				
Transformers		JDE_910_Process	×			
Processes	Description	Description for the new Process object definition				
Components Adapters	Description	Description for the new Process object delinition		$\hat{}$		
Decryptors Ebix	< Back Finish					
Emitters						
Encryptors Listeners Preemitters						

3. Specify a process name (for example, JDE_910_process) and click *Finish*.

The Construct Process pane opens, which allows you to construct the new process (for example, JDE_910_process) by adding supported components, as shown in the following image.

iWay Service Mana	ager			Management base	~	- 🔕	0	
Server <u>Registry</u> Deploy	ments							
Conduits I Channels I Inlets S Outlets Routes	Process Processe the syst Service Assig Belo	ses / JDE_910_Process es are stateless, lightweight em. Processes, typically aut Provider thru the iWay Adag on service object referenc w is a list of service objects	short-lived hored by th ter. es to proce currently de	I microflows that are executed by the iWay Service Manage e iWay Service Designer, can be bound to channels or expo ass JDE_910_Process fined on the server. Select one or more service objects and	r on messag sed as Web click Finish 1	es/docur Services o assign	nents as th by the iWa	ey pass thru ay Business
Transformers	F	ilter By Name Where Name	~	✓ Equals ∨				
Processes		Name	Туре	Description				
Components		JDEEnterpriseOne_910	Adapter	none				
Adapters Decryptors		move1	Service	The move1 service defines a move agent that moves output document stream. It represents the basic echo	the input d	ocumer iSM.	it stream to	o the
Ebix Emitters		Pictures	Adapter	The Pictures adapter defines appropriate configuration sample HSQL pictures database. This database is us	n informati ed in the P	on to co ictures s	nnect to th sample.	ie
Encryptors		pictures.img2xml	Service	converts the image to base64 and wraps it in a <pictu< td=""><td>ure> tag</td><td></td><td></td><td></td></pictu<>	ure> tag			
Listeners		pictures.iterator	Service	Iterate a loop for each portion of an XML document				
Preemitters Preparsers		RSSRead1	Service	Reads an RSS Document from url that is specified in	the origina	l incomr	ning docur	ment.
Reviewers Rules		Snip1	Service	Copies a subtree of the input document as defined by the output document as defined by PFIVPResponses	/ the PFIVF schema.	schem	a to the ro	ot of
Schemas Services Transforms	<< Ba	Finish						

- 4. Select JDEEnterpriseOne_910 and click Finish.
- 5. In the Conduits section on the left pane, click Routes, as shown in the following image.

iWay Service Ma	anager	Management base 🗸 🕢 🕢 🧐 7.0.3.2452
Server <u>Registry</u> De	ployments Tools	
Conduits Channels	Routes A route is used to define the path a particular n process, followed by another transformer, follow	nessage takes thru a channel. A Route is defined as a sequence of: a transformer, followed by a wed by zero or more outlets.
Iniets	Route Definitions	
Routes	Filter By Name Where Name	✓ Equals ✓
Transformers	Name View References	Description
Processes	🗆 move 🔿 🗸	The move route defines a simple route that moves the input stream to the output stream
Components		The office route defines a simple route that is used to invoke the PEIVP processes
Adapters		The prop route defines a simple route that is used to invoke the Prive process.
Decryptors	🗋 pfivpws 🙃 🛃	The pfivpws route defines a simple route that is used to invoke the PFIVPWS process. This version adds a transformer to the output segment of the route
Ebix	🗆 nictures loader 🔿 🖪	This route is used to invoke the nictures loader process
Encryptors		This route is used to invoke the pictures induce process.
Listeners	pictures.viewer	This route is used to invoke the pictures viewer process.
Preemitters	Add Delete Rename Conv	7
Preparsers	Add Delete Rename Copy	
Reviewers		
Rules		
Schemas		

6. Click Add.

The New Route Definition pane opens, as shown in the following image.

iWay Service Mar	lager	Management base	✓ Ø Ø Ø 7.0.3.2452
Server <u>Registry</u> Deplo	oyments Tools	Restart	Licenses About Logout
Conduits Channels	Routes A route is used to define the pati process, followed by another tran	n a particular message takes thru a channel. A Route is defined as a sequence of sformer, followed by zero or more outlets.	: a transformer, followed by a
Inlets	New Route Definition		
Outlets	Name *	Name of the new route	
Transformers		JDE_910_Route ×	
Processes	Description	Description for the new route	
Components			\bigcirc
Adapters			
Decryptors	< Back Finish		
Ebix			
Emitters			
Encryptors			
Listeners			
Preemitters			

7. Specify a route name (for example, JDE_910_Route) and click *Finish*.

The Construct Route pane opens, which allows you to construct the new route (for example, JDE_910_Route) by associating a configured process, as shown in the following image.

ager		Management	base	~ @	🧭 😨 7.0.3.2452
yments Tools					
Routes / JDE_910_Route A route is used to define the path process, followed by another tran Construct Route Below are the components cu	n a particular mes nsformer, followed	isage takes thru a channel. A Route d by zero or more outlets. in the route.	is defined as a sequence o	f: a transform	er, followed by a
Name	Туре	Conditions	Descriptio	n	
No data was found.					
< Back Add Dele	te View				
	Age Tools Routes / JDE_910_Route A route is used to define the pati process, followed by another transported	Agent Tools Routes / JDE_910_Route A route is used to define the path a particular mer process, followed by another transformer, followe	Management Inserts Tools Routes / JDE_910_Route A route is used to define the path a particular message takes thru a channel. A Route process, followed by zero or more outlets. Construct Route Below are the components currently registered in the route. Name Type No data was found. <	Management Dase Aments Tools Restart Routes / JDE_910_Route A route is used to define the path a particular message takes thru a channel. A Route is defined as a sequence or process, followed by another transformer, followed by zero or more outlets. Construct Route Below are the components currently registered in the route. Image: Conditions Description Name Type Conditions Description No data was found. View Image: Construct Route Image: Construct Route	Aggerr Management Dase Imagement mments Tools Restart Licenses Routes / JDE_910_Route A route is used to define the path a particular message takes thru a channel. A Route is defined as a sequence of: a transform process, followed by another transformer, followed by zero or more outlets. Construct Route Below are the components currently registered in the route. Imagement Description Name Type Conditions Description No data was found. View

8. Click Add.

The Select component type pane opens, as shown in the following image.

R A b	oute route y a pr Sele	s / Lawson9_routes e is used to define the rocess, followed by an ct component type	path a particular message takes thru a channel. A Route is defined as a sequence of: a transformer, followed other transformer, followed by zero or more outlets.
		Component Types	Description
	C	In Transformer	In Transformers are exit sequences that apply to the message before the process.
	¢	Process	Processes are stateless, lightweight, short-lived microflows that are executed by the iWay Service Manager on messages/documents as they pass through the system.
	C	Out Transformer	Out Transformers are exit sequences that apply to the message after the process.
	C	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter.
	<< B	lack Next >>	

9. Select Process and then click Next.

The Select a process definition pane opens, as shown in the following image.

iWay Service I ^{Server} Registry	Aanager Deployments Tools	Management 0886 🗸 🖉 🧿 7.0. Restart Licenses About Log
Conduits Channels Inlets Outlets	Routes / JDE_910_Route A route is used to define the path a particular r process, followed by another transformer, follo Select a process definition	nessage takes thru a channel. A Route is defined as a sequence of: a transformer, followed by a wed by zero or more outlets.
Transformers	Name	Description
Processes	JDE_910_Process	none
omponents	O move	The move1 service defines a move agent that moves the input document stream to the output document stream. It represents the basic echo pattern in iSM.
Decryptors Ebix	Samples.PFIVP.1	This sample process, delivered with iWay Designer, copies a subtree of the input document as defined by the PFIVP schema to the root of the output document as defined by PFIVPResponse schema.
Emitters Encryptors	Samples.PFIVPWS.1	This sample process, delivered with iWay Designer, illustrates the invocation of a simple iWay Business Service from a flow.
Listeners	O Samples.Pictures.Load.1	The Pictures.Load process is used to insert images into a RDBMS table.
Preemitters Preparsers	Samples Pictures.RetrieveAlbum.1	The Pictures.RetrieveAlbum process is used to get images from an RDBMS table and generate a photo album as an html page.
Reviewers Rules Schemas	<< Back Finish	

10. Select the configured process (for example, JDE_910_Process) and click Finish.

Procedure: How to Define an Inlet

1. In the Conduits section on the left pane, click Inlets.

The Inlets pane opens, as shown in the following image.

iWay Service	lanager Management base 🗸 🖉 😨 7.0324
Server <u>Registry</u>	eployments Tools Restart Licenses About Logout
Conduits Channels Inlets	Inlets Inlets are conduits which represent the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers. Inlet Definitions
Outlets	Filter By Name Where Name V Equals V
Routes	Distance Description
Transformers	Name References Description
Processes	file1 The file1 inlet contains the file1 listener and is a part of the file1 sample channel.
Components	pictures.loader A The pictures.loader inlet contains the pictures.loader listener and is a part of the pictures.loader channel.
Adapters	Dictures viewer The pictures viewer inlet contains the pictures viewer listener and is a part of the
Decryptors	pictures.viewer channel.
Ebix	SOAPIniet A This inlet is used by the stock SOAP channel
Emitters	
Encryptors	Add Dalata Panama Conv
Listeners	Aug Delete Rename Cupy

2. Click Add.

The New Inlet Definition pane opens, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deple	nager oyments Tools	Management <mark>base v</mark> Ø Restart Licenses	Image: Weight of the second se
Conduits Channels	Inlets Inlets are conduits which represe	in the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.	
Inlets	New Inlet Definition		
Outlets Routes	Name *	Name of the new inlet <u>JDE_910_Inlet</u> ×	
Transformers Processes	Description	Description for the new inlet	
Components			
Adapters Decryptors Ebix	<< Back Finish		

3. Specify an inlet name (for example, JDE_910_inlet) and click *Finish*.

The Construct Inlet pane opens, which allows you to construct the new inlet (for example, JDE_910_Inlet) by associating supported inlet components, as shown in the following image.

ln In	Inlets / Lawson9_inlets Inlets are conduits which represent the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.				
r.	Construct Inlet				
	Below are the components currently registered in the inlet. The order of decryptor and preparser components may be changed within each component type by checking a component and using the 'Move Up' and 'Move Down' buttons.				
	П	Name	Туре	Move	Description
	No data was found.				
	<< B	ack Add Delete			

4. Click Add.

The Select component type pane opens, as shown in the following image.

inie Inie	e ts / ets / Sele	/ Lawson9_inlets are conduits which rep ct component type	resent the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.
		Component Types	Description
	•	Listener	Listeners are protocol handlers, that receive input for a channel from a configured endpoint.
	\circ	Decryptor	Decrypts the document.
	0	Preparser	A logical process that handles documents before they are parsed by the system. Usually used to convert from non-XML to xml.
<	< B	ack Next >>	

5. Select Listeners and then click Next.

The Select a listener definition pane opens, as shown in the following image.

iWay Service Ma Server <u>Registry</u> Dep	INager Joyments Tools		Management base 🗸 🖉 🧭 7.0.3.2452 Restart Licenses About Logout
Conduits Channels Inlets Outlets	Inlets / JDE_910_Inlet Inlets are conduits which represent the Select a listener definition	he entry into a channel. Inle	ts contain a Listener, Decryptor, and Preparsers.
Routes Transformers	Name	Туре	Description
Processes	IDE 910 Event Listener	File	A default/sample file listener.
Components Adapters	pictures.loader	File	The pictures listener locates files with a variety of common image file
Decryptors Ebix	o <u>pictures.viewer</u>	HTTP 1.0 [deprecated]	The pictures viewer is used to kickoff the image retrieval process as defined by the pictures sample.
Emitters Encryptors	O SOAP2	SOAP	This listener is used by the stock SOAP channel.
Listeners Preemitters	< Back Finish		1

6. Select the configured listener (for example, JDE_910_Event_Listener) and click Finish.

Procedure: How to Construct a Channel

1. In the Conduits section on the left pane, click *Channels*.

Server <u>Registry</u> D	Deployments Tools Restart	Licenses About Logou
Conduits	Channels Channels are the pipes through which messages flow in iWay Service Manager. A Channel is defined as a named co	ntainer of Routes (Transforme
Channels	Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters).	
Inlets	Channel Definitions	
Outlets	Filter By Name Where Name V Equals V	
Routes		
Transformers	Name Type Regs Ebix View Description	
Processes	□ default □ 0 0 □ The default channel can be used as a starting point for a	uickly defining
Components	functionality in the system. This template defines the mil	es the minimal conduits and
Adapters	build and deploy.	channel, add a listerier,
Decryptors	☐ file1	s an inlet that contains a
Ebix	file listener and completes the sample file channel.	
Emitters	file2 is the file of	route that contains the
Encryptors	PFIVP process.	
Listeners	file3 <u>file3</u> <u>0</u> <u>0</u> The file3 channel is based on the file2 channel. It uses a previous of the files.	route that contains the
Preemitters	PFIVPWS process and adds a reviewer to the mix.	
Preparsers	file4	es routes as defined by
Reviewers	the file 1, mez and mes channels. This channel indistrates	s a multi-routed conduit.
Rules	<u>pictures.ioader</u> Log U U The pictures.ioader channel is used save image files to channels defined by the pictures sample which is built a	a database. It is one of the round the idea of tracking
Schemas	new images as they are recognized by the system.	round the laca of tracking
Services	pictures.viewer 🕼 0 0 🖝 The pictures.viewer channel is used retrieve image files	from a database. It is one
Transforms	of the channels defined by the pictures sample which is tracking new images as they are recognized by the syst	built around the idea of em.
ariables	SOAP2 0 0 This channel can be used to add IBSP (SOAP) services	to an iWay Application.
Parameters		, ,,
Registers	Add Delete Rename Copy Build	
Recovery		
Recycle Bin		

The Channels pane opens, as shown in the following image.

- 2. Click Add.
- 3. Specify a channel name (for example, JDE_Event_Channel), and click *Finish*, as shown in the following image.

iWay Service Mar Server <u>Registry</u> Deple	lager oyments Tools	Management base 🗸 🧭 🧭 7.0.3.2452 Restart Licenses About Logout						
Conduits Channels	Channels Channels are the pipes through Processes), controlled by Routing	which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + g Rules and bound to Ports (Listeners/Emilters).						
Inlets	New Channel Definition							
Outlets Routes Transformers	Name *	Name of the new channel JDE_Event_Channel ×						
Processes	Description for the new channel							
Components		\bigcirc						
Adapters Decryptors Ebix	< Back Finish							
Emitters								

The Construct Channel pane opens which allows you to construct the new channel (for example, JDE_Event_Channel) by associating supported channel components.

4. Click Add.

The Select component type pane opens, as shown in the following image.

Cł Cł (T	Channels / Lawson9_Channels Channels are the pipes through which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters). — Select component type							
		Channel Component Types	Description					
	۲	Inlet	Inlets are conduits which represent the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.					
	C Route A route is used to define the path a particular message takes through a channel. A Route is defined as a sequence of, a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.							
	C Outlet Outlets are conduits which contain Preemitters, Encryptors, and an Emitter							
	<< Back Next >>							

- 5. Select *Inlet* and then click *Next*.
- 6. Select the defined inlet (for example, JDE_910_Inlet) and click *Finish*, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deplo	ager yments Tools	Management <mark>base v</mark> 🕢 🖉 😨 7.0.3.2452 Restart Licenses About Logout
Conduits Channels Inlets Outlets Routes	Channels / JDE_910_C Channels are the pipes the Processes), controlled by Select an inlet definiti Filter By Name W	thanel ough which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Routing Rules and bound to Ports (Listeners/Emitters). on here Name Equals
Transformers	Name	Description
Processes	O file1	The file1 inlet contains the file1 listener and is a part of the file1 sample channel.
Components	JDE_910_Inlet	none
Adapters	o pictures.loader	The pictures.loader inlet contains the pictures.loader listener and is a part of the pictures.loader channel.
Decryptors		The nictures viewer inlet contains the nictures viewer listener and is a part of the nictures viewer channel
Ebix	O pietares.viewei	The pictures viewer lines contains the pictures viewer insterior and is a part of the pictures viewer channel.
Emitters	SOAPInlet	This inlet is used by the stock SOAP channel.
Encryptors		
Listeners	<< Back Finish	
Preemitters		

You are returned to the Construct Channel pane, as shown in the following image.

iWay Service Ma Server <u>Registry</u> Depl	nager oyments Tools		Management base	Restart	✓ Ø	About Log	
Conduits Channels Inlets Outlets Routes	Channels / JDE_910_Channel Channels are the pipes through which messa Processes), controlled by Routing Rules and Construct Channel Below are the components currently regis	iges flow in iWay Se bound to Ports (List stered in the channel	rvice Manager. A Channel is de eners/Emitters).	fined as a named	container of Ro	outes (Transforr	mers +
Transformers Processes	Name JDE_910_Inlet	Type Inlet	Conditions	Move	Description none		
Components Adapters Decryptors	< Back Add Delete Bui	ld View					

7. Click Add.

The Select component type pane opens.

8. Select *Route* and then click *Next*, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deploy	ag	er nts		Management base 🗸 🖉 🖉 7.0.3.2452 Restart Licenses About Logout				
Conduits Channels Inlets Outlets	Cha Pro	anne Innel cesse Selec	els / JDE_Event_Channel s are the pipes through which n is), controlled by Routing Rules at component type	nessages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + and bound to Ports (Listeners/Emitters).				
Routes	1		Channel Component Types	es Description				
Transformers		۲	Route	A route is used to define the path a particular message takes through a channel. A Route is				
Processes				defined as a sequence of: a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.				
Components		0	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter				
Adapters	Ľ	-						
Decryptors	<	< Ba	ck Next >>					
Ebix								

The select one or more route definitions pane opens, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deplo	lager syments Tools	Management base 🗸 🤕 📀 7.0.3.2452 Restart Licenses About Logout
Conduits Channels Inlets Outlets Routes	Channels / JDE_Event Channels are the pipes thr Processes), controlled by R Select one or more ro Filter By Name W	Channel Suph which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + doubling Rules and bound to Ports (Listeners/Emitters). ute definitions ere Name Equals
Transformers	□ Name	Description
Processes	JDE_910_Route	a none
Components	<u>move</u>	The move route defines a simple route that moves the input stream to the output stream.
Adapters	□ pfivp	The pfivp route defines a simple route that is used to invoke the PFIVP process.
Decryptors		The effective state of the second state in the the DEN/DWO second This second state
Ebix	<u>privpws</u>	a transformer to the output segment of the route
Emitters	The state of the state	This sends is used as in the single attention for decomposition
Encryptors	pictures.loader	This route is used to invoke the pictures loader process.
Listeners	pictures.viewer	This route is used to invoke the pictures viewer process.
Preemitters		
Preparsers	<< Back Finish	
Reviewers		

9. Select the defined route (for example, *JDE_910_Route*) and click *Finish*.

You are returned to the Construct Channel pane, as shown in the following image.

iWay Service Ma Server <u>Registry</u> Dep	nager	Tools		Management base	Restart	✓ Ø	Image: Weight of the second
Conduits Channels Inlets Outlets Routes	Channe Channel Process Cons Belo	els / JDE_Event_Channel Is are the pipes through which m es), controlled by Routing Rules struct Channel ow are the components currently	essages flow in iWay Ser and bound to Ports (Lister registered in the channel.	vice Manager. A Channel is hers/Emitters).	s defined as a named c	ontainer of Re	outes (Transformers +
Transformers		Name	Туре	Conditions	Move	Descriptio	n
Processes		JDE_910_Inlet	Inlet			none	
Components		JDE_910_Route	Route	à 4		none	
Adapters Decryptors Ebix	<< Ba	ack Add Delete	Build View				

10. In the Conditions column, click the minus sign icon for the route (to set as default) and then click *Add*.

The Select component type pane opens, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deploy	ager yments Tools	Management base 🗸 🖉 🧭 🥐 7.0.3.245: Restart Licenses About Logout
Conduits Channels Inlets Outlets	Channels / JDE_Event_Channel Channels are the pipes through which n Processes), controlled by Routing Rules Select component type	nessages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers and bound to Ports (Listeners/Emitters).
Routes	Channel Component Types	Description
Transformers	Route	A route is used to define the path a particular message takes through a channel. A Route is
Processes		defined as a sequence of: a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.
Components	 Outlet 	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter
Adapters		
Decryptors	<< Back Next >>	
Ebix		
Emitters		

11. Select Outlet and then click Next.

The Select one or more outlet definitions pane opens, as shown in the following image.

iWay Service Ma Server <u>Registry</u> Depl	nager oyments Tools	Management	base	✓ Ø Ø 2 7.0.3.245 rt Licenses About Logout		
Conduits Channels Inlets Outlets Routes	Channels / JDE_Ever Channels are the pipes the Processes), controlled by Select one or more of Filter By Name V	nt_Channel through which messages flow in iWay Service Manager. A Cha y Routing Rules and bound to Ports (Listeners/Emitters). outlet definitions Where Name V Equals V	annel is defined as a name	d container of Routes (Transformers		
Transformers	Name Description					
Processes	default outlet default outlet defines an empty outlet. An outlet that does not contain an emitter is conside outlet whose emitter is defined by the channels inlet listener.					
Adapters	pictures.outlet	t The pictures.outlet contains an emitter used to write a	n html page.			
Decryptors	CC Back Finish					
Ebix	SS Dack T IIIISII					
Emitters						
Encryptors						

12. Select *default.outlet* and then click *Finish*.

You are returned to the Construct Channel pane, as shown in the following image.

iWay Service Man Server <u>Registry</u> Deple	nager oyments Tools			Management base v 🕢 🖉 ? 0.3.2452 Restart Licenses About Logout
Conduits Channels Inlets Outlets Routes	Channels / JDE_Event_ Channels are the pipes thror Processes), controlled by R Construct Channel Below are the compone	Channel ough which messa outing Rules and b nts currently regis	ges flow in it round to Por pered in the c	Vay Service Manager. A Channel is defined as a named container of Routes (Transformers + is (Listeners/Emitters). hannel.
Transformers	Name	Type Condit	ions Move	Description
Processes	JDE_910_Inlet	Inlet		none
Components	JDE_910_Route	Route 🛛 🔒 🕼)	none
Adapters	default.outlet	Outlet 📑 🛃	,	The default.outlet defines an empty outlet. An outlet that does not contain an
Decryptors				emitter is considered a default outlet whose emitter is defined by the channels
Ebix				iniet listener.
Emitters	CC Back Add	Doloto Buil	d View	
Encryptors	Aug Aug	Delete	J VIEW	
Listeners				
Preemitters				

13. Select all three channel components and then click *Build*.

The build results for the channel are displayed, as shown in the following image.

Way Service I	Manager		Management base		• 🖉	0	
ierver <u>Registry</u>	Deployments Tools						
Conduits Channels Inlets	Click to manage che Channels are the p (Transformers + P	nel deployments across the various instances pes through which messages flow in iW ocesses), controlled by Routing Rules a	of the server. ay Service Manager. A Chann ind bound to Ports (Listeners/E	el is defined as mitters).	a named cor	ntainer of P	outes
Outlets	- Lawson9_Chann Build result for o	els nannel					
Routes	Message leve	Message					
Processes	Info	Start					
	Info	Validating Channel					
Components	Info	Channel is valid					
Adapters	Info	Validating Inlet					
Decryptors	Info	Inlet is valid					
Emitters	Info	Validating Routes					
Encryptors	Info	Routes are valid					
Listeners	Info	Validating Outlets					
Preemitters	Info	Outlets are valid					
Preparsers	Info	Build Successful					
Rules	Info	End					
Schemas Services	Info	Channel archive C:\PROGRA~2\iway \Lawson9_Channels.car has been cr	61\etc\repository\manager\ca eated/updated	\Lawson9_Cha	nnels\Lawso	n9_Channe	ls.1
Transforms	<< Back						

Chapter 7

Application Adapter for J.D. Edwards EnterpriseOne Troubleshooting

The following topics explain the limitations and workarounds when connecting to EnterpriseOne.

The adapter-specific errors listed in this section can occur when you are using the adapter with an iWay Business Services Provider (iBSP) configuration.

In this chapter:

- J.D. Edwards EnterpriseOne Troubleshooting
- Error Messages in iWay Explorer
- Error Messages in J.D. Edwards EnterpriseOne
- Error Messages in iWay Business Services Provider

J.D. Edwards EnterpriseOne Troubleshooting

This topic provides troubleshooting information for J.D. Edwards, separated into the following categories:

- iWay Explorer
- J.D. Edwards
- iBSP

Error Messages in iWay Explorer

The following table describes errors and corresponding solutions for iWay Explorer.

Error	Solution	
Cannot connect to the iWay Application	Ensure that:	
Adapter for J.D. Edwards EnterpriseOne from iWay Explorer.	J.D. Edwards is running.	
	The J.D. Edwards user ID and password are correct.	
	The port number is correct.	
	The custom Component Interface is properly installed.	
The following error message appears:	You provided invalid connection information	
java.lang.lllegalStateException:	for J.D. Edwards EnterpriseOne or the wrong JAR file is in the lib directory.	
Java.lang.Exception: Error Logon to J.D. Edwards EnterpriseOne System		
J.D. Edwards does not appear in the iWay Explorer Adapter node list.	Ensure that the J.D. Edwards JAR files are added to the lib directory.	

Error Messages in J.D. Edwards EnterpriseOne

The following table describes errors and corresponding causes and solutions for J.D. Edwards EnterpriseOne.

Error	Cause	Solution
Action code invalid.	In the Sales Order request, the Action code appears as "H," an invalid action code.	Use: Use:

Error	Cause	Solution
Invalid address number.	The address number does not exist in the Address Book Master file (F0101).	Enter an address number using the Address Book Revisions program (P01051). Ensure that the number entered is correct.
Record invalid	The record being processed either already exists for an ADD function or does not exist for an INQUIRY, CHANGE, or DELETE function.	If you are attempting to inquire, change, or delete a record you added previously, data base problems might exist in your production library. Contact your data processing department.
Item Branch record does not exist.	An Item Branch record (F4102) does not exist for this item in the Branch/ Plant specified.	Correct the Branch or enter an Item Branch record for this item in Branch Plant Item Information (P41026).
&1 does not match any of the valid values.	The &1 does not match any of the valid values specified in the Data Dictionary for this field.	Enter a valid value.
Date out of range.	The Last Service Date and the Inspection Date must be within the range of the effective dates of the Service Contract.	Change the date to be greater than or equal to the beginning effective date and less than or equal to the ending effective date of the Service Contract.

Error Messages in iWay Business Services Provider

This topic discusses the different types of errors that can occur when processing iWay Business Services through iWay Business Services Provider (iBSP).

General Error Handling in iBSP

iWay Business Services Provider (iBSP) 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 iBSP 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 iBSP 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 iBSP 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 iBSP 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>
```

This example shows that iBSP 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 iBSP 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 or not an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in iBSP, 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 is a description of how adapters handle common error conditions and how they are then exposed to the iWay Business Services consumer application.

Example: iWay Application Adapter for J.D. Edwards EnterpriseOne Invalid SOAP Request

When the adapter receives a SOAP request message that does not conform to the WSDL for the web service being executed, 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:CARRIERResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"</pre>
      xmlns="urn:schemas-iwaysoftware-com:iwse"
      cid="2A3CB42703EB20203F91951B89F3C5AF">
      <PS8>
        <error>Cannot find Component Interface {VARRIER}
             (91,2)Initialization failed
             (90,7)Not Authorized
             (90,6)Failed to execute PSSession request
             Cannot find Component Interface {VARRIER} (91,2)
        </error>
      </PS8>
    </m:CARRIERResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example: Invalid SOAP Request

When the adapter receives a SOAP request message that does not conform to the WSDL for the web service being executed, the following SOAP response is generated.

Example: Empty Result From a Request

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

When the adapter executes a SOAP request using input parameters passed that do not match records in the target system, the following SOAP response is generated.



Configuring the Application Adapter for J.D. Edwards EnterpriseOne in an iWay Environment

After you successfully configure the adapter to represent a particular adapter target, the adapter can be assigned to an iWay Service Manager channel.

In this appendix:

Configuring and Deploying the iWay Application System Adapter for J.D. Edwards EnterpriseOne

Configuring and Deploying the iWay Application System Adapter for J.D. Edwards EnterpriseOne

This section describes how to configure and deploy the iWay Application System Adapter for J.D. Edwards EnterpriseOne through the iWay Service Manager (iSM) Administration Console.

Procedure: How to Add the J.D. Edwards EnterpriseOne Adapter

1. Under the Components section on the left pane, click *Adapters*, as shown in the following image.



The Adapters pane opens and shows the adapter object, as shown in the following image.

iWay Service	Manager		Manag	ement base		<u>-</u> Ø	00	7.0.3.245
Server Registry	Deployments Tools				Restart	Licenses	About	Logout
Conduits	Adapters iWay Service Manager implement	ts an adapter container to conf	igure/invoke i)	Way Adapters. The av	fanter container	uses the iW	av Rusines	s Services
Channels	Provider to access configurationa	I metadata on behalf of its ada	pters. Listed b	elow are references	to adapters defin	ned in the re	gistry.	o o er mees
Inlets	Adapters							
Outlets	Filter Du Manua Mahara Ma	ana a di Manada a d						
Routes	Fitter By Name Where Name	me V Equais V	1					1
Transformers	□ Name	Target	References	Description				
Processes	JDEEnterpriseOne_9	10 JDEdwards One World	A	none				
Components	Pictures	RDBMS	a.	The Pictures adap	ter defines app	propriate co	nfiguratio	n
Adapters				information to con	nect to the san	nple HSQL	pictures	
Decryptors				database. This da	tabase is used	in the Picti	ires samp	le.
Ebix	Add Delete Penan	Conv						
Emitters	Aug Delete Kenain	Сору						
Encryptors								
Listeners								
Preemitters								
Preparsers								
Reviewers								
Rules								
Schemas								
Services								
Transforms								
Variables								
Parameters								
Registers								
Recovery								
Recycle Bin								

2. Select the configured J.D. Edwards EnterpriseOne adapter target that is available (for example, *JDEEnterpriseOne_910*) and click *OK*.

The following pane opens, which provides a summary of the configured connection parameters for the selected J.D. Edwards EnterpriseOne adapter target.

iWay Service Ma	nager	Management base		▼ ⊘	00	7.0.3.2452
Server <u>Registry</u> Dep	loyments Tools					
Conduits Channels	Adapters / JDEEnterprise iWay Service Manager impleme Provider to access configuratio	Dne_910 ents an adapter container to configure/invoke iWay Adapters. The adapt nal metadata on behalf of its adapters. Listed below are references to ac	er container dapters defir	uses the iWa ned in the reg	ay Busines gistry.	s Services
Inlets	Adapter JDEEnterpriseOne	_910				
Outlets	Adapter	JDEdwards One World				
Transformers	Target	JDEEnterpriseOne_910				
Processes	Description	Edit description				
Components				< >		
Decryptors	Create Error Document	If on, an error document will be returned when an error occurs				
Ebix		On				
Emitters	Persist Connection	If on, adapter connection will be reused between executes				
Encryptors		🗆 On				
Listeners	Repository					
Preparsers	Repository directory					
Reviewers		C:\E6420\JDE\GenJava\JDE91				
Rules	Sahama atula					
Schemas	Schema style					
Services		ATTRIBUTE_STYLE	~			
Transforms		Select voide	•			
Variables	Logon					
Parameters	User id *					
Registers		JDE				
Recovery	User password *					
Recycle Bin		•••••				
	JDE Environment *					
		DV910				
	Server IP address *					
		IWJDE9x				

3. Verify the configured connection parameters.

Procedure: How to Configure the Listener

1. Under the Components section on the left pane, click *Listeners*, as shown in the following image.

Way Service I	Manage	r				Management base		• @	0	
erver <u>Registry</u>	Deployme									
Conduits Channels Inlets Outlets Routes Transformans	Ada iWay Serv regis	pters Servi ices P try. dapter	ice Manager impl rovider to access s By Name Where	ements an a configurati	adapter contain onal metadata	er to configure/invoke iWay Adapters. on behalf of its adapters. Listed below	The adapte are referen	r container us ces to adapte	ses the iWa rs defined	ay Business in the
Processes			ame	Target	References	Description				
Components	1	La	wson9_adapters	Lawson9	2	none				
Adapters Decryptors		Pi	<u>ctures</u>	RDBMS	2	The Pictures adapter defines appropri the sample HSQL pictures database. sample.	iate configur This databa	ation informat ise is used in	tion to con the Pictu	nect to res
Ebix Emitters Encryptors	'	- <u>S</u>	ciFiBooks	RDBMS	A	The SciFiBooks adapter defines the a connect to the sample HSQL SciFiBo the SciFi Books sample.	appropriate o ooks databa	onfiguration i se. This data	nformation base is us	to ed in
Preemitters References to pro Preparsers	tocol based list	eners t	hat can be used thro	oughout the s	ystem					

The Listeners pane opens, as shown in the following image.

iste	Filter By Name W	Mere Name Eq	uais 💌	
Г	Name	Туре	References	Description
Г	file1	File	2	A default/sample file listener.
	javadoc	HTTP 1.0 [deprecated]	2	The javadoc listener is used to make the iWay Service Manager API available to a remote browser.
	pictures.loader	File	R	The pictures listener locates files with a variety of common imag file extensions (img, gif, jpg,).
	pictures.viewer	HTTP 1.0 [deprecated]	R	The pictures.viewer is used to kickoff the image retrieval process as defined by the pictures sample.
	scifibooks	Schedule Recurring Execution	A	This listener is defined for use by the SciFi Books sample. It wakes up daily and kicks off the update for the channel.
	SOAP2	SOAP	R	This listener is used by the stock SOAP channel.

2. Click Add.

The Select listener type pane opens, as shown in the following image.

Listeners Listeners are protocol that are defined in the	handlers, that receive input for a channel from a configured endpoint. Listed below are references to the listeners registry.
Select listener type	e
Type *	Type of the new listener
<< Back Next	Select a type Select a type AQ AS1 AS2 AS2 Backup Heartbeat Server ConnectDirect CS3 Email Exchange File FTP[S] Client FTP[S] Server HTTP 1.0 [deprecated] HTTP 1.1 [nonblocking] (nhttp) iEI Internal Queue Java Message Service (jmsq) Khalix

3. Select File from the Type drop-down list and click Next.

The Configuration parameters pane for the File listener opens, as shown in the following image.

	Configuration compared	for one listopon of two file					
Outlets	comparation parameters for new inscener of type rise						
Routes	Input Path *	Directory in which input messages are received. A specific file name or (DOS-style regular expression pattern) can be used. If you include the suffix in the pattern (such as ab*.xml) then be sure to configure the Suffix In to					
Transformers		allow any suffix. Multiple locations can be specified, separated by ',' or ',' character.					
Processes		C:\input Browse					
omponents	Destination *	Directory into which output files are stored. Specific file name is optional. Use * in file name to be replaced by timestamp. # by sequential counter if required directories are not present at runtime. ISM will attempt to create					
Adapters		them. At runtime, if it is unclear whether path names a directory or a filename, ISM will assume the path names a					
Ehiv		ine.					
Emitters		C:\output Browse					
Encryptors	Removal Destination	Full path file pattern asserting where input files will be moved. Use * in file name to be replaced by timestamp, #					
Listeners		by sequential counter					
Preemitters		Browse					
Preparsers	Suffix In Filter	The listener will match the extension exactly, plus the corresponding all-uppercase and all-lowercase extensions.					
Reviewers		List are supported: enter XML, in to accept files with extensions xml and in in either case. Do not use '.', use - to					
Rules		mean no extension, or * to mean any extension.					
Schemas		xml					
Services	Scan subdirectories	If true, all subdirectories will be scanned for files to process					
1101010110		false					
'ariables		Pick one					
Parameters							
Registers	Do not unzip ZIP files	Pass ZIP files as a single file for processing (requires ACCEPT FLAT turned on)					
ecovery		false					
Decycle Bio		Pick one					
Recycle bin	Suffix Out	Extension for output files (name is same as input file unless specified in destination parameter)					
		xml					

4. In the Input Path field, specify the directory where the input messages are received. For example:

C:/input

- In the Destination field, specify the directory where output files are stored. For example: C:/output
- 6. In the Suffix Out field, specify the extension for output files. For example:

xml

7. Click Next.

The following pane opens, which allows you to specify a name for the new listener.

iWay Service Mar	nager	Management base	
Server <u>Registry</u> Deple	oyments Tools		
Conduits Channels	Listeners Listeners are protocol handlers, defined in the registry.	that receive input for a channel from a configured endpoint. Listed below a	re references to the listeners that are
Inlets	Select listener type		
Outlets	Name *	Name of the new listener	
Transformers		JDE_910_Listener ×	
Processes	Description	Description for the new listener	
Components			\bigcirc
Adapters			
Decryptors	< Back Finish		
Ebix			
Emitters			
Encryptors			
Usteners			
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

Specify a listener name (for example, JDE_910_Listener) and click Finish.
Procedure: How to Configure a Process and Define a Route

1. Under the Conduits section on the left pane, click *Processes*, as shown in the following image.

Way Service Manag	ger			Manageme	nt base		• 🖉	0	xfoc.44054
ierver <u>Registry</u> Deploy	ment								
Conduits L	isten	ers							
Channels t	istene hat are	rs are protocol handlers, e defined in the registry.	that receive input for a chan	nel from a cor	ifigured endpoint. L	isted below	w are referen	ces to the	listeners
Inlets	Liste	ners							
Outlets		C'h							_
Routes		Filter By Name Where Nam	e 🗾 Equals 💆						
Transformers		Name	Туре	References	Description				
Processes	П	file1	File	A	A default/sample f	ile listener	r.		
compon References to processes t	hat car	n be used throughout the sys	tem		The investor listen	ar in used	to make the	Mar Can	ine
Adapters		Javauuu	HTTP 1.0 [deprecated]	2	Manager API avail	able to a r	emote brows	er. Br.	ice
Decryptors		Lawson9_file_listeners	File	2	none				
Ebix		nictures loader	File		The nictures lister	er locates	files with a v	ariety of co	ommon
Emitters		protores.rowger	1 10		image file extension	ons (img, g	gif, jpg,).	anery or co	011111011
Encryptors		pictures viewer	HTTP 1.0 [deprecated]	2	The pictures.viewe	r is used t	to kickoff the	image retr	ieval
Listeners					process as define	d by the pi	ictures sampl	е.	
Preparcers		scifibooks	Schedule Recurring	2	This listener is def	fined for us	se by the Scil	Fi Books s	ample.
Reviewers			Execution	-	it wakes up daily a	and kicks	of the update	for the ch	annei.
Rules		SOAP2	SOAP	2	This listener is use	ed by the s	stock SOAP	channel.	
Schemas		1							
Services	Add	Delete Renam	е Сору						

Processes Processes are stateless, lightweight, short-lived microflows that are executed by the iWay Service Manager on messages/documents as they pass thru the system. Processes, typically authored by the iWay Service Designer, can be bound to channels or exposed as Web Services by the iWay Business Service Provider thru the iWay Adapter. Process Definitions Filter By Name Where Name Ŧ Equals -Name **View References Description** B The move1 service defines a move agent that moves the input move document stream to the output document stream. It represents the basic echo pattern in iSM. 3 This sample process, delivered with iWay Designer, copies a Samples.PFIVP.1 ÷ subtree of the input document as defined by the PFIVP schema to the root of the output document as defined by PFIVPResponse schema. Samples.PFIVPWS.1 ÷ A This sample process, delivered with iWay Designer, illustrates the invocation of a simple iWay Business Service from a flow. A Samples.Pictures.Load.1 Ē The Pictures Load process is used to insert images into a RDBMS table Samples.Pictures.RetrieveAlburn.1 ÷ A The Pictures.RetrieveAlburn process is used to get images from an RDBMS table and generate a photo album as an html page. Samples.SciFiBooks.1 Q. ÷ The SciFiBooks process is used to define the business logic implemented by the SciFi Books sample. This sample is built around the concept of tracking new science fiction books as they are published and released. Сору Add Delete Rename

The Processes pane opens, as shown in the following image.

The New Process Definition pane opens, which allows you to specify a name for the new process definition, as shown in the following image.

iWay Service M	lanager	Management base 🗸 🖉 🖉 7.0.3.2452
Server <u>Registry</u> D	eployments Tools	
Conduits Channels Inlets	Processes Processes are stateless, ligi the system. Processes, typi Service Provider thru the iV	htweight, short-lived microflows that are executed by the iWay Service Manager on messages/documents as they pass thru ically authored by the iWay Service Designer, can be bound to channels or exposed as Web Services by the iWay Business Way Adapter.
Outlets	New Process Definition	
Routes	Name *	Name of the new Process object definition
Transformers	_	JDE_910_Process ×
Processes	Description	Description for the new Presson abject definition
Components	Description	Description for the new Process object delimition
Adapters		\bigcirc
Decryptors		
Ebix	<< Back Finish	
Emitters		
Encryptors		
Listeners		
Preemitters		
Preparsers		
Reviewers		
Rules		
Schemas		
Services		
Transforms		
Variabler		
Desemption		
Parameters		
Registers		
Recovery		
Recycle Bin		

- 3. Specify a process name (for example, JDE_910_process) and click Finish.
- 4. The Construct Process pane opens, which allows you to construct the new process (JDE_910_process) by adding supported components, as shown in the following image.

Processe Processe they pass Services	es / Lawson9_pr s are stateless, lin s thru the system by the iWay Busi	rocesses ghtweight, short-lived mi . Processes, typically aut ness Service Provider thr	croflows that are execute hored by the iWay Servic ru the iWay Adapter.	ed by the iWay Service Manager on messages/documents ce Designer, can be bound to channels or exposed as Wel	as
Constr	uct Process	hierts currently assigned to	the process. The order o	f centre object everytion may be channed by checking a	
comp	onent and using th	he 'Move Up' and 'Move Do	wn' buttons.	a service object execution may be changed by checking a	
	Name	Туре	Move	Description	
4 1	No data was found	d.			
<< Bac	k Add	Delete			

The Assign service object reference pane opens, as shown in the following image.

iWay Service M	anager			Management 📴 🗸 🔕 📀 7.0.3.245	
Server <u>Registry</u> De	ployments				
Conduits Channels Inlets Outlets	Process Processe the syste Service I	ses / JDE_910_Process es are stateless, lightweight, em. Processes, typically aut Provider thru the iWay Adag an service object reference	, short-lived hored by th oter. es to proce	I microflows that are executed by the iWay Service Manager on messages/documents as they pass the iWay Service Designer, can be bound to channels or exposed as Web Services by the iWay Busines ass JDE 910 Process	nru is
Routes	Belo	w is a list of service objects	currently de	fined on the server. Select one or more service objects and click Finish to assign.	
Transformers	_ F	Filter By Name Where Name	~	Equals V	
Processes		Name	Туре	Description	
Components		JDEEnterpriseOne_910	Adapter	none	
Adapters Decryptors		move1	Service	The move1 service defines a move agent that moves the input document stream to the output document stream. It represents the basic echo pattern in iSM.	
Ebix Emitters		Pictures	Adapter	The Pictures adapter defines appropriate configuration information to connect to the sample HSQL pictures database. This database is used in the Pictures sample.	
Encryptors		pictures.img2xml	Service	converts the image to base64 and wraps it in a <picture> tag</picture>	
Listeners		pictures.iterator	Service	Iterate a loop for each portion of an XML document	
Preparsers		RSSRead1	Service	Reads an RSS Document from url that is specified in the original incomming document.	
Reviewers Rules		Snip1	Service	Copies a subtree of the input document as defined by the PFIVP schema to the root of the output document as defined by PFIVPResponse schema.	
Schemas Services	<< Ba	ick Finish			
Transforms					
Variables Parameters					
Registers					
Recovery					
Recycle Bin					

6. Select JDEEnterpriseOne_910 and click Finish.

7. Under the Conduits section on the left pane, click *Routes*, as shown in the following image.

iWay Service	Manager		Management base 🗸 🕢 🖉 🕐 7.0.3.2452 -
Server Registry	Deployments Tools		
Conduits Channels Inlets Outlets	Routes A route is used to define th process, followed by anoth Route Definitions	he path a particular n er transformer, follov	nessage takes thru a channel. A Route is defined as a sequence of: a transformer, followed by a ved by zero or more outlets.
Routes	Filter By Name Wh	iere Name 🗸 🗸	Equals V
Transformers	Name	View References	Description
Processes	<u>move</u>	æ 🖪	The move route defines a simple route that moves the input stream to the output stream
Components	C ofwo	2 1	The offine route defines a simple route that is used to invoke the PEIVP process
Adapters		0 A	The proprote defines a simple route that is used to invoke the PTTVP process.
Decryptors	pfivpws	÷ 4	The pfivpws route defines a simple route that is used to invoke the PFIVPWS process. This version adds a transformer to the output segment of the route
Ebix	D nistures leader		This relation aloss a number to the output segment of the roles.
Emitters	pictures.toader	10° 44	This route is used to invoke the pictures loader process.
Encryptors	pictures.viewer	÷ 4	This route is used to invoke the pictures viewer process.
Disteners			
Prenarsers	Add Delete F	Rename Copy	
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

iWay Service Mar	nager	Management bose 🗸 🙆 📀 7.0.3.2452
Server <u>Registry</u> Deple	oyments Tools	Restart Licenses About Logout
Conduits	Routes	the environmental sector states a share of a Bristolic defined as a second of a two-former followed by a
Channels	process, followed by another tra	in a particular message takes thru a channel. A koute is defined as a sequence or: a transformer, followed by a Insformer, followed by zero or more outlets.
Inlets	New Route Definition	
Outlets	Name *	Name of the new route
Routes		IDE 010 Portal
Transformers		105-210-Kord
Processes	Description	Description for the new route
Components		\bigcirc
Adapters		
Decryptors	< Back Finish	
Ebix		
Emitters		
Encryptors		
Listeners		
Preemitters		
Preparsers		
Reviewers		
Rules		
Schemas		
Services		
Transforms		
Variables		
Parameters		
Registers		
Recovery		
Recycle Bin		

The New Route Definition pane opens, as shown in the following image.

9. Specify a route name (for example, JDE_910_Route) and click *Finish*.

The Construct Route pane opens, which allows you to construct the new route (JDE_910_Route) by associating a configured process, as shown in the following image.

iWay Service M	anager ployments Tools		Managemei	nt base Resta	✓ Ø Ø ? 7.0.3.24 rt Licenses About Logout	52
Conduits Channels Inlets Outlets	Routes / JDE_910_Roi A route is used to define t process, followed by anoth Construct Route Below are the compon	ute the path a particular mes her transformer, followed ents currently registered	isage takes thru a channel. A Roi d by zero or more outlets. in the route.	ute is defined as a sequence	of: a transformer, followed by a	
Routes		-	and the second			
Transformers	Name	Туре	Conditions	Descripti	on	
Processes	No data was for	und.				
Components						
Adapters	<< Back Add	Delete View				
Decryptors						
Ebix						
Emitters						
Encryptors						
Listeners						
Preemitters						
Preparsers						
Reviewers						
Rules						
Schemas						
Services						
Transforms						
Variables						
Parameters						
Registers						
Recovery						
Recycle Bin						

10. Click Add.

The Select component type pane opens, as shown in the following image.

lo.	et component type	
161	ct component type	
	Component Types	Description
5	In Transformer	In Transformers are exit sequences that apply to the message before the process.
•	Process	Processes are stateless, lightweight, short-lived microflows that are executed by the iWay Service Manager on messages/documents as they pass through the system.
5	Out Transformer	Out Transformers are exit sequences that apply to the message after the process.
5	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter.

11. Select Process and then click Next.



The Select a process definition pane opens, as shown in the following image.

12. Select the configured process (JDE_910_Process) and click Finish.

Procedure: How to Define an Inlet

1. In the Conduits section of the left pane, click *Inlets*.

iWay Service Ma	anager		Management 🔤 🗸 🖉 😨 7.0.3.2452
Server <u>Registry</u> De	ployments Tools		Restart Licenses About Logout
Conduits	Inlets		ante late a descal. Indete contain a listence. Descatter, and Descares
Channels	Inlets are conduits which	represent the	entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.
Inlets	Inlet Definitions		
Outlets	Filter By Name W	here Name	✓ Equais ✓
Routes			
Transformers	Name	References	Description
Processes	□ <u>file1</u>	2	The file1 inlet contains the file1 listener and is a part of the file1 sample channel.
Components	pictures loader	4	The pictures loader inlet contains the pictures loader listener and is a part of the pictures loader channel.
Adapters	pictures viewer		The pictures viewer inlet contains the pictures viewer listener and is a part of the
Decryptors			pictures.viewer channel.
Ebix	SOAPInlet	3	This inlet is used by the stock SOAP channel.
Emitters			
Encryptors	Add Delete	Rename	Copy
Listeners			
Preemitters			
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

The Inlets pane opens, as shown in the following image.

iWay Service Mar	nager	Management base	🗸 🖉 🧭 😨 7.0.3.2452
Server <u>Registry</u> Deple			
Conduits	Inlets Inlets are conduits which repr	esent the entry into a channel. Inlets contain a Listener, Decryptor, and Prepar	sers.
Inlets	New Inlet Definition		
Outlets	Name *	Name of the new inlet	
Routes		JDE_910_Inlet ×	
Transformers			
Processes	Description	Description for the new inlet	
Components			0
Adapters	<< Back Finish		
Decryptors	SS DOON THINKI		
Ebix			
Emitters			
Encryptors			
Listeners			
Preemitters			
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

The New Inlet Definition pane opens, as shown in the following image.

3. Specify an inlet name (for example, JDE_910_inlet) and click Finish.

The Construct Inlet pane opens, which allows you to construct the new inlet (JDE_910_Inlet) by associating supported inlet components, as shown in the following image.

in Ir	lets / lets a	Lawson9_inlets are conduits which repres	sent the entry into	a channel. Inlets contain a	Listener, Decryptor, and Pr	eparsers.
r.	Cons	struct Inlet				
	Belo com	ow are the components cu oponent type by checking a	rrently registered in a component and us	the inlet. The order of decry ing the 'Move Up' and 'Move	ptor and preparser componer Down' buttons.	nts may be changed within each
	П	Name	Туре	Move	Description	
	П	No data was found.				
L	-					
	<< B	ack Add Delet	te			

The Select component type pane opens, as shown in the following image.

Select component type				
	Component Types	Description		
•	Listener	Listeners are protocol handlers, that receive input for a channel from a configured endpoint.		
Ô	Decryptor	Decrypts the document.		
0	Preparser	A logical process that handles documents before they are parsed by the system. Usually used to convert from non-XML to xml.		

5. Select Listeners and then click Next.

The Select a listener definition pane opens, as shown in the following image.

iWay Service Man	1ager syments Tools		Management base V 🕢 🐼 📀 7.03.2452 Restart Licenses About Logout
Conduits Channels Inlets Outlets	Inlets / JDE_910_Inlet Inlets are conduits which repu Select a listener definition Filter By Name Where	resent the entry into a chan	nel. Inlets contain a Listener, Decryptor, and Preparsers.
Routes	Name	Туре	Description
Transformers	O file1	File	A default/sample file listener
Processes			
Components	JDE_910_Listener	File	none
Adapters	 pictures loader 	File	The pictures listener locates files with a variety of common image file extensions (img, gif, jpg,).
Decryptors	 pictures, viewer 	HTTP 1.0 [deprecated]	The pictures viewer is used to kickoff the image retrieval process as defined
Ebix			by the pictures sample.
Emitters	O SOAP2	SOAP	This listener is used by the stock SOAP channel.
Encryptors			,
Listeners	<< Back Finish		
Preemitters	Baar		
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

6. Select the configured listener (for example, JDE_910_Listener) and click Finish.

Procedure: How to Construct a Channel

1. Under the Conduits section in the left pane, click *Channels*.

iWay Service M Server <u>Registry</u> De	anager	Tools					Management base 🗸 🕢 🐼 😨 7.0.3.245 Restart Licenses About Logout
Conduits	Channel	S					in New York in the second in the Frederic and a second section of Particle (Transformers)
Channels	Processes), controlled by F	ougn w Routing	Rules	and bo	und to	Ports (Listeners/Emitters).
Inlets	- Chann	el Definitions					
Outlets							
Routes		ter By Name Wr	nere Nar	ne	~	Eq	uals 🗸
Transformers		Name	Туре	Regs	Ebix	View	Description
Processes		lefault		0	0	æ	The default channel can be used as a starting point for quickly defining
Components			-	-	Ē.,	Ť	functionality in the system. This template defines the minimal conduits and components required for deployment. You can copy this channel, add a listener,
Adapters							build and deploy.
Decryptors		ile1	6	0	<u>0</u>	•	The file1 channel is based on the default channel. It adds an inlet that contains a file listener and completes the sample file channel.
Emitters		ile2	6	<u>0</u>	<u>0</u>	6	The file2 channel is based on the file1 channel. It uses a route that contains the PFIVP process
Listeners		ile3	4	<u>0</u>	Q	6	The file3 channel is based on the file2 channel. It uses a route that contains the PEI/VPWS process and adds a reviewer to the mix
Preemitters Preparsers		ile4	4	<u>0</u>	<u>0</u>	÷	The file4 channel is based on the file3 channel. It includes routes as defined by the file1 file2 and file3 channels. This channel illustrates a multi-routed conduit
Reviewers Rules Schemas	p	bictures.loader	۵	Q	Q	()	The pictures loader channel is used save image files to a database. It is one of the channels defined by the pictures sample which is built around the idea of tracking new images as they are recognized by the system.
Services Transforms	£	bictures.viewer	4	Q	<u>0</u>	()	The pictures viewer channel is used retrieve image files from a database. It is one of the channels defined by the pictures sample which is built around the idea of tracking new images as they are recognized by the system.
Variables		SOAP2	1	<u>0</u>	<u>0</u>	6	This channel can be used to add IBSP (SOAP) services to an iWay Application.
Registers	Add	Delete	Renam	e	Сору		Build
Recovery							
Recycle Bin							

The Channels pane opens, as shown in the following image.

3. Specify a channel name (for example, *JDE_910_Channel*) and click *Finish*, as shown in the following image.

iWay Service Ma	nager	Management base	🗸 - 🕢 🤗 😨 7.0.3.2452 -
Server <u>Registry</u> Depl	loyments Tools		
Conduits Channels	Channels Channels are the pipes through Processes), controlled by Routin	which messages flow in iWay Service Manager. A Channel is defined as a n g Rules and bound to Ports (Listeners/Emitters).	amed container of Routes (Transformers +
Inlets	New Channel Definition		
Outlets	Name *	Name of the new channel	
Routes		JDE_910_Channel ×	1
Transformers	-		1
Processes	Description	Description for the new channel	
Components			0
Adapters			
Decryptors	<< Back Finish		
Ebix			
Emitters			
Encryptors			
Listeners			
Preemitters			
Preparsers			
Reviewers			
Rules			
Schemas			
Services			
Transforms			
Variables			
Parameters			
Registers			
Recovery			
Recycle Bin			

The Construct Channel pane opens, which allows you to construct the new channel (JDE_910_Channel) by associating supported channel components.

Channels / Lawson9_Channels Channels are the pipes through which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters).						
Construct Channel						
Below are the components	currently register	ed in the channel.				
□ Name T	□ Name Type Conditions Move Description					
No data was found.						
< Back Add Del	lete Build	View				

The Select component type pane opens, as shown in the following image.

ann anne ansf Sele	els / Lawson9_Channels els are the pipes through which i ormers + Processes), controlled ct component type	messages flow in iWay Service Manager. A Channel is defined as a named container of Routes d by Routing Rules and bound to Ports (Listeners/Emitters).
	Channel Component Types	Description
•	Inlet	Inlets are conduits which represent the entry into a channel. Inlets contain a Listener, Decryptor, and Preparsers.
0	Route	A route is used to define the path a particular message takes through a channel. A Route is defined as a sequence of: a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.
0	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter

- 5. Select Inlet and then click Next.
- 6. Select the defined inlet (for example, *JDE_910_Inlet*) and click *Finish*, as shown in the following image.

iWay Service Mana Server <u>Registry</u> Deploy	I ger ments Tools	Management 🐱 🕶 🕢 🐼 😨 70.3.2452 Restart Licenses About Logout
Conduits (C Channels) Inlets Outlets Boutes	Channels / JDE_910_C channels are the pipes the rocesses), controlled by R Select an inlet definiti	Channel Ough which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Routing Rules and bound to Ports (Listeners/Emitters). on here Name
Transformers	Name	Description
Processes	⊖ <u>file1</u>	The file1 inlet contains the file1 listener and is a part of the file1 sample channel.
Components	IDE_910_Inlet	none
Adapters	o pictures.loader	The pictures.loader inlet contains the pictures.loader listener and is a part of the pictures.loader channel.
Decryptors	 pictures viewer 	The pictures viewer inlet contains the pictures viewer listener and is a part of the pictures viewer channel
Ebix		This is let is used by the start COAD shares I
Emitters	O SOAPIniet	This inlet is used by the stock SOAP channel.
Encryptors		
Listeners	< Back Finish	
Preemitters		
Preparsers		
Reviewers		
Rules		
Schemas		
Services		
Transforms		
Variables		
Parameters		
Registers		
Recovery		
Recycle Bin		

You are returned to the Construct Channel pane, as shown in the following image.

Server Registry Deployments Tools Conduits Channels / JDE_910_Channel Channels are the pipes through which messages flow in iWay Service Manager. A Channel is define Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters). Inlets Outlets Construct Channel Construct Channel Routes Construct Channel Below are the components currently registered in the channel. Transformers Processes Inlet Conditions Processes JDE_910_Inlet Inlet Components Adapters JDE_910_Inlet Inlet Obscryptors Ebix Emitters Preparsers Preparsers Reviewers Rules Schemas Schwers Schwars Schemas Services Transforms Variables Parameters Registers		 @ • 📀	7.0.3.2
Conduits Channels / JDE_910_Channel Channels are the pipes through which messages flow in iWay Service Manager. A Channel is define Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters). Inlets Construct Channel Outlets Construct Channel Routes Below are the components currently registered in the channel. Processes JDE_910_Inlet Inlets JDE_910_Inlet Inlets JDE_910_Inlet Decryptors JDE_910_Inlet Ebix Emitters Preparsers Reviewers Rules Schemas Services Transforms	Restart	rt Licenses Ab	out Logoi
Channels Channels are the pipes through which messages how in IVay Service Manager: A Channel is define Processes; Controlled by Routing Rules and bound to Ports (Listeners/Emitters). Inlets Construct Channel Outlets Routes Transformers Processes; Outpined by Routing Rules and bound to Ports (Listeners/Emitters). Intets Outpine the components currently registered in the channel. Below are the components currently registered in the channel. Intet Components JDE_910_Intet Intet Adapters JDE_910_Intet Intet Decryptors Ebix Emitters Encryptors Listeners Preemitters Presentiters Presentiters Rules Schemas Services Transforms Variables Parameters Registers			17
Inlets Outlets Routes Construct Channel Below are the components currently registered in the channel. Processes Decryptors Decryptors Ebix Emitters Encryptors Listeners Preemitters Premitters Reviewers Reviewers Reviewers Raules Schemas Services Transforms Variables Parameters Registers	ed as a named	d container of Route	(Transforme
Outlets Below are the components currently registered in the channel. Routes Image: Type Conditions Transformers JDE_910_Inlet Processes JDE_910_Inlet JDE_910_Inlet Inlet Components Adapters Decryptors Ebix Emitters Proparsers Reviewers Rules Schemas Services Transforms Variables Parameters Registers			
Name Type Conditions P Processes JDE 910 Inlet Inlet Inlet Components <<			
Transformers Processes JDE_910_Inlet Inlet Components Adapters Components Adapters Components Compo			
Processes	1ove	Description	
Adapters Components Co		none	
Adapters << Back Add Delete Build View			
Decryptors Ebix Ebix Emitters Encryptors Listeners Preemitters Preparaers Reviewers Reviewers Rules Schemas Services Transforms Ariables Parameters Registers			
Ebix Emitters Emitters Encryptors Listeners Preemitters Preparsers Reviewes Rules Schemas Services Transforms Parameters Registers			
Emitters Encryptors Listeners Preemitters Preparsers Reviewers Rules Schemas Services Transforms Ariables Parameters Registers			
Encryptors Listeners Premitters Preparsers Reviewers Rules Schemas Services Transforms ariables Parameters Registers			
Listens Preemitters Preparsers Reviewers Reviewers Schemas Services Transforms ariables Parameters Registers			
Preemitters Preparsers Reviewers Rules Schemas Services Transforms Parameters Registers			
Preparsers Reviewers Rules Schemas Services Transforms Parameters Registers			
Reviewers Rules Schemas Services Transforms ariables Parameters Registers			
Rules Schemas Services Transforms Ariables Parameters Registers			
Schemas Services Transforms Ariables Parameters Registers			
Services Transforms ariables Parameters Registers			
Transforms ariables Parameters Registers			
fariables Parameters Registers			
Parameters Registers			
Registers			
ecovery			
Recycle Bin			

7. Click Add.

The Select component type pane opens, as shown in the following image.

Channels / Lawson9_Channels Channels are the pipes through which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters).								
	Description							
	•	Route	A route is used to define the path a particular message takes through a channel. A Route is defined as a sequence of: a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.					
	C	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter					
<< Back Next >>								

8. Select Route and then click Next.

The select one or more route definitions pane opens, as shown in the following image.

iWay Service	Manager Deployments Tools	Management base 🗸 🔕 📀 7.0.3.2452 Restart Licenses About Logout
Conduits	Channels / JDE_910_Channels are the pipes thro	nannel ugh which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers +
Channels	Processes), controlled by Re	outing Rules and bound to Ports (Listeners/Emitters).
Inlets	Select one or more rou	te definitions
Outlets	Filter By Name Whe	re Name V Equals V
Routes		
Transformers	Name	Description
Processes	JDE_910_Route	none
Components	move	The move route defines a simple route that moves the input stream to the output stream.
Adapters		The nfive route defines a simple route that is used to invoke the PFIVP process
Decryptors		The pitty route dennes a simple route maris used to invoke the PTTYP process.
Ebix	pfivpws	The pfivpws route defines a simple route that is used to invoke the PFIVPWS process. This version adds a transformer to the output segment of the route.
Emitters		This route is used to invoke the nictures leader process
Encryptors	precirco.iouder	
Listeners	pictures.viewer	This route is used to invoke the pictures viewer process.
Preemitters		
Preparsers	<< Back Finish	
Reviewers		
Rules		
Schemas		
Services		
Transforms		
Variables		
Parameters		
Registers		
Recovery		
Recycle Bin		

9. Select the defined route (for example, *JDE_910_Route*) and click *Finish*.

You are returned to the Construct Channel pane, as shown in the following image.

iWay Service	Vanager		Management base		
Server <u>Registry</u>	Deployments Tools				
Conduits	Channels / JDE_910_Channel				
Channels	Channels are the pipes through which me Processes), controlled by Routing Rules a	essages flow in iWay Se and bound to Ports (List	rvice Manager. A Channel i eners/Emitters).	s defined as a name	d container of Routes (Transformers
Inlets	Construct Channel	•			
Outlets	Below are the components currently r	egistered in the channel	L		
Routes					
Transformers	□ Name	Туре	Conditions	Move	Description
Processes	JDE_910_Inlet	Inlet			none
Components	JDE_910_Route	Route	6		none
Adapters					
Decryptors	<< Back Add Delete	Build View			
Ebix					
Emitters					
Encryptors					
Listeners					
Preemitters					
Preparsers					
Reviewers					
Rules					
Schemas					
Services					
Transforms					
Variables					
Parameters					
Registers					
Recovery					
Recycle Bin					

10. In the Conditions column, click the minus sign icon for the route (set as default) and then click *Add*.

The Select component type pane opens, as shown in the following image.

000	Chan Chann Trans - Sele	nels / Lawson9_Channels lels are the pipes through which formers + Processes), controlled ect component type	messages flow in iWay Service Manager. A Channel is defined as a named container of Routes I by Routing Rules and bound to Ports (Listeners/Emitters).				
		Channel Component Types	Description				
	C	Route	A route is used to define the path a particular message takes through a channel. A Route is defined as a sequence of: a transformer, followed by a process, followed by another transformer, followed by zero or more outlets.				
	•	Outlet	Outlets are conduits which contain Preemitters, Encryptors, and an Emitter				
	<< Back Next >>						

11. Select Outlet and then click Next.

The Select one or more outlet definitions pane opens, as shown in the following image.

Chann Chann (Trans	Channels / Lawson9_Channels Channels are the pipes through which messages flow in iWay Service Manager. A Channel is defined as a named container of Routes (Transformers + Processes), controlled by Routing Rules and bound to Ports (Listeners/Emitters).						
Sele	ect one or more	outlet definitions					
	Filter By Name V	Ahere Name 💌 Equals 💌					
	Name	Description					
9	default.outlet	The default outlet defines an empty outlet. An outlet that does not contain an emitter is considered a default outlet whose emitter is defined by the channels inlet listener.					
	pictures.outlet The pictures.outlet contains an emitter used to write an html page.						
<< E	<< Back Finish						

12. Select *default.outlet* and then click *Finish*.

You are returned to the Construct Channel pane, as shown in the following image.

iWay Service Mar	nager				Management base 🗸 🕢 🕢 🧭 😨 7.0.3.2452
Server <u>Registry</u> Deplo					
Conduits	Channels / JDE_910_Ch	annel			
Channels	Channels are the pipes throu Processes), controlled by Ro	ugh which uting Rule	messages flo es and bound	to Port	/ay Service Manager. A Channel is defined as a named container of Routes (Transformers - s (Listeners/Emitters).
Inlets	- Construct Channel				
Outlets	Below are the componen	its current	lv registered	in the cl	nannel
Routes			,,		
Transformers	Name	Туре	Conditions	Move	Description
Processes	JDE_910_Inlet	Inlet			none
Components	JDE_910_Route	Route	i -		none
Adapters	default.outlet	Outlet	👍 💫		The default outlet defines an empty outlet. An outlet that does not contain an
Decryptors					emitter is considered a default outlet whose emitter is defined by the channels
Ebix					iniet listener.
Emitters	CC Back Add [Doloto	Build	View	1
Encryptors	Nuu I	Delete	Duliu	VICW	
Listeners					
Preemitters					
Preparsers					
Reviewers					
Rules					
Schemas					
Services					
Transforms					
Variables					
Parameters					
Registers					
Recovery					
Recycle Bin					

13. Select all three channel components and then click *Build*.

The build results for the channel are displayed, as shown in the following image.

Way Service Mana	iger		Management	base		• 🙆	0	xfoc.44054
ierver <u>Registry</u> <u>Deplo</u>	yments Tools							
Conduits Channels Inlets Outlets	Click to manage chann Channels are the pip (Transformers + Pro Lawson9_Channel Build result for cha	el deployments across the various instances of es through which messages now in two cesses), controlled by Routing Rules an S innel	the server. y Service Manage d bound to Ports (r. A Channel i Listeners/Emi	s defined as tters).	a named co	ntainer of F	Routes
Routes	Message level	Message						
Transformers Processes	Info	Start						
Components	Info	Channel is valid						
Adapters Decryptors	Info	Validating Inlet						
Ebix Emitters	Info	Validating Routes						
Encryptors	Info	Routes are valid						
Preemitters	Info	Outlets are valid						
Reviewers	Info	Build Successful						
Rules Schemas	Info	End Channel archive C:\PROGRA~2\iway6	1\etc\repository\m	nanager\car\La	awson9_Cha	annels\Lawso	n9_Channe	els.1
Services Transforms	<< Back	\Lawson9_Channels.car has been cre	ated/updated					

Procedure: How to Deploy and Start a Channel

1. Click *Deployments* in the top pane, as shown in the following image.



The Channels pane is displayed by default, as shown in the following image.

Way Service	Manager				base		•	2 🥝 📀	
ierver Registry									
lanagement	Channels Manage Chappels which have	a been denlover	1.						
Channels									
Services Directory	The channels listed below	are deployed. Se	elect a channel to und	eploy, repair, :	start, stop, o	r deploy a ne	≪ channel fro	m the reposito	ny.
Web Services	Channel Name	Protocol	Deploy Date	Version	Status	Active	A-C-S-F	Descripti	on
	No deployed channel	ls were found.							
	Deploy Undeploy	Redeploy	Repair Start	Stop					

2. Click Deploy.

The Available Channels pane is displayed, as shown in the following image.

iWay Service M	anager ployments Tools	Man	agement base	✓ (Restart Licens	Image: Second State 7.0.3.2452 ses About Logout
Management Channels Services Directory	Channels Manage Channels which have been Available Channels This is a list of channels ready for channel that has already been di	deployed. r deployment into the selected Managed pployed in to the selected Managed Ser	d Server. Select the channels a ver. To Undeploy or Redeploy :	ind click deploy. Y a channel, go bac	ou can not deploy a k to the previous page.
Web Services	Channel Name JDE_910_Channel << Back Deploy Get*	Build Date Jun 29 2015 04:18 PM Versions	Built On http://GR10618	Version 1	Description

3. Select JDE_910_Channel and click Deploy.

The Channel Management pane is displayed, as shown in the following image.

iWay Service I Server Registry	Manager Deployments Tools		Managen	ent base		Resta	✓ 🤅 art License) 🔗 😨 7.0.3.2452 Es About Logout
Management Channels Services	Channels Manage Channels which have be Channel Management The channels listed below a	een deployed	d. Select a channel to undeploy, repair,	, start, stop,	or deploy	a new ch	nannel from th	e repository.
Directory	Filter By Name Where N	lame	✓ Equals ✓					
Web Services	Channel Name JDE 910 Channel	Protocol file	Deploy Date/By Jun 29 2015 04:20 PM / iway	Version	Status	Active	A-C-S-F	Description
	Deploy Undeploy	Redeploy	Repair Reset Statistics	Start	Stop	•		

4. Select JDE_910_Channel and click Start.

The JDE_910_Channel is started successfully, as shown in the following image.

iWay Service Ma Server Registry Dep	Nager loyments Tools		Manageme	nt base		Restart	✓ Ø	Image: Weight of the second
Management Channels Services	Channels Manage Channels which have b Channel Management The channels listed below a	een deployed	I. Select a channel to undeploy, repair, s	tart, stop, o	r deploy a	new char	nnel from the re	pository.
Web Services	Channel Name Deploy Undeploy	Protocol file Redeploy	Deploy Date/By Jun 29 2015 04:20 PM / iway Repair Reset Statistics	Version 1 Start	Status ✓ Stop	Active	A-C-S-F 0 - 0 - 0 - 0	Description

Procedure: How to Test and Validate the Channel

1. Copy the input payload for the GetEffectiveAddress.xml file in the following configured input location:

C:/input

😋 🔵 🗢 👪 🕨 Computer 🕨 OS (C:) 🕨 in						
File Edit View Tools Help						
Organize 🕶 🍘 Open 💌 Burn Ne	ew fold	ier				
Secent Places	•	Name	Date modified	Туре	Size	
Computer		GetEffectiveAddress.xml	10/28/2014 1:18 PM	XML Document		1 KB
G OS (C:) di app di Config.Msi di CSDsoftware di E910_1 di E6420 di ibi di T di Intel di ivay7 di jok1.7.0_55	н					

2. Check for a response document from the *GetEffectiveAddress.xml* file in the following configured destination location:

C:/output

Computer & OS(C) & out				
File Edit View Tools Help				
Organize Include in library Share v	vith Burn New folder			
	^ Name	Date modified	Туре	Size
4 🧊 Libraries	GetEffectiveAddress.xml	6/29/2015 4:22 PM	XML Document	1 KB
Documents				
Music				
Pictures				
P 🔄 VIGEOS				
A I Computer				
Config.Msi				
🛛 🎍 CSDsoftware				
> 🕌 E910_1				
Þ 🍻 E6420				
Þ 🇼 ibi				
Þ 🕌 AT				
🕌 in				
Þ 🕌 Intel	_			
Þ 🏰 iway7	E-			
Million (1997) 100				
Misocacne				
PerfLogs				
- i integr				

3. Open and view the successful response document from the GetEffectiveAddress.xml file, as shown in the following image.





Configuring the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools Designer

After you successfully configure the adapter to represent a particular adapter target, the adapter can be used within a process flow.

In this appendix:

Using the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools Designer

Using the Application Adapter for J.D. Edwards EnterpriseOne in iWay Integration Tools Designer

You can make an adapter available to a process flow created in iWay Integration Tools (iIT) Designer, a GUI-based tool, used to build stateless process flows that execute within iWay Service Manager (iSM). The adapter can be incorporated as a node, called an Adapter object, in an iWay process flow, allowing you to integrate it easily into a business process solution.

Before you begin, ensure that you have created a project and created a process for that project. You can create a project by right-clicking the Processes folder in your project and selecting New Process from the context menu. For more information, see the *iWay Integration Tools (iIT) Designer User's Guide*.

Procedure: How to Configure the Adapter Using iWay Integration Tools Designer

1. Create and connect to a J.D. Edwards EnterpriseOne target, as described in *Working With a Target* on page 40.



2. Right-click the *JDEEnterpriseOne_910* node (adapter target) and select *Create Registry* from the context menu, as shown in the following image.



A confirmation message is displayed, as shown in the following image.



3. Click OK.

4. Right-click the *Registry* node and select *Connect*, as shown in the following image.



5. Expand Registry, Components, and then Adapters.

6. Right-click the *JDEEnterpriseOne_910* node (adapter target) and select *Export* from the context menu, as shown in the following image.



The Export Resource dialog box opens, as shown in the following image.

A Export Resource	I	- 🗆 🗡
Export Dependencies		
Project name must be specified.		
Project:		
		_
•	Finish Car	ncel

7. Click the ellipsis button to the right of the Project field.

The Export dialog box opens, as shown in the following image.



8. Select the Schema folder.

Note: In this example, the Integration project is being called Schema, which is why the Schema folder must be selected.

9. Click OK.

The Export Resource dialog box opens, as shown in the following image.

roject:	Test			
elow is a utomati nem by ote that	a tree of node(s) and its dependence cally (you cannot uncheck these no checking checkbox next to node yo : all component will be exported to	es you are exporting from the server des). Nodes that are not checked by u want re-export. their default folders in the project. D	Checked nodes do not exist in your project and th default, exist in your project. You can choose whet efault folders can be customized in Export Folders p	us will be exported her you want to export preference page.
Only s	how existing components (compo	ents that exist both in project and o	n the server)	
V J	DEEnterpriseOne_910	adapter		
Sele	ct All Deselect All			

- 10. Select JDEEnterpriseOne_910 and click Finish.
- 11. Expand the Registry node, Components, and then Schemas.

- hervices & Events Applications 🖻 🌆 Registry 🗄 😴 Conduits 🗄 📳 Components . ⊕ · **(**) Adapters Emitters E G Listeners E- C Reviewers 🗄 🎯 Schemas 👸 Any 颜 PFI 🖬 Export 🔗 PFI 🤗 pict 🔗 pict 🔆 Filter... 🤗 pict 🗄 🕝 Service 😴 New iWay Resource... E Outline 🖾 Go Home An outline is not available. Go Back Go Into
- 12. Right-click *Any* and select *Export*, as shown in the following image.

The Export Resource dialog box opens, as shown in the following image.

A Export Resource	_ 🗆 X
Export Dependencies	
Project name must be specified.	
Project:	
(?)	Finish Cancel

13. Click the ellipsis button to the right of the Project field.

The Export dialog box opens, as shown in the following image.



14. Select the Schema folder and then click OK.

The Export Resource dialog box opens, as shown in the following image.

र्ट्रे Export Resource		
Export Dependencies Select nodes that you want to exports fro	m the registry.	
Project: Schema		
Below is a tree of node(s) and its depende (you cannot uncheck these nodes). Node: next to node you want re-export. Note that all component will be exported to Only show existing components (comp	ncies you are exporting from the server. Check that are not checked by default, exist in your p o their default folders in the project. Default fol onents that exist both in project and on the ser	ed nodes do not exist in your project and thus will be exported automatically project. You can choose whether you want to export them by checking checkbox ders can be customized in <u>Export Folders</u> preference page.
Name	Туре	Description
Any	schema	A schema representing a definition for any document in an
Select All Deselect All	1	
?		Finish Cancel

15. Select Any and then click Finish.

16. From the Integration Explorer tab, right-click *Flows*, select *New*, and then click *Process Flow* from the context menu, as shown in the following image.

🔬 Integration	Explorer 🔀	🤨 iWay Explorer	🛋 Library Manager	
 Schema Ada App Cha Ebit Ebit<td>apters olications annels kes we</td><td></td><td>🕐 🎻 Integration Pr</td><td>oject</td>	apters olications annels kes we		🕐 🎻 Integration Pr	oject
	Go Into Open in New Window Paste Duplicate Duplicate Move Rename Import	Project Application Channel Process Flow Transform		
		Register Set Schema Set f() IFL Expression f() Example	n	
	🐑 Refresh		📬 Other	Ctrl+N

New Process Flow Wize	ard	- 22			
General Properties Please select a project location and choose a name for the new Process Flow					
Project Folder	/Test/Flows B	rowse			
Name Description	JDE_910	•			
Target Server Version	7.0.3-SNAPSHOT Install additional Target Server Version Create in current folder Enable taps	~			
?	Finish Can	cel			

The New Process Flow Wizard opens, as shown in the following image.

17. Enter a name for the new process flow (for example JDE_910) and click Finish.

18. In the right pane, right-click the *Start* object and select *Properties* from the context menu, as shown in the following image.

	9 T	멻	Save	·	
	Sta	%	Search	nterpriseOn En	d
		2	Create Object +		
		*	Create Object		
		<₽.	Publish		
		4	Publish to		
			Deploy		
		ដា	Compile		
		0	Test Run		
		4	Undo		
		\Leftrightarrow	Redo		
		of	Cut		
esign Te	st Resul		Сору		
Properti	es 🛙	1	Paste	roblems	
tart Ob	iect	x	Delete		
ture obj			Rename		
Seneral	Only o		Edit	n any one process.	
chema	Name		Activate		
ebug	Start		Deactivate		
	Descr		Export to Library		
	Start		Properties		
				1	

19. The Properties tab is displayed, as shown in the following image.

Properties	s 🔀 🥺 Error Log 🕻	🛛 Console 🙀 Problems 😁 🏹	
Start Obje	ect		
General	Schema	Name of the schema which describes the input document to this process flow	-
Schema		X View	
Debug	Root Tag Element	Name of the root tag element of the input document to this process flow. (* Required if a Schema is selected)	
		×	
	Description	Description for the selected schema	-

20. Select the Schema tab and then the ellipsis button to the right of the Schema field.

🦼 Select Schema	
🖃 🗁 Schema	•
🕀 🗁 Adapters	
Channels	
Ebixes	
E Flows	
🖻 🗁 Schemas	
Any S Any	
AP20.1 Basic Invoice_request.xsd	
AP20.1 Basic Invoice_response.xsd	
S PA52_1_Individual Action_request.xsd	
S PA52_1_Individual Action_response.xs	
Transforms	-1
(?) ОК С	ancel

The Select Schema dialog box opens, as shown in the following image.

- 21. Select the schema called Any and click OK.
- 22. From the Root Tag Element drop-down list, select *request*, as shown in the following image.

Properties	s 🔀 🤨 Error Log	🗄 Console 🙀 Problems 😁 😁	
Start Obje	ect		
General	Schema	Name of the schema which describes the input document to this process flow	-
Schema		Any X View	
Debug Root Tag Element		Name of the root tag element of the input document to this process flow. (* Required if a Schema is selected)	_
		request 💌	
	Description	Description for the selected schema	-
23. From the Integration Explorer tab, expand the *Adapters* folder, and drag and drop the *JDEEnterpriseOne_910* node (adapter target) to the right pane, as shown in the following image.

Integration - Test/Flows/JDE 910.iwp/JDE 910.iwp - Way Integratio		
File Edit View Navigate Search Project Run Window He		
P □ D. 0. 0. 50 50 m 1 th - O - O	·) 8 · 8 · 6 · 6 · 6 · 16 B = 10 B = 10 B = 10 B	21 .39 100% v
d Internation E		
megration E A vivay copiorer at clorary mana	GetenectiveAddress	tores_requestmin \$0 DE 310 10
		to Palette p
Adapters		l∂ Select
IDEEnterpriseOne_910		L Marquee
Applications		Relation
Channels		Loop
Constant Con		🛅 Basic Objects 🗠
to ExchangeTestUsingHTTP		End
រូប្មី JDE_910		A Text
Equisters	_	Control Objects
S Any	*	Decision Switch
GetEffectiveAddress_request.xsd	Start JDEEnterpriseOn	End & Decision Test
GetEffectiveAddress_response.xsd	e_910	Hu Join
Iransforms XMI		3 Junction
GetEffectiveAddress_request.xml		* Set
		W Synch
		Secution Obia. (0)
		WR. BI
		@ DOS
		Q. Email
		Emitter
		EL ETI
		ti) File
E Outline 🛙 🗖	esign Test Results Trace	
	Properties 22 9 Error Lon Console 12 Problems	
	rocess Flow	
	nfo	
	Connection Router	
	and a second	
	•	

24. From the Root Tag Element drop-down list, select *response* and click *Finish*, as shown in the following image.

🔬 New End Object		l ×
End Object Scher Select a schema to be a	ma assigned to the End Object	
Schemas available in th	he current server configuration are shown below. To view an existing schema click on the 'View Schema' buttr	on
Schema	Name of the schema which describes the output document to this process flow	
	Any X View	
Root Tag Element	Name of the root tag element of the output document to this process flow. (* Required if a Schema is selected)	
	request]
Terminate	request response	
Description	Description for the selected schema	
	A schema representing a definition for any document in and any document out. This schema has two possible root elements, appropriately named 'request' and 'response'.	- -
?	< Back Next > Finish Cancel	

25. From the palette, select *Relation* and create a relation between the Start object and the J.D. Edwards EnterpriseOne adapter target object.

The Relation Properties dialog box opens, as shown in the following image.

Arelation Properties	5			
Relation Configur Use this dialog to config	ation jure a relationship betwee	en two objects using st	ock or custom ev	
Event: OnCompletion				•
Case of:				÷ /* ×
Case	Туре	Description		
Description:	ays create relations using	g default event type.		
?			Finish	Cancel

26. From the Event drop-down list, select OnCompletion and click Finish.

27. Create a new relation between the J.D. Edwards EnterpriseOne adapter target object and the End object, as shown in the following image.



The Relation Properties dialog box opens, as shown in the following image.

A Relation Properties	5			
Relation Configura Use this dialog to config	ation ure a relationship betwee	n two objects using stock	or custom event	
Event: OnSuccess				•
Case of:			+ /	* ×
Case	Туре	Description		
Description:				
🔲 Don't ask again. Alw	ays create relations using	default event type.		
?			Finish Ca	ancel

- 28. From the Event drop-down list, select OnSuccess and click Finish.
- 29. Click Save.

30. From the Flows folder, right-click the *JDE_910* process flow, select *Run As*, and then click *Process Flow* from the context menu, as shown in the following image.



31. Copy the input request document for the GetEffectiveAddress function and paste the document in the Input Document area, as shown in the following image.

lit configura	ition and launch.			
lame: JDE_91	0			
Main Test Se	ver Registers Runtime Options			
Process Flow	vt			
/Test/Flow	s/JDE_910.iwp			Browse
Input Docur	nent:			
_				
				Import
Cide Deces	one wo encounty on the th			<u> </u>
< callMetho	st type="callmethod" user="JDE" pwd="Jl od name="GetEffectiveAddress" runOnErro	E" environment="DV910" session="" sess ="no">	sionidle="">	•
< callMetho	<pre>st type="cellmethod" user="JDE" pwd="JI wd name="GetEffectiveAddress" runOnErro pnment:</pre>	E" environment="DV910" session="" sess ="no">	sionidle="">	
<pre></pre> Server Envir URL:	st type="callmethod" user="JDE" pwd="JI wd name="GetEffectiveAddress" runOnErro pnment: http://localhost:9000	E" environment="DV910" session="" sess ="no">	sionidle="">	•
<pre></pre> Server Envir URL: User Name:	st type="callmethod" user="JDE" pwd="JI wd name="GetEffectiveAddress" runOnErro pnment: http://localhost:9000	E" environment="DV910" session="" sess ="no">	sionidle="">	
<pre></pre> callMetho Server Envir URL: User Name: Password:	st type="callmethod" user="JDE" pwd="JI wd name="GetEffectiveAddress" runOnErro pnment: http://localhost.9000	E" environment="DV910" session="" sess ="no">	sionidle="">	•
<pre><goexeque <callmetho="" envir="" mini<="" name:="" password:="" pre="" server="" url:="" use="" user=""></goexeque></pre>	nt type="callmethod" user="JDE" pwd="Jl d name="GetEffectiveAddress" runOnErro nment: http://localhost:9000 mal runtime configuration with dependen	e" environment="DV910" session="" sess ="no">	sionidle="">	•
 Server Envir URL: User Name: Password: Use mini Use test s 	nt type="callmethod" user="JDE" pwd="Jl d name="GetEffectiveAddress" runOnErro nment: http://localhost:9000 mal runtime configuration with dependenerver	e" environment="DV910" session="" sess ="no">	sionidle="">	•
 Server Enviro Server Enviro URL: User Name: Password: Use mini Use test s 	nt type="cellmethod" user="JDE" pwd="Jl d name="GetEffectiveAddress" runOnErro nment: http://localhost:9000 mal runtime configuration with dependen erver	e" environment="DV910" session="" sess ="no">	sionidle="">	•
<pre><gereque <callmetho Server Envir URL: User Name: Password: ① Use mini ④ Use test s</callmetho </gereque </pre>	tt type="callmethod" user="JDE" pwd="JI kd name="GetEffectiveAddress" runOnErro pnment: http://localhost9000 mal runtime configuration with dependen erver	E" environment="DV910" session="" sess ="no">	sionidle="">	• •

32. Click Apply and then click Run.

The response document for the GetEffectiveAddres function can be viewed by opening the End.xml file in the Output folder under the Test Results, as shown in the following image.



33. Right-click the *JDE_*910 process flow, select *Integration Tools* and then click *Publish* from the context menu, as shown in the following image.



The published process flow can now be associated with a channel route in the iWay Service Manager Administration Console.



Configuring EnterpriseOne for Outbound Transaction Processing

EnterpriseOne enables you to specify outbound functionality for Master Business Functions (MBF).

This section describes how to enable outbound transaction processing in EnterpriseOne and how to modify the jde.ini file for XML and XML List support.

In this appendix:

- Specifying Outbound Functionality for a Business Function
- Configuring an Event Listener for the iWay Application Adapter for J.D. Edwards EnterpriseOne
- XML List Method Support
- Modifying the EnterpriseOne jde.ini File

Specifying Outbound Functionality for a Business Function

You can specify outbound functionality for business functions and manage the flow of data. You enable outbound transaction processing using a processing option that controls how a transaction is written.

Outbound Transaction Processing

To process outbound data, you use the:

- Data Export Control table
- Processing Log table

The Data Export Control table manages the flow of the outbound data to third-party applications. The Processing Log table contains all the information about the EnterpriseOne event.

Procedure: How to Configure Outbound Transaction Processing

To configure outbound transaction processing:

- 1. Set an environment variable called JAVA_HOME on the J.D. Edwards Enterprise Server.
- 2. Set this to the location of your JDK installation, for example:

iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide

SET JAVA_HOME=D:\jdk1.3

3. Add the following to your PATH variable.

%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin;%JAVA_HOME%\jre\bin\classic;

- Add the installation location of the iwoevent.dll file to your PATH variable, for example: d:\mylocation
- 5. Add the Connector. jar and kernel. jar files to your CLASSPATH, for example: d:\b7\system\classes\Connector.jar; d:\b7\system\classes\kernel.jar

Note: You must add the required .JAR files as specified in the following table.

J.D. Edwards EnterpriseOne Version	Required .JAR Files
B733	Connector.jar
	kernel.jar
ERP8 (B7334)	Connector.jar
	kernel.jar
EnterpriseOne B9	Connector.jar
	kernel.jar
	jdeutil.jar
	log4.jar
EnterpriseOne 8.10	Connector.jar
	kernel.jar
	jdeutil.jar
	log4.jar

J.D. Edwards EnterpriseOne Version	Required .JAR Files
EnterpriseOne 8.11	Base_JAR.jar
	Connector.jar
	JdeNet_JAR.jar
	log4.jar
	System_JAR.jar
EnterpriseOne 8.12 (Tools	Base_JAR.jar
Release 8.96.2.0)	JdeNet_JAR.jar
	System_JAR.jar
	Connector.jar
	EventProcessor_EJB.jar
	EventProcessor_JAR.jar
	log4.jar
EnterpriseOne 8.12 (Tools	Base_JAR.jar
Release 8.97.1.2 and 8.97.2.0)	JdeNet_JAR.jar
	System_JAR.jar
	Connector.jar
	EventProcessor_EJB.jar
	EventProcessor_JAR.jar
	commons-httpclient-3.0.jar
	jmxri.jar
	ManagementAgent_JAR.jar
	log4.jar

^{6.} Set a system variable called IWOEVENT_HOME on the J.D. Edwards Enterprise Server and set this location to a folder, for example, Outbound, where iwoevent.dll and iwoevent.cfg are located. For more information, see *How to Create a System Variable on Windows* on page 156.

For more information on configuring EnterpriseOne for outbound processing, see Detailed Tasks for EnterpriseOne Operations in the J.D. Edwards Interoperability Guide for EnterpriseOne.

Procedure: How to Create a System Variable on Windows

To create a new system variable:

1. Click Start, select Settings, and click Control Panel.

The Control Panel window opens as shown in the following image.



2. Double-click the System icon.

The System Properties window opens as shown in the following image.

System Properties
General Network Identification Hardware User Profiles Advanced
Performance Performance options control how applications use memory, which affects the speed of your computer.
Performance Options
Environment Variables
Environment variables tell your computer where to find certain types of information.
Environment Variables
Startup and Recovery
Startup and recovery options tell your computer how to start and what to do if an error causes your computer to stop.
Startup and Recovery
OK Cancel Apply

3. Click the Advanced tab and click Environment Variables.

variable	Value
CLASSPATH	C:\Program Files\Altova\xmlspy\XMLSpy
	New Edit Delete
ystem variables -	
ystem variables - Variable	Value
ystem variables - Variable ANT_HOME	Value C:\Program Files\Ant\;%ANT_HOME%\bin C:\WINIT\custom22\cmd ove
ystem variables - Variable ANT_HOME ComSpec TWAYS1	Value C:\Program Files\Ant\;%ANT_HOME%\bin C:\WINNT\system32\cmd.exe C:\Program Files\WAV51
ystem variables - Variable ANT_HOME ComSpec IWAY51 IWAY50	Value C:\Program Files\Ant\;%ANT_HOME%\bin C:\WINNT\system32\cmd.exe C:\Program Files\Way51 C:\iway60t
ystem variables - Variable ANT_HOME ComSpec IWAY51 IWAY60 JAVA_HOME	Value C:\Program Files\Ant\;%ANT_HOME%\bin C:\WINNT\system32\cmd.exe C:\Program Files\iWay51 C:\iway60t C:\j2sdk1.6.0_13

The Environment Variables dialog box opens as shown in the following image.

4. Click *New* in the System variables section.

The New System Variable dialog box opens as shown in the following image.

New System Varia	ble	<u>? ×</u>
Variable Name:	IWOEVENT_HOME	
Variable Value:	d:\B7333\JDEdwardsOneWorld\Out	bound
	OK Car	ncel

- 5. Perform the following steps:
 - a. Type a name for the system variable in the Variable Name field.
 - b. Type a valid path for the system variable in the Variable Value field.

c. Click OK.

Procedure: How to Create a System Variable on UNIX

To create a new system variable:

1. You must define and set the following variable to the location of your JDK installation: JAVA_HOME

For example:

JAVA_HOME=D:/jdk1.3/ export JAVA_HOME

Note: Be sure to include the trailing slash.

2. The PATH variable must contain the JDK bin directory, for example:

```
PATH=$PATH:/D:/jdk1.3/bin/
export PATH
```

Procedure: How to Enable Outbound Transaction Processing

To enable outbound transaction processing:

1. Right-click the application that contains the processing options for the Master Business Functions of the transaction.

For a list of these options, see Appendix B of the J.D. Edwards Interoperability Guide.

- 2. From the shortcut menu, select Prompt for Values.
- 3. Click either the Outbound tab or the Interop tab.
- 4. Enter the transaction type.

The EnterpriseOne Event listener processes only the after image for the business function.

You are not required to set the before image function.

The Data Export Control Table and the Processing Log Table

The Data Export Control table manages the flow of the outbound data to third-party applications. EnterpriseOne allows for the subscription of multiple vendor-specific objects for an interoperability transaction.

The records in the Data Export Control table are used to determine the vendor-specific objects to call from the Outbound Subsystem batch process (R00460) or the Outbound Scheduler batch process (R00461).

The Processing Log table contains all the information about the EnterpriseOne event including the transaction type, order type, and sequence number from the Data Export Control table.

Procedure: How to Use the Data Export Controls

To use the data export controls:

- 1. On the Work With Data Export Controls pane, click *Add*.
- 2. Type values in the Transaction Type and Order Type fields.
- 3. For each detail row, enter either a batch process name or version or a function name and the library.
- 4. To launch the vendor-specific object for an add or insert, type 1.
- 5. For the update, delete, and inquiry actions, type 1.
- 6. In the Launch Immediately column, type 1.
- 7. Click OK.

Configuring an Event Listener for the iWay Application Adapter for J.D. Edwards EnterpriseOne

This section describes how to install and configure an event listener for the iWay Application Adapter for J.D. Edwards EnterpriseOne on an AS/400 platform. Outbound (event) processing on AS/400 requires the configuration of an Event Stub and a configuration file (iwoevent.cfg).

Configuring the iwoevent.cfg File

The iwoevent.cfg file contains connectivity information that is read by the Event Stub to communicate with J.D. Edwards EnterpriseOne.

Procedure: How to Configure the iwoevent.cfg File

To configure the iwoevent.cfg file:

1. Using the CRTDIR command, create a directory on AS/400 that is accessible by the J.D. Edwards EnterpriseOne application server. For example:

CRTDIR DIR('/e810sys/outbound') DTAAUT(*RW)

- Copy the iwoevent.cfg configuration file to the directory you just created, for example: /e810sys/outbound
- 3. Create a directory using the alias name that is specified in the iwoevent.cfg file.
- 4. Add an environment variable called IWOEVENT_HOME for the J.D. Edwards EnterpriseOne application server.

The value should be a full path to the directory that you specified in Step 1, for example:

ADDENVVAR ENVVAR(IWOEVENT_HOME) VALUE('/e810sys/outbound') LEVEL(*SYS)

If the iwoevent.cfg configuration file has tracing enabled (trace=on), the iwoevent.log trace file is created in the IWOEVENT_HOME directory.

Reference: Sample iwoevent.cfg Configuration File

You can use the following sample iwoevent.cfg configuration file for reference purposes:

```
common.trace=on
alias.Gopi=172.30.244.136:1234,trace=on
trans.JDEAB=Gopi
```

Configuring the Event Stub

The Event Stub for the iWay Application Adapter for J.D. Edwards EnterpriseOne must be installed on the AS/400 platform by an administrator.

Procedure: How to Configure the Event Stub

To configure the Event Stub:

1. In the iSeries green console, use the CRTLIB command to create a temporary library. For example:

CRTLIB IWAYTEMP

Use the CRTSAV command to create an online save file in the library you just created, for example:

CRTSAVF IWAYTEMP/IWAYSAV

3. Using FTP, upload the iwaysav.sav file to your iSeries system. For example:

```
FTP YourSystemNameLoginBIN
PUT IWAYSAV.SAV IWAYTEMP/IWAYSAV
```

where:

YourSystemNameLogin

Is the name of your iSeries system.

4. In the iSeries green console, enter the following command:

RSTLIB SAVLIB(IWAYPLUGIN) DEV(*SAVF) SAVF(IWAYTEMP/IWAYSAV)

A new library named IWAYPLUGIN is created, which contains one object named EVENTPLUG.

Note: Using the POO47 (Work With Data Export Controls) application, you must specify the Function Library as IWAYPLUGIN/EVENTPLUG.

XML List Method Support

The iWay Application Adapter for J.D. Edwards EnterpriseOne uses the XML List method to collect a list of records from EnterpriseOne. XML List is built on the EnterpriseOne Table Conversion (TC) engine. It takes an XML document as a request and returns an XML document containing data. A list can represent data in a table, a business view, or data from a table conversion. Using data from a table conversion allows you to use multiple tables.

You can send the request through JDENET to perform any of the following operations:

- Create List
- Get Template
- Get Group
- Delete List

The following diagram illustrates the enterprise server side architecture of the XML List kernel.



For more information on the Create List, Get Template, Get Group, and Delete List operations, refer to the *J.D. Edwards Interoperability Guide*.

List Retrieval Engine Table Conversion Wrapper

The List Retrieval Engine is an optimized database engine that provides and manages access to XML repository files. Each XML list repository file is a pair of index and data files with *.idb and *.ddb extensions. The IDB file keeps an index generated on a data file, and the DDB file keeps data generated by the table conversion engine. The Table Conversion Wrapper is a system module that aggregates list-retrieval and list-processing APIs from the Table Conversion Engine and the List Retrieval Engine, and provides a uniform access to it for XML List.

To keep and manage repository files, the List Retrieval Engine uses a predefined folder as its system directory. This system directory must be configured in the jde.ini file as described in *Modifying the jde.ini File for the List Retrieval Engine*.

Modifying the EnterpriseOne jde.ini File

Because the iWay Application Adapter for J.D. Edwards EnterpriseOne uses XML for the transfer of information to and from EnterpriseOne, you must configure the EnterpriseOne environment to support XML. You can do this by modifying the EnterpriseOne jde.ini file.

To enable support for the XML List method, you must modify the EnterpriseOne jde.ini file accordingly, as described in *Modifying the jde.ini File for XML List Support*.

Example: Modifying the jde.ini File for XML Support

The following is a sample of the modifications required to implement XML support on Windows NT. Add the following blocks of code:

```
[JDENET_KERNEL_DEF6]
;krnlName=CALL OBJECT KERNEL
;dispatchDLLName=jdekrnl.dll
;dispatchDLLFunction=_JDEK_DispatchCallObjectMessage@28
;maxNumberOfProcesses=10
;numberOfAutoStartProcesses=0
krnlName=CALL OBJECT KERNEL
dispatchDLLName=XMLCallObj.dll
dispatchDLLFunction=_XMLCallObjectDispatch@28
maxNumberOfProcesses=10
numberOfAutoStartProcesses=0
```

```
[JDENET_KERNEL_DEF15]
krnlName=XML TRANSACTION KERNEL
dispatchDLLName=XMLTransactions.dll
dispatchDLLFunction=_XMLTransactionDispatch@28
maxNumberOfProcesses=1
numberOfAutoStartProcesses=1
```

Note: The @28 and the underscore character "_" in the example above are for Windows NT only.

For more information on establishing your EnterpriseOne environment for XML support, see Setting the jde.ini File for XML in the J.D. Edwards Interoperability Guide.

Example: Modifying the jde.ini File for XML List Support

The following is a sample of the modifications required to implement XML List support on Windows NT. Add the following blocks of code:

```
[JDENET_KERNEL_DEF16]
krnlName=XML List
dispatchDLLName=xmllist.dll
dispatchDLLFunction=_XMLListDispatch@28
maxNumberOfProcesses=3
beginningMsgTypeRange=5257
endingMsgTypeRange=5512
newProcessThresholdRequest=0
numberOfAutoStartProcesses=3
```

Note: The @28 and the underscore character "_" in the example above are for Windows NT only.

Reference: DLL Extensions for Other Platforms

The following table lists different DLL extensions for other platforms.

	XML List dispatchDLLName=	Call Object dispatchDLLName=	XML Trans dispatchDLLName=
AS400	XMLLIST	XMLCALLOBJ	XMLTRANS
HP9000B	libxmllist.sl	libxmlcallobj.sl	libxmltransactions.sl
SUN or RS6000	libxmllist.so	libxmlcallobj.so	libxmltransactions.so

Example: Modifying the jde.ini File for the List Retrieval Engine

To keep and manage repository files, the List Retrieval Engine uses a predefined folder as its system directory. This system directory should be configured in the jde.ini file as follows:

```
[LREngine]
System=C:\output
Repository_Size=20
Disk_Monitor=No
```



Sample J.D. Edwards EnterpriseOne Files

The iWay Application Adapter for J.D. Edwards EnterpriseOne supports the jdeRequest and jdeResponse XML structures for executing business functions within EnterpriseOne. Using EnterpriseOne XML, you can:

- □ Aggregate business function calls into a single object.
- Use the EnterpriseOne ThinNet API.
- □ Access both Z files and business functions.

This section provides examples of the jdeRequest and jdeResponse XML structures for executing business functions within EnterpriseOne.

In this appendix:

- Issuing a Single-Function Request
- Issuing a Multiple-Function Request
- Sample Sales Order Request
- □ Sample Sales Order Response

Issuing a Single-Function Request

The following example, GetEffectiveAddress, is a single-function call to EnterpriseOne, and the result of this request is a standard jdeResponse document. In a single-function request, only one callMethod within the XML object is specified.

Example: Executing a Business Function With a Single-Function Call

The following code is a sample GetEffectiveAddress jdeRequest.

```
<jdeRequest type="callmethod" user="JDE" pwd="JDE" environment="DV7333"</pre>
session="">
 <callMethod name="GetEffectiveAddress" app="BSE" runOnError="no">
 <params>
                       <param name="mnAddressNumber">1001</param>
                       <param name="jdDateBeginningEffective"></param>
                       <param name="cEffectiveDateExistence10"></param>
                       <param name="szAddressLine1"></param></param>
                       <param name="szAddressLine2"></param></param>
                       <param name="szAddressLine3"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="szAddressLine4"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="szZipCodePostal"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="szCity"></param>
                       <param name="szCountyAddress"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="szState"></param>
                       <param name="szCountry"></param>
                       <param name="szUserid"></param></param>
                       <param name="szProgramid"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="jdDateupdated"></param></param>
                       <param name="szWorkstationid"></param></param>
                       <param name="mnTimelastupdated"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
                       <param name="szNamealpha"></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param></param>
 </params>
<onError abort="yes"></onError>
 </callMethod>
</jdeRequest>
```

The following code is a sample GetEffectiveAddress jdeResponse.

```
<?xml version="1.0"?>
<!DOCTYPE jdeResponse>
<jdeResponse environment="DV7333"
            pwd="JDE"
             session="516.1029417972.68"
             type="callmethod"
             user="JDE">
  <callMethod app="BSE"
             name="GetEffectiveAddress"
             runOnError="no">
    <returnCode code="0"/>
    <params>
      <param name="mnAddressNumber">1001</param>
      <param name="jdDateBeginningEffective"/>
      <param name="cEffectiveDateExistence10"/>
      <param name="szAddressLine1">8055 Tufts Avenue, Suite 1331</param>
      <param name="szAddressLine2">
                                                                </param>
      <param name="szAddressLine3">
                                                                </param>
      <param name="szAddressLine4">
                                                                </param>
      <param name="szZipCodePostal">80237 </param>
      <param name="szCity">Denver </param>
      <param name="szCountyAddress">
                                        </param>
      <param name="szState">CO
                                         </param>
      <param name="szCountry"/>
      <param name="szUserid"/>
      <param name="szProgramid"/>
      <param name="jdDateupdated"/>
      <param name="szWorkstationid"/>
      <param name="mnTimelastupdated">0
                                        </param>
      <param name="szNamealpha">J.D. Edwards & Company
                                                               </param>
    </params>
  </callMethod>
</jdeResponse>
```

Issuing a Multiple-Function Request

The following example, GetEffectiveAddress, is a multiple-function call to EnterpriseOne, and the result of this request is a standard jdeResponse document with multiple sections. In a multiple-function request, more than one callMethod within the XML object is specified.

Example: Executing a Business Function With a Multiple-Function Call

The following code is a sample Purchase Order in the jdeRequest format. The XML contains return parameter specifications as well as file cleanup logic.

```
<?xml version='1.0' encoding='utf-8' ?>
<ideRequest pwd='password' type='callmethod' user='user' session=''</pre>
environment='DV7333' sessionidle=''>
   <callMethod app='XMLTest' name='GetLocalComputerId'
      runOnError='no'>
   <params>
      <param name='szMachineKey' id='machineKey'></param>
   </params>
   <onError abort='yes'>
   </onError>
   </callMethod>
   <callMethod app='XMLTest' name='F4311InitializeCaching'
      runOnError='no'>
   <params>
      <param name='cUseWorkFiles'>2</param>
   </params>
   </callMethod>
   <callMethod app='XMLTest' name='F4311FSBeginDoc' runOnError='no'</pre>
      returnNullData='yes'>
   <params>
      <param name='mnJobNumber' id='jobNumber'></param>
      <param name='szComputerID' idref='machineKey'></param>
      <param name='cHeaderActionCode'>A</param>
      <param name='cProcessEdits'>1</param>
      <param name='cUpdateOrWriteToWorkFile'>2</param>
      <param name='cRecordWrittenToWorkFile'>0</param>
      <param name='szOrderCOmpany' id='orderCompany'>00200</param>
      <param name='szOrderType'>OP</param>
      <param name='szOrderSuffix'>000</param>
      <param name='szBranchPlant'>M30</param>
      <param name='mnSupplierNumber'</pre>
         id='supplierNumber'>4343</param>
      <param name='mnShipToNumber'>0.0</param>
      <param name='jdOrderDate'>2000/03/02</param>
      <param name='cEvaluatedReceiptsFlag'>N</param>
      <param name='cCurrencyMode'>D</param>
      <param name='szTransactionCurrencyCode'>USD</param>
```

```
<param name='mnCurrencyExchangeRate'>0.0</param>
                  <param name='szOrderedPlacedBy'>SUBSTITUTE</param>
                  <param name='szProgramID'>EP4310</param>
                  <param name='szPurchaseOrderPrOptVersion'</pre>
                     id='Version'>ZJDE0001</param>
                  <param name='szUserID'>SUBSTITUTE</param>
                  <param name='mnProcessID' id='processID'></param>
                  <param name='mnTransactionID' id='transactionID'></param>
               </params>
               <onError abort='yes'>
               <callMethod app='XMLTest' name='F4311ClearWorkFiles'
                  runOnError='yes' returnNullData='yes'>
               <params>
                  <param name='szComputerID' idref='jobNumber'></param>
                  <param name='mnJobNumber' idref='machineKey'></param>
                  <param name='cClearHeaderFile'>1</param>
                  <param name='cClearDetailFile'>1</param>
                  <param name='mnLineNumber'>0</param>
                  <param name='cUseWorkFiles'>2</param>
                  <param name='mnProcessID' idref='processID'></param>
                  <param name='mnTransactionID' idref='transactionID'></param>
               </params>
               </callMethod>
               </onError>
               </callMethod>
               <!-- This is the first EditLine entry -->
               <callMethod app='XMLTest' name='F4311EditLine' runOnError='yes'</pre>
                  returnNullData='no'>
               <params>
                  <param name='mnJobNumber' idref='jobNumber'></param>
                  <param name='szComputerID' idref='machineKey'></param>
                  <param name='cDetailActionCode'>A</param>
                  <param name='cProcessEdits'>1</param>
                  <param name='cUpdateOrWriteWorkFile'>2</param>
                  <param name='cCurrencyProcessingFlag'>Y</param>
                  <param name='szPurchaseOrderPrOptVersion'</pre>
                  idref='version'></param>
                  <param name='szOrderCompany' idref='orderCompany'></param>
                  <param name='szOrderType'>OP</param>
                  <param name='szOrderSuffix'>000</param>
                  <param name='szBranchPlant'>M30</param>
                  <param name='mnSupplierNumber'</pre>
                     idref='supplierNumber'></param>
                  <param name='mnShipToNumber'>0.0</param>
                  <param name='jdRequestedDate'>2000/03/02</param>
                  <param name='jdTransactionDate'>2000/03/02</param>
                  <param name='jdPromisedDate'>2000/03/02</param>
                  <param name='jdGLDate'>2000/03/02</param>
                  <param name='szUnformattedItemNumber'>1001</param>
                  <param name='mnQuantityOrdered'>1</param>
                  <param name='szDetailLineBranchPlant'>M30</param>
                  <param name='szLastStatus'>220</param>
                  <param name='szNextStatus'>230</param>
                  <param name='cEvaluatedReceipts'>N</param>
                  <param name='szTransactionCurrencyCode'>USD</param>
               </callMethod>
               <!-- This is the second EditLine entry -->
               <callMethod app='XMLTest' name='F4311EditLine' runOnError='yes'
iWay Application Adapter for MD1 Edwards EnterpriseOne User's Guide
               <params>
                  <param name='mnJobNumber' idref='jobNumber'></param>
                  <param name='szComputerID' idref='machineKey'></param>
                  <param name='cDetailActionCode'>A</param>
                   maram name-laprogagaEdital>1</nam
```

```
169
```

The following code shows the Purchase Order response document, which contains individual return codes for each callMethod executed. In addition, this method returns the order number assigned to the Purchase Order.

```
<?xml version="1.0" encoding="utf-8" ?>
<jdeResponse environment="DV7333" user="JDE" type="callmethod"</pre>
sessionidle="" session="2612.1026498135.5" pwd="JDE">
   <callMethod name="GetLocalComputerId" runOnError="no"
     app="XMLTest">
   <returnCode code="0"/>
   <params>
     <param name="szMachineKey" id="machineKey">XEENT</param>
   </params>
   </callMethod>
   <callMethod name="F4311InitializeCaching" runOnError="no"
     app="XMLTest">
   <returnCode code="0"/>
   <params>
      <param name="cUseWorkFiles">2</param>
   </params>
   </callMethod>
   <callMethod name="F4311FSBeginDoc" returnNullData="yes"
      runOnError="no" app="XMLTest">
   <returnCode code="0"/>
   <params>
      <param name="mnJobNumber" id="jobNumber">3</param>
      <param name="szComputerID" idref="machineKey">XEENT</param>
      <param name="cHeaderActionCode">1</param>
      <param name="cProcessEdits">1</param>
      <param name="cUpdateOrWriteToWorkFile">2</param>
      <param name="cRecordWrittenToWorkFile">1</param>
      <param name="cCurrencyProcessingFlag">Z</param>
      <param name="szOrderCOmpany" id="orderCompany">00200</param>
      <param name="mnOrderNumber">0</param>
```

```
<param name="szOrderType">OP</param>
                  <param name="szOrderSuffix">000</param>
                  <param name="szBranchPlant">M30</param>
                  <param name="szOriginalOrderCompany"/>
                  <param name="szOriginalOrderNumber"/>
                  <param name="szOriginalOrderType"/>
                  <param name="szRelatedOrderCompany"/>
                  <param name="szRelatedOrderNumber"/>
                  <param name="szRelatedOrderType"/>
                  <param name="mnSupplierNumber"</pre>
                     id="supplierNumber">17000</param>
                   <param name="mnShipToNumber">6074</param>
                   <param name="jdRequestedDate">2002/07/12</param>
                   <param name="jdOrderDate">2000/03/02</param>
                   <param name="jdPromisedDate">2002/07/12</param>
                   <param name="jdCancelDate"/>
                   <param name="szReference01"/>
                   <param name="szReference02"/>
                   <param name="szDeliveryInstructions01">
               </param>
                     <param name="szDeliveryInstructions02">
               </param>
                   <param name="szPrintMessage"/>
                   <param name="szSupplierPriceGroup"/>
                   <param name="szPaymentTerms"/>
                   <param name="szTaxExplanationCode"/>
                   <param name="szTaxRateArea"/>
                   <param name="szTaxCertificate">
                                                                        </param>
                   <param name="cAssociatedText"/>
                   <param name="szHoldCode"/>
                   <param name="szFreightHandlingCode"/>
                   <param name="mnBuyerNumber">0</param>
                   <param name="mnCarrierNumber">0</param>
                   <param name="cEvaluatedReceiptsFlag">N</param>
                   <param name="cSendMethod"/>
                   <param name="szLandedCostRule">
                                                      </param>
                   <param name="szApprovalRouteCode"/>
                   <param name="mnChangeOrderNumber">0</param>
                   <param name="cCurrencyMode">D</param>
                   <param name="szTransactionCurrencyCode">USD</param>
                   <param name="mnCurrencyExchangeRate">0</param>
                   <param name="szOrderedPlacedBy">SUBSTITUTE</param>
                   <param name="szOrderTakenBy"/>
                   <param name="szProgramID">EP4310</param>
                   <param name="szApprovalRoutePO"/>
                   <param name="szPurchaseOrderPrOptVersion"</pre>
                      id="Version">ZJDE0001</param>
                   <param name="szBaseCurrencyCode">USD</param>
                   <param name="szUserID">SUBSTITUTE</param>
                   <param name="cAddNewLineToExistingOrder"/>
                   <param name="idInternalVariables">0</param>
                   <param name="cSourceOfData"/>
                   <param name="mnSODOrderNumber">0</param>
                   <param name="szSODOrderType"/>
                   <param name="szSODOrderCompany"/>
                   <param name="szSODOrderSuffix"/>
                   <param name="mnRetainage">0</param>
                   <param name="szDescription"/>
iWay Application Adapter for J. Datedwards EnterpriseOne User's Guide
                  <param name="jdEffectiveDate"/>
                  <param name="jdPhysicalCompletionDate"/>
                  <param name="mnTriangulationRateFromCurrenc">0</param>
                  <param name="mnTriangulationRateToCurrency">0</param>
                   name = " a Curren au Convergion Method " /
```

171

Sample Sales Order Request

The following is a sample Sales Order request.

Example: Executing a Sales Order Request

The following code is an example of a Sales Order request.

```
<?xml version='1.0' encoding='utf-8' ?>
<jdeRequest type='callmethod' user='JDE' pwd='JDE' environment='DV7333'>
   <callMethod name='GetLocalComputerId' app='XMLInterop'
        runOnError='no'>
   <params>
    <param name='szMachineKey' id='2'></param>
   </params>
   <onError abort='yes'>
   </onError>
   </callMethod>
   <callMethod name='F4211FSBeginDoc' app='XMLInterop'
        runOnError='no'>
   <params>
    <param name='mnCMJobNumber' id='1'></param>
     <param name='cCMDocAction'>A</param>
     <param name='cCMProcessEdits'>1</param>
     <param name='szCMComputerID' idref='2'></param>
    <param name='cCMUpdateWriteToWF'>2</param>
    <param name='szCMProgramID'>XMLInterop</param>
    <param name='szCMVersion'>ZJDE0001</param>
    <param name='szOrderType'>SO</param>
    <param name='szBusinessUnit'>M30</param>
    <param name='mnAddressNumber'>4242</param>
     <param name='jdOrderDate'>2000/03/29</param>
     <param name='szReference'>10261</param>
     <param name='cApplyFreightYN'>Y</param>
     <param name='szCurrencyCode'></param>
     <param name='cWKSourceOfData'></param>
     <param name='cWKProcMode'></param>
     <param name='mnWKSuppressProcess'>0</param>
   </params>
```

```
<onError abort='yes'>
               <callMethod name='F4211ClearWorkFile' app='XMLInterop'
                  runOnError='ves'>
               <params>
            <param name='mnJobNo' idref='1'></param>
            <param name='szComputerID' idref='2'></param>
            <param name='mnFromLineNo'>0</param>
            <param name='mnThruLineNo'>0</param>
            <param name='cClearHeaderWF'>2</param>
            <param name='cClearDetailWF'>2</param>
            <param name='szProgramID'>XMLInterop</param>
            <param name='szCMVersion'>ZJDE0001</param>
               </params>
               </callMethod>
               </onError>
               </callMethod>
               <callMethod name='F4211FSEditLine' app='XMLInterop'
                  runOnError='yes'>
               <params>
                 <param name='mnCMJobNo' idref='1'></param>
                 <param name='cCMLineAction'>A</param>
                 <param name='cCMProcessEdits'>1</param>
                 <param name='cCMWriteToWFFlag'>2</param>
                 <param name='szCMComputerID' idref='2'></param>
            <!-- param name='mnLineNo'>10261</param -->
                 <param name='szItemNo'>1001</param>
                 <param name='mnQtyOrdered'>1</param>
                 <param name='cSalesTaxableYN'>N</param>
                 <param name='szTransactionUOM'>EA</param>
                 <param name='szCMProgramID'>XMLInterop</param>
                 <param name='szCMVersion'>ZJDE0001</param>
                 <param name='cWKSourceOfData'></param>
               </params>
               <onError abort='no'>
               </onError>
               </callMethod>
               <callMethod name='F4211FSEditLine' app='XMLInterop'</pre>
                  runOnError='yes'>
               <params>
                 <param name='mnCMJobNo' idref='1'></param>
                 <param name='cCMLineAction'>A</param>
                 <param name='cCMProcessEdits'>1</param>
                 <param name='cCMWriteToWFFlag'>2</param>
                 <param name='szCMComputerID' idref='2'></param>
            <!-- param name='mnLineNo'>10262</param -->
                 <param name='szItemNo'>1001</param>
                 <param name='mnQtyOrdered'>10</param>
                 <param name='cSalesTaxableYN'>N</param>
                 <param name='szTransactionUOM'>EA</param>
                 <param name='szCMProgramID'>XMLInterop</param>
                 <param name='szCMVersion'>ZJDE0001</param>
                 <param name='cWKSourceOfData'></param>
               </params>
               <onError abort='no'>
               </onError>
               </callMethod>
               <callMethod name='F4211FSEndDoc' app='XMLInterop'
                  runOnError='no'>
iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide
                 <param name='mnCMJobNo' idref='1'></param>
                 <param name='szCMComputerID' idref='2'></param>
                 <param name='szCMProgramID'>XMLInterop</param>
                 <param name='szCMVersion'>ZJDE0001</param>
```

arom name - I a CMU a o Work Eil a a L > 2 < / na

Sample Sales Order Response

This is the corresponding response document for the Sales Order request. There are error messages returned in the document. The error messages can be used within a workflow. The following shows sample error codes:

```
<prvor code="2597">Warning: WARNING: Duplicate Customer Order Number
</prvor>
<error code="4136">Warning: Pick date is less than todays date
</prvor>
```

Example: Using the Sales Order Response

The following code is the jdeResponse document.

```
<?xml version="1.0" encoding="utf-8" ?>
<jdeResponse environment="DV7333" user="JDE" type="callmethod" pwd="JDE">
  <callMethod name="GetLocalComputerId" runOnError="no"
     app="XMLInterop">
  <returnCode code="0"/>
  <params>
      <param name="szMachineKey" id="2">XEENT</param>
  </params>
  </callMethod><callMethod name="F4211FSBeginDoc" runOnError="no"
     app="XMLInterop">
  <returnCode code="1"/>
  <params>
     <param name="mnCMJobNumber" id="1">3</param>
     <param name="cCMDocAction">A</param>
    <param name="cCMProcessEdits">1</param>
    <param name="szCMComputerID" idref="2">XEENT</param>
     <param name="cCMErrorConditions">1</param>
     <param name="cCMUpdateWriteToWF">2</param>
     <param name="szCMProgramID">XMLInterop</param>
     <param name="szCMVersion">ZJDE0001</param>
     <param name="szOrderCo">00200</param>
     <param name="szOrderType">SO</param>
     <param name="szBusinessUnit">M30</param>
     <param name="mnAddressNumber">4242</param>
     <param name="mnShipToNo">4242</param>
     <param name="jdRequestedDate">2000/03/29</param>
     <param name="jdOrderDate">2000/03/29</param>
     <param name="jdPromisedDate">2000/03/29</param>
     <param name="szReference">10261</param>
     <param name="szDeliveryInstructions1">
                                                         </param>
     <param name="szDeliveryInstructions2">
                                                         </param>
     <param name="szPrintMesg">
                                       </param>
     <param name="szPaymentTerm"> </param>
     <param name="cPaymentInstrument"> </param>
     <param name="mnTradeDiscount">,000</param>
     <param name="szTaxExplanationCode">S </param>
```

```
<param name="szTaxArea">DEN
                                                      </param>
                 <param name="szCertificate">
                                                      </param>
                 <param name="szHoldOrdersCode"> </param>
                 <param name="cPricePickListYN">Y</param>
                 <param name="szRouteCode"> </param>
                 <param name="szStopCode">
                                             </param>
                 <param name="szZoneNumber"> </param>
                 <param name="szFreightHandlingCode"> </param>
                 <param name="cApplyFreightYN">Y</param>
                 <param name="mnCommissionCode1">6001</param>
                 <param name="mnCommissionRate1">5,000</param>
                 <param name="mnCommissionRate2">,000</param>
                 <param name="szWeightDisplayUOM"> </param>
                 <param name="szVolumeDisplayUOM"> </param>
                 <param name="cMode">D</param>
                 <param name="szCurrencyCode">USD</param>
                 <param name="jdDateUpdated">2002/07/12</param>
                 <param name="szWKBaseCurrency">USD</param>
                 <param name="cWKAdvancedPricingYN">N</param>
                 <param name="szWKCreditMesg"> </param>
                 <param name="szWKTempCreditMesg"> </param>
                 <param name="cWKSourceOfData"/>
                 <param name="cWKProcMode"/>
                 <param name="mnWKSuppressProcess">0</param>
                 <param name="szPricingGroup">PREFER </param>
                 <param name="mnProcessID">2252</param>
                 <param name="mnTransactionID">4</param>
               </params><errors><error code="2597">Warning: WARNING: Duplicate
                  Customer Order Number</error><error code="4136">Warning: Pick
                 date is less than todays date</error></errors>
               </callMethod><callMethod name="F4211FSEditLine" runOnError="yes"</pre>
                  app="XMLInterop">
            <returnCode code="1"/><params>
                 <param name="mnCMJobNo" idref="1">3</param>
                 <param name="cCMLineAction">A</param>
                 <param name="cCMProcessEdits">1</param>
                 <param name="cCMWriteToWFFlag">2</param>
                 <param name="cCMRecdWrittenToWF">1</param>
                 <param name="szCMComputerID" idref="2">XEENT</param>
                 <param name="cCMErrorConditions">1</param>
                 <param name="szOrderCo">00200</param>
                 <param name="szOrderType">SO</param>
                 <param name="szBusinessUnit">M30</param>
                 <param name="mnShipToNo">4242</param>
                 <param name="jdRequestedDate">2000/03/29</param>
                 <param name="jdPromisedDate">2000/03/29</param>
                 <param name="jdPromisedDlvryDate">2000/03/29</param>
                 <param name="szItemNo">1001
                                                              </param>
                 <param name="szLocation"> .
                                               . </param>
                 <param name="szDescription1">Bike Rack Trunk Mount </param>
                 <param name="szDescription2">
                                                                     </param>
                 <param name="szLineType">S</param>
                 <param name="szLastStatus">900</param>
                 <param name="szNextStatus">540</param>
                 <param name="mnQtyOrdered">1</param>
                 <param name="mnQtyBackordered">1</param>
                 <param name="mnUnitPrice">44,99</param>
                 <param name="mnUnitCost">32,1000</param>
iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guiden>
                 <param name="cPaymentInstrument"> </param>
                 <param name="cSalesTaxableYN">N</param>
                 <param name="cAssociatedText"> </param>
                 <param name="szTransactionUOM">EA</param>
                  area nome-larpriginguoMINEA (norma
```

Feedback

Customer success is our top priority. Connect with us today!

Information Builders Technical Content Management team is comprised of many talented individuals who work together to design and deliver quality technical documentation products. Your feedback supports our ongoing efforts!

You can also preview new innovations to get an early look at new content products and services. Your participation helps us create great experiences for every customer.

To send us feedback or make a connection, contact Sarah Buccellato, Technical Editor, Technical Content Management at Sarah_Buccellato@ibi.com.

To request permission to repurpose copyrighted material, please contact Frances Gambino, Vice President, Technical Content Management at *Frances_Gambino@ibi.com*.

Inf%rmation Builders

iWay

iWay Application Adapter for J.D. Edwards EnterpriseOne User's Guide

Version 7.0.x and Higher

DN3502254.0418

Information Builders, Inc. Two Penn Plaza New York, NY 10121-2898