

JD Edwards EnterpriseOne Applications

Business Interfaces Implementation Guide

9.2

Copyright © 2011, 2024, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	i
<hr/>	
1 Introduction to JD Edwards EnterpriseOne Business Interfaces	1
Understanding this Guide	1
Overview of Business Interfaces	3
2 Introduction to JD Edwards EnterpriseOne Business Services	7
Understanding JD Edwards EnterpriseOne Business Services	7
JD Edwards EnterpriseOne Business Services Integration Process	8
JD Edwards EnterpriseOne Business Services Implementation	9
3 Introduction to Real-Time Events	11
JD Edwards EnterpriseOne Application Real-Time Events Overview	11
JD Edwards EnterpriseOne Application Real-Time Events Implementation	12
4 Introduction to Batch Import and Export Programs	15
Batch Import and Export Programs Overview	15
5 Accessing Additional Information for Business Interface Components	17
Using the Oracle Technical Catalog	17
Accessing Javadoc for Business Services	35
6 Business Services for Pre-Built Integrations	69
Overview of Business Services for Pre-Built Integrations	69
Agile Product Lifecycle Management Integration with JD Edwards EnterpriseOne	69
Mobile Smartphone Applications	70
Direct Connect for Requisition Self Service	74
Global Order Promising Integration Using Web Service Callout	75
Purchase Order Dispatch for Requisition Self Service	75
Rate Shopping Integration Using Web Service Callout	76

EU VAT ID Validation with VIES Using Web Service	76
Brazil NF-e Transmission Web Services	77
7 JD Edwards EnterpriseOne Orchestrations	79
Understanding JD Edwards EnterpriseOne Orchestrations	79
Additional Orchestrations	79
8 ALM: Asset Master - Equipment Extensions	87
Asset Master - Equipment Extensions Overview	87
Asset Master - Equipment Extensions Business Services - EquipmentManager	87
Asset Master - Equipment Extensions Batch Export Programs	91
9 ALM: Condition Based Alert	93
Condition Based Alert Overview	93
Condition Based Alert Business Services - CapitalAssetManager	93
Condition Based ALert Batch Import Programs	99
10 ALM: Equipment Locations	101
Equipment Locations Overview	101
Equipment Locations Real-Time Events	101
Equipment Locations Business Services - LatLongProcessor (Release 9.2 Update)	102
11 ALM: Meter Reading	105
Meter Reading Overview	105
Meter Reading Batch Import Programs	105
12 ALM: Work Order - Equipment	107
Work Order - Equipment Overview	107
Work Order - Equipment Business Services - Equipment Work Order (JP130001)	107
Work Order - Equipment Real-Time Events	108
13 ALM: Work Order - Service	109
Work Order - Service Overview	109
Work Order - Service Business Services - CustomerServiceManager	109
Work Order - Service Business Services - Service Work Order (JP90CD00)	119
Work Order - Service Real-Time Events	120

14 CRM: Service Timecard	121
Service Timecard Overview	121
15 CRM: Work Order - Supplier Recovery Claim	123
Work Order - Supplier Recovery Claim Overview	123
Work Order - Supplier Recovery Claim Batch Import Programs	123
Work Order - Supplier Recovery Claim Export Programs	124
16 FMS: Account Ledger	127
Account Ledger Overview	127
Account Ledger Real-Time Events	127
Account Ledger Batch Import Programs	129
17 FMS: Account Master	131
Account Master Overview	131
Account Master Real-Time Events	131
18 FMS: Accounts Payable Draft	133
Accounts Payable Draft Overview	133
Accounts Payable Draft Real-Time Events	133
Accounts Payable Draft Batch Export Programs	135
19 FMS: Accounts Payable Payment	137
Accounts Payable Payment Overview	137
Accounts Payable Payment Real-Time Events	137
Accounts Payable Payment Batch Export Programs	139
20 FMS: Accounts Payable Voucher	141
Accounts Payable Voucher Overview	141
Accounts Payable Voucher Business Services - AccountsPayableManager	142
Accounts Payable Voucher Batch Import Programs	146
21 FMS: Accounts Receivable Deduction Activities	149
Accounts Receivable Deduction Activities Overview	149
Accounts Receivable Deduction Activities Real-Time Events	149

22	FMS: Accounts Receivable Invoice	151
	Accounts Receivable Invoice Overview	151
	Accounts Receivable Invoice Business Services - ARInvoiceManager	152
	Accounts Receivable Invoice Real-Time Events	158
	Accounts Receivable Invoice Batch Import Programs	160
23	FMS: Accounts Receivable Receipts and Drafts	161
	Accounts Receivable Receipts and Drafts Overview	161
	Accounts Receivable Receipts and Drafts Batch Import Programs	161
24	FMS: Asset Master	163
	Asset Master Overview	163
	Asset Manager Business Services - FixedAssetManager	163
	Asset Master Batch Import Programs	165
	Asset Master Batch Export Programs	165
25	FMS: Bank Statement	167
	Bank Statement Overview	167
	Bank Statements Batch Import Programs	167
26	FMS: Budgets	169
	Budgets Overview	169
	Budgets Batch Import Programs	169
27	FMS: Cost Analyzer Balances	171
	Cost Analyzer Balances Overview	171
	Cost Analyzer Batch Import Programs	171
28	FMS: Credit Card Data	173
	Credit Card Data Overview	173
29	FMS: Customer Master Data	175
	Customer Master Data Overview	175

Customer Master Data Business Services - CustomerManager	176
Customer Master Data Business Services - CustomerAndContactManager	183
Customer Master Data Real-Time Events	184
Customer Master Data Batch Import Programs	191
30 FMS: Employee Profile	193
Employee Profile Overview	193
31 FMS: Expense Report	195
Expense Report Overview	195
32 FMS: Expense Report Approvals	197
Expense Report Approvals Overview	197
33 FMS: Supplier Master Data	199
Supplier Master Data Overview	199
Supplier Master Data Business Services - SupplierManager	200
Supplier Master Data Real-Time Events	205
Supplier Master Data Batch Import Programs	210
34 FMS: Financial Management Solutions - Additional Interface Components	213
Additional Financial Management Solutions Interface Component Overview	213
FinancialComplianceManager Business Services	213
35 Foundation: Address Book	227
Address Book Overview	227
Address Book Business Services - AddressBookManager	228
Address Book Business Services - LatLongProcessor (Release 9.2 Update)	237
Address Book Real-Time Events	238
Address Book Batch Import Programs	246
36 Foundation: Business Unit	249
Business Unit Overview	249

Business Unit Business Services - BusinessUnitManager	250
Business Unit Real-Time Events	253
Business Unit Batch Import Programs	257
Business Unit Batch Export Programs	258
37 Foundation: Company	259
Company Overview	259
Company Business Services - CompanyManager	260
Company Real-Time Events	263
Company Batch Import Programs	265
Company Batch Export Programs	265
38 Foundation: Currency Information	267
Currency Information Overview	267
Currency Information Real-Time Events	267
Currency Information Batch Import Programs	268
39 Foundation: Additional Interface Components	269
Foundation Additional Components Overview	269
FoundationEnvironmentManager Business Services	269
40 HCM: Competency	279
Competency Overview	279
Competency Real-Time Events	279
41 HCM: Employee Master	281
Employee Master Overview	281
EmployeeManager Business Services	281
Employee Master Real-Time Events	290
Employee Master Batch Import Programs	294
42 HCM: Job	295
Job Overview	295
Job Real-Time Events	295

43	HCM: Resource Competency	299
	Resource Competency Overview	299
	Resource Competency Real-Time Events	299
	Resource Competency Batch Import Programs	302
44	HCM: Human Resource Requisitions	303
	Requisition Components Overview	303
45	HCM: Timecard	307
	Timecard Overview	307
	Timecard Batch Import Programs	307
46	Project Management: Bid Master	309
	Bid Master Overview	309
	Bid Master Real-Time Events	309
47	Project Management: Community Master	313
	Community Master Overview	313
	Community Master Real-Time Events	313
48	Project Management: Construction Lot Master	319
	Construction Lot Master Overview	319
	Construction Lot Master Real-Time Events	319
49	Project Management: Project	323
	Project Overview	323
50	Project Management: Purchase Order - Homebuilder Extensions	325
	Purchase Order - Homebuilder Extensions Overview	325
	Purchase Order - Homebuilder Extensions Real-Time Events	325
51	REM: Lease (Release 9.2 Update)	327
	Lease Overview	327

Lease Batch Import Programs	327
52 REM: Tenant Work Order	329
Tenant Work Order Overview	329
Tenant Work Order Business Services - Tenant Work Order (JP150000)	329
53 SCM and Manufacturing: Demand	331
Demand Overview	331
Demand Batch Import Programs	331
54 SCM and Manufacturing: Engineering Change Orders	333
Engineering Change Orders Overview	333
55 SCM and Manufacturing: Forecast	335
Forecast Overview	335
Forecast Batch Import Programs	335
56 SCM and Manufacturing: Inventory Cycle Counts	337
Inventory Cycle Counts Overview	337
Inventory Cycle Counts Batch Import Programs	337
57 SCM and Manufacturing: Inventory Location Master	339
Inventory Location Master Overview	339
Inventory Location Master Real-Time Events	339
Inventory Location Master Batch Import Programs	342
58 SCM and Manufacturing: Inventory Tag Counts	343
Inventory Tag Counts Overview	343
Inventory Tag Counts Batch Import Programs	343
59 SCM and Manufacturing: Inventory Transactions	345
Inventory Transactions Overview	345
Inventory Transactions Batch Import Programs	345

Inventory Transactions Batch Export Programs	346
60 SCM and Manufacturing: Item Branch Master	347
Item Branch Master Overview	347
Item Branch Master Business Services - InventoryManager	348
61 SCM and Manufacturing: Item Cost	361
Item Cost Overview	361
Item Cost Real-Time Events	361
Item Cost Batch Import Programs	362
62 SCM and Manufacturing: Item Master	363
Item Master Overview	363
Item Master Real-Time Events	364
Item Master Batch Import Programs	366
Item Master Batch Export Programs	366
63 SCM and Manufacturing: Kanban	369
Kanban Overview	369
Kanban Batch Import Programs	369
64 SCM and Manufacturing: Shop Floor	371
Shop Floor Overview	371
Shop Floor Real-Time Events	371
65 SCM and Manufacturing: Planning Schedule	375
Planning Schedule Overview	375
Planning Schedule Batch Import Programs	375
Planning Schedule Batch Export Programs	376
66 SCM and Manufacturing: Pricing	379
Pricing Overview	379
Pricing Real-Time Events	380
Item Base Price Batch Import Program	381

67	SCM and Manufacturing: Product Data Management	383
	Product Data Management Overview	383
	Product Data Management Batch Import and Export Programs	383
68	SCM and Manufacturing: Sales Invoice	385
	Sales Invoice Overview	385
	Sales Invoice Real-Time Events	385
	Sales Invoice Batch Export Programs	388
69	SCM and Manufacturing: Sales Order	389
	Sales Order Overview	389
	Sales Order Business Services - SalesOrderManager	390
	Sales Order Real-Time Events	413
	Sales Order Batch Import Programs	417
	Sales Order Batch Export Programs	418
70	SCM and Manufacturing: Ship and Debit	421
	Ship and Debit Overview	421
	Ship and Debit Batch Import Programs	421
71	SCM and Manufacturing: Style Item Master	423
	Style Item Master Overview	423
	Style Item Master Batch Import Programs	423
72	SCM and Manufacturing: Transportation Shipment	425
	Transportation Shipment Overview	425
	Transportation Shipment Business Services	426
	Transportation Shipment Real-Time Events	429
	Transportation Shipment Batch Import Programs	435
	Transportation Shipment Batch Export Programs	436
73	SCM and Manufacturing: Test Result	437
	Test Result Overview	437
	Test Result Batch Import Programs	437

74	SCM and Manufacturing: UOM Conversions	439
	UOM Conversions Overview	439
	UOM Conversions Real-Time Events	439
75	SCM and Manufacturing: Warehouse Suggestion	441
	Warehouse Suggestion Overview	441
	Warehouse Suggestion Batch Import Programs	441
76	SCM and Manufacturing: Work Order - Food and Beverage	443
	Work Order - Food and Beverage Overview	443
	Work Order - Food and Beverage Batch Import Programs	443
77	SCM and Manufacturing: Work Order - Manufacturing	445
	Work Order - Manufacturing Overview	445
	Work Order - Manufacturing Business Services - WorkOrderManager (JP480000)	446
	Work Order Business Services - Manufacturing Work Order (JP310000)	446
	Work Order - Manufacturing Real-Time Events	451
	Work Order - Manufacturing Batch Import Programs	453
	Work Order - Manufacturing Batch Export Programs	454
78	SCM and Manufacturing: Additional Interface Components	455
	SCM Additional Components Overview	455
	SCM Business Services - IntegrationTimeStampManager	455
79	Supply Management: Purchase Orders	457
	Purchase Orders Overview	457
	Purchase Orders Business Services - ProcurementManager	458
	Purchase Orders Real-Time Events	468
	Purchase Orders Batch Import Programs	474
	Purchase Orders Batch Export Programs	475
80	Supply Management: Purchase Order Receipts	477
	Purchase Order Receipts Overview	477
	Purchase Order Receipts Business Services - PurchaseOrderReceiptsManager	478

Purchase Order Receipts Real-Time Events	482
Purchase Order Receipts Batch Import Programs	483
Purchase Order Receipts Batch Export Programs	483

81 Supply Management: Procurement Requisition Approval **485**

Procurement Requisition Approval Overview	485
---	-----

82 Supply Management: Procurement Voucher **487**

Procurement Voucher Overview	487
Procurement Voucher Business Services - VoucherMatchManager	487
Procurement Voucher Batch Import Programs	494

Index	495
--------------------	------------

Preface

Welcome to the JD Edwards EnterpriseOne documentation.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Information

For additional information about JD Edwards EnterpriseOne applications, features, content, and training, visit the JD Edwards EnterpriseOne pages on the JD Edwards Resource Library located at:

<http://learnjde.com>

Conventions

The following text conventions are used in this document:

Convention	Meaning
Bold	Boldface type indicates graphical user interface elements associated with an action or terms defined in text or the glossary.
<i>Italics</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
Monospace	Monospace type indicates commands within a paragraph, URLs, code examples, text that appears on a screen, or text that you enter.
> Oracle by Example	Indicates a link to an Oracle by Example (OBE). OBEs provide hands-on, step-by-step instructions, including screen captures that guide you through a process using your own environment. Access to OBEs requires a valid Oracle account.

1 Introduction to JD Edwards EnterpriseOne Business Interfaces

Understanding this Guide

This guide discusses and describes the business interfaces that are available for use in the JD Edwards EnterpriseOne system. Business interfaces include business services, real-time events, and import and export batch programs.

This guide contains the following chapters:

Part I: Business Interfaces Overview contains the following chapters:

- *Chapter 1 - Introduction to JD Edwards EnterpriseOne Business Interfaces*
- *Chapter 2 - Introduction to JD Edwards EnterpriseOne Business Services*
- *Chapter 3 - Introduction to Real-Time Events*
- *Chapter 4 - Introduction to Batch Import and Export Programs*
- *Chapter 5 - Accessing Additional Information for Business Interface Components*
- *Chapter 6 - Business Services for Pre-Built Integrations*
- *Chapter 7 - Understanding JD Edwards EnterpriseOne Orchestrations*

Part II Asset Lifecycle Management (ALM) contains the following chapters:

- *Chapter 8 - ALM: Asset Master - Equipment Extensions*
- *Chapter 9 - ALM: Condition Based Alert*
- *Chapter 10 - ALM: Equipment Locations*
- *Chapter 11 - ALM: Meter Reading*
- *Chapter 12 - ALM: Work Order - Equipment*
- *Chapter 13 - ALM: Work Order - Service*

Part III Customer Relationship Management (CRM) contains the following chapters:

- *Chapter 14 - CRM: Service Timecard*
- *Chapter 15 - CRM: Work Order - Supplier Recovery Claim*

Part IV Financial Management Solutions (FMS) contains the following chapters:

- *Chapter 16 - FMS: Account Ledger*
- *Chapter 17 - FMS: Account Master*
- *Chapter 18 - FMS: Accounts Payable Draft*
- *Chapter 19 - FMS: Accounts Payable Payment*
- *Chapter 20 - FMS: Accounts Payable Voucher*
- *Chapter 21 - FMS: Accounts Receivable Deduction Activities*
- *Chapter 22 - FMS: Accounts Receivable Invoice*
- *Chapter 23 - FMS: Accounts Receivable Receipts and Drafts*

- *Chapter 24 - FMS: Asset Master*
- *Chapter 25 - FMS: Bank Statement*
- *Chapter 26 - FMS: Budgets*
- *Chapter 27 - FMS: Cost Analyzer Balances*
- *Chapter 28 - FMS: Credit Card Data*
- *Chapter 29 - FMS: Customer Master Data*
- *Chapter 30 - FMS: Employee Profile*
- *Chapter 31 - FMS: Expense Report*
- *Chapter 32 - FMS: Expense Report Approvals*
- *Chapter 33 - FMS: Supplier Master Data*
- *Chapter 34 - FMS: Financial Management Solutions - Additional Interface Components*

Part V Foundation contains the following chapters:

- *Chapter 35 - Foundation: Address Book*
- *Chapter 36 - Foundation: Business Unit*
- *Chapter 37 - Foundation: Company*
- *Chapter 38 - Foundation: Currency Information*
- *Chapter 39 - Foundation: Additional Interface Components*

Part VI Human Capital Management (HCM) contains the following chapters:

- *Chapter 40 - HCM: Competency*
- *Chapter 41 - HCM: Employee Master*
- *Chapter 42 - HCM: Job*
- *Chapter 43 - HCM: Resource Competency*
- *Chapter 44 - HCM: Human Resource Requisitions*
- *Chapter 45 - HCM: Timecard*

Part VII Project Management contains the following chapters:

- *Chapter 46 - Project Management: Bid Master*
- *Chapter 48 - Project Management: Community Master*
- *Chapter 49 - Project Management: Construction Lot Master*
- *Chapter 50 - Project Management: Project*
- *Chapter 51 - Project Management: Purchase Order - Homebuilder Extensions*

Part VIII Real Estate Management (REM) contains the following chapters:

- *Chapter 52 - REM: Lease (Release 9.2 Update)*
- *Chapter 53 - REM: Tenant Work Order*

Part IX Supply Chain Management (SCM) and Manufacturing contains the following chapters:

- *Chapter 54 - SCM and Manufacturing: Demand*
- *Chapter 55 - SCM and Manufacturing: Engineering Change Orders*
- *Chapter 56 - SCM and Manufacturing: Forecast*

- *Chapter 57 - SCM and Manufacturing: Inventory Cycle Counts*
- *Chapter 58 - SCM and Manufacturing: Inventory Location Master*
- *Chapter 59 - SCM and Manufacturing: Inventory Tag Counts*
- *Chapter 60 - SCM and Manufacturing: Inventory Transactions*
- *Chapter 61 - SCM and Manufacturing: Item Branch Master*
- *Chapter 62 - SCM and Manufacturing: Item Cost*
- *Chapter 63 - SCM and Manufacturing: Item Master*
- *Chapter 64 - SCM and Manufacturing: Kanban*
- *Chapter 65 - SCM and Manufacturing: Shop Floor*
- *Chapter 66 - SCM and Manufacturing: Planning Schedule*
- *Chapter 67 - SCM and Manufacturing: Pricing*
- *Chapter 68 - SCM and Manufacturing: Product Data Management*
- *Chapter 69 - SCM and Manufacturing: Sales Invoice*
- *Chapter 70 - SCM and Manufacturing: Sales Order*
- *Chapter 71 - SCM and Manufacturing: Ship and Debit*
- *Chapter 72 - SCM and Manufacturing: Style Item Master*
- *Chapter 73 - SCM and Manufacturing: Transportation Shipment*
- *Chapter 74 - SCM and Manufacturing: Test Result*
- *Chapter 75 - SCM and Manufacturing: UOM Conversions*
- *Chapter 76 - SCM and Manufacturing: Warehouse Suggestion*
- *Chapter 77 - SCM and Manufacturing: Work Order - Food and Beverage*
- *Chapter 78 - SCM and Manufacturing: Work Order - Manufacturing*
- *Chapter 79 - SCM and Manufacturing: Additional Interface Components*

Part X Supply Management contains the following chapters:

- *Chapter 80 - Supply Management: Purchase Orders*
- *Chapter 81 - Supply Management: Purchase Order Receipts*
- *Chapter 82 - Supply Management: Procurement Requisition Approval*
- *Chapter 83 - Supply Management: Procurement Voucher*

Overview of Business Interfaces

A business interface is a set of components that implementation teams can use to create an integration between JD Edwards EnterpriseOne and an external system. Business interfaces can include one or more of these business interface components:

- Business Services
- Real-Time Events
- Batch Import and Export Programs
- Orchestrations

See:

- [Understanding JD Edwards EnterpriseOne Business Services](#)
- [JD Edwards EnterpriseOne Application Real-Time Events Overview](#)
- [Batch Import and Export Programs Overview](#)
- [Understanding JD Edwards EnterpriseOne Orchestrations](#)

Each of these interface components can be used to send data to or from the EnterpriseOne system. While some of these interface components are used within a specific process, most have been designed as stand-alone components that can assist implementation teams with building customized integration solutions.

Note:

Most of the business services available in the EnterpriseOne system are stand-alone components. Those business services are documented in detail in this guide.

In addition to those business services, there are others that have been created for use with a specific pre-built integration. Those services are documented within the context of the integration process to which they belong.

For additional information about the services that have been created as part of a pre-built integration process, see [Business Services for Pre-Built Integrations](#)

Business Objects

This guide describes each business interface component available in the EnterpriseOne system, and organizes these components by business object. A business object is a specific piece of business data, such as a customer, sales order, or business unit.

For the purposes of this guide, the business objects have been grouped under their respective product areas, which include:

- Asset Lifecycle Management (ALM)
- Customer Relationship Management (CRM)
- Financial Management Solutions (FMS)
- Foundation
- Human Capital Management (HCM)
- Project Management
- Real Estate Management
- Supply Chain Management and Manufacturing (SCM and MFG)
- Supply Management

Technical Catalog

Additional detailed technical information about these interface components is available in the Oracle Technical Catalog, which is accessible at:

<https://apex.oracle.com/pls/apex/f?p=50180>

For additional information about the information provided in the catalog, and instructions regarding how to use the catalog, see *Using the Oracle Technical Catalog*.

2 Introduction to JD Edwards EnterpriseOne Business Services

Understanding JD Edwards EnterpriseOne Business Services

Business services are objects that enable interoperability between JD Edwards EnterpriseOne and other Oracle applications or third-party applications and systems. Business services enable software applications that are written in various programming languages and running on various platforms to exchange information.

A web service is a standardized way of integrating web-based applications. In JD Edwards EnterpriseOne, published business services are exposed to consumers as web services. These web services enable JD Edwards EnterpriseOne to expose native transactions to other applications and systems.

Note: In the JD Edwards EnterpriseOne system, published business services are exposed as web services. Therefore, in many cases, you will see the terms business service and web service used interchangeably.

Published Business Services

A published business service is a JD Edwards EnterpriseOne object that is managed in Object Management Workbench (OMW). The published business service represents one Java class that publishes multiple business services. When you create a business service or web service, you identify the Java class. The published business service also contains value object classes that make up the signature for the published business service.

Internal Business Services

A business service is a JD Edwards EnterpriseOne object that is managed by OMW. A business service represents one or more classes that expose public methods. Each method performs a business process. A business service also contains internal value object classes that make up the signature for the business service methods. These public methods can be called from other business service classes and published business service classes.

Published business services are exposed to consumers. Internal business services are created for internal use by JD Edwards EnterpriseOne published business services. They are made available to consumers as a call from a published business service. Business services that are exposed to consumers are exposed as web service operations.

Note: In the context of EnterpriseOne, business services are exposed as web service operations. Therefore, in many cases, you will see the terms business service and web service used interchangeably.

Business Service Properties

Many business services use business services properties. Business service properties are similar to processing options for an interactive or batch program in the JD Edwards EnterpriseOne system. Properties enable you to easily change values or processing information about a business service without having to modify the programming code associated with that business service.

For example, many business services include a business service property that enables the user to specify the maximum number of records to return for a query. Other business services include properties that enable users to specify the version of a specific JD Edwards EnterpriseOne program to use during processing.

This documentation includes details about all of the business service properties that are used by each business service.

Utilities

Utilities are generic, reusable business services that perform standard operations that are used by many business services.

Utilities enable multiple business services to complete the same process in a uniform manner. For example, both the ProcessPurchaseOrder and ProcessCustomer business services retrieve entity information from the JD Edwards EnterpriseOne system. Rather than coding the retrieval of entity information separately in both of the business services, each business service uses the Entity Processor utility to retrieve the entity information. By using the Entity Processor utility, the complexity and the amount of code associated with each of the business services is reduced.

Business services use these utilities:

- Entity Processor (J0100010)
- GL Account Processor (J0900010)
- Inventory Item ID Processor (J4100010)
- Net Change Processor (J0000020)
- Processing Version Processor (J0000010)
- User Defined Code Utility (J0000050)

See *JD Edwards EnterpriseOne Tools Business Services Development Guide* .

JD Edwards EnterpriseOne Business Services Integration Process

Published business services transfer information between JD Edwards EnterpriseOne and other Oracle or third-party systems. Internal business services (web service operations) process that information within EnterpriseOne. Because these systems may be using different programming languages or running on different platforms, data must be formatted during transfer so that it can be read and processed by each system.

The user of a third-party system, also known as the consumer, initiates a business services integration by entering data into an input interface. An input interface is based on JD Edwards EnterpriseOne data structure, and is used to hold the

information associated with the user's request. For example, the consumer might specify that they want to inquire on a particular sales order in the JD Edwards EnterpriseOne system. Therefore, they can enter the sales order number in the input interface and specify that they want to inquire on that order.

The published business service reads the consumer's request to determine the action that the system takes, and calls the appropriate business service. In this example the SalesOrderManager published business service calls the GetSalesOrder business service. Additionally, the published business service formats the data in the input interface so that it can be read and processed by the JD Edwards EnterpriseOne system.

The business service passes the data into the EnterpriseOne system and tells the system what actions to perform. After the EnterpriseOne system processes or retrieves the specified information, it sends the results back to the business service. The business service receives the processed data from the EnterpriseOne system and passes it back to the published business service. The published business service then formats the data so that it can be read by the third-party system and passes it back to the consumer using the response interface.

JD Edwards EnterpriseOne Business Services Implementation

Before you can use JD Edwards EnterpriseOne business services, you must be sure that the technical setup of your system supports Service Oriented Architecture (SOA) integration processing. All of the system setup steps that you must perform are outlined in the JD Edwards EnterpriseOne 8.98 Documentation Map, which you can access on the JD Edwards Service Oriented Architecture (SOA) page from the My Oracle Support web site.

In addition to technical setup, you must set up the application systems that are associated with the business services you will use before you can use the business services that are included in this documentation:

- JD Edwards EnterpriseOne Accounts Payable
- JD Edwards EnterpriseOne Accounts Receivable
- JD Edwards EnterpriseOne Address Book
- JD Edwards EnterpriseOne Customer Relationship Management
- JD Edwards EnterpriseOne Human Capital Management
- JD Edwards EnterpriseOne Inventory Management
- JD Edwards EnterpriseOne Procurement
- JD Edwards EnterpriseOne Real Estate Management
- JD Edwards EnterpriseOne Sales Order Management

The implementation steps that are required for each of these systems are documented in detail in the corresponding implementation guide.

Note:

- *"JD Edwards EnterpriseOne Accounts Payable Preface" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide .*
- *"JD Edwards EnterpriseOne Accounts Receivable Preface" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide .*
- *"JD Edwards EnterpriseOne Address Book Preface" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide .*
- *"JD Edwards EnterpriseOne CRM Application Fundamentals Preface" in the JD Edwards EnterpriseOne Applications Customer Relationship Management Fundamentals Implementation Guide .*
- *"JD Edwards EnterpriseOne Human Capital Management Fundamentals" in the JD Edwards EnterpriseOne Applications Human Capital Management Fundamentals Implementation Guide .*
- *"JD Edwards EnterpriseOne Inventory Management Preface" in the JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide .*
- *"JD Edwards EnterpriseOne Procurement Management Preface" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide .*
- *"JD Edwards Real Estate Management Preface" in the JD Edwards EnterpriseOne Applications Real Estate Management Implementation Guide .*
- *"JD Edwards EnterpriseOne Sales Order Management Preface" in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide .*

3 Introduction to Real-Time Events

JD Edwards EnterpriseOne Application Real-Time Events Overview

A real-time event (RTE) is a notification to a third-party system that a business transaction has occurred in the JD Edwards EnterpriseOne system. Third-party systems can subscribe to the JD Edwards EnterpriseOne system to receive notification when a specific transaction occurs. You can use any JD Edwards EnterpriseOne interface, such as HTML, WIN32, and terminal servers to generate real-time events. Real-time events can be used for both synchronous and asynchronous processing.

A XAPI event is similar to a real-time event in that an outbound notification is sent to subscribers when a specified transaction occurs in the JD Edwards EnterpriseOne system. The difference between a real-time event and a XAPI event is that the subscriber to a XAPI event returns a reply to the originator. XAPI events use the same infrastructure as real-time events. XAPI events also provide a way for two different JD Edwards EnterpriseOne systems to communicate with each other.

The purpose of this guide is to provide information about the JD Edwards EnterpriseOne real-time and XAPI events.

Naming Conventions

Real-time events use this naming standard:

RTXYYYY where *XX* is the product area (for example AB, PO, or SO), and *YYY* is the type (for example, OUT, HDR, DTL).

Transaction-Specific Instructions

The discussions of the real-time events in this guide include a table similar to this table. This table explains the data that is contained in these tables.

Event Type

The name of the event.

Event Description

A description of the event.

Event Category

RTE for real-time events; **XAPI** for XAPI events.

Event Aggregate

Defined as single or container event.

A single event is defined as one data structure. For example, RTABHDR (D0100085A), whereas a container event is a group of single events that are included in the RTE output. For example, RTABOUT may include the single events RTABHDR, RTABEAOUT, and RTABPHOUT.

Product Code

The system code.

Data Structure

The JD Edwards EnterpriseOne data structure that is used for the current single event.

Event Relationship

The relationship of the current single event to other real-time events.

Defining Events

Events are defined in the Interoperability Event Definition table (F90701). You use the Event Definition Workbench program (P90701A) to add new single and container events to the Interoperability Event Definition table and to review the existing events. You add single events by event name. When you add a single event, you must include the JD Edwards EnterpriseOne data structure. A container event contains single events, aggregate events, or both.

Note: The system publishes container events, which contain the data from the contained single events, rather than publishing the single contained events. You can use the Connector Events Client to view published events.

Note: The real-time events that are described in this guide have already been set up in the data that is shipped with the software. You can use the Event Definition Workbench program to review these real-time events.

Note:

- *JD Edwards EnterpriseOne Tools Interoperability Guide , "Using Guaranteed Events."*

JD Edwards EnterpriseOne Application Real-Time Events Implementation

This section provides an overview of the steps that are required to use application real-time events.

In the planning phase of your implementation, take advantage of all JD Edwards EnterpriseOne sources of information, including the installation guides and troubleshooting information. A complete list of these resources appears in the preface with information about where to find the most current version of each.

Implementation Steps

Implementing any application real-time events must be preceded by setting up the corresponding JD Edwards EnterpriseOne system.

The Getting Started chapter in the implementation guide for each of these systems contains specific implementation information for that system.

After you have set up the relevant JD Edwards EnterpriseOne systems for the application real-time events you are using, you must complete this step to set up the application real-time events:

- Define and activate real-time events.

See *JD Edwards EnterpriseOne Tools Interoperability Guide*, "Using Guaranteed Events," *Defining Events*.

4 Introduction to Batch Import and Export Programs

Batch Import and Export Programs Overview

You use batch import and export programs to facilitate the sharing of data between JD Edwards EnterpriseOne systems and external software systems. There are several different kinds of batch processes used by the EnterpriseOne system to import and export data. These include:

- Electronic Data Interface (EDI) programs
- Interoperability programs
- Temporary table batch programs

EDI Programs

Electronic Data Interchange (EDI) is the paperless, computer to computer exchange of business transactions, such as purchase orders and invoices, in a standard format with standard content. As such, it is an important part of an electronic commerce strategy.

Electronic commerce is a means to extend business processes to include suppliers, customers, and employees in a fully integrated supply chain. It can yield shorter cycle times, more efficient inventory management, and better knowledge sharing throughout the extended enterprise. To see these benefits, you need an integrated enterprise application system that you can extend and customize for the supply chain, while still maintaining enterprise information integrity.

EDI is a crucial part of business to business commerce. When computers exchange data using EDI, the data is transmitted in EDI Standard format so that it is recognizable by other systems using the same EDI Standard format. Companies who use EDI have their own translator software package to convert the data from the EDI Standard format to their computer system's format. Companies that exchange EDI data are called trading partners.

Note: *Introduction to JD Edwards EnterpriseOne Data Interface for Electronic Data Interchange*

Interoperability Programs

To fully cover the information requirements of an enterprise, companies sometimes use products from different software and hardware providers. Interoperability among different products is important to successfully implementing the enterprise solution. Full interoperability among different systems results in a flow of data that is seamless to the user. The interoperability function provides an interface that facilitates exchanging transactions, both inbound and outbound, with external systems.

Interoperability for inbound transactions consists of these processes:

- External systems send information to the interface tables using either an external program or flat files and the Inbound Flat File Conversion program. The party sending the information is responsible for conforming to format and other requirements for the interface tables.

- You run a transaction process (a batch program) that validates the data, updates valid data to the JD Edwards EnterpriseOne application tables, and sends action messages about incorrect data to the Work Center.
- You use an inquiry function to interactively review and revise the incorrect data, and then run the transaction process again. You repeat this step as often as needed to correct errors.

Interoperability for outbound transactions loads data into tables for third-party consumption. The data is loaded into the tables by applications and processes that add or modify data within EnterpriseOne. Interoperability for outbound transactions requires that you set a processing option specifying a transaction type. Additionally, some entry programs enable you to specify a version of the Master Business Function Processing Options program that, in turn, enables you to specify a version of the Interoperability Processing Options program. This is useful if you need to create a personal version of the Interoperability Processing Options program to designate different transaction types.

Note: *Introduction to JD Edwards EnterpriseOne Application Fundamentals Interoperability*

Temporary Table Batch Processing

Many product areas in the EnterpriseOne software system use batch programs to import data into, or export data from, the EnterpriseOne database. These programs either write data into, or retrieve data from temporary tables, which are often referred to as Z tables because the table IDs end with Z. These Z tables store the imported or exported data so it can be reviewed, and if necessary, manipulated, before being used by the target system.

The temporary table batch import and export programs function like any other EnterpriseOne batch program, with regard to copying, versioning, processing options, data selection, and processing.

Note: *Introduction to JD Edwards EnterpriseOne Batch Versions*

Note: These batch import and export programs are documented within the product guides that describe the processes in which the batch programs are used in the EnterpriseOne system. Links to each program are provided in the chapters that discuss the specific business object associated with the programs.

5 Accessing Additional Information for Business Interface Components

Using the Oracle Technical Catalog

You can use the Oracle Technical Catalog to access detailed technical information about available business interface components. The components in the catalog are organized by business object.

For each business object included in the technical catalog, you can access a functionality matrix. This matrix lists all of the available components of a business object, and provides details about the functionality of each component. This graphic illustrates the information included in a functionality matrix document:

AddressBook	Service Operations		Real Time Events			Batch Import/Export	Comments
	Get	Process	Address Book RTABOUT	Org Structure RTPCOUT	Branch Plant RTSHPOUT	Z File In	
Functionality							
Single	✓	✓	✓	✓	✓	✓	
Multiple	✓	✗				✓	
Add		✓				✗	
Update without Net Change		✓				✗	
Update with Net Change		✓				✗	
Delete		✓				✗	
Localizations		✗				✗	
Component							
Postal Codes	✗	✗	✗	✗	✗	✗	
Alternate Address	✓	✓	✗	✗	✗	✗	
Address Organization Structure Master	✓	✓	✗	✓	✗	✗	
Enhanced Parent Child	✗	✗	✗	✗	✗	✗	
Postal Code Address	✗	✗	✗	✗	✗	✗	
Activity Master	✗	✗	✗	✗	✗	✗	
Calendar Master	✗	✗	✗	✗	✗	✗	
Action Plan	✗	✗	✗	✗	✗	✗	
Supplier Master	✗	✗	✗	✗	✗	✗	
Customer Master	✗	✗	✗	✗	✗	✗	
Address Book - Who's Who	✓	✓	✓	✗	✗	✓	
Address by Date	✓	✓	✓	✗	✗	✗	
Service/WarrantyManagement	✗	✗	✗	✗	✗	✗	
Supplemental Data	✗	✗	✗	✗	✗	✗	
Bank Account Info	✗	✗	✗	✗	✗	✗	
GLN	✓	✓	✗	✗	✗	✗	GLN Entry Enhancement is included in Version 2 of BSSV
Address Book Master	✓	✓	✓	✗	✓	✓	
Related Person	✗	✓	✗	✗	✗	✗	
Address Book - Phone Numbers	✓	✓	✓	✗	✗	✓	
Electronic Address	✓	✓	✓	✗	✗	✓	
Attachments	✗	✗	✗	✗	✗	✗	

The functionality matrix lists all business services, real-time events, and batch import and export programs available for the business object. It also includes details about what functions and information each service, event, or program includes.

In addition to the functionality matrix, there is additional information available for each component. This table describes the information that is available for each type of business interface component:

Object Type	Available Information
Business Services	<p>For each business service, you can access a list of methods associated with the business service. For each associated method, you can view:</p> <ul style="list-style-type: none"> • Flow diagrams • Method descriptions • Input details • Output details
Real-Time Events	<p>For each real-time event, you can access flow diagrams and data structure details, which include the following information about each field in the data structure:</p> <ul style="list-style-type: none"> • Item Alias • Item Name • Item Type • Item Size
Batch Import and Export Programs	<p>For each batch import and export program, the technical catalog provides a list of the tables used by each program, and the columns contained in each of those tables.</p>

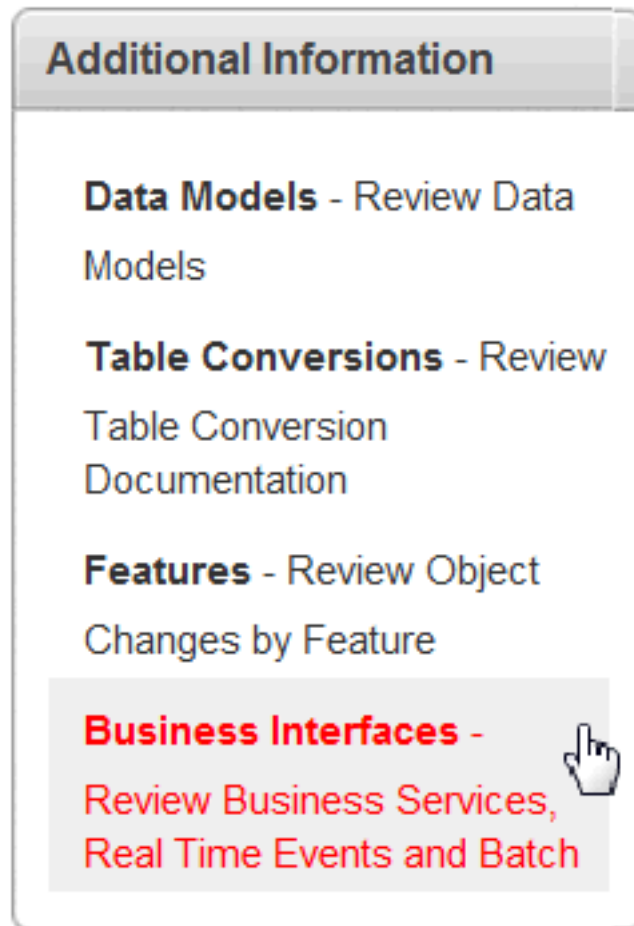
You can access the Oracle Technical Catalog at: <https://apex.oracle.com/pls/apex/f?p=50180>

Accessing Technical Catalog Information for Business Services

To access technical information for business services:

1. Log in to the Oracle Technical Catalog, and accept the user agreement notice. The catalog is located at: <https://apex.oracle.com/pls/apex/f?p=50180>
2. From the Oracle Technical Catalog Home screen, select JD Edwards EnterpriseOne.

3. In the Additional Information section, click the Business Interfaces link.



- To sort and search the data in the detail area of the form, you can click on any of the column headings and use the text search field to narrow your displayed results, or use the sort arrows to display the data in ascending or descending order.

The following example shows the search and sort feature that appears when you click the Object Type heading. The user wants to display only business services.

JD Edwards EnterpriseOne Business Interfaces

Q- Go Actions < Back

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
- AP Voucher	JP040000	Accounts Payable Manager			9.1 Update 2	04	Accounts Payable	Financial Management
Address Book Master	JP010000	Address Book Manager			9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTABOUT	Address Book Real Time Event Contai			9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMSQUT	Address Single Real Time Event	Business Service		9.1 Update 2	01	Address Book	System Foundation
Address Book Master	JP010050	Customer And Contact Manager	Real Time Event		9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT3	RTCMOUT3	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
- Customer Master Data	RTCMOUT	Customer RTCMOUT	Real Time Event	Yes	9.1 Update 2	04	Accounts Payable	Financial Management

- To search for a specific component, enter the object ID in the search field, and then click Go. The following example shows the search for the JP01000 business service.

ORACLE Technical Catalog Feedback Email this Page

Home Releases Objects System Codes Compare Releases Reporting

JD Edwards EnterpriseOne Business Interfaces

Q- JP01000 Go Actions < Back

Row text contains 'JP01000'

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
Address Book Master	JP010000	Address Book Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation

1 - 1



- To review the functionality matrix for a business object, click on the text in the Business Object column.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component	
-	A/P Voucher	JP040000	Accounts Payable Manager	Business Service	Yes	9.1 Update 2	04	Accounts Payable	Financial Management
	Address Book Master	JP010000	Address Book Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation
	Address Book Master	JP010050	Customer And Contact Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation

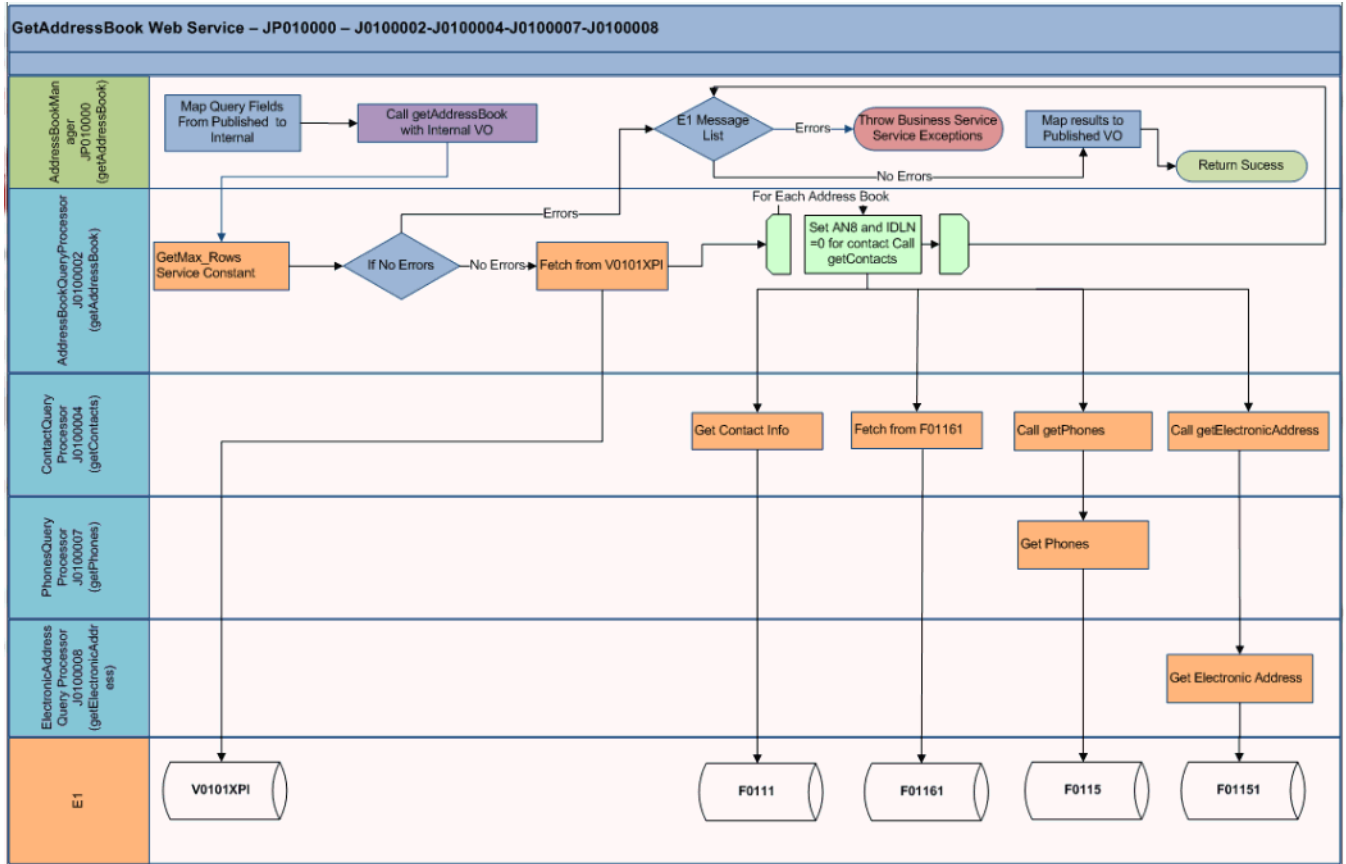
- To review additional details about the business service, click the text in the Object Name column.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component	
-	A/P Voucher	JP040000	Accounts Payable Manager	Business Service	Yes	9.1 Update 2	04	Accounts Payable	Financial Management
	Address Book Master	JP010000	Address Book Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation
	Address Book Master	JP010050	Customer And Contact Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation

8. The system displays information about each method associated with the selected business service. This example illustrates the methods shown for the AddressBookManager business service:

Method Name	Method Description	Input	Output
 getAddressBook	The getAddressBook is a database query operation that enables consumers to retrieve and review address, phone, and electronic mail information. The operation returns zero to many records if it completes successfully. If the operation encounters errors while processing address information, processing stops and those errors and warnings are returned to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted to warnings, processing continues, and the warnings are returned to the consumer.	GetAddressBook	ShowAddressBook
 processAddressBook	The processAddressBook is an inbound transaction operation that enables consumers to Add, Delete or Change records in the JD Edwards EnterpriseOne Address Book system. The operation uses the AddressBookMasterMBF business function (N0100041) to process information. If the operation encounters errors while processing address book information, processing stops and all errors and warnings are returned to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted into warnings, processing continues, and the warnings are returned to the consumer.	ProcessAddressBook	ConfirmProcessAddressBook

9. To view the flow diagram for the method, click on the text in the Method Name column. This graphic shows an example of a flow diagram:



10. To view the input details for the method, click on the text in the Input column. This graphic shows a portion of the input details for a business service method:

Field Name	Data Type	Required				Key	Function or Table	Data Item	Description
		Query	Add	Change	Delete				
getAddressBook	GetAddressBook	-	-	-	-	-	-	-	-
entityName	String	-	-	-	-	-	F0101	ALPH	NameAlpha
businessUnit	String	-	Yes	-	-	-	F0101	MCU	CostCenter
industryClassificationCode	String	-	Yes	-	-	-	F0101	SIC	StandardIndustryCode
languageCode	String	-	Yes	-	-	-	F0101	LNGP	LanguagePreference
entityTypeCode	String	-	-	-	-	-	F0101	AT1	AddressType1
entity	Entity	-	-	-	-	-	-	-	-
entityId	Integer	-	Yes	-	-	Yes	F0101	ANB	Address Number
entityLongId	String	-	Yes	-	-	-	F0101	ALKY	AlternateAddressKey
entityTaxId	String	-	Yes	-	-	-	F0101	TAX	TaxId
address	GetAddress	-	-	-	-	-	-	-	-
postalCode	String	-	Yes	-	-	-	F0116	ADDZ	ZipCodePostal
countyCode	String	-	Yes	-	-	-	F0116	COUN	CountyAddress
stateCode	String	-	Yes	-	-	-	F0116	ADDS	State
countryCode	String	-	Yes	-	-	-	F0116	CTR	Country
categoryCodesAddressBook	CategoryCodesAddressBook	-	-	-	-	-	-	-	-
categoryCode001	String	-	Yes	-	-	-	F0101	AC01	ReportCodeAddBook001
categoryCode010	String	-	Yes	-	-	-	F0101	AC10	ReportCodeAddBook010
categoryCode011	String	-	Yes	-	-	-	F0101	AC11	ReportCodeAddBook011
categoryCode012	String	-	Yes	-	-	-	F0101	AC12	ReportCodeAddBook012

1 - 20

- To view the output details for the method, click on the text in the Output column. This graphic shows a portion of the output details for a business service method:

Field Name	Data Type	Function or Table	Data Item	Description
addressBookResult	AddressBookResultV2[]	-	-	-
entityTypeCode	String	F0101	AT1	AddressType1
personCorporationCode	String	F0101	TAXC	PersonCorporationCode
subledgerInactiveCode	String	F0101	SBLI	SubledgerInactiveCode
entityTaxIdAdditional	String	F0101	TX2	TaxId2
taxExemptCertificate	String	F0101	TXCT	CertificateTaxExempt
isEntityTypeNettingIndicator	Boolean	F0101	AT2	AddressType2
isEntityType3	String	F0101	AT3	AddressType3
isEntityType4	String	F0101	AT4	AddressType4
isEntityType5	String	F0101	AT5	AddressType5
isEntityTypePayables	Boolean	F0101	ATR	AddressTypeReceivables
isEntityTypeReceivables	Boolean	F0101	ATP	AddressTypePayables
businessUnit	String	F0101	MCU	CostCenter
isEntityTypePurchaser	Boolean	F0101	ATPR	AddTypeCode4Purch
isEntityTypeEmployee	Boolean	F0101	ATE	AddressTypeEmployee
employeeGroupApprovalsCode	String	F0101	AEMPGP	EmployeeGroupApprovals
globalLocatorNumber	String	F0101	GAN8	GlobalLocationNumber
entity	Entity	-	-	-
entityId	Integer	F0101	AN8	Address Number
entityLongId	String	F0101	ALKY	AlternateAddressKey

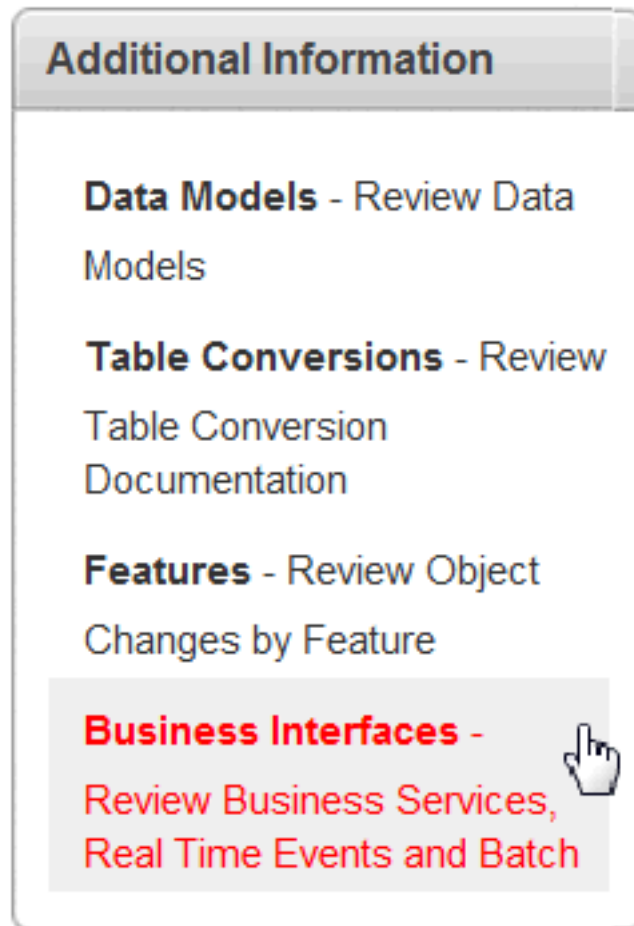
1 - 20 >

Accessing Technical Catalog Information for Real-Time Events

To access technical information for real-time events:

- Log in to the Oracle Technical Catalog, and accept the user agreement notice. The catalog is located at:
<https://apex.oracle.com/pls/apex/f?p=50180>
- From the Oracle Technical Catalog Home screen, select JD Edwards EnterpriseOne.

- In the Additional Information section, select click the Business Interfaces link.



- To sort and search the data in the detail area of the form, you can click on any of the column headings and use the text search field to narrow your displayed results, or use the sort arrows to display the data in ascending or descending order.

The following example shows the search and sort feature that appears when you click the Object Type heading. The user wants to display only Real-Time Events.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
Address Book Master	RTCMSOUT	Address Single Real Time Event	Real		9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT3	RTCMOUT3			9.1 Update 2	01	Address Book	System Foundation
Customer Master Data	RTCMOUT	Customer RTCMOUT			9.1 Update 2	04	Accounts Payable	Financial Management
Address Book Master	RTABOUT	Address Book Real Time Event Conta	Real Time Event		9.1 Update 2	01	Address Book	System Foundation
Address Book Master	JP010050	Customer And Contact Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	JP010000	Address Book Manager	Business Service	Yes	9.1 Update 2	01	Address Book	System Foundation

- To search for a specific component, enter the object ID in the search field. The following example shows the search for the RTABOUT event.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
Address Book Master	RTABOUT	Address Book Real Time Event Container1	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT	Address Single Real Time Event	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT3	RTCMOUT3	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Customer Master	RTCMOUT	RTCMOUT	Real Time Event	Yes	9.1 Update 2	04	Accounts Payable	Financial Management

- To review the functionality matrix for a business object, click on the text in the Business Object column.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
Address Book Master	RTABOUT	Address Book Real Time Event Container1	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT	Address Single Real Time Event	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT3	RTCMOUT3	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation

- To review additional details about the event, click the text in the Object Name column.

Business Object	Object Name	Object Description	Object Type	Business Interface	Release	System Code	Module	Component
Address Book Master	RTABOUT	Address Book Real Time Event Container1	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT	Address Single Real Time Event	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTCMOUT3	RTCMOUT3	Real Time Event	Yes	9.1 Update 2	01	Address Book	System Foundation

8. The system displays the data structure details of the selected event.

Attachment(s)
 [Address Book Master Flow-RTABOUT.pdf](#)

Q-

<u>Real Time Event Name</u>	<u>Real Time Event Description</u>	<u>Data Structure Name</u>	<u>Data Structure Description</u>
RTABEAOUT	AB Electronic Address	D0100085C	Address Book Master Email Wrap
RTABHDR	AB Header	D0100085A	Address Book Master Real Time
RTABPHOUT	AB Phone	D0100085B	Address Book Master Phones Wra

1 - 3

- Click on the attachment to view the flow diagram, or click the Data Structure name to view the data structure details.

Real Time Event Data Structure Detail

Description: Customer Master Real Time Reco

System Code: 01 (Address Book)

Component: System Foundation

▼
Real Time Event Data Structure

Q-

Go
Actions ▼

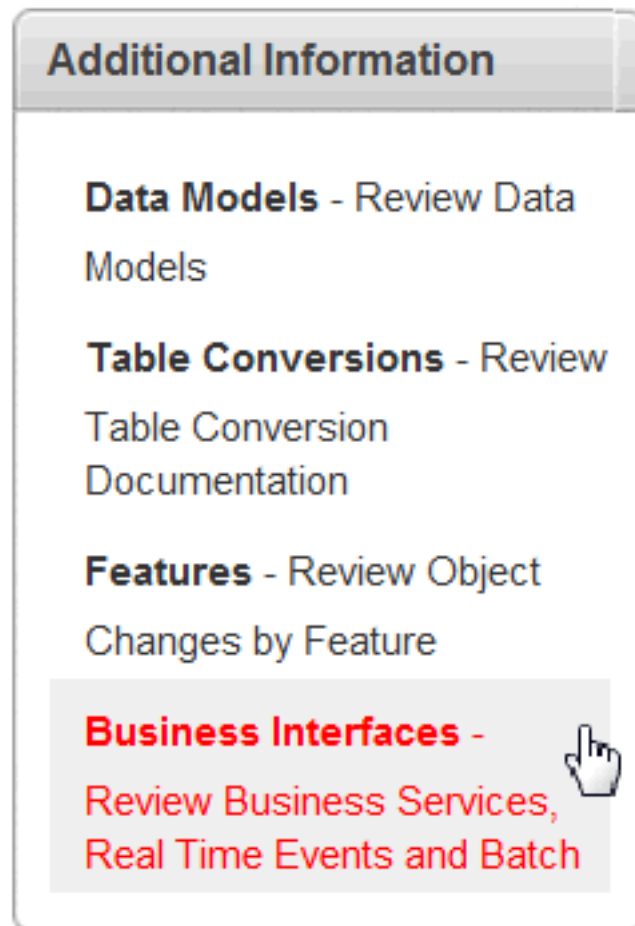
Item Alias	Item Name	Item Type	Item Size
ACTN	cActionCode	Character	1
AN8	mnAddressNumber	Numeric	8
CO	szCompany	String	5
ALPH	szNameAlpha	String	40
MCU	szBusinessUnit	String	12
AT1	szSearchType	String	3
BADT	cShipToSoldToFlag	Character	1
SIC	szStandardIndustryCode	String	10
RMK	szRemark	String	30
LNGP	szLanguageCode	String	2
POPN	szCreatedBy	String	10
MLNM	szMailingName	String	40
ADD1	szAddressLine1	String	40
ADD2	szAddressLine2	String	40
ADD3	szAddressLine3	String	40

1 - 15 ▶

Accessing Technical Catalog Information for Batch Programs

To access technical information in the Oracle Technical Catalog:

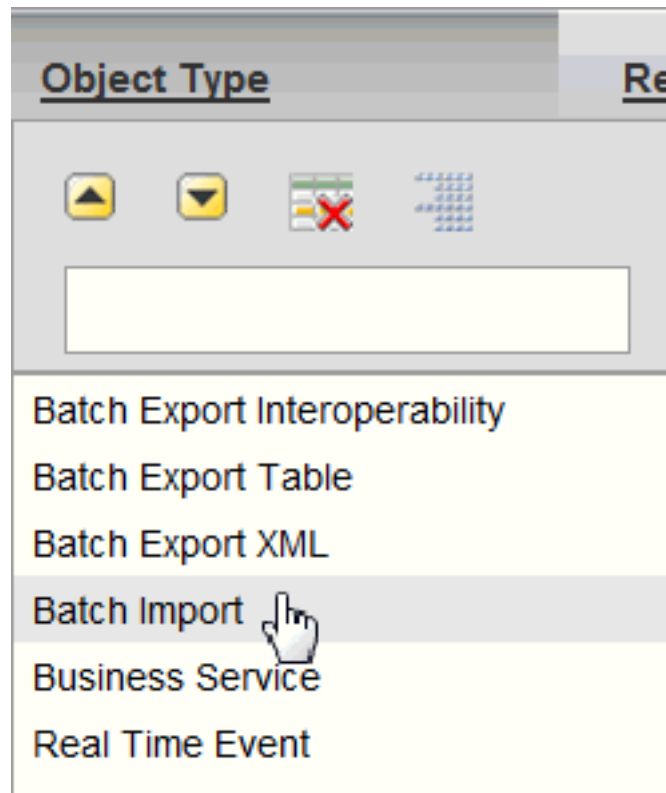
1. Log in to the Oracle Technical Catalog, and accept the user agreement notice. The catalog is located at:
<https://apex.oracle.com/pls/apex/f?p=50180>
2. From the Oracle Technical Catalog Home screen, select JD Edwards EnterpriseOne.
3. In the Additional Information section, select click the Business Interfaces link.



4. To sort and search the data in the detail area of the form, you can click on any of the column headings and use the text search field to narrow your displayed results, or use the sort arrows to display the data in ascending or descending order.

The following example shows the search and sort feature that appears when you click the Object Type heading. There are 4 batch program object types you can search or sort on:

- o Batch Import
- o Batch Export Interoperability
- o Batch Export Table
- o Batch Export XML



- To search for a specific batch program, enter the object ID in the search field.

Q- R011110Z

Business Object	Object Name	Object Description	Object Type	Release	System Code	Module	Component
A/P Payment	RTRAOUT	Accounts Payable Payment	Real Time Event	9.1 Update 2	04	Accounts Payable	Financial Management
A/P Payment	F0413Z1	F0413 Interoperability Table	Batch Export Interoperability	9.1 Update 2	04	Accounts Payable	Financial Management
A/P Payment	F0414Z1	F0414 Interoperability Table	Batch Export Interoperability	9.1 Update 2	04	Accounts Payable	Financial Management
A/P Payment	R47058	Update EDI Transaction as Sent	Batch Export Table	9.1 Update 2	04	Accounts Payable	Financial Management
A/P Voucher	JP040000	Accounts Payable Manager	Business Service	9.1 Update 2	04	Accounts Payable	Financial Management
Address Book Master	R01010Z	Address Book Batch Upload	Batch Import	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	JP010000	Address Book Manager	Business Service	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTABOUT	Address Book Master	Real Time Event	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	RTABOUT2	Address Book Master Version 2	Real Time Event	9.1 Update 2	01	Address Book	System Foundation
Address Book Master	R011110Z	Who's Who Batch Upload	Batch Import	9.1 Update 2	01	Address Book	System Foundation
Company Information	R0010Z0	Company Batch Outbound	Batch Export XML	9.1 Update 2	09	General Ledger	Financial Management

- To review the details of the tables used by the batch program, click the text in the Object Name column.

JD Edwards EnterpriseOne Business Interfaces

Q-

Row text contains 'R011110Z'

Business Object	Object Name	Object Description	Object Type	Release	System Code	Module	Component
Address Book Master	R011110Z	Who's Who Batch Upload	Batch Import	9.1 Update 2	01	Address Book	System Foundation

7. The system displays the tables used by the batch program, along with the columns in each of the tables.

Note:

For batch import programs, you will see details of the table the import program is pulling data from.

For batch export interoperability and batch export table programs, you will see information about the tables the program is putting data into.

For batch export XML programs, you will see limited information about the tables used by the program, but you will have access to .xsl and .xsd files to review schema details of the XML that is generated by the program.

The screenshot shows the JD Edwards EnterpriseOne Business Interfaces Batch Import Export screen. At the top, there are breadcrumb navigation links: JD Edwards EnterpriseOne > Business Interfaces > Batch Import Export. Below the breadcrumbs is a search bar with a magnifying glass icon, a 'Go' button, an 'Actions' dropdown menu, and a red '< Back' button. The main content is a table with the following columns: Table Name, Table Description, Column Alias, Column Description, Column Type, and Column Size. The table lists 15 columns for the F01112Z1 table, all with the description 'Batch - Related Person'. The columns and their details are as follows:

Table Name	Table Description	Column Alias	Column Description	Column Type	Column Size
F01112Z1	Batch - Related Person	EDUS	EDI - User ID	String	10
F01112Z1	Batch - Related Person	EDBT	EDI - Batch Number	String	15
F01112Z1	Batch - Related Person	EDTN	EDI - Transaction Number	String	22
F01112Z1	Batch - Related Person	EDLN	EDI - Line Number	Numeric	7
F01112Z1	Batch - Related Person	EDCT	EDI - Document Type	String	2
F01112Z1	Batch - Related Person	TYTN	Type - Transaction	String	8
F01112Z1	Batch - Related Person	EDFT	EDI - Translation Format	String	10
F01112Z1	Batch - Related Person	EDDT	EDI - Transmission Date	Date	6
F01112Z1	Batch - Related Person	DRIN	Direction Indicator	Character	1
F01112Z1	Batch - Related Person	EDDL	EDI - Detail Lines Processed	Numeric	5
F01112Z1	Batch - Related Person	EDSP	EDI - Successfully Processed	Character	1
F01112Z1	Batch - Related Person	PNID	Trading Partner ID	String	15
F01112Z1	Batch - Related Person	TNAC	Transaction Action	String	2
F01112Z1	Batch - Related Person	AN8	Address Number	Numeric	8
F01112Z1	Batch - Related Person	IDLN	Who's Who Line Number - ID	Numeric	5

At the bottom right of the table, there is a pagination indicator '1 - 15' with a right-pointing arrow.

Accessing Javadoc for Business Services

This section provides an overview of Javadoc for business services and discusses how to:

- Access Javadoc.
- View value object and field information.

- View the uses of an object in Javadoc.
- View hierarchy information.

Understanding Javadoc for Business Services

Javadoc is a tool that parses the declarations and documentation comments in a set of Java source files and produces a corresponding set of HTML pages. These pages describe the public and protected classes, nested classes, interfaces, constructors, methods, and fields. Javadoc generates the Application Programming Interface (API) documentation for the JD Edwards EnterpriseOne business services source files.

Javadoc is installed automatically when you install the JD Edwards EnterpriseOne system. You access Javadoc by navigating to the javadoc folder that is located in your JD Edwards EnterpriseOne install directory. This is an example of the path you would follow to access javadoc:

```
C:\B9\STAGINGA\java\javadoc
```

Note: Javadoc generates several main pages of documentation, each of which can be launched by double clicking on the corresponding html file in the javadoc folder. This documentation provides instructions for accessing and navigating through the API documentation by launching the overview-summary.html file. If you choose to access the documentation by launching a different file, the instructions might be slightly different.

Javadoc is organized into several main pages of documentation, each of which is described in this table:

Page Title	Description
Overview	The overview page is the main page of API documentation. This page contains a list of all of the available packages and a link to each package.
Package	Each package has its own page that contains a list of each class and interface. The page also contains a summary of each class and interface and provides a link to each.
Class	Each class, interface, nested class and nested interface has its own page. Each page contains a brief description of the class/interface. The page also includes summary and detail tables for field, constructor and method information. The page also includes information about inherited methods.
Tree	Each package has a tree or hierarchy page. The information that appears in the Tree page depends on the page from which the Tree option is selected. Clicking the Tree option from the overview page displays the hierarchy for all packages. Clicking the Tree option from a particular package, class or interface page displays the hierarchy for only that package.
Index	The index page contains an alphabetical list of all classes, interfaces, constructors, methods and fields. You access the index by clicking the Index link in the navigation bar from any page.
Help	The help page provides general instructions about how to use javadoc and overview information about how the documentation is organized. You access the help page by clicking the Help link in the navigation bar from any page.

Display and Navigation Options

To navigate through the Javadoc documentation, you can click on any of the package, class, method, value object or field names that are live links. When you click on these links a new page opens with additional detail about that object. You can also use the links in the navigation bar at the top of each page. The standard navigation bar includes these links:

- Overview
- Package
- Class
- Use
- Tree
- Deprecated
- Index
- Help

All javadoc pages are available with or without HTML frames. To view the page with or without frames, click on one of these options at the top of any page:

- Frames
- No Frames

You can also use the previous and next options at the top of each page to view the next package, class, or interface. Depending on the page you are on, the names of the options change. For example, when you are viewing a package, you can click NEXT PACKAGE or PREV PACKAGE to navigate to another package.

Package Naming Conventions

To access javadoc for a specific business service, you must locate the appropriate package. The naming convention for JD Edwards EnterpriseOne business service packages is:

```
oracle.e1.bssv.<OMW object name>
```

In package naming, objects can be published business services or business services. Published business service object names begin with JP followed by a numeric identification code. For example, the AddressBookManager object name is JP010000. Business service object names begin with J followed by a numeric identification code. For example, the processAddressBook business service object name is J0100001. Therefore, to access javadoc for the AddressBookManager business service, you would select one of these packages from the overview page:

- `oracle.e1.bssv.JP010000`

Selecting this package brings you to the main package page for the AddressBookManager business service.

- `oracle.e1.bssv.JP010000.valueobject`

Selecting this package brings you directly to a page that lists and describes each value object that is associated with the AddressBookManager business service.

Note: Javadoc for several reference implementations has also been provided. Reference implementation package names include the letter *R* after the *J* or *JP*.

Uses of a Business Service Object

You can also use javadoc to determine where a business service or related object is used. Each documented package, class and interface has its own Use page. The page describes what packages, classes, methods, constructors and fields use any part of the selected class or package. You access the Use page for an object by clicking the Use link in the navigation bar.

For example, if you are viewing class ABC, and you click on Use, the Use page for class ABC displays:

- Subclasses of ABC.
- Fields declared as ABC.
- Methods that return ABC.
- Methods and constructors with parameter type ABC.

Class Hierarchy

You can use Javadoc to display the hierarchy of all packages, or of individual packages, classes or interfaces. You use the Tree link in the navigation bar to access hierarchy information. If you click the Tree link from the main overview page, the system displays a hierarchical view of all packages along with a hierarchical view of each package.

Alternatively, you can click the Tree link from any package, class or interface page to view the hierarchy for only that package.

Classes are organized by inheritance structure, starting with `java.lang.Object`. The interfaces do not inherit from `java.lang.Object`.

Prerequisite

Before you can access Javadoc, you must install your JD Edwards EnterpriseOne system and identify your install directory. Contact your system administrator for assistance.

Accessing Javadoc

Access the Javadoc folder in your JD Edwards EnterpriseOne install directory and double click on the overview-summary.html file.

Overview Package Class Use **Tree** **Deprecated** **Index** **Help**

[PREV](#) [NEXT](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

EnterpriseOne
Service Business Functions

EnterpriseOne Business Service (BSSV) API Specification

Packages	
oracle.e1.bssv.J0100001	
oracle.e1.bssv.J0100001.valueobject	
oracle.e1.bssv.J0100002	
oracle.e1.bssv.J0100002.valueobject	
oracle.e1.bssv.J0100003	
oracle.e1.bssv.J0100003.valueobject	
oracle.e1.bssv.J0100004	
oracle.e1.bssv.J0100004.valueobject	
oracle.e1.bssv.J0100005	
oracle.e1.bssv.J0100005.valueobject	
oracle.e1.bssv.J0100006	
oracle.e1.bssv.J0100006.valueobject	
oracle.e1.bssv.J0100007	
oracle.e1.bssv.J0100007.valueobject	
oracle.e1.bssv.J0100008	
oracle.e1.bssv.J0100008.valueobject	
oracle.e1.bssv.J0100021	
oracle.e1.bssv.J0100021.valueobject	
oracle.e1.bssv.J0100022	

Your web browser opens the main overview page. To view information about a specific business service, click on the corresponding package name. For example, to view information about the AddressBookManager business service, click on package oracle.e1.bssv.JP010000.

The web browser displays the package page, which includes the Class Summary table for the selected package.

Overview	Package	Class	Use Tree	Deprecated	Index	Help	<i>EnterpriseOne Service Business Functions</i>	
PREV PACKAGE	NEXT PACKAGE				FRAMES	NO FRAMES	All Classes	

Package oracle.e1.bssv.JP010000

Interface Summary	
Oracle	E1
SBF	SEI
PkgBldFile	AddressBookManager

Class Summary	
AddressBookManager	The AddressBook Published Business Service shall manage the execution of the following SBFs: - Process Address Book - Process Contact - Get Address Book - Get Contact

Review the information in the Class Summary table and click on the link in the table to view field, constructor and method summary and detail information, along with inherited method information for the business service.

Note: You can access value object information by clicking on the links in the method summary table. For additional information about viewing value objects, proceed to the next task in this section.

[Overview](#) [Package](#) **[Class](#)** [Use Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
[SUMMARY: NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

EnterpriseOne
Service Business Functions

oracle.e1.bssv.JP010000

Class AddressBookManager

[java.lang.Object](#)

```
└ oracle.e1.bssvfoundation.base.BusinessService
  └ oracle.e1.bssvfoundation.base.PublishedBusinessService
    └ oracle.e1.bssv.JP010000.AddressBookManager
```

All Implemented Interfaces:

[ServiceLifecycle](#)

```
public class AddressBookManager
extends oracle.e1.bssvfoundation.base.PublishedBusinessService
```

The AddressBook Published Business Service shall manage the execution of the following SBFs: - Process Address Book - Process Contact - Get Address Book - Get Contact

Constructor Summary

[AddressBookManager](#) ()
Published Business Service Public Constructor

Method Summary

ShowAddressBook	getAddressBook (GetAddressBook vo) Published method for getAddressBook This exposed method Acts as wrapper method, passing null context and null connection, will call protected Published Business ServiceMethodName.
protected ShowAddressBook	getAddressBook (oracle.e1.bssvfoundation.base.IContext context, oracle.e1.bssvfoundation.connection.IConnection connection, GetAddressBook vo) Protected method for AddressBookManager Published Business Service.
ShowContact	getContact (GetContact vo) Published method for Get Contact Acts as wrapper method, passing null context and null connection, will call protected getContact.
protected ShowContact	getContact (oracle.e1.bssvfoundation.base.IContext context, oracle.e1.bssvfoundation.connection.IConnection connection, GetContact vo) Protected method for extends Published Business Service.
protected ConfirmProcessAddressBook	processAddressBook (oracle.e1.bssvfoundation.base.IContext context, oracle.e1.bssvfoundation.connection.IConnection connection, ProcessAddressBook vo) Protected method for AddressBookManager Published Business Service.
ConfirmProcessAddressBook	processAddressBook (ProcessAddressBook vo) Published method for process address book.
protected ConfirmProcessContact	processContact (oracle.e1.bssvfoundation.base.IContext context, oracle.e1.bssvfoundation.connection.IConnection connection, ProcessContact vo) Protected method for AddressBookManager Published Business Service.
ConfirmProcessContact	processContact (ProcessContact vo) Published method for process contact.

Methods inherited from class oracle.e1.bssvfoundation.base.PublishedBusinessService

[close](#), [destroy](#), [finishPublishedMethod](#), [init](#), [startPublishedMethod](#), [startPublishedMethod](#)

Methods inherited from class oracle.e1.bssvfoundation.base.BusinessService

[finishInternalMethod](#), [startInternalMethod](#)

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

AddressBookManager

```
public AddressBookManager()
```

Published Business Service Public Constructor

Method Detail

processAddressBook

```
public ConfirmProcessAddressBook processAddressBook(ProcessAddressBook vo)  
throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Published method for process address book. processAddressBook method shall process adds, changes and deletes of Address Book information in EnterpriseOne Acts as wrapper method, passing null context and null connection, will call protected processAddressBook.

Parameters:

vo - the value object representing input data for

Returns:

confirmVO the response data from the business process

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

processAddressBook

```
protected ConfirmProcessAddressBook processAddressBook (oracle.e1.bssvfoundation.base.IContext context,  
                                                    oracle.e1.bssvfoundation.connection.IConnection connection,  
                                                    ProcessAddressBook vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Protected method for AddressBookManager Published Business Service. processAddressBook will call make calls to the AddressBookProcessor BSSV class for completing business process.

Parameters:

vo - the value object representing input data for
context - conditionally provides the connection for the database operation and logging information
connection - can either be an explicit connection or null. If null the default connection is used.

Returns:

response value object is the data returned from the business process

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

processContact

```
public ConfirmProcessContact processContact (ProcessContact vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Published method for process contact. The processContact method shall process adds, changes and deletes of Contact information in EnterpriseOne. Acts as wrapper method, passing null context and null connection, will call protected processAddressBook.

Parameters:

vo - the value object representing input data for

Returns:

confirmVO the response data from the business process

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

processContact

```
protected ConfirmProcessContact processContact(oracle.e1.bssvfoundation.base.IContext context,  
                                             oracle.e1.bssvfoundation.connection.IConnection connection,  
                                             ProcessContact vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Protected method for AddressBookManager Published Business Service. processContact will make calls to the ContactProcessor BSSV class for completing business process.

Parameters:

vo - the value object representing input data for
context - conditionally provides the connection for the database operation and logging information
connection - can either be an explicit connection or null. If null the default connection is used.

Returns:

response value object is the data returned from the business process

Throws:

[oracle.e1.bssvfoundation.exception.BusinessServiceException](#)

getContact

```
public ShowContact getContact(GetContact vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Published method for Get Contact Acts as wrapper method, passing null context and null connection, will call protected getContact.

Parameters:

vo - the value object representing input data for

Returns:

confirmVO the response data from the business process

Throws:

[oracle.e1.bssvfoundation.exception.BusinessServiceException](#)

getContact

```
protected ShowContact getContact(oracle.e1.bssvfoundation.base.IContext context,  
                                oracle.e1.bssvfoundation.connection.IConnection connection,  
                                GetContact vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Protected method for extends Published Business Service. getContact will call make calls to the ContactQueryProcessor BSSV class for querying on AddressBook info

Parameters:

vo - the value object representing input data for querying on AddressBook info.
context - conditionally provides the connection for the database operation and logging information
connection - can either be an explicit connection or null. If null the default connection is used.

Returns:

confirmVO the response data from the business process AddressBook query.

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

getAddressBook

```
public ShowAddressBook getAddressBook(GetAddressBook vo)  
    throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Published method for getAddressBook This exposed method Acts as wrapper method, passing null context and null connection, will call protected Published Business ServiceMethodName.

Parameters:

vo - the value object representing input data for getAddressBook

Returns:

showVO the response data from the query of V0101XPI

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

getAddressBook

```
protected ShowAddressBook getAddressBook (oracle.e1.bssvfoundation.base.IContext context,  
                                           oracle.e1.bssvfoundation.connection.IConnection connection,  
                                           GetAddressBook vo)  
throws oracle.e1.bssvfoundation.exception.BusinessServiceException
```

Protected method for AddressBookManager Published Business Service. getAddressBook will call make calls to BSSV classes for completing business process.

Parameters:

vo - the value object representing input data for getAddressBook.
context - conditionally provides the connection for the database operation and logging information
connection - can either be an explicit connection or null. If null the default connection is used.

Returns:

confirmVO the response data from the business process getAddressBook.

Throws:

oracle.e1.bssvfoundation.exception.BusinessServiceException

Viewing Value Object and Field Information

Access the javadoc folder in your JD Edwards EnterpriseOne install directory and double click on the overview-summary.html file.

Click the link for the business service you want to review. For example, to view the value objects for the AddressBookManager business service, click oracle.e1.bssv.JP010000.valueobject.

Your web browser opens a new page that lists each value object that is associated with the business service.

[Overview](#) **Package** [Class](#) [Use Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV PACKAGE](#) [NEXT PACKAGE](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

EnterpriseOne
Service Business Functions

Package oracle.e1.bssv.JP010000.valueobject

Class Summary	
Address	Address is part of the Exposed Input value object
AddressBook	
AddressBookResult	
AlternateAddress	Exposed Input Value Object that represents an alternate address and the action to be performed for it.
AlternateAddressRecord	Exposed Input Value Object that represents an Alternate Address record
CategoryCodesAddressBook	
CategoryCodesContact	
Classifications	
ConfirmProcessAddressBook	TODO: Java Doc comments for Value Object here
ConfirmProcessContact	ConfirmProcessContact Output VO
Contact	Exposed input value object of Contact information
ContactRecord	Exposed Input Value Object that represents a contact record that will be returned to the consumer
ElectronicAddress	Published Value Object that represents a single electronic address and the action to be performed on it.
ElectronicAddressRecord	Exposed Input Value Object that represents an Electronic Address Record
GetAddress	
GetAddressBook	
GetContact	Exposed Input Value Object that provides input fields used to query the contact table.
Parent	
PhoneNumber	Exposed Input Value Object that represents a single phone number and the action to be performed for it.
PhoneNumberRecord	Exposed Input Value Object that represents an Phone Number record
ProcessAddressBook	

Click on the value object that you want to view. The web browser displays a summary of the value object, along with a summary of each field that is included in the value object.

Overview Package Class Use Tree Deprecated Index Help
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

*EnterpriseOne
Service Business Functions*

oracle.e1.bssv.JP010000.valueobject

Class ContactRecord

[java.lang.Object](#)

- └ oracle.e1.bssv.foundation.base.ValueObject
 - └ [oracle.e1.bssv.JP010000.valueobject.GetContact](#)
 - └ oracle.e1.bssv.JP010000.valueobject.ContactRecord

All Implemented Interfaces:

[Serializable](#)

```
public class ContactRecord
extends GetContact
implements Serializable
```

Exposed Input Value Object that represents a contact record that will be returned to the consumer

See Also:

[Serialized Form](#)

Field Summary	
<code>private AlternateAddressRecord []</code>	<code>alternateAddress</code> Alternate Address
<code>private ElectronicAddressRecord []</code>	<code>electronicAddress</code> Entity
<code>private PhoneNumberRecord[]</code>	<code>phone</code> Phone

Constructor Summary
<code>ContactRecord()</code> Default public constructor for instantiating: ContactRecord

Method Summary	
AlternateAddressRecord []	getAlternateAddress ()
AlternateAddressRecord	getAlternateAddress (int i)
ElectronicAddressRecord []	getElectronicAddress ()
ElectronicAddressRecord	getElectronicAddress (int i)
PhoneNumberRecord []	getPhone () <small>class in oracle.e1.bssv.JP010000.valueobject</small>
PhoneNumberRecord	getPhone (int i)
void	setAlternateAddress (AlternateAddressRecord [] alternateAddress)
void	setAlternateAddress (int i, AlternateAddressRecord alternateAddress)
void	setElectronicAddress (ElectronicAddressRecord [] electronicAddress)
void	setElectronicAddress (int i, ElectronicAddressRecord electronicAddress)
void	setPhone (int i, PhoneNumberRecord phone)
void	setPhone (PhoneNumberRecord [] phone)

Methods inherited from class oracle.e1.bssv.JP010000.valueobject.[GetContact](#)

[getCategoryCodesContact](#), [getContactID](#), [getContactTitle](#), [getContactTypeCode](#), [getDayOfBirth](#), [getDescription1](#), [getDisplaySequence](#), [getEntity](#), [getEntityNameContact](#), [getEntityNameSecondary](#), [getFunctionCode](#), [getGenderCode](#), [getGiveName](#), [getMailingNameSecondary](#), [getMiddleName](#), [getMonthOfBirth](#), [getNameMailing](#), [getNickName](#), [getPreferredContactMethod](#), [getPrimaryContactCode](#), [getRemark](#), [getSalutationName](#), [getSurname](#), [getTypeCode](#), [getYearOfBirth](#), [setCategoryCodesContact](#), [setContactID](#), [setContactID](#), [setContactTitle](#), [setContactTypeCode](#), [setDayDateOfBirth](#), [setDayOfBirth](#), [setDescription1](#), [setDisplaySequence](#), [setDisplaySequence](#), [setEntity](#), [setEntityNameContact](#), [setEntityNameSecondary](#), [setFunctionCode](#), [setGenderCode](#), [setGiveName](#), [setMailingNameSecondary](#), [setMiddleName](#), [setMonthDateOfBirth](#), [setMonthOfBirth](#), [setNameMailing](#), [setNickName](#), [setPreferredContactMethod](#), [setPrimaryContactCode](#), [setRemark](#), [setSalutationName](#), [setSurname](#), [setTypeCode](#), [setYearDateOfBirth](#), [setYearOfBirth](#)

Methods inherited from class oracle.e1.bssv.foundation.base.ValueObject

[toString](#), [transformBoolean01](#), [transformBooleanYN](#), [transformToBoolean](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

alternateAddress

```
private AlternateAddressRecord[] alternateAddress
```

Alternate Address

This object contains and array of alternate addresses

phone

```
private PhoneNumberRecord[] phone
```

Phone

This object contains and array of Phones

electronicAddress

```
private ElectronicAddressRecord[] electronicAddress
```

Entity

`class in oracle.e1.bssv.JP010000.valueobject`

This object contains and array of entities

Constructor Detail

ContactRecord

```
public ContactRecord()
```

Default public constructor for instantiating: ContactRecord

Method Detail

setAlternateAddress

```
public void setAlternateAddress(AlternateAddressRecord[] alternateAddress)
```

setAlternateAddress

```
public void setAlternateAddress(int i,  
                               AlternateAddressRecord alternateAddress)
```

getAlternateAddress

```
public AlternateAddressRecord[] getAlternateAddress()
```

getAlternateAddress

```
public AlternateAddressRecord getAlternateAddress(int i)
```

setPhone

```
public void setPhone(PhoneNumberRecord[] phone)
```

setPhone

```
public void setPhone(int i,  
    PhoneNumberRecord phone)
```

getPhone

```
public PhoneNumberRecord[] getPhone()
```

getPhone

```
public PhoneNumberRecord getPhone(int i)
```

setElectronicAddress

```
public void setElectronicAddress(ElectronicAddressRecord[] electronicAddress)
```

setElectronicAddress

```
public void setElectronicAddress(int i,  
    ElectronicAddressRecord electronicAddress)
```

To view detailed information about a field or method, click on the field or method name link.

Note: When viewing fields, the field names are in alphabetical order in the Field Summary table. The Field Detail table lists the fields in the order in which they appear in the source code. This preserves the logical groupings that were established by the programmer.

Class AlternateAddressRecord

[java.lang.Object](#)
└ oracle.e1.bssvfoundation.base.ValueObject
└ oracle.e1.bssv.JP010000.valueobject.AlternateAddressRecord

All Implemented Interfaces:

[Serializable](#)

```
public class AlternateAddressRecord  
extends oracle.e1.bssvfoundation.base.ValueObject  
implements Serializable
```

Exposed Input Value Object that represents an Alternate Address record

See Also:

[Serialized Form](#)

Field Summary	
private String	addressLine1 Address Line 1
private String	addressLine2 Address Line 2
private String	addressLine3 Address Line 3
private String	addressLine4 Address Line 4
private String	city City
private String	contactAddressTypeCode Type - Address

Field Detail

dateEffective

private [Calendar](#) dateEffective

Date - Beginning Effective

The date on which an address, item, transaction, or table record becomes active. The meaning of this field differs, depending on the program. For example, the effective date could represent the following:

- o When a change of address becomes effective
- o When a lease becomes effective
- o When a price becomes effective
- o When the currency exchange rate becomes effective
- o When a tax rate becomes effective

contactAddressTypeCode

private [String](#) contactAddressTypeCode

Type - Address

A user defined code (01/AT) that identifies the type of address, such as a home address or an office address.

EnterpriseOne Key Field: true
EnterpriseOne Alias: ATYPE
EnterpriseOne field length: 5

Viewing the Uses of an Object in Javadoc

Using any of the previous tasks in this section, access the object on which you want to inquire. Click the Use link in the navigation bar on the Javadoc page.

[Overview](#) [Package](#) [Class](#) [Use](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV](#) [NEXT](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

EnterpriseOne
Service Business Functions

Uses of Package `oracle.e1.bssv.JP010000.valueobject`

Packages that use `oracle.e1.bssv.JP010000.valueobject`

oracle.e1.bssv.J0100001.valueobject	
oracle.e1.bssv.J0100002.valueobject	
oracle.e1.bssv.J0100003.valueobject	
oracle.e1.bssv.J0100004.valueobject	
oracle.e1.bssv.J0100005.valueobject	
oracle.e1.bssv.J0100006.valueobject	
oracle.e1.bssv.J0100024.valueobject	
oracle.e1.bssv.JP010000	
oracle.e1.bssv.JP010000.valueobject	
oracle.e1.bssv.JP010020.valueobject	

Classes in `oracle.e1.bssv.JP010000.valueobject` used by `oracle.e1.bssv.J0100001.valueobject`

[ProcessAddressBook](#)

Classes in `oracle.e1.bssv.JP010000.valueobject` used by `oracle.e1.bssv.J0100002.valueobject`

[GetAddressBook](#)

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.J0100003.valueobject](#)

[ProcessContact](#)

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.J0100004.valueobject](#)

[GetContact](#)

Exposed Input Value Object that provides input fields used to query the contact table.

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.J0100005.valueobject](#)

[PhoneNumber](#)

Exposed Input Value Object that represents a single phone number and the action to be performed for it.

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.J0100006.valueobject](#)

[ElectronicAddress](#)

Published Value Object that represents a single electronic address and the action to be performed on it.

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.J0100024.valueobject](#)

[AlternateAddress](#)

Exposed Input Value Object that represents an alternate address and the action to be performed for it.

Classes in oracle.e1.bssv.JP010000.valueobject used by oracle.e1.bssv.JP010000
ConfirmProcessAddressBook TODO: Java Doc comments for Value Object here
ConfirmProcessContact ConfirmProcessContact Output VO
GetAddressBook
GetContact Exposed Input Value Object that provides input fields used to query the contact table.
ProcessAddressBook
ProcessContact
ShowAddressBook
ShowContact ShowContact Output VO

Classes in <code>oracle.e1.bssv.JP010000.valueobject</code> used by <code>oracle.e1.bssv.JP010000.valueobject</code>	
<u>Address</u>	Address is part of the Exposed Input value object
<u>AddressBook</u>	
<u>AddressBookResult</u>	
<u>AlternateAddress</u>	Exposed Input Value Object that represents an alternate address and the action to be performed for it.
<u>AlternateAddressRecord</u>	Exposed Input Value Object that represents an Alternate Address record
<u>CategoryCodesAddressBook</u>	
<u>CategoryCodesContact</u>	
<u>Classifications</u>	
<u>Contact</u>	Exposed input value object of Contact information
<u>ContactRecord</u>	Exposed Input Value Object that represents a contact record that will be returned to the consumer
<u>ElectronicAddress</u>	Published Value Object that represents a single electronic address and the action to be performed on it.
<u>ElectronicAddressRecord</u>	Exposed Input Value Object that represents an Electronic Address Record
<u>GetAddress</u>	
<u>GetContact</u>	Exposed Input Value Object that provides input fields used to query the contact table.

Classes in [oracle.e1.bssv.JP010000.valueobject](#) used by [oracle.e1.bssv.JP010020.valueobject](#)

[ConfirmProcessAddressBook](#)

TODO: Java Doc comments for Value Object here

[ElectronicAddress](#)

Published Value Object that represents a single electronic address and the action to be performed on it.

[ElectronicAddressRecord](#)

Exposed Input Value Object that represents an Electronic Address Record

[PhoneNumber](#)

Exposed Input Value Object that represents a single phone number and the action to be performed for it.

[PhoneNumberRecord](#)

Exposed Input Value Object that represents an Phone Number record

[ProcessAddressBook](#)

[Processing](#)

Viewing Hierarchy Information

Access the javadoc folder in your JD Edwards EnterpriseOne install directory and double click on the overview-summary.html file. From the main overview page, click the Tree link in the navigation bar. The system displays the hierarchy of all packages.

Note: You can access hierarchy information for a selected package by clicking the Tree link from any package, class or interface page.

[Overview](#) [Package Class Use](#) **[Tree](#)** [Deprecated](#) [Index](#) [Help](#)

[PREV](#) [NEXT](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

EnterpriseOne
Service Business Functions

Hierarchy For All Packages

Package Hierarchies:

[oracle.e1.bssv.J0100001](#), [oracle.e1.bssv.J0100001.valueobject](#), [oracle.e1.bssv.J0100002](#), [oracle.e1.bssv.J0100002.valueobject](#), [oracle.e1.bssv.J0100003](#),
[oracle.e1.bssv.J0100003.valueobject](#), [oracle.e1.bssv.J0100004](#), [oracle.e1.bssv.J0100004.valueobject](#), [oracle.e1.bssv.J0100005](#),
[oracle.e1.bssv.J0100005.valueobject](#), [oracle.e1.bssv.J0100006](#), [oracle.e1.bssv.J0100006.valueobject](#), [oracle.e1.bssv.J0100007](#),
[oracle.e1.bssv.J0100007.valueobject](#), [oracle.e1.bssv.J0100008](#), [oracle.e1.bssv.J0100008.valueobject](#), [oracle.e1.bssv.J0100021](#),
[oracle.e1.bssv.J0100021.valueobject](#), [oracle.e1.bssv.J0100022](#), [oracle.e1.bssv.J0100022.valueobject](#), [oracle.e1.bssv.J0100023](#),
[oracle.e1.bssv.J0100023.valueobject](#), [oracle.e1.bssv.J0100024](#), [oracle.e1.bssv.J0100024.valueobject](#), [oracle.e1.bssv.J4200010](#),
[oracle.e1.bssv.J4200010.valueobject](#), [oracle.e1.bssv.J4200020](#), [oracle.e1.bssv.J4200020.valueobject](#), [oracle.e1.bssv.J4200030](#),
[oracle.e1.bssv.J4200030.valueobject](#), [oracle.e1.bssv.J4200040](#), [oracle.e1.bssv.J4200040.valueobject](#), [oracle.e1.bssv.J4200050](#),
[oracle.e1.bssv.J4200050.valueobject](#), [oracle.e1.bssv.J4200060](#), [oracle.e1.bssv.J4200060.valueobject](#), [oracle.e1.bssv.J4200070](#),
[oracle.e1.bssv.J4200070.valueobject](#), [oracle.e1.bssv.J4300010](#), [oracle.e1.bssv.J4300010.valueobject](#), [oracle.e1.bssv.J4300020](#),
[oracle.e1.bssv.J4300020.valueobject](#), [oracle.e1.bssv.J4300030](#), [oracle.e1.bssv.J4300030.valueobject](#), [oracle.e1.bssv.JP010000](#),
[oracle.e1.bssv.JP010000.valueobject](#), [oracle.e1.bssv.JP010020](#), [oracle.e1.bssv.JP010020.valueobject](#), [oracle.e1.bssv.JP420000](#),
[oracle.e1.bssv.JP420000.valueobject](#), [oracle.e1.bssv.JP430000](#), [oracle.e1.bssv.JP430000.valueobject](#), [oracle.e1.bssv.JPR01000](#),
[oracle.e1.bssv.JPR01000.valueobject](#), [oracle.e1.bssv.JPR01001](#), [oracle.e1.bssv.JPR01002](#), [oracle.e1.bssv.JPR01002.valueobject](#),
[oracle.e1.bssv.JPR01020](#), [oracle.e1.bssv.JPR01020.valueobject](#), [oracle.e1.bssv.JPRCUST0](#), [oracle.e1.bssv.JPRCUST1](#), [oracle.e1.bssv.JPRCUST2](#),
[oracle.e1.bssv.JR010010](#), [oracle.e1.bssv.JR010010.valueobject](#), [oracle.e1.bssv.JR010011](#), [oracle.e1.bssv.JR010020](#),
[oracle.e1.bssv.JR010020.valueobject](#), [oracle.e1.bssv.JR010023](#), [oracle.e1.bssv.JR010023.valueobject](#), [oracle.e1.bssv.JR010030](#),
[oracle.e1.bssv.JR010030.valueobject](#), [oracle.e1.bssv.JR010040](#), [oracle.e1.bssv.JR010040.valueobject](#), [oracle.e1.bssv.JR010050](#),
[oracle.e1.bssv.JR010050.valueobject](#), [oracle.e1.bssv.util.J0000010](#), [oracle.e1.bssv.util.J0000010.valueobject](#), [oracle.e1.bssv.util.J0000020](#),
[oracle.e1.bssv.util.J0100010](#), [oracle.e1.bssv.util.J0100010.valueobject](#), [oracle.e1.bssv.util.J0900010](#), [oracle.e1.bssv.util.J0900010.valueobject](#),
[oracle.e1.bssv.util.J4100010](#), [oracle.e1.bssv.util.J4100010.valueobject](#)

Class Hierarchy

- java.lang.[Object](#)
 - oracle.e1.bssvfoundation.base.BusinessService
 - oracle.e1.bssv.J0100001.[AddressBookProcessor](#)
 - oracle.e1.bssv.J0100002.[AddressBookQueryProcessor](#)
 - oracle.e1.bssv.J0100024.[AlternateAddressesProcessor](#)
 - oracle.e1.bssv.J0100003.[ContactProcessor](#)
 - oracle.e1.bssv.J0100004.[ContactQueryProcessor](#)
 - oracle.e1.bssv.J0100023.[CustomerCreditQueryProcessor](#)
 - oracle.e1.bssv.J4200040.[CustomerItemPriceQueryProcessor](#)
 - oracle.e1.bssv.J0100021.[CustomerProcessor](#)
 - oracle.e1.bssv.J0100022.[CustomerQueryProcessor](#)
 - oracle.e1.bssv.J0100006.[ElectronicAddressesProcessor](#)
 - oracle.e1.bssv.J0100008.[ElectronicAddressesQueryProcessor](#)
 - oracle.e1.bssv.util.J0100010.[EntityProcessor](#)
 - oracle.e1.bssv.util.J0900010.[GLAccountProcessor](#)
 - oracle.e1.bssv.util.J4100010.[InventoryItemIdProcessor](#)
 - oracle.e1.bssv.J4200060.[ItemListPriceQueryProcessor](#)
 - oracle.e1.bssv.J4200030.[ItemPriceAndAvailabilityQueryProcessor](#)
 - oracle.e1.bssv.util.J0000020.[NetChangeProcessor](#)
 - oracle.e1.bssv.J0100005.[PhonesProcessor](#)
 - oracle.e1.bssv.J0100007.[PhonesQueryProcessor](#)
 - oracle.e1.bssv.util.J0000010.[ProcessingVersionProcessor](#)
 - oracle.e1.bssvfoundation.base.PublishedBusinessService (implements javax.xml.rpc.server.[ServiceLifecycle](#))
 - oracle.e1.bssv.JP010000.[AddressBookManager](#)
 - oracle.e1.bssv.JP010020.[CustomerManager](#)
 - oracle.e1.bssv.JP430000.[ProcurementManager](#)
 - oracle.e1.bssv.JPR01000.[RI AddressBookManager](#)
 - oracle.e1.bssv.JPRCUST0.[RI CustomABManager](#)
 - oracle.e1.bssv.JPR01002.[RI AddressBookStagingManager](#)
 - oracle.e1.bssv.JPRCUST2.[RI CustomAddressBookStagingManager](#)
 - oracle.e1.bssv.JPR01001.[RI AddressBookTransactionManager](#)
 - oracle.e1.bssv.JPRCUST1.[RI CustomAddressBookManager](#)
 - oracle.e1.bssv.JPR01020.[RI CustomerManager](#)
 - oracle.e1.bssv.JP420000.[SalesOrderManager](#)

Hierarchy For Package oracle.e1.bssv.JP010000.valueobject

Package Hierarchies:

[All Packages](#)

Class Hierarchy

- o java.lang.[Object](#)
 - o oracle.e1.bssv.foundation.base.ValueObject (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[Address](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[AddressBook](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[AddressBookResult](#)
 - o oracle.e1.bssv.JP010000.valueobject.[AlternateAddress](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[AlternateAddressRecord](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[CategoryCodesAddressBook](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[CategoryCodesContact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[Classifications](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[Contact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ElectronicAddress](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ElectronicAddressRecord](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[GetAddress](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[GetAddressBook](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[GetContact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ContactRecord](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.foundation.base.MessageValueObject (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ConfirmProcessAddressBook](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ConfirmProcessContact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ShowAddressBook](#)
 - o oracle.e1.bssv.JP010000.valueobject.[ShowContact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[Parent](#)
 - o oracle.e1.bssv.JP010000.valueobject.[PhoneNumber](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[PhoneNumberRecord](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ProcessAddressBook](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[ProcessContact](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[Processing](#) (implements java.io.[Serializable](#))
 - o oracle.e1.bssv.JP010000.valueobject.[RelatedAddress](#)

6 Business Services for Pre-Built Integrations

Overview of Business Services for Pre-Built Integrations

The business services that are documented in this guide have been built to enable you to create your own integration processes that meet your business requirements.

In addition to these business services, there are several other business services that have been built to support specific integrations between JD Edwards EnterpriseOne and other third-party systems. Though these business services were created as part of a specific solution, you can use them just as you would any other business service to meet the needs of your organization.

In most instances, those business services are documented in the solution-specific documentation that was created for the specific integration solution. This chapter lists all of the business services that are documented outside of this reference guide, provides you with a description of each service, and includes a reference to the detailed documentation for that service.

Agile Product Lifecycle Management Integration with JD Edwards EnterpriseOne

Agile Product Lifecycle Management (PLM) enables companies to manage individual product life cycles and to complete product portfolios and programs tied to product conception, design, launch, maturity, and phase-out. Primary users of Agile PLM are involved directly or indirectly in the product design. They manage content in Agile PLM that often directly affects processes managed in other enterprise systems, such as JD Edwards EnterpriseOne, Manufacturing Execution Systems (MES), Customer Relationship Management (CRM), and so forth.

The integration of Agile PLM and JD Edwards EnterpriseOne is designed to synchronize product content information between Agile Product Collaboration and JD Edwards EnterpriseOne.

The Agile PLM integration uses these business services:

Business Service	Description
EngineeringChangeOrderManager (JP300000)	The EngineeringChangeOrderManager Published Business Service (PBSSV) manages the processing of the following: <ul style="list-style-type: none"> EngineeringChangeOrdersProcessor (J3000010) EngineeringChangeOrderPartsListProcessor (J3000020) InventoryItemsProcessor (J4100020) InventoryItemBranchProcessor (J4100030)
EngineeringChangeOrdersProcessor (J3000010)	The EngineeringChangeOrderManager calls the EngineeringChangeOrdersProcessor from the provider when you create an ECO. For the Design to Release PIP, the external system is Agile PLM. You create an ECO in Agile PLM and releases the same to JD Edwards EnterpriseOne. This processor calls

Business Service	Description
	the create ECO (B3004100) business function to add a header record in F4801. This action generates an ECO number, which the system uses to create a BOM in JD Edwards EnterpriseOne.
EngineeringChangeOrderPartsListProcessor (J3000020)	The EngineeringChangeOrdersProcessor calls the EngineeringChangeOrderPartsListProcessor Web service. This processor calls the create ECO (B3004100) business function to add a parts list and related items in JD Edwards EnterpriseOne.
InventoryItemsProcessor (J4100020)	The EngineeringChangeOrdersProcessor calls the InventoryItemsProcessor Web service. This processor calls the F4101 ItemMasterAddition (B4101062) business function to add items in JD Edwards EnterpriseOne.
InventoryItemBranchProcessor (J4100030)	The EngineeringChangeOrdersProcessor calls the InventoryItemsProcessor Web service is called. This processor calls the F4102 ItemBranchAddition (B4101072) business function to add item branch records in JD Edwards EnterpriseOne.
IntegrationTimeStampManager (JP300010)	The IntegrationTimeStampManager PBSSV manages the processing of the ProcessIntegrationTimeStamp (J3000030) business service.
ProcessIntegrationTimeStamp (J3000030)	The IntegrationTimeStampManager PBSSV calls the ProcessIntegrationTimeStamp Web service. This processor adds and modifies the Integration Time Stamp table (F0095) with the last successful runtime.

For additional information about these business services, see *Oracle Design to Release Integration pack for Agile Product Lifecycle Management and JD Edwards EnterpriseOne Implementation Guide*, "Configuring the Oracle Design to Release Integration Pack for Agile PLM and JD Edwards EnterpriseOne."

Mobile Smartphone Applications

In previous releases of the software, EnterpriseOne offered several mobile smartphone applications. These applications are no longer available as of the current release. They have been replaced by more technologically advanced mobile applications. However, many of the business services that were created to support the mobile smartphone applications are still available for use, if needed.

Note: For additional information about any of the business services listed in this section, see *Business Services for Mobile Applications in the Release 9.1 version of the JD Edwards EnterpriseOne Applications Functionality for Mobile Devices Implementation Guide*.

This table lists the business services that were created for use with the Expense Management mobile smartphone application:

Business Service	Description
ExpenseReportManager (JP09E000)	This published business service manages the processing of these business service operations: <ul style="list-style-type: none"> ExpenseReportQueryProcessor (J09E0001)

Business Service	Description
	<ul style="list-style-type: none"> • ExpenseManagementEmployeeProfileQueryProcessor (J09E0002) • ExpenseReportWorkflowProcessor (J09E0003) • CreditCardTransactionProcessor (J09E0004) • ExpenseReportProcessor (J09E0005) • ExpenseReportRecordReservation (J09E0006) • getExpenseManagementPolicy (J09E0007) • ExpenseReportAttachmentManager (JP09E001) • ExpenseReportAttachmentsQueryProcessor (J09E0008), which contains the getExpenseReportAttachment method. • ExpenseReportAttachmentsProcessor (J09E0009), which contains the processExpenseReportAttachment method. • getExpenseManagementValuesList (J0000030), which is a method contained by the getUserDefined code operation, which is managed by the FoundationEnvironment published business service (J000000) • lookupBusinessUnits (J0000040), which is managed by the FoundationEnvironment published business service (J000000) • lookupCurrencyCodes (J0000060), which is managed by the FoundationEnvironment published business service (J000000) • lookupAddressBook (J0100009), which is managed by the ManageAddressBook published business service • lookupWorkOrders (J4800001, which is managed by the WorkOrderManager published business service (JP480000) • lookupJobCost (J5100001), which is managed by the JobCostManager published business service (JP510000)
ExpenseReportQueryProcessor (J09E0001)	<p>This operation manages the retrieval of expense report data for expense report entry, review, and approval processing on a mobile device. This operation contains these methods:</p> <ul style="list-style-type: none"> • getAllExpenseReports • getExpenseReportsToApprove • getExpenseReports • getExpenseReportHeaders
ExpenseManagementEmployeeProfileQuery (J09E0002)	<p>This operation contains the getExpenseManagementEmployeeProfile method that the system uses to retrieve the expense report profile and EnterpriseOne profile information that is associated with the user name or address number entered in the mobile device at sign on.</p>
ExpenseReportWorkflowProcessor (J09E0003)	<p>This operation is used to approve, reject, and submit expense reports from a mobile device, and contains these methods:</p> <ul style="list-style-type: none"> • approveExpenseReport • rejectExpenesReport • submitExpenseReport
CreditCardTransactionProcessor (J09E0004)	<p>This operation contains the getCreditCardTransactions method that the system uses to retrieve credit card transaction data from the EnterpriseOne database so that it can be added to an expense report that is created using a mobile device.</p>

Business Service	Description
ExpenseReportProcessor (J09E0005)	Use this operation to process expense report data that was entered on a mobile device in the EnterpriseOne database. This operation contains these methods: <ul style="list-style-type: none"> processExpenseReports processExpenseReportHeader processExpenseReportDetail
ExpenseReportRecordReservation (J09E0006)	Use this operation to reserve and release records in the EnterpriseOne database when you update the record. This operation contains these methods: <ul style="list-style-type: none"> reserveExpenseReport releaseExpenseReport
getExpenseManagementValuesList (J0000030)	This operation is managed by the Foundation Environment published business service. This operation contains the getUserDefinedCode method, which retrieves UDC values from the EnterpriseOne database. The system uses these values to process expense report data on the mobile device.
lookupBusinessUnits (J0000040)	This operation is managed by the FoundationEnvironment published business service. This operation retrieves business unit values from the EnterpriseOne database when you search for expense reports by business unit using a mobile device.
lookupCurrencyCodes (J0000060)	This operation is managed by the FoundationEnvironment published business service. This operation retrieves currency code values from the EnterpriseOne database when you search for valid currency codes when entering or editing expense report data on a mobile device.
lookupAddress Book (J0100009)	This operation is managed by the ManageAddressBook published business service. This operation is used to retrieve valid values from the EnterpriseOne database when you search for an expense report or for a manager, when working with expense reports on a mobile device.
lookupWorkOrders (J4800001)	This operation is managed by the WorkOrderManager published business service. This operation retrieves work order numbers from the EnterpriseOne database when you search for a valid work order using their mobile device during expense report entry.
lookupJobCost (J5100001)	This operation is managed by the JobCostManager published business service. This operation retrieves valid job cost data from the EnterpriseOne database when you search for a valid job cost using their mobile device during expense report entry.

This table lists the business services that were create for use by the Purchase Order Approval mobile smartphone application:

Business Service	Description
ProcurementManager (JP430000)	This published business service manages the processing of these mobile purchase order operations: <ul style="list-style-type: none"> PurchaseOrderApprovalQueryProcessor (J4300040) GetPurchaseOrderDetailForApprover (J4300050) PurchaseOrderEmployeeProfileQueryProcessor (J4300060) GetMobilePurchaseOrderCountsProcessor (J4300070)

Business Service	Description
	<ul style="list-style-type: none"> PurchaseOrderApproveRejectProcessor (J4300080).
PurchaseOrderApprovalQueryProcessor (J4300040)	This operation contains the <code>getPurchaseOrdersForApprover</code> method, which the system uses to retrieve purchase order header records for all orders that are currently awaiting approval by you.
GetPurchaseOrderDetailForApprover (J4300050).	This operation contains the <code>getPurchaseOrderDetailForApprover</code> method, which the system uses to retrieve purchase order detail records for all orders that are currently awaiting approval by you.
PurchaseOrderEmployeeProfileQueryProcessor (J4300060).	This operation contains the <code>getPurchaseOrderEmployeeProfile</code> method, and is used to retrieve the procurement profile, as well as additional EnterpriseOne profile information for the name or address number that you entered in the mobile device during signing on.
GetMobilePurchaseOrderCountsProcessor (J4300070).	This operation contains the <code>getPurchaseOrderCounts</code> method, which the system uses to retrieve the purchase order descriptions and counts for each order type that is currently awaiting approval by you.
PurchaseOrderApproveRejectProcessor (J4300080).	This operation contains the <code>processPurchaseOrderApproveReject</code> method, which the system uses to approve or reject a specific purchase order, and to update any remarks that you enter.

This table lists the business services that were created for use by the Sales Inquiries mobile smartphone application:

Business Service	Description
getUDCDescription (J0000050)	This operation is managed by the FoundationEnvironment published business service and retrieves valid document type values from user-defined code table 00/DT. The <code>getDocumentTypeDescription</code> method calls this operation.
InventoryManager (JP410000)	This published business service manages these operations, which are used by the mobile RSS Approval application: <ul style="list-style-type: none"> <code>getItemPrice</code> (J4100040) <code>getItemSearch</code> (J4100050) <code>getItemBranchSearch</code> (J4100060) <code>getCalculatedAvailability</code> (J4100007)
getItemPrice (J4100040)	Use this operation to retrieve item pricing information from the EnterpriseOne database. The operation uses the user-defined search criteria to retrieve records.
getItemSearch (J4100050)	Use this operation to retrieve valid item number values that can be used in the search criteria when searching for item price and availability records.
getItemBranchSearch (J4100060)	Use this operation to retrieve valid branch plant values that can be used in the search criteria when searching for item price and availability records.
getCalculatedAvailability (J4100007)	Use this operation to retrieve item availability records from the EnterpriseOne database. The operation uses the user-defined search criteria to retrieve records.

This table lists the business services that were created for use by the Mobile Service Time Entry smartphone application:

Operation/Method	Description
Service Time Card Manager published business service (JP170002)	This published business service manages the processing of the Web service operations listed below in this table.
getServiceTimeCards (J1700050)	Use this operation to retrieve service timecards for a specific person. You can apply additional filtering on specific search values.
getServiceOperationSequence (J1700060)	Use this operation to retrieve operation sequence, which is attached to a specific service order.
getServiceTimeCardProcessingOptions (J1700070)	Use this operation to retrieve the service timecard mobile processing options.
processServiceTimeCard (J1700080)	Use this operation to add, update, and delete a service timecard.

Direct Connect for Requisition Self Service

The Direct Connect feature in the JD Edwards Requisition Self Service system enables you to select goods and services from approved vendor websites using the JD Edwards EnterpriseOne Shopping Cart program, and add them to a purchase requisition. The JD Edwards EnterpriseOne system processes this requisition by using standard processing.

The Direct Connect solution uses these business services:

Business Service	Description
Direct Connect Authentication Processor (J43E0010)	The Direct Connect Authentication Processor business service (J43E0010) is a transaction service that invokes the HTTP service to connect to the vendor's website and authenticates your credentials.
Shopping Cart Processor (J43E0020)	After you access a vendor's website and select the products or services that you want to purchase, you complete the steps on the vendor's website to check out the products or services. When a you initiates the checkout process, the Shopping Cart Processor business service (J43E0020) is invoked. The Shopping Cart Processor is an inbound transaction business service that transfers your checkout data from the vendor's website to the JD Edwards EnterpriseOne database.

For additional information about these business services, see *"Setting Up Direct Connect and Purchase Order Dispatch."*

Global Order Promising Integration Using Web Service Callout

The Web Services Callout integration uses the GOP JAXRPC Processor (JC34A010) or the GOP JAXWS Processor (JC34A020) business services to consume the Global Order Promising Web service. When the system transmits sales order events from EnterpriseOne to Global Order Promising, the Global Order Promising business service converts the events into the XML format required by the Global Order Promising Server.

The Global Order Promising integration uses these business services:

Business Service	Description
GOP JAXRPC Processor (JC34A010)	<p>The GOP JAXRPC Processor business service (JC34A010) runs on the Oracle WebLogic Server (WLS) platform and reads the data stored in specific JD Edwards EnterpriseOne tables.</p> <p>The E1 Enterprise server collects the data required for determining promised delivery date from the JD Edwards EnterpriseOne tables and then calls the JC34A010 business service. The JC34A010 business service makes the necessary calls to the GOP Web methods and returns the promised dates received from GOP to the E1 Enterprise server.</p>
GOP JAXWS Processor (J34A020)	<p>The GOP JAXWS Processor business service (JC34A020) runs on the Oracle WebLogic Server (WLS) platform or the Web Application Server (WAS) platform, and reads the data stored in specific JD Edwards EnterpriseOne tables.</p> <p>The E1 Enterprise server collects the data required for determining promised delivery date from the JD Edwards EnterpriseOne tables, and then calls the JC34A020 business service. The JC34A020 business service makes the necessary calls to the GOP Web methods and returns the promised dates received from GOP to the E1 Enterprise server.</p>

Purchase Order Dispatch for Requisition Self Service

Many organizations and vendors require purchase orders before they complete a purchasing transaction. The Purchase Order (PO) Dispatch solution enables you to send purchase orders electronically from the JD Edwards EnterpriseOne system to the vendor.

The PO Dispatch solution uses this business service:

Business Service	Description
PO Dispatch Processor (J43E0030)	<p>The PO Dispatch Processor business service (J43E0030) transforms the data that is generated by the Purchase Order Print program (R43500) into cXML 1.2.016 OrderRequest schema format. The PO Dispatch Processor business service then posts that cXML message so that the vendor's website can receive it. The PO Dispatch Processor business service then receives the response from the vendor in cXML 1.2.016 OrderResponse schema format.</p>

For additional information about these business services, see *JD Edwards EnterpriseOne Applications Requisition Self Service Implementation Guide*, "Setting Up Direct Connect and Purchase Order Dispatch."

Rate Shopping Integration Using Web Service Callout

The Web Services Callout integration uses the Rate Shopping JAXWS Processor (JC49T020) business services to consume the Rate Inquiry Web service of the Oracle Transportation Management system. When accessing rate shopping from the Sale Order programs, P4210 or P42101, the business service converts the requested date, quantity, weight, and volume information into the XML format required by the Oracle Transportation Management rate inquiry Web service.

Business Service	Description
Rate Shopping JAXWS Processor (JC49T020)	The Rate Shopping JAXWS Processor business service (JC49T020) runs on the Weblogic Server platform or the Websphere Application Server (WAS) and receives the data that is passed from the Rate Shopping application (P49T10) through the call to the JC49T020 method getRateShoppingJAXWS. The data is loaded into the proxy classes generated from the Oracle Transportation Management WLS wsdl to build an XML message. The Rate Shopping JAXWS Processor business service consumes the Oracle Transportation Management Rate Inquiry Query (RIQ) Web service with the XML message and in turn receives the RIQ response from the Oracle Transportation Web service. The Rate Shopping JAXWS Processor business service converts the response received through the Oracle Transportation Management Web services proxy generated classes and returns the data to the Rate Shopping application.

For additional information about this business services, see *JD Edwards EnterpriseOne Applications Integration with Oracle Transportation Management Implementation Guide*, "Rate Inquiry Process Using Web Services Callout."

EU VAT ID Validation with VIES Using Web Service

You use the VAT Information Exchange System (VIES) Web service to enable quick and easy validation of VAT identification numbers throughout the European Union community. JD Edwards EnterpriseOne is set up to communicate with the VIES Web service, enabling you to validate VAT identification numbers for customers or suppliers directly in the JD Edwards EnterpriseOne system.

The aim of this business service is to enable you to run the Web service proxy on a weblogic server without having to run the migration tool.

This functionality runs on three supported applications servers:

- Weblogic Server (WLS)
- Websphere Application Server (WAS)
- Oracle Application Server (OAS)

The type of business service (BSSV) build that you are using or that you want to use with this service determines which business service object you must configure:

Business Service	Description
VIIES VAT ID Validation (WLS)-EU – 01 (J740105)	The VIIESAppServerOASorWLS business service (J740105) allows running the JAXRPC web service proxy on a weblogic server instead of OAS without having to run the migration tool.
VIIES VAT ID Validation JAXWS – EU-01 (J740120)	This business service allows running a JAXWS based proxy for both WLS and WAS.
VIIES VAT ID Validation – EU- 01 (J740102)	This business service allows running the Web service proxy on OAS/WAS.

The EU VAT ID validation solution uses these business services:

Server	Build Type	Business Service
Weblogic Server (WLS)	JAX-RPC with JDeveloper 11g	J740105
Weblogic Server (WLS) Note: For JD Edwards EnterpriseOne release Tools 9.1.2 or later.	JAX-WS with JDeveloper 11g	J740120
Websphere Application Server (WAS)	JAX-RPC with JDeveloper 10g & RAD	J740102
Websphere Application Server (WAS) Note: For JD Edwards EnterpriseOne Tools 9.1.2 release or later.	JAX-WS with JDeveloper 11g	J740120
Oracle Application Server (OAS) Note: Releases prior to JD Edwards EnterpriseOne Tools 9.1 use this option.	JAX-RPC with JDeveloper 10g	J740102

For additional information about this business service, see [JD Edwards EnterpriseOne Applications Tax Processing Implementation Guide](#)

Brazil NF-e Transmission Web Services

The Web Services Callout integration uses the NF-e Lot Processing Processor (J76B0021) business services to orchestrate calls to the BSSV consumer proxies, which in turn call the Web service of the Brazil fiscal authority.

It uses internal functions to read the input XML and convert it into the Simple Object Access Protocol (SOAP) message format required by the Web service, and invokes the corresponding Web service consumer proxy.

Business Service	Description
NfeRecepcao (JC76B001)	The NfeRecepcao service is called by the J76B0021 service to dispatch the NFe lot to the NfeRecepcaoLote2 Web service provided by SEFAZ (Brazilian fiscal authority), and to receive the receipt number or error code sent by SEFAZ.
NfeRetRecepcao (JC76B002)	The NfeRetRecepcao service returns a status message from SEFAZ.
NfeCancelamento (JC76B003)	The NfeCancelamento service is called by the J76B0021 service to dispatch the cancelled NFe numbers to the SEFAZ (Brazilian fiscal authority).
NfeInutilizacao (JC76B005)	The NfeInutilizacao service is called by the J76B0021 service to dispatch the unused or destroyed NFe numbers to the SEFAZ (Brazilian fiscal authority).

Note: *"Using the NFe Transmission Process" in the JD Edwards EnterpriseOne Applications Localizations for Brazil Implementation Guide . "Business Services for Brazil" in the JD Edwards EnterpriseOne Applications Localizations for Brazil Implementation Guide .*

7 JD Edwards EnterpriseOne Orchestrations

Understanding JD Edwards EnterpriseOne Orchestrations

The JD Edwards EnterpriseOne orchestrations are prebuilt processes that can transform data that is collected from external devices into actionable business processes within JD Edwards EnterpriseOne. The JD Edwards EnterpriseOne Orchestrator Studio processes these orchestrations to enable the immediate, real-time transformation of raw data into valuable data that can be used by the JD Edwards EnterpriseOne system.

An orchestration comprises the following components: service requests, rules, cross-references, and white lists. You define the metadata for each of these components in XML files, and the Orchestrator Studio uses the metadata in these XML files to perform AIS server calls that invoke JD Edwards EnterpriseOne transactions.

This chapter focuses on the predefined JD Edwards EnterpriseOne orchestrations that are available to address common use cases. You can use these orchestrations as examples to design your own orchestrations based on your business requirements.

Additional Orchestrations

The following prebuilt orchestrations are available:

- Add Condition-Based Maintenance Alert
- Update Meter Readings
- Update Equipment Location
- Kanban Check In
- Kanban Check Out
- Add Blend Operation

The Orchestrator Studio uses the input message defined for each of these orchestrations to make a service call to the AIS server which, in turn, invokes the respective JD Edwards EnterpriseOne application.

The AIS Server exposes each orchestration as an endpoint at `http://<server>:<host>/jderest/orchestrator/<orchestration name>`.

Note: *JD Edwards EnterpriseOne Tools Orchestrator Guide for Studio Version 8 and Prior*

Add Condition-Based Maintenance Alert

You can use the Add Condition-Based Maintenance Alert orchestration to process a condition-based maintenance alert from a sensor reading associated with an asset.

For example, a sensor attached to a piece of equipment to monitor its health may send an alert when the tolerance level is crossed. The Orchestrator Studio uses the Add Condition-Based Maintenance Alert orchestration metadata to process the data from the sensor reading and convert it into transaction-capable information in the JD Edwards EnterpriseOne system.

This table includes information that the Add Condition-Based Maintenance Alert orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Condition-Based Alerts Revision (P1311)	ZJDE0001	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_12C_AddCBMAAlert</code>

This is an example of an input message defined for the Add Condition-Based Maintenance Alert orchestration:

```
{
  "inputs": [
    {
      "name": "EquipmentNumber",
      "value": "34665"
    },
    {
      "name": "MeasurementLocation",
      "value": "EG-01"
    },
    {
      "name": "Description",
      "value": "Orchestration Test"
    },
    {
      "name": "AlertLevel",
      "value": "1"
    },
    {
      "name": "Date",
      "value": "062415"
    },
    {
      "name": "Time",
      "value": "13:15:01"
    },
    {
      "name": "NotificationRecipient",
      "value": "6002"
    },
    {
      "name": "NotificationStructureType",
      "value": "WF1"
    },
    {
      "name": "PlannedStartDate",
      "value": "062615"
    },
    {
      "name": "AutomatedResponseType",
      "value": "1"
    }
  ]
}
```



```

    }
  ]
}

```

Update Meter Readings

You can monitor meter readings on a piece of equipment and collect the data by using an IoT device. You can then use the Update Meter Readings orchestration to process the meter readings from the IoT device. For example, you can use meter readings to determine whether to initiate maintenance tasks for a machine or equipment based on the accumulated statistical units.

This table includes information that the Update Meter Readings orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Speed Meter Readings (P12120U)	ZJDE0001	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_13_UpdateMeterReadings</code>

This is an example of an input message defined for the Update Meter Readings orchestration:

```

{
  "inputs": [
    {
      "name": "EquipmentNumber",
      "value": "34665"
    },
    {
      "name": "OdometerNewReading",
      "value": "61"
    },
    {
      "name": "FuelMeterNewReading",
      "value": "10"
    },
    {
      "name": "HourMeterNewReading",
      "value": "94"
    },
    {
      "name": "Meter4NewReading",
      "value": "4"
    },
    {
      "name": "Meter5NewReading",
      "value": "6"
    },
    {
      "name": "Meter6NewReading",
      "value": "8"
    }
  ]
}

```

Update Equipment Location

You use the Update Equipment Location orchestration to update the location of equipment. This orchestration processes the data forwarded by a location sensor and updates the Equipment Master Location Details table (F17311).

For example, a service technician might need to know the physical location of equipment to quickly and easily locate the equipment that requires maintenance. In such a situation, you can use an IoT location sensor attached to the equipment to record the equipment's geopositioning data, including the latitude and longitude. The Update Equipment Location orchestration then processes the sensor data and updates the F17311 table.

This table includes information that the Update Equipment Location orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Equipment Master Address Location Revisions (P1704)	ZJDE0001	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_17C_UpdateEquipmentLocation</code>

This is an example of an input message defined for the Update Equipment Location orchestration:

```
{
  "inputs": [
    {
      "name": "EquipmentNumber",
      "value": "34665"
    },
    {
      "name": "CustomerNumber",
      "value": "4244"
    },
    {
      "name": "SiteNumber",
      "value": "4244"
    },
    {
      "name": "Remark",
      "value": "Orchestration Test"
    },
    {
      "name": "Latitude",
      "value": "39.632080"
    },
    {
      "name": "Longitude",
      "value": "-104.900376"
    }
  ]
}
```

Kanban Check In and Check Out

You can use the Kanban Check In and Check Out orchestrations to initiate kanban check-in and check-out transactions in the JD Edwards EnterpriseOne system.

You can use an IoT device to monitor the location of a kanban container or the inventory levels in a kanban container. For example, when a kanban container is removed from the consuming location indicating that it is empty, the Orchestrator Studio processes the data forwarded by the device by using the Kanban Check Out orchestration metadata, and initiates the check-out transaction in the Kanban Processing program (P3157). A kanban check-out transaction creates an order based on the source type and/or updates the kanban status.

Similarly, when replenishment occurs and the container is delivered to the consuming location, the Orchestrator Studio initiates the check-in transaction in the Kanban Processing program. A kanban check-in transaction performs an order completion or receipt and/or a transfer to the consuming location and kanban status update.

Note: This prebuilt orchestration is designed to support the one-phase kanban transfer method where the completion of the kanban and transfer of inventory are performed concurrently (Kanban Transfer Method in the Kanban Master = 1).

This table includes information that the Kanban Check Out orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Kanban Processing (P3157)	ZJDE0001	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_31_KanbanCheckOut</code>

Note: You must verify the following processing option values for the ZJDE0001 version of the Kanban Processing program:

- In the Mode tab, the processing option **1. Enter a '1' to set mode to Kanban Supply. If left blank, Kanban Consumption mode is assumed** is set to a blank value.
- In the Mode tab, the processing option **2. Enter a '1' to prompt the confirmation of a transaction** is set to a blank value.

This table includes information that the Kanban Check In orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Kanban Processing (P3157)	ZJDE0002	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_31_KanbanCheckIn</code>

Note: You must verify the following processing option values for the ZJDE0002 version of the Kanban Processing program:

- In the Mode tab, the processing option **1. Enter a '1' to set mode to Kanban Supply. If left blank, Kanban Consumption mode is assumed** is set to **1**.
- In the Mode tab, the processing option **2. Enter a '1' to prompt the confirmation of a transaction** is set to a blank value.
- In the Process tab, the processing options **2. Enter a '1' to perform a blind execution of Hours Quantities, 3. Enter a '1' to perform a blind execution of Material Issues, 4. Enter a '1' to perform a blind execution of Work Order Completions, 5. Enter a '1' to perform a blind execution of Shipment Confirmation, and 6. Enter a '1' to perform a blind execution of Inventory Transfers** are set to **1** to perform blind executions.

This is an example of an input message defined for the Kanban Check In and Check Out orchestrations:

```
{
  "inputs": [
    {
      "name": "KanbanID",
      "value": "664"
    },
    {
      "name": "CardNumber",
      "value": "1"
    }
  ]
}
```

Add Blend Operation

You use the Add Blend Operation orchestration to create a quality operation based on sensor readings. For example, when a sensor indicates that the temperature inside a wine tank has crossed the tolerance level, you can use the Add Blend Operation orchestration to process the sensor readings and trigger the creation of a quality operation in the JD Edwards EnterpriseOne system.

This table includes information that the Add Blend Operation orchestration uses to process IoT data:

Application Invoked	Application Version	Endpoint URL
Operation Search (P31B94)	ZJDE0001	<code>http://<server>:<host>/jderest/orchestrator/JDE_ORCH_31B_AddBlendOperation</code>

This is an example of an input message defined for the Add Blend Operation orchestration:

```
{
  "inputs": [
    {
      "name": "OperationID",
      "value": "QAV"
    }
  ]
}
```

```
    },  
    {  
      "name": "OperationDescription",  
      "value": "Orchestration Test"  
    },  
    {  
      "name": "Winery",  
      "value": "W10"  
    },  
    {  
      "name": "VesselNumber",  
      "value": "W10-1"  
    },  
    {  
      "name": "TestID",  
      "value": "BRIX"  
    },  
  ],  
}
```


8 ALM: Asset Master - Equipment Extensions

Asset Master - Equipment Extensions Overview

This chapter provides detailed information about the business interface components that are available for the Asset Master - Equipment Extensions business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The EquipmentManager published business service (JP17000) manages these business services:</p> <ul style="list-style-type: none"> • J1700001 (equipmentProcessor) • J1700002 (equipmentQueryProcessor)
Real-Time Events	<p>This list includes the real-time events for the Asset Master - Equipment Extensions business object:</p> <ul style="list-style-type: none"> • RTEQPOUT, which is a container event for RTEQPHDR. <p>This event is documented in the Equipment Location chapter. For detailed information, see Equipment Locations Real-Time Events.</p>
Batch Export Programs	<p>This list includes the batch export programs for the Asset Master - Equipment Extensions business object:</p> <ul style="list-style-type: none"> • Outbound Asset Master Processor program (R1201Z10)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Asset Master - Equipment Extensions Business Services - EquipmentManager

The EquipmentManager web service (JP170001) manages the processing of equipment-related web service operations. This table includes a description of the EquipmentManager web service operations:

Operation	Description
EquipmentProcessor	Use this operation to add new equipment records, or change existing equipment records in the JD Edwards EnterpriseOne system.

Operation	Description
EquipmentQueryProcessor	Use this operation to retrieve and review equipment records that are stored in the JD Edwards EnterpriseOne database.

Accessing Javadoc for the EquipmentManager Web Service Operations

To access Javadoc for the EquipmentManager web service and its related operations, review these Javadoc packages:

- JP170001 (EquipmentManager)
- J1700001 (equipmentProcessor)
- J1700002 (equipmentQueryProcessor)

equipmentProcessor

The equipmentProcessor web service operation is an inbound transaction operation that enables consumers to add new equipment records to the JD Edwards EnterpriseOne system, or to change existing equipment records. The operation uses the CRMInstalledBaseProcessing business function (N1702710) to process equipment information.

If the operation completes successfully, all data additions and changes are updated in these JD Edwards EnterpriseOne tables:

- Asset Master table (F1201)
- Equipment Master Extension table (F1217)

If the operation encounters errors while processing, no updates are made to the JD Edwards EnterpriseOne tables, and the error messages are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the EquipmentProcessor operation uses:

Group	Key	Description	Default Value
J1700001	J1700001_EQ_MBF_VERSION	Use this property to specify the version of the Equipment Master Revisions application (P1702) that the CRMInstalledBase business function (N1702710) uses to process equipment information.	ZJDE0001

Implementation Details

This table includes information that can help determine whether the EquipmentProcessor operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the operation completes successfully, all changes or additions are updated in the JD Edwards EnterpriseOne system.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • actionType • assetId • description • unitNumber • serialNumber • Customer entity, which includes one or more of these fields: <ul style="list-style-type: none"> ○ entityId ○ entityLongId, ○ entityTaxId • Site entity, which includes one or more of these fields: <ul style="list-style-type: none"> ○ entityId ○ entityLongId, ○ entityTaxId
<p>If I encounter errors while process a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

equipmentQueryProcessor

The EquipmentQueryProcessor web service operation is a database query operation that enables consumers to retrieve and review equipment records from the JD Edwards EnterpriseOne system. The operation retrieves data from these EnterpriseOne tables:

- Asset Master table (F1201)

- Equipment Master Extension table (F1217)

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the EquipmentQueryProcessor operation uses:

Group	Key	Description	Default Value
J1700002	J1700002_V1201R_MAX_GRID_ROWS_RETURNED	Use this business service property to specify the maximum number of records the operation can return for a query.	20

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records, which could impact performance.

Implementation Details

This table includes information that can help determine whether the EquipmentQueryProcessor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records that match the selection criteria are returned to the consumer. If the operation completes successfully, but finds no matching records, a record count of zero (0) is returned.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • assetId • description • unitNumber • serialNumber • Customer entity, which includes one or more of these fields: <ul style="list-style-type: none"> ○ entityId ○ entityLongId, ○ entityTaxId

Question	Answer
	<ul style="list-style-type: none"> • Site entity, which includes one or more of these fields: <ul style="list-style-type: none"> ○ entityld ○ entityLongld, ○ entityTaxld
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Asset Master - Equipment Extensions Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Asset Master - Equipment Extensions business object:

Program and Program Type	Description	Available Documentation
Outbound Asset Master Processor program (R1201Z10) This is an interoperability program.	The Outbound Asset Master Processor program fetches the F986113 record and retrieves the name and version of the custom UBE from the F0047 table. The Outbound Asset Master Processor program passes information about the transactions to the custom UBE. The custom UBE then retrieves the records from the F1201Z1 table and processes that information.	See this topic: <ul style="list-style-type: none"> • <i>Processing Interoperability for Fixed Assets</i>

9 ALM: Condition Based Alert

Condition Based Alert Overview

This chapter provides detailed information about the business interfaces that are available for the Condition Based Alert business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The CapitalAssetManager published web service (JP130000) manages these methods: <ul style="list-style-type: none"> • J1300001 (createCapitalAssetConditionBasedAlert) • J1300002 (getCapitalAssetConditionBasedAlert)
Batch Import Programs	Available batch import programs: <ul style="list-style-type: none"> • Inbound Condition-Based Alerts Processor program (R1310Z11)
Orchestration	Add Condition-Based Maintenance Alert For additional information about the Add Condition-Based Maintenance Alert orchestration, see Add Condition-Based Maintenance Alert .

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Condition Based Alert Business Services - CapitalAssetManager

The CapitalAssetManager web service (JP130000) manages the processing of capital asset-related web service operations. This table includes a description of the capital asset web service operations:

Operation	Description
createCapitalAssetConditionBasedAlert	Use this operation to add capital asset condition-based alerts within the JD Edwards EnterpriseOne Plant and Equipment Management system.

Operation	Description
getCapitalAssetConditionBasedAlert	Use this operation to query the JD Edwards EnterpriseOne database for existing capital asset condition-based alerts.

Accessing Javadoc for the Capital Asset Manager Web Service Operations

To access Javadoc for the Capital Asset Manager web service and its related operations, review these Javadoc packages:

- JP130000 (CapitalAssetManager)
- J1300001 (createCapitalAssetConditionBasedAlert)
- J1300002 (getCapitalAssetConditionBasedAlert)

Prerequisites

Before using the Capital Asset Manager web service, or any of the related web service operations, you must install and configure the JD Edwards EnterpriseOne Capital Asset Management product including the JD Edwards EnterpriseOne Plant and Equipment Management and JD Edwards EnterpriseOne Condition-Based Maintenance systems.

See [JD Edwards EnterpriseOne Applications Capital Asset Management Implementation Guide](#) .

See [JD Edwards EnterpriseOne Applications Condition-Based Maintenance Implementation Guide](#) .

createCapitalAssetConditionBasedAlert

The createCapitalAssetConditionBasedAlert web service operation is an inbound transaction operation that enables consumers to process condition-based alert information within the JD Edwards EnterpriseOne system. The consumer can add capital asset condition-based alerts in the JD Edwards EnterpriseOne Plant and Equipment Management system.

If the operation is successful, the system creates condition-based alerts and commits the records.

If the operation fails, the system returns error messages to the consumer.

The createCapitalAssetConditionBasedAlert web service operation calls the XPIConditionBasedAlert master business function (N1301150) to process the capital asset record. When a condition-based alert is successfully created, a condition-based alert key is returned to the caller through the Alert ID. Exceptions are sent to the caller. The minimum required fields to create a condition-based alert in JD Edwards EnterpriseOne are equipmentNumber, alertLevel, eventDate, eventTime, description, and automatedResponseType.

Supported Functionality

This section discusses the functionality that the createCapitalAssetConditionBasedAlert operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The createCapitalAssetConditionBasedAlert operation supports adding capital asset condition-based alerts in the JD Edwards EnterpriseOne Plant and Equipment Management system.

The createCapitalAssetConditionBasedAlert operation does not support updating or deleting capital asset condition-based alerts.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes condition-based alert information. This table includes information about the business service properties that the createCapitalAssetConditionBasedAlert web service operation uses:

Group	Key	Description	Default Value
J1300001	J1300001_CBA_MBF_VERSION	Use this business service property to specify which version of the XPI Condition-Based Alerts program (P1301150) the operation uses.	ZJDE0001
J1300001	J1300001_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages.	Capital Asset Send In

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

The following table includes information that can help determine whether the createCapitalAssetConditionBasedAlert operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system does not return an error message.</p> <p>If the operation succeeds and returns records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • alertId • description • alertLevelCode • alertStatusCode • automatedResponseTypeCode • notificationRecipient

Question	Answer
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.
Does this operation use record reservation?	No. This operation does not reserve records.

Setting Processing Options for XPI Condition-Based Alerts (P1301150)

Processing options enable you to specify the default processing for programs and reports.

For programs, you can specify options such as the default values for specific transactions, whether fields appear on a form, and the version of the program that you want to run.

Process

1. Equipment Number Format

Specify how the system validates the equipment number. Values are:

- 1: Use the equipment number. This is the default value.
- 2: Use the unit number.
- 3: Use the serial number.

Versions

1. Condition-Based Alerts Revisions (P1311) Version

Specify the version that the system uses for the Condition-Based Alerts Revisions program (P1311). If you leave this processing option blank, the system uses the ZJDE0001 version.

2. Condition-Based Alerts Processor (R1312) Version

Specify which version of the Condition-Based Alerts Processor program (R1312) the system uses when processing the automated responses of condition-based alerts. If you leave this processing option blank, the system does not process the automated responses of condition-based alerts. Note: If you leave this processing option blank, you must run a version of the Condition-Based Alerts Processor program (R1312) in order to complete the automated response-type processing.

getCapitalAssetConditionBasedAlert

The getCapitalAssetConditionBasedAlert web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Plant and Equipment Management system to retrieve existing capital asset data. The getCapitalAssetConditionBasedAlert operation invokes a database operation to retrieve capital asset information from the Condition-Based Alerts table (F1310) in JD Edwards EnterpriseOne based on the selection criteria specified in the value object.

The `getCapitalAssetConditionBasedAlert` web service operation allows an open query.

Note: An open query of the JD Edwards EnterpriseOne database impacts system performance. It is strongly recommended that you set the Max Rows business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database.

If the operation is successful, the system returns zero to many records to the consumer. You can specify the maximum number of records to return during a query using the Max Rows business service property.

If the operation fails, the system returns an error message to the consumer.

The `getCapitalAssetConditionBasedAlert` web service operation queries the JD Edwards EnterpriseOne condition-based alert information in a real-time fashion. The web service operation, `getCapitalAssetConditionBasedAlert`, is called by the source system. This web service operation consumes a published interface, `GetCapitalAssetConditionBasedAlert`, which exposes the selection criteria for retrieving the condition-based alert information from the F1310 table in JD Edwards EnterpriseOne. The fields in `GetCapitalAssetConditionBasedAlert` are used to set the selection in the `getCapitalAssetConditionBasedAlert` integration. The web service operation returns results in the `ShowCapitalAssetConditionBasedAlert` published interface.

Note: The system adjusts the date time stamp entered to Coordinated Universal Time (UTC), which the system determines using the time zone on your business service server, unless an offset is used. For example, if the business service server is GMT-7, entering 2007-01-18T00:00:00.000 results in an adjustment to this value of -7 hours, and the actual value that passes in to JD Edwards EnterpriseOne becomes 2007-01-17T17:00:00.000. An offset can be passed in with the original value so that these adjustments do not occur, such as 2007-01-18T00:00:00.000-7:00, which offsets by 7 hours from UTC, resulting in the date being passed into JD Edwards EnterpriseOne as 2007-01-18T00:00:00.000.

Supported Functionality

This section discusses the functionality that the `getCapitalAssetConditionBasedAlert` operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The `getCapitalAssetConditionBasedAlert` operation supports a database query operation that enables consumers to query the JD Edwards EnterpriseOne Plant and Equipment Management system to retrieve existing capital asset data.

The `getCapitalAssetConditionBasedAlert` operation does not support the wildcard (asterisk `*`) for search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes capital asset information. This table includes information about the business service properties that the `getCapitalAssetConditionBasedAlert` web service operation uses:

Group	Key	Description	Default Value
J1300002	J1300002_F1310_MAX_GRID_ROWS_RETURNED	Use this business service property to define the maximum	100

Group	Key	Description	Default Value
		number of rows that the operation returns when querying the JD Edwards EnterpriseOne database.	

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database.

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

The following table includes information that can help determine whether the getCapitalAssetConditionBasedAlert operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system does not return an error message.</p> <p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • alertId • description • alertLevelCode • alertStatusCode • automatedResponseTypeCode • dateRequested • timeRequested • eventTimeUTC • itemAsset <p>The operation may complete successfully without returning rows because the selection criteria did not match any records in the database or an open query was performed and there were no records. This is considered successful and provides valid information.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

Question	Answer

Condition Based ALert Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Condition Based Alerts business object:

Program and Program Type	Description	Available Documentation
<p>Inbound Condition-Based Alerts Processor program (R1310Z1I)</p> <p>This is an interoperability batch program.</p>	<p>Use this program to copy the information from the F1310Z1 table to the Condition-Based Alerts table (F1310). The program also performs these functions:</p> <ul style="list-style-type: none"> Processes the inbound condition-based alerts transactions. Updates the F1310 table. Initiates any required downstream processing that is based on the transaction information or the alert action rules by running a version of the Condition-Based Maintenance Alerts program (R1312). 	<p>See this topic for additional information:</p> <ul style="list-style-type: none"> <i>Processing Inbound Interoperability for Condition-Based Maintenance</i>

10 ALM: Equipment Locations

Equipment Locations Overview

This chapter provides detailed information about the business interface components that are available for the Equipment Locations business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Equipment Locations business object:</p> <ul style="list-style-type: none"> RTEQPOUT, which is a container event for RTEQPHDR.
Orchestration	<p>Update Equipment Location</p> <p>For additional information about the Update Equipment Location orchestration, see Update Equipment Location.</p>
Business Interface	<p>(Release 9.2 Update) The LatLongProcessor web service (JC011500) uses the Oracle Location Service to retrieve and store the latitude and longitude for each record of equipment with the an Address Line 1, City, and Postal code. The following business service is available for the Customer Equipment business service:</p> <ul style="list-style-type: none"> getProcessLatLong

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Equipment Locations Real-Time Events

This section describes the available real-time events associated with the Equipment Locations business object, which include:

- RTEQPOUT, which is a container event for RTEQPHDR.

RTEQPOUT

RTEQPOUT is a container event that is generated by the JD Edwards EnterpriseOne system to provide equipment information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTEQPOUT	Equipment Details	RTE	Container event for RTEQPHDR (D1702940).	H13

See the real-time event that is contained by this event for information about the conceptual approach, and processes that can initiate this event.

Objects That Can Initiate The Event

These objects can initiate the RTEQPOUT event, and the single event contained within the event:

- N120010 - F1201 Process Master Data
- N120090 - F1217 Process Master Data

RTEQPHDR

RTEQPHDR is a single event that is generated by the JD Edwards EnterpriseOne system to provide equipment information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTEQPHDR	Equipment Header Details	RTE	Single	H13	D1702940: Equipment Real Time Template	Included in container event RTEQPOUT.

Conceptual Approach

An equipment record added, modified or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Asset Master File table (F1201) and the Equipment Master Extension table (F1217).

Equipment Locations Business Services - LatLongProcessor (Release 9.2 Update)

The LatLongProcessor web service (JC011500) processes the latitude and longitude for each record of equipment with an Address Line 1, City, and Postal code. This table includes a description of the equipment location business service.

Operation	Description
getProcessLatLong	Use this business service to retrieve and update the latitude and longitude for each record of equipment with the an Address Line 1, City, and Postal code information from the Equipment Master Information.

Prerequisite

You must also set the value of Enable Location Services to Yes in the Work with EnterpriseOne System Control (P99410) program.

getProcessLatLong

The getProcessLatLong business service runs on the Oracle WebLogic Server (WLS) or the Web Application Server (WAS) platform and reads the data stored in the Equipment Master Location Details (F17311) table. The EnterpriseOne Enterprise Server collects the data required from the JD Edwards EnterpriseOne F17220W table and then calls the getProcessLatLong business service that communicates with Oracle Location Services to update the Equipment Master Location Details (F17311) table with latitude and longitude for each record of equipment with an Address Line 1, City, and Postal code.

11 ALM: Meter Reading

Meter Reading Overview

This chapter provides detailed information about the business interfaces that are available for the Meter Reading business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	This list includes the batch import programs for the Meter Reading business object: <ul style="list-style-type: none"> Inbound Meter Readings Processor program (R12120Z1I)
Orchestration	Update Meter Readings For additional information about the Update Meter Readings orchestration, see <i>Update Meter Readings</i> .

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Meter Reading Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Meter Reading business object:

Program and Program Type	Description	Available Documentation
Inbound Meter Readings Processor program (R12120Z1I) This is an interoperability program.	Run the Inbound Meter Readings Processor program (R12120Z1I) to validate the data in the F12120Z1 table. If validation fails, the system issues errors and does not update the records to the meter reading database.	See this topic: <ul style="list-style-type: none"> <i>Understanding Meter Reading Interoperability</i>

12 ALM: Work Order - Equipment

Work Order - Equipment Overview

This chapter provides detailed information about the business interfaces that are available for the Work Order - Equipment business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>These are the published business services available for the Work Order - Equipment business object:</p> <ul style="list-style-type: none"> • Equipment Work Order (JP130001), which manages these business services: <ul style="list-style-type: none"> ○ Work Order Processor (J4800010) ○ Work Order Query Processor (J4800020)
Real-Time Events	<p>This list includes the real-time events for the Work Order - Equipment business object:</p> <ul style="list-style-type: none"> • RTEQORDOUT, which is a container event for RTEQORDHDR.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Work Order - Equipment Business Services - Equipment Work Order (JP130001)

This published business service processes equipment work order information using these business services:

- Work Order Processor (J4800010)
- Work Order Query Processor (J4800020)

These business services are documented under the Work Order Processor published business service, as they are also managed by that service. See [Work Order Processor \(J4800010\)](#) and [Work Order Query Processor \(J4800020\)](#).

Work Order - Equipment Real-Time Events

This section describes the available real-time events associated with the Work Order business unit, which include:

- RTEQORDOUT, which is a container event for RTEQORDHDR.

RTEQORDOUT

RTEQORDOUT is a container event that is generated by the JD Edwards EnterpriseOne system to provide work order information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTEQORDOUT	Equipment Order	RTE	Contains the RTEQORDHDR (D1702960) event.	H13

Conceptual Approach

A work order record added, modified or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Work Order Master File table (F4801) and the Work Order Master Tag File table (F4801T).

Objects That Can Initiate The Event

The X4801 - X4801 Work Order Entry Master business function can initiate the RTEQORDOUT event, and the single event contained within the event.

RTEQORDHDR

RTEQORDHDR is a single event that is generated by the JD Edwards EnterpriseOne system to provide work order information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTEQORDHDR	Equipment Order Header	RTE	Single	H13	D1702960: Equipment Order Real Time Template	Included in container event RTEQORDOUT.

Conceptual Approach

A work order record added, modified or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Work Order Master File table (F4801) and the Work Order Master Tag File table (F4801T).

13 ALM: Work Order - Service

Work Order - Service Overview

This chapter provides detailed information about the business interfaces that are available for the Work Order - Service business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The CustomerServiceManager published business service (JP170000) manages these business services:</p> <ul style="list-style-type: none"> • J1700010 (processServiceOrder) • J1700010 (processServiceOrderV2) • J1700020 (getServiceOrder) • 1700020 (getServiceOrderV2) • J1700030 (getCommitmentDateTime) • J1700030 (getCommitmentDateTimeV2) <p>The Service Work Order (JP90CD00), which manages these business services:</p> <ul style="list-style-type: none"> • Work Order Processor (J4800010) • Work Order Query Processor (J4800020)
Real-Time Events	<p>This list includes the real-time events for the Work Order - Service business object:</p> <ul style="list-style-type: none"> • RTLBR0UT, which is a container event for RTLBRHDR.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Work Order - Service Business Services - CustomerServiceManager

The CustomerServiceManager web service (JP170000) manages the processing of customer service-related web service operations. This table includes a description of the customer service web service operations:

Operation	Description
getServiceOrder (J1700020)	Use this operation to query the JD Edwards EnterpriseOne database for existing service orders, along with labor and parts information.
getServiceOrderV2 (J1700020)	Use this operation to query the JD Edwards EnterpriseOne database for existing service orders along with labor, parts, and global locator number (GLN) information.
processServiceOrder (J1700010)	Use this operation to complete these tasks within the JD Edwards EnterpriseOne Service Management system: <ul style="list-style-type: none"> • Add, change or delete service orders. • Add, change or delete related labor lines. • Add, change or delete related parts lines.
processServiceOrderV2 (J1700010)	Use this operation to add, update, or delete service order information to include labor, parts, and GLN.
getCommitmentDateTime (J1700030)	Use this operation to retrieve information from the JD Edwards EnterpriseOne Service Agreement system to calculate the commitment date and time for a service order. The system retrieves information such as customer and specified site.
getCommitmentDateTime (J1700030)	Use this operation to retrieve information from the JD Edwards EnterpriseOne Service Agreement system to calculate the commitment date and time for a service order. The system retrieves information such as customer, specified site, and GLN.

Note:

In the EnterpriseOne system, work orders are stored in the Work Order Master table (F4801). When you create a work order, you must specify the document type that is associated with the work order. The system uses the value in the Document Type field (DCTO) to determine the order type of the work order.

You use the Document Type Maintenance program (P40040) to maintain document type information. Document type records are stored in the Document Type Master table (F40039). When you enter a document type into the system, you can associate that document type with an order type. Order types might include manufacturing orders, equipment orders, and service work orders. To specify that the document type is associated with service work orders, enter **05** in the Order Type field.

The getServiceOrder and processServiceOrder operations process only those work orders that have been defined as service work orders. Service work orders must include a document type that is associated with an order type of **05**.

See *"Setting Up Service Management" in the JD Edwards EnterpriseOne Applications Service Management Implementation Guide* .

Prerequisite

Before using the Customer Service Manager web service, or any of the related web service operations, you must install and configure the Service Management system. Additionally, you must verify that you are using document types associated with order type **05** to identify service work orders.

See *"Setting Up Service Management" in the JD Edwards EnterpriseOne Applications Service Management Implementation Guide* .

Accessing Javadoc for the Customer Service Manager Web Service Operations

To access Javadoc for the Customer Service Manager web service and its related operations, review these Javadoc packages:

- JP170000 (CustomerServiceManager)
- J1700010 (processServiceOrder)
- J1700010 (processServiceOrderV2)
- J1700020 (getServiceOrder)
- 1700020 (getServiceOrderV2)
- J1700030 (getCommitmentDateTime)
- J1700030 (getCommitmentDataTimeV2)

getServiceOrder

The `getServiceOrder` web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Service Management system to retrieve existing service orders, along with related parts and labor information.

If the operation is successful, the system returns zero to many records to the consumer. You can specify the maximum number of service orders to return during a query using the Max Rows business service property. If the operation fails, the system returns an error message to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns when you query the EnterpriseOne database. This table includes information about the business service properties used by the `getServiceOrder` operation:

Group	Key	Description	Default Value
J1700020	J1700020_MAX_ROWS	Use this business service property to specify the maximum number of service order records that	0 :Return all rows.

Group	Key	Description	Default Value
		<p>the operation returns when querying the EnterpriseOne database.</p> <p>Note: This business service property limits the number of service order header records that are retrieved. It does not limit the number of labor or parts lines that are retrieved. Therefore, the number of rows that the system returns, including order headers, parts lines and labor lines, could be significantly higher than the value in this constant.</p>	

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching header records, along with any associated labor and parts records. Additionally, it is recommended that you specify selection criteria when you query the EnterpriseOne database. The system returns an error if you perform a query with no selection criteria and this business service property is set to **0**.

Implementation Details

The following table includes information that can help determine whether the `getServiceOrder` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • laborLineNumber • operationTypeCode • businessUnit • partsLineNumber <p>Note: If the system does not return any errors or records, verify that you have entered selection criteria and that the Max Rows business service property is set to a value other than 0.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again.</p>

Question	Answer
	If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

getServiceOrderV2

Review the information in the `getServiceOrder` section of this chapter before using the `getServiceOrderV2` web service operation. All of the information in the `getServiceOrder` section also applies to the `getServiceOrderV2` operation.

The `getServiceOrderV2` operation is a version of the `getServiceOrder` web service operation. This web service operation is used to retrieve service order information with associated GLNs. This version executes the `getServiceOrder` (J1700020) web service operation.

processServiceOrder

The `processServiceOrder` web service operation is an inbound transaction operation that enables consumers to process service order information within the JD Edwards EnterpriseOne system. The consumer can complete these tasks in the JD Edwards EnterpriseOne Service Management system:

- Add, change or delete service orders.
- Add, change or delete labor lines that are associated with a service order.
- Add, change or delete parts lines that are associated with a service order.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes service order information. This table includes information about the business service properties that the `processServiceOrder` web service operation uses:

Group	Key	Description	Default Value
J1700010	J1700010_SERVICE_ORDER_INT	Use this business service property to specify which version of the CRM Service Order Integration Processing Options program (P1702650) the operation uses.	ZJDE0001
J1700010	J1700010_BYPASS_WARNINGS	Use this business service property to specify whether the system ignores warnings that are generated in the EnterpriseOne system during processing.	1: Bypass warnings.

Group	Key	Description	Default Value
		If you want the operation to stop processing upon receipt of a warning, set this property to 0 or leave it blank. Using this setting, the operation stops processing and returns the warnings to the consumer.	

Implementation Details

The following table includes information that can help determine whether the processServiceOrder operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • laborLineNumber • operationTypeCode • businessUnit • partsLineNumber
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing with auto-commit. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.
Does this operation use record reservation?	This operation calls the Work Order Integration business function (N1702650). This function reserves records in the EnterpriseOne system when the operation is changing or deleting existing data.

Setting Processing Options for CRM Service Order Integration Processing Options (P1702650)

You use processing options to specify default values that a program uses when processing information.

Service Order Tab

1. Order Type

Use this processing option to specify the default document type that the system uses when you enter a work order. This code also indicates the origin of the transaction. Enter a value from UDC 00/DT.

2. Service Type

Use this processing option to specify the default order type that the system uses when you enter a work order. The order type indicates the type classification of a work order or engineering change order. You can use work order type as a selection criterion for work order approvals. Enter a value from UDC 00/TY.

3. Priority

Use this processing option to specify the default work order priority that the system uses when you enter a work order. This code indicates the priority of a work order or engineering change order in relation to other orders. Enter a value from UDC 00/PR.

4. Primary Service Provider

Use this processing option to specify the Address Book number of a manager or a planner that the system uses as a default when you enter a work order.

5. Secondary Service Provider

Use this processing option to specify the address book number of the supervisor that the system uses as a default when you enter a work order.

6. Primary Technician

Use this processing option to specify the address book number of a person that the system assigns to do the work when you enter a work order.

7. Secondary Technician

Use this processing option to specify the address book number of an inspector that the system uses as a default when you enter a work order.

8. Type Bill of Material

Use this processing option to specify the default parts list that the system uses when you enter a work order. The parts list is a user defined code that designates the type of bill of material. Enter a value from UDC 40/TB.

9. Type Routing

Use this processing option to specify the labor detail that the system uses as a default when you enter a work order. The system retrieves the labor detail for the work order header and uses it to identify the requested service. Enter a value from UDC 40/TR.

10. Entitlement Checking

Use this processing option to determine whether the system performs entitlement checking and the preferred method. Values are:

Blank: Bypass entitlement checking.

1: Check entitlements using the Entitlement Dates table (F1791).

2: Check entitlements without using the Entitlement Dates table.

11. Category Code 01 through 20. Category Code 10

Use these processing options to specify user-defined default information that the system uses when creating a work order.

21. Business Unit

Use this processing option to specify the default business unit, which the system uses when it creates a new service order. You must specify a business unit that exists in the Business Unit Master table (F0006).

22. Branch

Use this processing option to specify the default branch that the system uses when it creates a service order. You must specify a branch that exists in the Business Unit Master table (F0006).

23. Default Service Provider

Use this processing option to specify the default values for manager and supervisor addresses on the work order. Values are:

Blank: The system does not supply defaults for the manager and supervisor addresses.

1: The system uses defaults for the manager and supervisor addresses based on this hierarchy:

- First: The system uses the values for the manager and supervisor from processing options.
- Second: The system uses the manager value from the SWM Address Book extension record for the customer.
- Third: The system uses the manager value from the equipment record.
- Fourth: The system uses the values for the manager and supervisor from the Work Order Default Coding File table (F48001), based on category codes 1, 2, and 3 from the work order.

Note: The system uses the Assigned To address value from processing options first, and information from the equipment record second.

processServiceOrderV2

Review the information in the processServiceOrder section of this chapter before using the processServiceOrderV2 web service operation. All of the information in the processServiceOrder section also applies to the processServiceOrderV2 operation.

The processServiceOrderV2 operation is a version of the processServiceOrder web service operation. This web service operation is used to add, update, and delete service order information with associated GLNs. This version executes the processServiceOrder (J1700010) web service operation.

getCommitmentDateTime

The getCommitmentDateTime web service operation is an inbound transaction operation that enables consumers to calculate the commitment date and time for a given service order. The system calculates the commitment information using the date and time that the specified service order was entered into the system along with the commitment information from the customer or site's service contract, which is stored in the JD Edwards EnterpriseOne Service Management system.

If the system encounters errors during processing, those errors are returned to the consumer.

Prerequisites

Before using the getCommitmentDateTime operation, service contracts with commitment information must exist in the EnterpriseOne system. Additionally, you must set up service and warranty constants, time zone information, and contact coverage.

See *"Managing Contracts"* and *"Setting Up Service and Warranty Constants"* in the *JD Edwards EnterpriseOne Applications Service Management Implementation Guide*.

Supported Functionality

This section discusses the functionality that the getCommitmentDateTime operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

This operation supports time calculations using time zones and daylight savings rules. Thresholds are supported only if the consumer does not pass in a start date.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system calculates commitment information. This table includes information about the business service properties that the getCommitmentDateTime web service operation uses:

Group	Key	Description	Default Value
J1700030	J1700030.UTC_TIMEZON_DAYLIGHTSAVINGSRULE	Use this business service property to specify the UTC time zone and daylight savings rule that the operation uses when displaying the commitment date and time to the consumer. Note: You can specify the time zone only, or the time zone and the daylight savings rule. To specify both pieces of information, enter the time zone followed by " ", and then enter the daylight savings rule. For example, enter 26 USA for time zone 26 and daylight savings rule USA .	26: Greenwich Mean Time Note: Other values for this business service property are stored in UDC table H91/TZ.
J1700030	J1700030.BYPASS_BSFN_WARNINGS	Use this business service property to specify whether the system ignores warnings that are generated in the EnterpriseOne system during processing. If you want the operation to stop processing upon receipt of a warning, set this property to 0 or leave it blank. Using this setting, the operation stops processing and returns the warnings to the consumer.	1: Bypass warnings.

Implementation Details

The following table includes information that can help determine whether the getCommitmentDateTime operation is functioning correctly:

Question	Answer
How do I know if the operation completes successfully?	If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the EnterpriseOne system. At a minimum, the system returns non-zero values for these fields:

Question	Answer
	<ul style="list-style-type: none"> • dateCommitment • timeCommitment
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.
Does this operation use record reservation?	No. This operation does not reserve records that are used to calculate commitment information.

getCommitmentDateTimeV2

Review the information in the `getCommitmentDateTime` section of this chapter before using the `getCommitmentDateTimeV2` web service operation. All of the information in the `getCommitmentDateTime` section also applies to the `getCommitmentDateTimeV2` operation.

The `getCommitmentDateTimeV2` operation is a version of the `getCommitmentDateTime` web service operation. This web service operation is used retrieve information, along with associated GLNs, for the purpose of calculating commitment date and time for a service order. This version executes the `getCommitmentDateTime (J1700020)` web service operation.

lookupServiceOrder

The `lookupServiceOrders` web service operation is a database query operation that enables consumers to retrieve and review service order information from the JD Edwards EnterpriseOne database. This operation only fetches the records from the F40039 table that contain a value of 05 (service orders) in field A201. This operation retrieves these items, if the data exists in the database:

- OrderNumber
- AddressBookNumber
- OrderDescription
- OrderStatus
- OrderType
- AssignedTo
- CategoryCodesServiceOrder, which includes CategoryCode001 through CategoryCode020.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns when you query the EnterpriseOne database. This table includes information about the business service properties used by the `lookupServiceOrders` operation:

Group	Key	Description	Default Value
J1700020	J1700020_MAX_ROWS	Use this business service property to specify the maximum number of V4801JS records the operation can return for a query.	100

Implementation Details

The following table includes information that can help determine whether the lookupServiceOrder operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero or more records are returned, based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • OrderNumber • OrderStatus
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again.</p> <p>If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

Work Order - Service Business Services - Service Work Order (JP90CD00)

This published business service processes service work order information using these business services:

- Work Order Processor (J4800010)
- Work Order Query Processor (J4800020)

These business services are documented under the WorkOrderManager published business service, as they are also managed by that service. See [Work Order Processor \(J4800010\)](#) and [Work Order Query Processor \(J4800020\)](#).

Work Order - Service Real-Time Events

This section describes the available real-time events associated with the Work Order - Service business unit, which include:

- RTLBROUT, which is a container event for RTLBRHDR.

RTLBROUT

RTLBROUT is a container event that is generated by the JD Edwards EnterpriseOne system to provide work order labor information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTLBROUT	Labor Header	RTE	Container event for TLBRHDR (D1703000).	H13

Conceptual Approach

A parts record added, modified or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Work Order Routing table (F3112) and the Routing Instructions - Extension table (F31172).

Objects that Can Initiate the Event

This object can initiate the RTLBROUT event, and the single events contained within the event:

N1700410 – F31172 Process Master Data

RTLBRHDR

RTLBRHDR is a single event that is generated by the JD Edwards EnterpriseOne system to provide work order labor information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTLBRHDR	Routing Header	RTE	Single	H13	D1703000: Routing Real Time Template	Included in container event RTLBROUT.

Conceptual Approach

A parts record added, modified or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Work Order Routing table (F3112) and the Routing Instructions - Extension table (F31172).

14 CRM: Service Timecard

Service Timecard Overview

This chapter provides detailed information about the business interfaces that are available for the Service Timecard business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	Service Time Card Manager published business service (JP170002), and its related methods. This service was created for a specific integration for mobile applications. See <i>Mobile Smartphone Applications</i> .

15 CRM: Work Order - Supplier Recovery Claim

Work Order - Supplier Recovery Claim Overview

This chapter provides detailed information about the business interfaces that are available for the Work Order - Supplier Recovery Claim business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Work Order - Supplier Recovery Claim business object:</p> <ul style="list-style-type: none"> Inbound Warranty Claim Parts Detail Processing program (R17311Z2I) Inbound Warranty Claim Processing program (R174801Z2I) Inbound Warranty Claim Labor Detail Processing Program (R17311Z22I)
Batch Export Programs	<p>This list includes the batch export programs for the Work Order - Supplier Recovery Claim business object:</p> <ul style="list-style-type: none"> (R174801Z2O)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Work Order - Supplier Recovery Claim Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Work Order - Supplier Recovery Claim business object:

Program and Program Type	Description	Available Documentation
Inbound Warranty Claim Parts Detail	A dealer or authorized service provider performs work that is covered under a manufacturer's warranty. Inbound	See this topic:

Program and Program Type	Description	Available Documentation
<p>Processing program (R173111Z21)</p> <p>This is an interoperability program.</p>	<p>Warranty Claim Parts Detail Processing (R173111Z21) processes the parts detail of incoming flat file claims that are sent in from dealers. The data is uploaded to the work order file, and the claim is then processed through JD Edwards EnterpriseOne Service Management.</p>	<ul style="list-style-type: none"> • Processing Inbound Interoperability for Service Management
<p>Inbound Warranty Claim Processing program (R174801Z21)</p> <p>This is an interoperability program.</p>	<p>A dealer or authorized service provider performs work that is covered under a manufacturer's warranty. Use Inbound Warranty Claims Processing (R174801Z21) to receive flat file claims that are sent in from dealers. For each claim, the program creates a new warranty claim number, and processes parts detail and labor detail if they exist.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Processing Inbound Interoperability for Service Management
<p>Inbound Warranty Claim Labor Detail Processing Program (R173112Z21)</p> <p>This is an interoperability program.</p>	<p>A dealer or authorized service provider performs work that is covered under a manufacturer's warranty. Inbound Warranty Claim Parts Detail Processing (R173111Z21) processes the parts detail of incoming flat file claims that are sent in from dealers. The data is uploaded to the work order file, and the claim is then processed through JD Edwards EnterpriseOne Service Management.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Processing Inbound Interoperability for Service Management

Work Order - Supplier Recovery Claim Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Work Order - Supplier Recovery Claim business object:

Program and Program Type	Description	Available Documentation
<p>Outbound Supplier Recovery Processing (R174801Z20)</p> <p>This is an interoperability program.</p>	<p>A manufacturer submits a reimbursement claim to the supplier of a warranted part that was found to be defective. Use Outbound Supplier Recovery Processing (R174801Z20) to send flat file claims to suppliers. For each claim, the program creates a new supplier recovery claim number. Each claim can contain several parts detail and labor detail records.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Processing Outbound Interoperability for Service Management

16 FMS: Account Ledger

Account Ledger Overview

This chapter provides detailed information about the business interfaces that are available for the Account Ledger business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Account Ledger business object:</p> <ul style="list-style-type: none"> • RTPOSTOUT • RTBJESOUT
Batch Import Programs	<p>This list includes the batch import programs for the Account Ledger business object:</p> <ul style="list-style-type: none"> • Store and Forward Journal Entry Upload (R0911Z1) • Journal Entries Batch Processor program (R09110Z)
Additional Interfaces	<p>The Account Ledger business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • Journal Entry Master Business Function (P0900049) <p>Depending on the processing option settings on the Interoperability tab, this program can write outbound records that can be used by an external software system.</p> <p>See Setting Processing Options for the Journal Entry Master Business Function (P0900049).</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Account Ledger Real-Time Events

This section describes the available real-time events associated with the Account Ledger, business object which includes:

- RTPOSTOUT
- RTBJESOUT

RTPOSTOUT

RTPOSTOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems when a batch has been successfully posted to the Account Balances table (F0902).

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure
RTPOSTOUT	Fixed Asset GL Post	RTE	Single	00	D0900220B: a Journal Entry Post RTE Notification Message

Conceptual Approach

This event delivers a real time event notification when a batch has been successfully posted to the Account Balances table (F0902) using the General Ledger Post program (R09801). The status indicates only that the batch of journal entries has been successfully posted. If the batch fails to post, this event is not generated.

Objects that Can Initiate the Event

R09801 - General Ledger Post

RTBJESOUT

RTBJESOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems of the status of batch journal entries being processed through the Journal Entry Batch Processor (R09110Z).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTBJESOUT	Journal Entry	RTE	Single	00	D0900210B: Batch Journal Entry Status RTE Notification Message

Conceptual Approach

This event delivers a real time event notification of the status of batch journal entries being processed through the Journal Entry Batch Processor (R09110Z). The status indicates whether the journal entry document was created successfully. The reason for the failure is not included. The notification also includes the journal entry document number and general ledger date if the document is successfully added to the Account Ledger table (F0911).

Objects that Can Initiate the Event

R09110Z - Journal Entry Batch

Account Ledger Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Account Ledger business object:

Program and Program Type	Description	Available Documentation
Store and Forward Journal Entry Upload (R0911Z1)	If you use either the Store and Forward or the Batch Journal Entry programs, you might want to process journal entries with VAT taxes. When you process batch journal entries, you must create records in the F0911Z1 table, and then process them using the R0911Z1 program (depending on the menu, this program is referenced as either Journal Entries Batch Processor or Store & Forward Journal Entry Upload). When you run the R0911Z1 program, provided there are no errors, the system updates transactions in the F0911 table.	See this topic: <ul style="list-style-type: none"> • Understanding Batch Journal Entries with VAT
Journal Entries Batch Processor program (R09110Z)	Run the Journal Entries Batch Processor program (R09110Z) to upload the journal entries from the F0911Z1 table to the F0911 table. The Journal Entries Batch Processor program verifies that the information in the F0911Z1 table is formatted correctly before you transfer it to the F0911 table.	See this topic: <ul style="list-style-type: none"> • Understanding Journal Entry Batch Processing

17 FMS: Account Master

Account Master Overview

This chapter provides detailed information about the business interfaces that are available for the Account Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	This list includes the real-time events for the Account Master business object: <ul style="list-style-type: none"> RTAMOUT

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Account Master Real-Time Events

This section describes the available real-time events associated with the Account Master, business object which includes the RTAMOUT event.

RTAMOUT

RTAMOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of item additions, changes, or deletions of accounts in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTAMOUT	Account Master	RTE	Single	H09	D0900170B: GL Account Real Time Event Notification Outbound Data

Conceptual Approach

A new general ledger account added in the JD Edwards EnterpriseOne system creates a new record in the Account Master (F0901) table. The user can create an alternate description for this account by language preference that is

stored in the Account Master - Alternate Description (F0901D) table. If an account is added or modified in a system, the information is sent to a third-party system for add or update. If an account is deleted, the pertinent information is sent to the third-party, which will take the appropriate action. If applicable, the third-party integration must update the cross-reference table in the process in order to tie together the key values from both systems.

Objects that Can Initiate the Event

These objects can initiate the RTAMOUT event:

- P0901 - Accounts
- R0006QD - Update F0901 as Model Base on Table F0006
- R09803 - Global Account Number Update
- R09805 - Global Subsidiary Update
- R09813 - Change Account Information
- R09814 - Delete Account Master Records
- R10861 - Delete Prior Consolidation
- B0900053 - Duplicate Business Unit
- B0900073 - Dynamic Account Creation
- B0900075 - F0901D Update Account Translation
- B0900077 - F0901D Edit Account Translation
- B0900155 - Delete Account F0901/F0901D/F0902/F0911 Records
- B1400009 - Global Budget Pattern Code Update
- F0901/F0901D, F0902, F0911 Records
- B5100017 - F5109 Copy Accounts
- B5100140 - Update Project Account

18 FMS: Accounts Payable Draft

Accounts Payable Draft Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Payable Draft business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Accounts Payable Draft business object:</p> <ul style="list-style-type: none"> • RTRAOUT, which is a container event for these single events: <ul style="list-style-type: none"> ◦ RTRAHDR ◦ RTRADTL
Batch Export	<p>This list includes the batch export programs for the Accounts Payable Draft business object:</p> <ul style="list-style-type: none"> • Update as Sent program (R47058)
Additional Components	<p>The Accounts Payable Draft business object also uses the F0413Z1 Retrieve Interoperability Processing Options (P0400297) to create outbound records.</p> <p>See Setting Processing Options for F0413Z1 Retrieve Interoperability Processing Options (P0400297) for additional information.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Accounts Payable Draft Real-Time Events

This section describes the available real-time events associated with the Accounts Payable Draft business object, which include:

- RTRAOUT, which is a container event for these single events:
 - RTRAHDR
 - RTRADTL

RTRAOUT

RTRAOUT is a container event that provides AP Payment Remittance information to third-party system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTRAOUT	RTRAOUT Header	RTE	Container event for RTRAHDR and RTRADTL	04

Conceptual Approach

A payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A real-time notification event is triggered in the payments system after this payment is made in JD Edwards EnterpriseOne tables.

The Send Method field (A6CRMD) in the Supplier Master (F0401) table controls whether JD Edwards EnterpriseOne sends a RemittanceAdviceNotify real-time event notification when RTRAOUT is enabled. A real-time event notification is sent only if the supplier has a value of 5 in the Send Method field.

Objects That Can Initiate The Event

These objects can initiate the RTRAOUT event, and the single events contained within it:

- B0400016 - A/P Payments Master Business Function
- B0400169 - Remittance Advice Real-Time Event Notification
- B0400047 - Voucher Entry Master Business Function
- P0413M - A/P Manual Payments
- N0400061 - Remittance Advice Notification For Manual Payments

RTRAHDR

RTRAHDR is a single event that contains payment remittance advice header information that can be used by another system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTRAHDR	RA Header	RTE	Single	04	D0400169A: Remittance Advice - Header	Included in container event RTRAOUT.

Conceptual Approach

A payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A real-time notification event is triggered in the payments system after payment is made in JD Edwards EnterpriseOne tables.

RTRADTL

RTRADTL is a single event that contains payment remittance advice detail information that can be used by another system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTRADTL	RA Detail	RTE	Single	04	D0400169B: Remittance Advice - Detail	Included in container event RTRAOUT.

Conceptual Approach

A payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A detailed real-time notification event is triggered in the payments system after a payment is made in JD Edwards EnterpriseOne tables.

Accounts Payable Draft Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Accounts Payable Draft business object:

Program and Program Type	Description	Available Documentation
Update as Sent program (R47058) This is an EDI program.	You run the Update as Sent program when your trading partner notifies you that they received your EDI payment transmission.	See this topic: <ul style="list-style-type: none"> Understanding the Outbound EDI Payments Process

19 FMS: Accounts Payable Payment

Accounts Payable Payment Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Payable Payment business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Accounts Payable Payment business object:</p> <ul style="list-style-type: none"> • RTRAOUT, which contains these single events: <ul style="list-style-type: none"> ◦ RTRAHDR ◦ RTRADTL
Batch Export Programs	<p>This list includes the batch export programs for the Accounts Payable Payment business object:</p> <ul style="list-style-type: none"> • Update as Sent program (R47058)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Accounts Payable Payment Real-Time Events

This section describes the available real-time events associated with the Accounts Payable Payment business object which include:

- RTRAOUT, which contains these single events:
 - RTRAHDR
 - RTRADTL

RTRAOUT

RTRAOUT is a container event that provides AP Payment Remittance information to third-party system. RTRAOUT is an aggregate event that contains these single events:

RTRAHDR - contains AP Payment Remittance Advice header information that can be used by another system.

RTRADTL - contains AP Payment Remittance Advice detail information that can be used by another system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTRAOUT	Accounts Payable Payment	RTE	Container Container event for: <ul style="list-style-type: none"> RTRAHDR (D0400169A) RTRADTL (D0400169B) 	04

Please see the appropriate real-time events that are contained by this event for information about the event fields.

Conceptual Approach

An payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A real-time notification event is triggered in the payments system after this payment is made in JD Edwards EnterpriseOne tables.

The Send Method field (A6CRMD) in the Supplier Master (F0401) table controls whether JD Edwards EnterpriseOne sends a RemittanceAdviceNotify real-time event notification when RTRAOUT is enabled. A real-time event notification is sent only if the supplier has a value of 5 in the Send Method field.

Objects that Can Initiate the Event

These objects can initiate the RTRAOUT event, and the single events contained within it:

- N0400061 - Remittance Advice Notification For Manual Payments

Suppliers who want to receive remittance advice notification must be set up to receive external events. To receive external events, set Send Methods equal to **5** (External Events) on the EDI information tab in the Supplier Master program (P04012).

RTRAHDR

RTRAHDR is a single event that contains payment remittance advice header information that can be used by another system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTRAHDR	AP Payment Header	RTE	Single	04	D0400169A: Remittance Advice - Header	Included in container event RTRAOUT.

Conceptual Approach

A payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A real-time notification event is triggered in the payments system after payment is made in JD Edwards EnterpriseOne tables.

RTRADTL

RTRADTL is a single event that contains payment remittance advice detail information that can be used by another system.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTRADTL	AP Payment Details	RTE	Single	04	D0400169B: Remittance Advice - Detail	Included in container event RTRAOUT.

Conceptual Approach

A payment made in the JD Edwards EnterpriseOne system updates new records in the Accounts Payable - Matching Document (F0413) and Accounts Payable Matching Document Detail (F0414) tables. A detailed real-time notification event is triggered in the payments system after a payment is made in JD Edwards EnterpriseOne tables.

Accounts Payable Payment Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Accounts Payable Payment business object:

Program and Program Type	Description	Available Documentation
Update as Sent program (R47058) This is an EDI program.	<p>You run the Update as Sent program when your trading partner notifies you that they received your EDI payment transmission.</p> <p>When you run the Update as Sent program, the system updates the Processed (Y/N) field (EDPSP) to Y on the payment transactions in your EDI payment interface tables (F47056, F470561, F470562, F470563, and F47057). This prevents the translator software from reprocessing the same transactions.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Updating Processed Transactions

20 FMS: Accounts Payable Voucher

Accounts Payable Voucher Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Payable Voucher business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The AccountsPayableManager published business service (JP040000) manages these web service operations:</p> <ul style="list-style-type: none"> • J0400001 (processVoucher) • J0400002 (getVoucher)
Batch Import Programs	<p>This list includes the batch import programs for the Accounts Payable Voucher business object:</p> <ul style="list-style-type: none"> • Batch Voucher Processor Report (R04110ZA) • Store and Forward Voucher Batch Processor Report (R04110Z2) • Edit/Create Voucher (R47041) • Voucher Batch Processor program (R04110Z) • Inbound AP Claim program (R04110ZB) • Store and Forward Upload (R0411Z1)
Additional Interfaces	<p>The Accounts Payable Voucher business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • F0411 Interoperability Processing Options program (P0400048) <p>Use this program to enable outbound processing of payments. This is an interoperability program. See Understanding Outbound Interoperability Setup for Accounts Payable.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Accounts Payable Voucher Business Services - AccountsPayableManager

This section describes the available business services associated with the AccountsPayableManager published business service (JP040000), which is part of the Accounts Payable Voucher business object. The business services include:

Operation	Description
getVoucher	Use this operation to retrieve and review accounts payable vouchers from the JD Edwards EnterpriseOne Accounts Payable system.
processVoucher	Use this operation to add, delete or void accounts payable vouchers in the JD Edwards EnterpriseOne Accounts Payable system.

Accessing Javadoc for the AccountsPayableManager Web Service

To access Javadoc for the AccountsPayableManager web service and its related operations, review these Javadoc packages:

- JP040000 (AccountsPayableManager)
- J0400001 (processVoucher)
- J0400002 (getVoucher)

Prerequisites

Before you can use any of the accounts payable web service operations, you must install and configure the JD Edwards EnterpriseOne Accounts Payable system.

See *"Setting Up the Accounts Payable System" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide*.

getVoucher

The getVoucher web service operation is a database query operation that enables consumers to retrieve accounts payable vouchers from the JD Edwards EnterpriseOne Accounts Payable system.

The operation retrieves records based on the selection criteria that is passed in. If no selection criteria is passed in, the operation attempts to retrieve all records from the EnterpriseOne database, using the business service property to limit the number of records returned. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data. This table includes information about the business service properties that the getVoucher operation uses:

Group	Key	Description	Default Value
J0400002	J0400002_MAX_ROWS	Use this business service property to specify the maximum number of records the operation returns for the query.	0: Return All Records

Note:

It is recommended that this business service property is configured and used. Also, it is strongly recommended that the consumer pass in selection criteria when performing a query. If these two recommendations are not followed, the operation attempts to fetch all the records from the database, which can significantly impact performance.

Also be aware that this operation performs multiple data retrievals from the EnterpriseOne database. Initially, records are selected from the Accounts Payable Ledger table (F0411) based on the selection criteria that is passed in by the consumer. After the operation retrieves records from the F0411, processing continues, and additional records are retrieved from the Account Ledger table (F0911), based on the data in the F0411 records. This business service property limits the number of records initially retrieved from the F0411. Therefore, the total number of records that are returned to the consumer might be greater than the value that you enter in this property.

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

This table includes information that can help determine whether the getVoucher operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned to the consumer based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany

Question	Answer
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

processVoucher

The processVoucher web service operation is an inbound transaction operation that enables consumers to add, delete or void vouchers in the JD Edwards EnterpriseOne Accounts Payable system.

If the operation completes successfully, the consumer receives a success message. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes vouchers. This table includes information about the business service properties that the processVoucher operation uses:

Group	Key	Description	Default Value
J0400001	J0400001_GL_VERSION	Use this business service property to specify the version of the P0400074 that the operation uses to process voucher information.	ZJDE0001
J0400001	J0400001_GL_VCHR_MBF_VERSION	Use this business service property to specify the version of the P0900049 that the operation uses to process voucher information.	ZJDE0001

Note:

The published value object includes a field in the general ledger array that enables the user to enter a version of the P0900049 program for the F0911FSEditLine function. Though the user can enter a different version for each general ledger record, the operation uses only the first version. The operation first tries to retrieve the processing version from the value object. If no value is entered in the value object, the operation tries to retrieve a version from the business service properties. If no version is found, the operation uses version ZJDE0001.

The header class of the published value object also enables the user to specify a version of the P0400047 program, which is used for the F0411FSBeginDoc and F0411FSEditLine functions. Again, the operation first tries to retrieve the version from the value object. If no version is found, the operation then tries to retrieve a version from the business service properties. If no value is found, the operation uses version ZJDE0001.

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

This table includes information that can help determine whether the processVoucher operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the operation completes successfully, a success message is returned to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany <p>Additionally, if you are adding a voucher, the operation returns two arrays; one for the voucher records added and one for the general ledger records. In addition to the fields listed, the voucher array returns non-zero values for the DocumentPayItem field. It also returns a value for the DocumentPayItemExtension field, but that value can be 0.</p> <p>The general ledger array returns a value in the DocumentLineNumber field.</p>
<p>If I encounter errors while process a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

Accounts Payable Voucher Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Accounts Payable Voucher business object:

Program and Program Type	Description	Available Documentation
<p>Batch Voucher Processor Report (R04110ZA)</p> <p>This is a temporary table batch program.</p>	<p>After your custom program loads the transaction information into the Voucher Transaction - Batch Upload (F0411Z1) and Journal Entry Transactions - Batch File (F0911Z1) tables, run this program to process the information in the F0411Z1 and F0911Z1 tables and load it into the Accounts Payable Ledger (F0411) and Account Ledger (F0911) tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Batch Voucher Processor Report (R04110ZA)
<p>Store and Forward Voucher Batch Processor Report (R04110ZZ)</p> <p>This is a temporary table batch program.</p>	<p>Use this program to move the uploaded vouchers from the Voucher Transactions - Batch Upload table (F0411Z1) and the Journal Entry Transactions - Batch File (F0911Z1) tables to the Accounts Payable Ledger (F0411) and the Account Ledger (F0911) tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Store and Forward Voucher Batch Processor Report (R04110ZZ)
<p>Edit/Create Voucher (R47041)</p>	<p>Run the Edit/Create Voucher program to transfer inbound EDI vouchers to the Voucher Transactions - Batch Upload and the Journal Entry Transactions - Batch File tables (F0411Z1 and F0911Z1) without matching the voucher to a purchase order.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Transferring Inbound EDI Voucher Transactions
<p>Voucher Batch Processor program (R04110Z)</p> <p>This is an EDI program.</p>	<p>Run the Voucher Batch Processor program (R04110Z) to process invoice information. You can run the Voucher Batch Processor program in either proof or final mode. In proof mode, the program edits the transaction and sends any errors to the Employee Work Center. In final mode, the program updates the records in the F0411Z1 table and F0911Z1 table as processed. This action prevents further processing.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Understanding Voucher Processing
<p>Inbound AP Claim program (R04110ZB)</p>	<p>The Inbound AP Claim program (R04110ZB) also creates records in the Voucher Transactions - Batch Upload</p>	<p>See this topic:</p>

Program and Program Type	Description	Available Documentation
	table (F0411Z1) and the Journal Entries Transactions - Batch table (F0911Z1). You run the Inbound AP Claim program if your JD Edwards EnterpriseOne system is integrated with the Oracle Demantra Predictive Trade Planning software.	<ul style="list-style-type: none"> • <i>Understanding the Batch Voucher Process</i>
Store and Forward Upload (R0411Z1)	When you enter vouchers, the system updates the values from the pay item, tax rate area, tax explanation code, and tax item number fields from the F0411 or F0411Z1 tables to the F0911 or F0911Z1 tables.	See this topic: <ul style="list-style-type: none"> • <i>Understanding Programs that Update the Account Ledger Table (F0911)</i>

21 FMS: Accounts Receivable Deduction Activities

Accounts Receivable Deduction Activities Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Receivable Deduction Activities business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Accounts Receivable Deduction Activities business object:</p> <ul style="list-style-type: none"> RTDAOUT, which contains the RTDADTL single event.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Accounts Receivable Deduction Activities Real-Time Events

This section describes the available real-time events associated with the Accounts Receivable Deduction Activities business object, which includes the RTDAOUT event, which contains the RTDADTL single event.

RTDAOUT

RTDAOUT is used by the JD Edwards EnterpriseOne system to provide Accounts Receivable deduction information to third-party systems.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTDAOUT	AR Deduction Activities	RTE	<p>Container</p> <p>Container event for: RTDADTL (D03B0208B)</p>	H03B

Please see the real-time event that is contained by this event for information about the event fields.

Conceptual Approach

Deduction processing creates a new record in the A/R Deduction Management table (F03B40). The R45730 program is run to select the appropriate deduction records and to trigger a real-time notification. The system then publishes a notify event indicating that deductions were created in the A/R Deduction Management table. This event contains the recently selected deduction and its pertinent information.

Objects that Can Initiate the Event

The following object can initiate the event, and all single events contained within the event:

- R03B0208 - Extract and Export Open Siebel Deductions

RTDADTL

RTDADTL is used by the JD Edwards EnterpriseOne system to provide Accounts Receivable deduction information to third-party systems.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTDADTL	AR Deduction Activities Details	RTE	Single	H03B	D03B0208B: Deductions Activity Record	Included in container event RTDAOUT.

Conceptual Approach

Deduction processing creates a new record in the A/R Deduction Management table (F03B40). The R45730 program is run to select the appropriate deduction records and to trigger a real-time notification for the WSG integration. The WSG integration then publishes a notify event indicating that deductions were created in the A/R Deduction Management table. This event contains the recently selected deduction and its pertinent information.

22 FMS: Accounts Receivable Invoice

Accounts Receivable Invoice Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Receivable Invoice business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The ARInvoiceManager published business service (JP03B000) manages these business services:</p> <ul style="list-style-type: none"> • ProcessARInvoiceProcessor (J03B0001), which contains the method processARInvoice. • ARInvoiceQueryProcessor (J03B0002), which contains the method getARInvoice. • ARInvoiceRevisionsQueryProcessor (J03B0003), which contains the method getARInvoiceRevisions. • ProcessARInvoiceRecurringInformationProcessor (J03B0004), which contains the processARInvoiceRecurringInformation method.
Real-Time Events	<p>This list includes the real-time events for the Accounts Receivable Invoice business object:</p> <ul style="list-style-type: none"> • RTARIVOUT, which contains these single events: <ul style="list-style-type: none"> ○ RTARIVOUTA ○ RTARIVOUTB
Batch Import Programs	<p>This list includes the batch import programs for the Accounts Receivable Invoice business object:</p> <ul style="list-style-type: none"> • Batch Invoice Processor program (R03B11Z1A)
Additional Interfaces	<p>The Accounts Receivable Invoice business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • Invoice Entry MBF Processing Options (P03B0011) <p>Depending on the processing option settings on the Versions tab, this program can write outbound records that can be used by an external software system.</p> <p>See Setting Processing Options for Invoice Entry MBF Processing Options (P03B0011).</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Accounts Receivable Invoice Business Services - ARInvoiceManager

This section describes the ARInvoiceManager published business service (JP03B000), and the associated business services, which includes:

Operation	Description
processARInvoice (J03B0001)	Use this operation to process the AR Invoice information in the JD Edwards Enterprise One AR Management system. This operation enables you to add, change, delete/void invoices.
getARInvoice (J03B0002)	Use this operation to retrieve AR information from the Enterprise One database. The operation uses the user-defined search criteria to retrieve the records.
getARInvoiceRevisions (J03B0003)	Use this operation to retrieve AR revisions information from the Enterprise One database. The operation uses the user-defined search criteria to retrieve the records.

Accessing Javadoc for the ARInvoiceManager Web Service Operations

To access Javadoc for the ARInvoiceManager web service and its related operations, review these Javadoc packages:

- JP03B000 (ARInvoiceManager)
- J03B0001 (ProcessARInvoiceProcessor)
- J03B0002 (ARInvoiceQueryProcessor)
- J03B0003 (getARInvoiceRevisions)

processARInvoiceProcessor

processARInvoiceProcessor (J03B0001) is a web service operation that calls the processARInvoice method, based on the action of the user. This method enables consumers to add, change, or delete/void invoice records in the Enterprise One database.

When the processARInvoice method is called, the operation calls the BSFNs - ProcessInvoiceHeader_Wrapper (B03B1000), ProcessInvoiceDetail_Wrapper (B03B1000), ProcessGLInvoiceHeader_Wrapper (B03B1000), ProcessGLInvoiceDetail_Wrapper (B03B1000), CommitInvoice_Wrapper (B03B1000), and ARGLClearCache_Wrapper (B03B1000), to add, change, or delete/void Invoice records.

If the operation is successful, the system updates the AR invoice information in the JD Edwards Enterprise One Accounts Receivable system. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Note: Be aware of the following items when using this method:

- GL information, with the exception of the Remark field, cannot be updated if the invoice is posted.
- PayItem information, with the exception of the Remark field, cannot be updated if the invoice is fully-paid.
- Correct amounts should be used when modifying the invoice amount. The system recalculates the amounts if they are sent blank.

Setup Considerations

Before you can use this operation, you can set business properties to specify how the system processes the AR Information. This table includes information about the business service properties that the processARInvoice web service operation uses:

Group	Key	Description	Default Value
J03B0001	J03B0001_AR_GL_VERSION	Use this business service property to specify the version of the Invoice Entry program that will be used while processing the data.	ZJDE0001
J03B0001	J03B0001_AR_GL_VERSION	Use this business service property to specify the version of the Journal Entry program that will be used while processing the data.	ZJDE0001

Implementation Details

This table includes information that can help determine whether the ARInvoiceProcessor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer.</p> <p>The message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns a non-zero value for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentType • documentCompany • Entity - Customer

Question	Answer
	<ul style="list-style-type: none"> batchNumber payItem
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. In the event that the operation encounters errors, all JD Edwards Enterprise One tables are returned to their original state, and no updates are performed. No manual updates are necessary.
Does this operation use record reservation?	This operation reserves records in the JD Edwards Enterprise One system during the change or cancellation of an AR invoice.

ARInvoiceQueryProcessor

The getARInvoice (J03B0002) is a database query operation that enables consumers to retrieve and review AR Invoice records from the JD Edwards Enterprise One database. The operation returns one or more records to the consumers from the Customer Ledger (F03B11) and Account Ledger (F0911) tables that meet the specified search criteria.

Setup Considerations

Before you can use this operation, you can set business properties to specify how the system processes the invoice information. This table includes information about the business service properties that the getARInvoice web service operation uses:

Group	Key	Description	Default Value
J03B0002	J03B0002_MAX_ROWS_F03B11	Use this business service property to specify the maximum number of AR records the operation can return for a query. The default value is set to 100; however, this might limit the returned data when searching for AR records. If you find that limiting the number of results returned to 100 does not enable you to find the records you are searching for, it is acceptable to increase this value to a number that fits your business needs..	100
J03B0002	J03B0002_MAX_ROWS_F0911	Use this business service property to specify the maximum number of GL records	100

Group	Key	Description	Default Value
		<p>the operation can return for a query.</p> <p>The default value is set to 100; however, this might limit the returned data when searching for GL Records. If you find that limiting the number of results returned to 100 does not enable you to find the GL records you are searching for, it is acceptable to increase this value to a number that fits your business needs.</p>	

Implementation Details

The following table includes information that can help determine whether the ARInvoiceQueryProcessor (J03B0002) operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned, based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards Enterprise One system.</p> <p>At a minimum, the system returns a non-zero value for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentType • documentCompany • customerNumber • company • batchNumber • batchType • batchDate
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again.</p> <p>If the operation does not return any records, verify that records matching your query exist in the JD Edwards Enterprise One database.</p>
Does this operation use record reservation?	No. This operation does not reserve records.

Question	Answer

ARInvoiceRevisionsQueryProcessor

The getARInvoiceRevisions (J03B0003) is a database query operation that enables consumers to retrieve and review AR invoice revisions records from the JD Edwards Enterprise One database. The operation returns one or more records to the consumers from the Invoice Revisions (F03B112) table that meets the specified search criteria.

Setup Considerations

Before you can use this operation, you can set business properties to specify how the system processes the AR Revisions Information. This table includes information about the business service properties that the getARInvoiceRevisions web service operation uses:

Group	Key	Description	Default Value
J03B0003	J03B0003_MAX_ROWS_F03B112	<p>Use this business service property to specify the maximum number of AR records the operation can return for a query.</p> <p>The default value is set to 100; however, this might limit the returned data when searching for AR revision records. If you find that limiting the number of results returned to 100 does not enable you to find the records you are searching for, it is acceptable to increase this value to a number that fits your business needs..</p>	100

Implementation Details

The following table includes information that can help determine whether the ARInvoiceRevisionsQueryProcessor (J03B0003) operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, zero to many records are returned, based on the selection criteria.

Question	Answer
	<p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards Enterprise One system.</p> <p>At a minimum, the system returns a non-zero value for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentType • documentCompany • customerNumber • company • batchNumber • batchType • batchDate • payItem • payItemExtnNumber • invoiceRevisionCode
<p>If I encounter errors while process a transaction, do I need to reverse the transaction?</p>	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again.</p> <p>If the operation does not return any records, verify that records matching your query exist in the JD Edwards Enterprise One database.</p>
<p>Does this operation use record reservation?</p>	<p>No. This operation does not reserve records.</p>

ProcessARInvoiceRecurringInformationProcessor

ProcessARInvoiceRecurringInformationProcessor (J03B0004) is a web service operation that calls the processARInvoiceRecurringInformation method. This method enables consumers to change the recurring information of AR recurring invoice records in the JD Edwards Enterprise One database.

When the processARInvoiceRecurringInformation method is called, the operation calls the BSFN - ProcessInvoiceRecurringInfo (B03B1001) to change the recurring information of invoice records.

If the operation is successful, the system updates the AR invoice recurring information in the JD Edwards Enterprise One Accounts Receivable system. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Implementation Details

The following table includes information that can help determine whether the ProcessARInvoiceRecurringInformationProcessor (J03B0004) operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, no errors are returned to the consumer.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. In the event that the operation encounters errors, all JD Edwards Enterprise One tables are returned to their original state, and no updates are performed. No manual updates are necessary.
Does this operation use record reservation?	No. This operation does not reserve records.

Accounts Receivable Invoice Real-Time Events

This section describes the available real-time events associated with the Accounts Receivable Invoice business object which include:

- RTARIVOUT, which contains these single events:
 - RTARIVOUTA
 - RTARIVOUTB

RTARIVOUT

RTARIVOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems when an Invoice has been added, changed, or deleted in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTARIVOUT	Accounts Receivable Invoice	RTE	Container Container event for: RTARIVOUTA (D03B1011) RTAVIROUTB (D0901011)	03B

Conceptual Approach

An invoice is added, modified or deleted in the JD Edwards EnterpriseOne system, which creates, modifies, or deletes a record in the Customer Ledger table (F03B11) and Account Ledger table (F0911).

Objects that Can Initiate the Event

The following objects can initiate the event, and all single events contained within the event:

- B0000080 - Update A/R Post Status Flags
- B03B0011- Invoice Entry Master Business
- B03B0012 - F03B14 Apply Receipt to Invoice
- B03B0013 - Delete Invoice from F03B11
- B03B0015 - Void Invoice
- B03B0018 - A/R Speed Status Change
- B03B0019 - F03B13 Void/NSF Receipt
- B03B0023 - F03B14 Void/Delete Receipt
- B03B0112 - F03B21 - Write Notification Detail from Cache
- B03B0116 - F03B23 - BF Write Fee Journal
- B03B0179 - Void NSF Draft At Any Status
- N03B0077 - F03B41 - Void all Activity Records
- N03B0198 - Update A/R from Address Book
- P03B120 - Recurring Invoice Review
- P03B40 - Deduction Processing
- R15117 - Lease Global Update
- R155001 - Real Estate Management Statement
- R15903 - Priority Code Reapplication
- R15906 - Billing Detail Service/Tax Date

RTARIVOUTA

RTAVIROUTA is used by the JD Edwards EnterpriseOne system to inform third-party systems when an invoice pay item has been added, changed, or deleted in order to keep the two systems synchronized.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTARIVOUTA	AR Invoice Details	RTE	Single		D03B1011 - AR Inv Real Time Event	Included in container event RTARIVOUT.

Conceptual Approach

An Invoice PayItem is added, modified or deleted in the JD Edwards EnterpriseOne system, which creates, modifies, or deletes a record in the Customer Ledger table (F03B11).

RTARIVOUTB

RTARIVOUTB is used by the JD Edwards EnterpriseOne system to inform third-party systems when an GL distribution line for an invoice has been added, changed, or deleted in order to keep the two systems synchronized.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTARIVOUTB	AR Invoice GL Details	RTE	Single		D0901011 - GL Inv Real Time Event	Included in container event RTARIVOUT.

Conceptual Approach

The GL distribution line is added, modified or deleted for an invoice in the JD Edwards EnterpriseOne system, which creates, modifies, or deletes a record in the Account Ledger table (F0911).

Accounts Receivable Invoice Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Accounts Receivable Invoice business object:

Program and Program Type	Description	Available Documentation
Batch Invoice Processor program (R03B11Z1A). This is a temporary table batch program.	When you create invoices using an external system, you can transfer them to the JD Edwards EnterpriseOne Accounts Receivable system using the Batch Invoice Processor program (R03B11Z1A).	See this topic: <ul style="list-style-type: none"> Understanding Invoice Batch Processing

23 FMS: Accounts Receivable Receipts and Drafts

Accounts Receivable Receipts and Drafts Overview

This chapter provides detailed information about the business interfaces that are available for the Accounts Receivable Receipts and Accounts Receivable Drafts business objects. Both of these business objects use the same interfaces.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Accounts Receivable Receipts and Drafts business objects:</p> <ul style="list-style-type: none"> Update to Electronic Receipts program (R47111) Update Receipts Header Report (R03B551)

Note: The A/R Receipts business object also includes the `GetWriteOffProcessingOptions` method, which is managed by the `FinancialComplianceManager` published business service. See *FinancialComplianceManager Business Services*.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Accounts Receivable Receipts and Drafts Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Accounts Receivable Receipts and Drafts business objects:

Program and Program Type	Description	Available Documentation
Update to Electronic Receipts program (R47111)	To transfer inbound transmissions from the EDI tables to the F03B13Z1 table, run the Update to Electronic	See this topic:

Program and Program Type	Description	Available Documentation
<p>This is an inbound temporary table batch program.</p>	<p>Receipts program (R47111) from the appropriate menu. When you run the program from the Lock Box menu, the system processes 823 transactions. When you run the program from the Inbound Payment Order menu, the system processes 820 transactions.</p> <p>You can run this program in proof or final mode.</p>	<ul style="list-style-type: none"> • Understanding the Transfer of Inbound Transmissions
<p>Update Receipts Header Report (R03B551)</p>	<p>When you run this program, the system creates either unapplied receipts or drafts, or it creates logged receipt records. The system creates logged receipts when it cannot locate a payor or customer; otherwise, it creates unapplied receipts.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Update Receipts Header Report (R03B551)

24 FMS: Asset Master

Asset Master Overview

This chapter provides detailed information about the business interfaces that are available for the Asset Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The FixedAssetManager web service (JP120000) manages the processing of the getFixedAsset web service operation (J1200001).
Batch Import Programs	This list includes the batch import programs for the Asset Master business object: <ul style="list-style-type: none"> Inbound Asset Master Processor (R1201Z1I)
Batch Export Programs	This list includes the batch export programs for the Asset Master business object: <ul style="list-style-type: none"> Outbound Asset Master Processor program (R1201Z1O)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Asset Manager Business Services - FixedAssetManager

The FixedAssetManager web service (JP120000) manages the processing of the getFixedAsset web service operation (J1200001). You use this operation to query the JD Edwards EnterpriseOne Fixed Asset system, and review fixed asset records. If the system encounters errors during processing, the Fixed Asset Manager returns those errors to the consumer.

Accessing Javadoc for the Fixed Asset Manager Web Service Operations

To access Javadoc for the Fixed Asset Manager web service and its related operations, review these Javadoc packages:

- JP120000 (Fixed Asset Manager)
- J1200001 (getFixedAsset)

Prerequisites

Before you use the Fixed Asset Manager web service or the `getFixedAsset` operation, you must install and configure the JD Edwards EnterpriseOne Fixed Asset system. Additionally, fixed asset records must exist in the EnterpriseOne database.

See *JD Edwards EnterpriseOne Applications Fixed Assets Implementation Guide* .

getFixedAsset

The `getFixedAsset` web service operation is a database query operation that enables consumers to retrieve and review fixed asset records that exist in the Asset Master table (F1201) in the JD Edwards EnterpriseOne Fixed Asset system.

If the operation completes successfully, fixed asset records, along with any warnings encountered, are returned to the consumer. If the operation encounters errors during processing, the operation stops processing and returns the errors to the consumer.

Note: If no records, warnings, or errors are returned to the consumer, it is likely that the operation completed successfully, but no records matching the selection criteria exist in the JD Edwards EnterpriseOne database.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system `getFixedAsset` operation processes and returns data. This table includes information about the business service properties that the `getFixedAsset` operation uses:

Group	Key	Description	Default Value
J1200001	J1200001_MAX_ROWS	Use this business service property to specify the maximum number of rows that the operation returns to the consumer.	100

Note: It is recommended that you set this property to a value other than **0** (zero). Setting this property to **0** enables the operation to retrieve all records that match the selection criteria. Additionally, it is recommended that you specify selection criteria when performing a query. Sending a blank query and setting this option to **0** could significantly impact system performance.

Implementation Details

This table includes information that can help determine whether the `getFixedAsset` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns fixed asset records to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns a non-zero value for the assetID field.</p>
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and your business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

Asset Master Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Asset Master business object:

Program and Program Type	Description	Available Documentation
<p>Inbound Asset Master Processor (R1201Z1I)</p> <p>This is an inbound interoperability program.</p>	<p>You run the Inbound Asset Master Processor program (F1201Z1I) to copy the information from the unedited transaction table to the Asset Master File table (F1201).</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Processing Inbound Interoperability for Fixed Assets

Asset Master Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Asset Master business object:

Program and Program Type	Description	Available Documentation
<p>Outbound Asset Master Processor program (R1201Z10)</p> <p>This is an outbound interoperability program.</p>	<p>The Outbound Asset Master Processor program fetches the F986113 record and retrieves the name and version of the custom UBE from the F0047 table.</p> <p>The Outbound Asset Master Processor program passes information about the transactions to the custom UBE. The custom UBE then retrieves the records from the F1201Z1 table and processes that information.</p>	<p>See this topic:</p> <ul style="list-style-type: none">• <i>Processing Outbound Interoperability for Fixed Assets</i>

25 FMS: Bank Statement

Bank Statement Overview

This chapter provides detailed information about the business interfaces that are available for the Bank Statement business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	Available batch import programs: <ul style="list-style-type: none"> Load Bank Statement report (R09612)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Bank Statements Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Bank Statements business object:

Program and Program Type	Description	Available Documentation
Load Bank Statement report (R09612)	The Load Bank Statement program loads data from the F09610 and F09611 staging tables to the F0916 and F0917 bank statement tables in the JD Edwards EnterpriseOne General Accounting system. This program formats data according to the type of electronic format used (BAI2 or SWIFT) and edits values in the bank statement tables to ensure that they are valid. The Load Bank Statement report (R09612) shows the number of records loaded and, if applicable, not loaded.	See this topic for additional information: <ul style="list-style-type: none"> Understanding the Load Bank Statement Program

26 FMS: Budgets

Budgets Overview

This chapter provides detailed information about the business interfaces that are available for the Budgets business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	Available batch import programs: <ul style="list-style-type: none"> • Upload/Conversion program (R14110)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Budgets Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Budgets business object:

Program and Program Type	Description	Available Documentation
Upload/Conversion program (R14110) This is a temporary table import program.	After you review and, if necessary, revise your budget data on the Revise Batch Account Balances form, run the Upload/Conversion program (R14110) to copy the data from the Account Balance - Batch File (F0902Z1) to the F0902 table.	See this topic: <ul style="list-style-type: none"> • Understanding the Upload/Conversion Program

27 FMS: Cost Analyzer Balances

Cost Analyzer Balances Overview

This chapter provides detailed information about the business interfaces that are available for the Cost Analyzer Balances business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	Available batch import programs: <ul style="list-style-type: none"> External Cost Analyzer Balances Processor program (R1602Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Cost Analyzer Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Cost Analyzer business object:

Program and Program Type	Description	Available Documentation
External Cost Analyzer Balances Processor program (R1602Z1)	<p>After you review and revise the balance records in the External Cost Analyzer Balances - Interoperability Table (F1602Z1), run the External Cost Analyzer Balances Processor program (R1602Z1) to upload the records to the Cost Analyzer Balances table (F1602).</p> <p>Before the External Cost Analyzer Balances Processor program uploads the records to the F1602 table, it edits the records to ensure that the information is in the correct format. After the program edits the records, it generates a report that lists the number of records selected from the F1602Z1</p>	<p>See this topic for additional information:</p> <ul style="list-style-type: none"> <i>Understanding External Cost Analyzer Balances Uploads</i>

Program and Program Type	Description	Available Documentation
	table, the number of records added to and updated in the F1602 table, and the number of records that failed.	

28 FMS: Credit Card Data

Credit Card Data Overview

This chapter provides detailed information about the business interfaces that are available for the Credit Card Data business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The Credit Card Data business object uses the CreditCardTransactionQueryProcessor (J09E0004) business service. This service was created for use with the Mobile Expense Management application, and is documented with that application.</p> <p>For details about this service, see <i>Mobile Expense Management Business Services</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

29 FMS: Customer Master Data

Customer Master Data Overview

This chapter provides detailed information about the business interfaces that are available for the Customer Master Data business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The CustomerManager published business service (JP010020) manages these business services:</p> <ul style="list-style-type: none"> • <code>getCustomer</code> (J0100022) • <code>getCustomerV2</code> (J0100022) • <code>getCustomerV3</code> (J0100022) • <code>getCustomerCreditInformation</code> (J0100023) • <code>getCustomerCreditInformationV2</code> (J0100023) • <code>processCustomer</code> (J0100021) • <code>processCustomerV2</code> (J0100021) • <code>processCustomerV3</code> (J0100021) <p>The CustomerAndContactManager published business service (JP010050) manages these business services:</p> <ul style="list-style-type: none"> • <code>processCustomer</code> (J0100021) <p>This method is documented under the CustomerManager published business service (JP010020), which is documented in this chapter.</p> <ul style="list-style-type: none"> • <code>processContact</code> (J0100003) <p>This method is documented under the AddressBookManager published business service.</p> <p>See Address Book Business Services - AddressBookManager.</p> <p>The Customer Master Data business object also uses the <code>getCustomerCreditLimits</code> method (J000044), which is managed by the FinancialComplianceManager. See FinancialComplianceManager Business Services.</p>
Real-Time Events	<p>This list includes the real-time events for the Customer Master Data business object:</p> <ul style="list-style-type: none"> • RTCMOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTCMHDR ○ RTCMPHOUT ○ RTCMEAOUT ○ RTCMCCOUT • RTCMOUT2, which is a container event for these single events:

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> ○ RTCMHDR2 ○ RTCMEAOUT2 ○ RTCMAAOUT ○ RTCMPHOUT2 ○ RTCMCCOUT2 • RTCMOUT3, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTCMHDR3 ○ RTCMAAOUT ○ RTCMEAOUT2 ○ RTCMPHOUT2 ○ RTCMCCOUT2 ○ RTABCCOUT <p>Note that several of the single events contained within RTCMOUT3 are also contained within RTCMOUT2. The documentation for those events is located under the RTCMOUT2 container event.</p>
Batch Import Programs	<p>This list includes the batch import programs for the Customer Master Data business object:</p> <ul style="list-style-type: none"> • Customer Master Batch Upload program (R03010Z) • SCP Customer Master Information Extract program (R34A530)
Additional Interfaces	<p>The Customer Master Data business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • Customer Master MBF - PO (P0100042) <p>Depending on the processing option settings on the Outbound tab, this program can write outbound interoperability records that can be used by an external software system.</p> <p>See Setting Processing Options for Customer Master MBF - PO (P0100042)</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Customer Master Data Business Services - CustomerManager

This section describes the CustomerManager (JP010020) business services associated with the Customer Master Data business object, which include:

Operation	Description
getCustomer (J0100022)	Use this operation to retrieve and review customer information, including address, phone numbers, electronic addresses, and contact information, from the JD Edwards EnterpriseOne database.
getCustomerV2 (J0100022)	Use this operation to retrieve and review customer information such as address, phone number, electronic address, contact information, and global locator number (GLN) from the JD Edwards EnterpriseOne database.
getCustomerV3 (J0100022)	Use this operation to retrieve and review customer information such as address, phone number, electronic address, contact information, and global locator number (GLN) from the JD Edwards EnterpriseOne database. You can also search for customers by address using this operation. This is a LIKE query method that is case insensitive.
getCustomerCreditInformation (J0100023)	Use this operation to retrieve and review customer credit information from the JD Edwards EnterpriseOne Accounts Receivable system.
getCustomerCreditInformationV2 (J0100023)	Use this operation to retrieve and review customer credit information, including GLN, from the JD Edwards EnterpriseOne Accounts Receivable system.
processCustomer (J0100021)	Use this operation to add, change, or delete customer information in the JD Edwards EnterpriseOne system.
processCustomerV2 (J0100021)	Use this operation to add, change, or delete customer information, including GLN, in the JD Edwards EnterpriseOne system.
processCustomerV3 (J0100021)	Use this operation to add, change, or delete customer information, including GLN, in the JD Edwards EnterpriseOne system. Using this method, the user can provide the state description instead of the state code, and the system will derive the state code.

Accessing Javadoc for the CustomerManager Web Service Operations

To access Javadoc for the CustomerManager web service and its related operations, review these Javadoc packages:

- JP010020 (CustomerManager)
- J0100022 (getCustomer)
- J0100022 (getCustomerV2)
- J0100022 (getCustomerV3)
- J0100023 (getCustomerCreditInformation)
- J0100023 (getCustomerCreditInformationV2)
- J0100021 (processCustomer)
- J0100021 (processCustomerV2)
- J0100021 (processCustomerV3)

Prerequisites

Before using this web service, or any of the related operations, you must install and configure the JD Edwards EnterpriseOne Address Book and Accounts Receivable systems. Additionally, you should be familiar with customer and contact information that is stored in the Address Book system.

See *"Setting Up the JD Edwards EnterpriseOne Address Book System" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide* , and *"Entering Address Book Records" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide* .

See *"Setting Up the JD Edwards EnterpriseOne Accounts Receivable System" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* , and *"Setting Up Customer Master Information" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* .

getCustomer

The getCustomer web service operation is a database query operation that enables consumers to retrieve and review customer information from the JD Edwards EnterpriseOne database. This operation retrieves these items for each customer, if the data exists in the database:

- Address book information, including:
 - Phone numbers.
 - Electronic addresses.
- Customer master information.
- Who's Who line 0 contact information.
- Parent address information.

The operation uses the business view V03012JA to select records. Then the operation uses the AB – Get Parent Address business function (B0100002) to retrieve the parent address for each record the operation returns. If the operation is successful, zero to many records are returned. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Supported Functionality

This section discusses the functionality that the getCustomer operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The getCustomer operation does not retrieve related-person information for a specified customer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the database. This table includes information about the business service properties that the getCustomer operation uses:

Group	Key	Description	Default Value
J0100022	J0100022_MAX_ROWS	Use this business service property to specify the maximum number of records the operation can return for a query.	100

Note: It is recommended that this business service property is configured and used. Also, it is strongly recommended that the consumer pass in selection criteria when performing a query. If these two recommendations are not followed, the operation attempts to fetch all the records from the database, which can significantly impact performance.

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the `getCustomer` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned, based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityName entityTypeCode company
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

getCustomerV2

Review the information in the `getCustomer` section of this chapter before using the `getCustomerV2` web service operation. All of the information in the `getCustomer` section also applies to the `getCustomerV2` operation.

The `getCustomerV2` operation is a version of the `getCustomer` web service operation. This web service operation is used to retrieve customer information with associated GLNs. This version executes the `getCustomer (J0100022)` web service operation.

getCustomerV3

Review the information in the `getCustomer` and `getCustomerV2` sections of this chapter before using the `getCustomerV3` web service operation. All of the information in those sections also applies to the `getCustomerV3` operation.

The `getCustomerV3` operation is a version of the `getCustomer` web service operation. This web service operation is used to retrieve customer information using a LIKE query that is case insensitive. This method also enables users to search for customers by address. This version executes the `getCustomer (J0100022)` web service operation.

getCustomerCreditInformation

The `getCustomerCreditInformation` web service operation is an inbound transaction operation that enables consumers to retrieve customer credit information from the JD Edwards EnterpriseOne database.

The operation uses several business functions, including the `CreditCheckProcessing` business function (B4200420,) to process credit information. If the operation is successful, credit information for the specified entity is returned to the consumer. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Prerequisite

Before you use this operation, you must enter customer credit information.

See *"Setting Up Credit and Collections Management" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide*.

Implementation Details

This table includes information that can help determine whether the `getCustomerCreditInformation` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, records containing credit information for the specified entities are returned to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityID creditHoldExempt

Question	Answer
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation does not perform transactions. No data in the EnterpriseOne system is updated when this operation processes. If you encounter errors when processing this operation, review the error messages, along with your selection criteria, and retry your query. Additionally, you can verify that records matching your selection criteria exist in the EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

getCustomerCreditInformationV2

Review the information in the `getCustomerCreditInformation` section of this chapter before using the `getCustomerCreditInformationV2` web service operation. All of the information in the `getCustomerCreditInformation` section also applies to the `getCustomerCreditInformationV2` operation.

The `getCustomerCreditInformationV2` operation is a version of the `getCustomerCreditInformation` web service operation. This web service is operation is used to retrieve customer credit information with associated GLNs. This version executes the `getCustomerCreditInformation (J0100023)` web service operation.

processCustomer

The `processCustomer` web service operation is an inbound transaction operation that enables consumers to add, change, or delete customer records from the EnterpriseOne database.

The operation uses the `CustomerMasterMBF` business function (N0100042) to process customer information. If the operation is successful, the system returns a completion message to the consumer. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Note: This operation calls the `processAddressBook` web service operation to process customer address information. If the operation encounters errors while processing address information, processing stops and those errors are returned to the consumer.

See *Address Book Business Services - AddressBookManager*.

Supported Functionality

This section discusses the functionality that the `processCustomer` operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes customer and address information. This table includes information about the business service properties that the processCustomer operation uses:

Group	Key	Description	Default Value
J0100021	J0100021_CUS_MBF_VERSION	Use this business service property to specify the version of the CustomerMasterMBF business function (N0100042) that the operation uses when processing customer data.	ZJDE0001
J0100021	J0100021_AB_MBF_VERSION	Use this business service property to specify the version of the AddressBookMasterMBF (N0100041) that the operation uses when processing customer address information. You should enter a different version than you use in the business service properties of the processAddressBook business service.	ZJDE0001

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the processCustomer operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the consumer receives a successful return message.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityID entityName

Question	Answer
	<ul style="list-style-type: none"> entityTypeCode
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

processCustomerV2

Review the information in the processCustomer section of this chapter before using the processCustomerV2 web service operation. All of the information in the processCustomer section also applies to the processCustomerV2 operation.

The processCustomerV2 operation is a version of the processCustomer web service operation. This web service operation is used to add, update, or delete customer information with associated GLNs. This version executes the getCustomerCreditInformation (J0100021) web service operation.

processCustomerV3

Review the information in the processCustomer and processCustomerV2 sections of this chapter before using the processCustomerV3 web service operation. All of the information in the those sections also applies to the processCustomerV3 operation.

The processCustomerV3 operation is a version of the processCustomer web service operation. This web service operation is used to add, update, or delete customer information with associated GLNs. This version allows the user to pass in the state description rather than the state code, and the system will derive the state code from the entered state description. This version executes the getCustomerCreditInformation (J0100021) web service operation.

Customer Master Data Business Services - CustomerAndContactManager

This section describes CustomerAndContactManager published business service (JP010050). This published business service processes two methods, which are both contained under other published business services.

The CustomerAndContactManager published business service manages these business services:

- processCustomer (J0100021)
This method is documented under the CustomerManager published business service (JP010020, which is documented in this chapter).
- processContact(J0100003)

This method is documented under the AddressBookManager published business service.

See *Address Book Business Services - AddressBookManager*.

Customer Master Data Real-Time Events

This section describes the available real-time events associated with the Customer Master Data business object, which include:

- RTCMOUT, which is a container event for these single events:
 - RTCMHDR
 - RTCMPHOUT
 - RTCMEAOUT
 - RTCMCCOUT
- RTCMOUT2, which is a container event for these single events:
 - RTCMHDR2
 - RTCMEAOUT2
 - RTCMAAOUT
 - RTCMPHOUT2
 - RTCMCCOUT2
- RTCMOUT3, which is a container event for these single events:
 - RTCMHDR3
 - RTCMAAOUT *
 - RTCMEAOUT2
 - RTCMPHOUT2 *
 - RTCMCCOUT2 *
 - RTABCCOUT *

* Note that many of the single events contained within RTCMOUT3 are also contained within RTCMOUT2. Documentation for those single events can be found in the RTCMOUT2 section.

RTCMOUT

RTCMOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Note: There are three customer-related RTEs, which include RTCMOUT, RTCMOUT2, and RTCMOUT3. RTCMOUT is documented in this section, and was the original customer RTE. RTCMOUT2 and RTCMOUT3 were created to handle additional customer-related changes that were not included in RTCMOUT. RTCMOUT2 and RTCMOUT3 are discussed in the next sections.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTCMOUT	Customer Master	RTE	Container Container event for: <ul style="list-style-type: none"> • RTCMHDR (D0100077A) • RTCMPHOUT (D0100077B) • RTCMEAOUT (D0100077C) • RTCMCCOUT (D0100077D) 	01

Please see the appropriate real-time events that are contained by this event for more information about the fields, conceptual approach, and processes that can initiate this event.

Special Setup

RTCMOUT has some additional special setup requirements due to the logic in the real-time wrapper.

The processing option for version ZJDE001 of the Customer Master Real Time Event Wrapper Application program (P0100077) determines whether RTCMOUT is always sent or is sent only if certain fields are changed. The purpose of the processing option is to enable filtering of the outbound event, RTCMOUT.

When the processing option is 0, blank, or null, no filtering occurs, and RTCMOUT is sent whenever a change is made to a customer master record. If the processing option is 1, an event is sent only if one of the five territory alignment fields have been changed. Events always occur on an Add or Delete. The five territory alignment fields are Industry Classification Code, City, State, Zip, and Country. If one of these fields have changed and the filter is turned off, a flag is set in a cTerritoryAlign parameter in the event data structure. This flag is a signal that something has changed in those five fields.

Objects that Can Initiate the Event

These objects can initiate the RTCMOUT event and the single events contained within this event:

- N0100041 - Address Book MBF
- N0100042 - Customer Master MBF
- P0111 - Whos Who application
- B0100087 - Who's Who MBF
- B0100089 - Electronic Address MBF
- B0100090 - Phones MBF

RTCMHDR

RTCMHDR is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMHDR	Customer Master Header	RTE	Single	01	D0100077A: Customer Master Real Time Record	Included in container event RTCMOUT.

Conceptual Approach

A customer record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in a number of tables such as Customer Master by Line of Business (F03012) and Address Book Master (F0101) tables. A real-time notification event is triggered in the Customer Master or Address Book Master Business Function after data is modified to any customer table.

RTCMPHOUT

RTCMPHOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer phone number changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMPHOUT	Customer Master Phone	RTE	Single	01	D0100077B: Customer Phones Wrapper Data Structure	Included in container event RTBUOUT

Conceptual Approach

A phone number is changed in the JD Edwards EnterpriseOne system for a customer. This change is sent out using real-time events. A given customer may have an infinite number of phone numbers.

RTCMEAOUT

RTCMEAOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer email address changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMEAOUT	Customer Electronic Address	RTE	Single	01	D0100077C: Customer Master Real Time Email Structure	Included in container event RTCMOUT

Conceptual Approach

An email address is added or changed in the JD Edwards EnterpriseOne system for a customer. This change is sent out using real-time events. A given customer may have an infinite number of email addresses.

RTCMCCOUT

RTCMCCOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer category code changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMCCOUT	Customer Category Codes	RTE	Single	01	D0100077D: Customer Master Real Time Cat Codes Structure	Included in container event RTCMOUT.

Conceptual Approach

A customer's category code data is added or changed in the JD Edwards EnterpriseOne system. This change is sent out using real-time events.

RTCMOUT2

This section provides the details of the RTCMOUT2 RTE.

The RTCMOUT2 RTE structure publishes customer data in a format that can be easily consumed by external systems. RTCMOUT2 is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTCMOUT2	Customer Master Version 2	RTE	Container Container event for: <ul style="list-style-type: none"> • RTCMHDR2 • RTCMEAOUT2 • RTCMAAOUT • RTCMCCOUT2 • RTABCCOUT 	01

Special Setup

RTCMOUT2 has some additional special setup requirements due to the logic in the real-time wrapper.

The processing option for version ZJDE0001 of the Customer Master Real Time Event Wrapper Application program (P0100077) determines whether RTCMOUT2 is always sent or is sent only if certain fields are changed. The purpose of the processing option is to enable filtering of the outbound event, RTCMOUT2.

When the processing option is 0, blank, or null, no filtering occurs, and RTCMOUT2 is sent whenever a change is made to a customer master record. If the processing option is 1, an event is sent only if one of the five territory alignment

fields has been changed. Events always occur on an Add or Delete. The five territory alignment fields are Industry Classification Code, City, State, Zip, and Country. If one of these fields has changed and the filter is disabled, a flag is set in a cTerritoryAlign parameter in the event data structure. This flag is a signal that something has changed in those five fields.

Objects that Can Initiate the Event

These objects can initiate the RTCMOUT2 event, and the single events contained within the event:

- N0100041 -Address Book MBF
- N0100042 -Customer Master MBF
- B0100087 - Who's Who MBF
- R01840 - Effective Address Update

RTCMHDR2

RTCMHDR2 is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMHDR2	Customer Header Version 2	RTE	Single	01	D0100096A	Included in container event RTCMOUT2.

Conceptual Approach

A customer record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in a number of tables such as the Customer Master by Line of Business (F03012) and Address Book Master (F0101) tables. A real-time notification event is triggered in the Customer Master or Address Book Master Business Function after data is modified in any customer table.

RTCMEAOUT2

RTCMEAOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer email address changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMEAOUT2	Customer Elec Addr Version 2	RTE	Single	01	D0100096C	Included in container event RTCMOUT2.

Conceptual Approach

An email address is added or changed in the JD Edwards EnterpriseOne system for a customer. This change is sent using RTEs. A given customer may have an infinite number of email addresses.

RTCMAAOUT

RTCMAAOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of alternate address data changes for customers.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMAAOUT	Customer Alternate Address	RTE	Single	01	D0100096D	Included in container event RTCMOUT2.

Conceptual Approach

An alternate address is added or changed in the JD Edwards EnterpriseOne system for a customer. This change is sent using RTEs. A given customer may have an infinite number of alternate addresses.

RTCMPHOUT2

RTCMPHOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer phone number changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMPHOUT2	Customer Phone Version 2	RTE	Single	01	D0100096E	Included in container event RTCMOUT2.

Conceptual Approach

A phone number is changed in the JD Edwards EnterpriseOne system for a customer. This change is sent using RTEs. A given customer may have an infinite number of phone numbers.

RTCMCCOUT2

RTCMCCOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of customer category code changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMCCOUT2	Customer Cat Codes Version 2	RTE	Single	01	D0100096I	Included in container event RTCMOUT2.

Conceptual Approach

A customer's category code data is added or changed in the JD Edwards EnterpriseOne system. This change is sent out using RTEs.

RTABCCOUT

RTABCCOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of address book category code changes.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABCCOUT	Address Book Category Codes	RTE	Single	01	D0100096H	Included in container event RTCMOUT2.

Conceptual Approach

A customer's address-book category-code data is added or changed in the JD Edwards EnterpriseOne system. This change is sent using real-time events.

RTCMOUT3

RTCMOUT3 is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTCMOUT3	Customer Master Version 3	RTE	Container Container event for: <ul style="list-style-type: none"> • RTCMHDR3 (D0100091A) • RTCMAAOUT • RTCMEAOUT2 • RTCMPHOUT2 • RTCMCCOUT2 • RTABCCOUT 	01

Please see the appropriate real-time events that are contained by this event for more information about the fields, conceptual approach, and processes that can initiate this event.

Special Setup

RTCMOUT3 has some additional special setup requirements due to the logic in the real-time wrapper.

The processing option for version ZJDE001 of the Customer Master Real Time Event Wrapper Application program (P0100077) determines whether RTCMOUT3 is always sent or is sent only if certain fields are changed. The purpose of the processing option is to enable filtering of the outbound event, RTCMOUT3.

When the processing option is 0, blank, or null, no filtering occurs, and RTCMOUT3 is sent whenever a change is made to a customer master record. If the processing option is 1, an event is sent only if one of the five territory alignment

fields have been changed. Events always occur on an Add or Delete. The five territory alignment fields are Industry Classification Code, City, State, Zip, and Country. If one of these fields have changed and the filter is turned off, a flag is set in a cTerritoryAlign parameter in the event data structure. This flag is a signal that something has changed in those five fields.

Objects that Can Initiate the Event

These objects can initiate the RTCMOUT3 event, and the single events contained within the event:

- N0100041 - Address Book MBF
- N0100042 - Customer Master MBF
- B0100078 - Who's Who MBF
- B0100089 - Electronic Address MBF
- B0100090 - Phones MBF

RTCMHDR3

RTCMHDR3 is used by the JD Edwards EnterpriseOne system to notify third-party systems of customer additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCMHDR3	Customer Header Version 3	RTE	Single	01	D0100091A: Customer Master Real Time Record	Included in container event RTCMOUT.

Conceptual Approach

A customer record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in a number of tables such as Customer Master by Line of Business (F03012) and Address Book Master (F0101) tables. A real-time notification event is triggered in the Customer Master or Address Book Master Business Function after data is modified to any customer table.

Customer Master Data Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Customer Master Data business object:

Program and Program Type	Description	Available Documentation
Customer Master Batch Upload program (R03010Z),	The Customer Master Batch Upload program accesses the processed address book records, retrieves the corresponding customer information in	See this topic:

Program and Program Type	Description	Available Documentation
	the F03012Z1 table and uploads it to the Customer Master by Line of Business table.	<ul style="list-style-type: none"><li data-bbox="781 287 1068 338">• <i>Uploading Customer Master Batch Information</i>
SCP Customer Master Information Extract program (R34A530)	The SCP Customer Master Information Extract program (R34A530) extracts customer information to a text file. The system also extracts the customer's parent address number and description.	From Interactive Versions, (fast path iv), search for R34A530.

30 FMS: Employee Profile

Employee Profile Overview

This chapter provides detailed information about the business interfaces that are available for the Employee Profile business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The Employee Profile business object uses the getExpenseManagementPolicy (J09E0007) business service. This service was created for use with the Mobile Expense Management application, and is documented with that application.</p> <p>For details about this service, see <i>Mobile Expense Management Business Services</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

31 FMS: Expense Report

Expense Report Overview

This chapter provides detailed information about the business interfaces that are available for the Expense Report business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>ExpenseReportManager (J09E000)</p> <p>This published business service, along with the associated business services, were created as part of a pre-built integration for Mobile Expense Management. The documentation for this service is included in the solution-specific documentation for JD Edwards EnterpriseOne Mobile Applications.</p> <p>See these topics:</p> <ul style="list-style-type: none">• Mobile Expense Management Business Services• Business Services for Pre-Built Integrations.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

32 FMS: Expense Report Approvals

Expense Report Approvals Overview

This chapter provides detailed information about the business interfaces that are available for the Expense Report Approvals business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The Expense Report Approvals business object uses the ExpenseReportProcessor (J09E0005) business service. This service was created for use with the Mobile Expense Management application, and is documented with that application.</p> <p>For details about this service, see <i>Mobile Expense Management Business Services</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

33 FMS: Supplier Master Data

Supplier Master Data Overview

This chapter provides detailed information about the business interfaces that are available for the Supplier Master Data business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The SupplierManager published web service (JP010030) manages the processing of these web service operations:</p> <ul style="list-style-type: none"> • processSupplier (J0100031) • processSupplierV2 (J0100031) • processSupplierV3 (J0100031) • getSupplier (J0100032) • getSupplierV2 (J0100032) • getSupplierV3 (J0100032)
Real-Time Events	<p>This list includes the real-time events for the Supplier Master Data business object:</p> <ul style="list-style-type: none"> • RTSMOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSMHDR ○ RTSMPHOUT ○ RTSMEAOUT • RTSMOUT2, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSMHDR2 ○ RTSMPHOUT2 ○ RTSMEAOUT2
Batch Import Programs	<p>This list includes the batch import programs for the Supplier Master Data business object:</p> <ul style="list-style-type: none"> • Supplier Master Batch Upload program (R04010Z)
Additional Interfaces	<p>The Supplier Master Data business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • Supplier Master MBF - PO (P0100043) <p>Depending on the processing option settings on the Outbound tab, this program can write outbound interoperability records that can be used by an external software system.</p> <p>See Setting Processing Options for Supplier Master MBF - PO (P0100043).</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Supplier Master Data Business Services - SupplierManager

This section describes the available business services associated with the Supplier Master Data business object.

The SupplierManager web service (JP010030) manages the processing of supplier-related web service operations. This table includes a description of the supplier manager web service operations:

Operation	Description
processSupplier (J0100031)	Use this operation to complete these tasks within the JD Edwards EnterpriseOne Address Book, JD Edwards EnterpriseOne Accounts Payable, and JD Edwards EnterpriseOne Accounts Receivable systems. <ul style="list-style-type: none"> • Add, change or delete supplier records. • Add, change or delete supplier address book records. • Add, change or delete supplier phone numbers. • Add, change or delete supplier electronic address records.
processSupplierV2 (J0100031)	Use this operation to add, update, or delete supplier information in the JD Edwards EnterpriseOne Address Book, Accounts Payable, and Accounts Receivable systems. This version includes global locator number (GLN) in the process.
processSupplierV3 (J0100031)	Use this operation to add, update, or delete supplier information in the JD Edwards EnterpriseOne Address Book, Accounts Payable, and Accounts Receivable systems. This version includes global locator number (GLN) and the voucher match automation attributes in the process.
getSupplier (J0100032)	Use this operation to query the JD Edwards EnterpriseOne database for existing supplier information.
getSupplierV2 (J0100032)	Use this operation to retrieve supplier information, including GLN, from the JD Edwards EnterpriseOne database.
getSupplierV3 (J0100032)	Use this operation to retrieve supplier information, including GLN and the voucher match automation attributes, from the JD Edwards EnterpriseOne database.

Prerequisites

Before using the Supplier Manager web service, or any of the related web service operations, you must install and configure the JD Edwards EnterpriseOne Address Book and JD Edwards EnterpriseOne Accounts Payable systems.

See [JD Edwards EnterpriseOne Applications Address Book Implementation Guide](#) .

See [JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide](#) .

Accessing Javadoc for the Supplier Manager Web Service Operations

To access Javadoc for the Supplier Manager web service and its related operations, review these Javadoc packages:

- JP010030 (SupplierManager)
- J0100031 (processSupplier)
- J0100031 (processSupplierV2)
- J0100031 (processSupplierV3)
- J0100032 (getSupplier)
- J0100032 (GetSupplierV2)
- J0100032 (getSupplierV3)

processSupplier

The processSupplier web service operation is an inbound transaction operation that enables consumers to process supplier information within the JD Edwards EnterpriseOne system. The processSupplier web service operation enables source systems to process supplier information in JD Edwards EnterpriseOne in a real-time fashion. The web service uses JD Edwards EnterpriseOne master business functions to process the supplier information to add, change or delete supplier related records.

The consumer can complete these tasks in the JD Edwards EnterpriseOne Address Book, JD Edwards EnterpriseOne Accounts Payable, and JD Edwards EnterpriseOne Accounts Receivable systems:

- Add, change or delete supplier records.
- Add, change or delete address book records.
- Add, change or delete phone number records.
- Add, change or delete electronic address records.

After processing supplier information, the processSupplier web service operation calls the processAddressBook web service operation to process the address book record. If the operation encounters no errors then the processAddressBook web service operation calls the processPhones and processElectronicAddresses web service operations. The AddressBookProcessor web service operation will return either success messages or error messages if the transaction failed.

If AddressBookProcessor returns success, the processSupplier operation calls the SupplierMaster business function (N0100043) to process the supplier records regardless of the success or failure of the phone and electronic address processing.

Note: The processSupplier operation processes address book information and supplier records even if the processPhones and processElectronicAddresses web service operations fail. If the system encounters errors while processing the address book or supplier record information the system rolls back all transactions.

Every change in the supplier address book is a net change. If a null is passed in a variable field the system will check to see if that field has a value in the database. If the field does have a value in the database then the system will not overwrite the field with the null.

Supported Functionality

This section discusses the functionality that the processSupplier operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

When adding, changing or deleting supplier records, the processSupplier operation updates the Supplier Master (F0401), Address Book Master (F0101), Address Book -Who's Who (F0111), Address Book - Phone Numbers (F0115), Address by Date (F0116), Electronic Address (F01151), and Address Organization Structure Master (F0150) tables.

The processSupplier operation does not support localizations.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes supplier information. This table includes information about the business service properties that the processSupplier web service operation uses:

Group	Key	Description	Default Value
J0100031	J0100031_Supplier_MBF_VERSION	Use this business service property to specify which version of the Supplier Master MBF – PO program (P0100043) the operation uses. <i>See "Setting Processing Options for Supplier Master MBF - PO (P0100043)" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide .</i>	ZJDE0001
J0100001	J0100031_AB_MBF_VERSION	Use this business service property to specify which version of the Address Book MBF – PO (for N0100041) program (P0100041) the operation uses. <i>See "Setting Processing Options for Address Book MBF (P0100041)".</i>	ZJDE0001

Implementation Details

The following table includes information that can help determine whether the processSupplier operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • EntityId • EntityName • EntityTypeCode <p>If the operation completes successfully, the AN8 (Address Number) and ALPH (Alpha Name) fields in the F0101 and F0111 tables and the AN8 field in the F0401 table will be populated.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. The processSupplier operation processes address book information and supplier records even if the processPhones and processElectronicAddresses operations fail. If the system encounters errors while processing the address book or supplier record information the system rolls back all transactions. No manual update is necessary.</p>
Does this operation use record reservation?	<p>No. This operation does not reserve records.</p>

processSupplierV2

Review the information in the processSupplier section of this chapter before using the processSupplierV2 web service operation. All of the information in the processSupplier section also applies to the procesSupplierV2 operation.

The processSupplierV2 operation is a version of the processSupplier web service operation. This web service operation is used to add, update, and delete supplier information with associated GLNs. This version executes the processSupplier (J0100031) web service operation.

processSupplierV3

Review the information in the processSupplier section of this chapter before using the processSupplierV3 web service operation. All of the information in the processSupplier section also applies to the procesSupplierV3 operation.

The processSupplierV3 operation is a version of the processSupplier web service operation. This web service operation is used to add, update, and delete supplier information with associated GLNs and the voucher match automation attributes.

This version executes the processSupplier (J0100031) web service operation.

getSupplier

The getSupplier web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne database to retrieve existing supplier information, phone numbers, and electronic address records.

When the getSupplier web service operation is called, it first performs a select on the V0401XPI view (business view over the F0101, F0116, and F0401 tables) which returns zero or more supplier records. If the V0401XPI view produces errors then the program stops processing. Otherwise, the program loops through the returned supplier records and calls the getContact web service operation for each of the supplier records returned. Finally, the getSupplier operation exits and returns any errors to the Supplier Manager web service.

If the operation is successful, the system returns zero to many supplier records to the consumer. You can specify the maximum number of records to return during a query using the Max Rows business service property. It is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database. If the operation fails, the system returns an error message to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns when you query the JD Edwards EnterpriseOne database. This table includes information about the business service properties used by the getSupplier operation:

Group	Key	Description	Default Value
J0100032	J0100032_MAX_ROWS	Use this business service property to define the maximum number of rows that the operation returns when querying the JD Edwards EnterpriseOne database.	100

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database. However, open queries are allowed.

Implementation Details

The following table includes information that can help determine whether the getSupplier operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation.

Question	Answer
	<p>However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • MailingName • EntityId • EntityTypeCode <p>Also, the operation completes successfully if no errors are returned.</p>
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
<p>Does this operation use record reservation?</p>	<p>No. This operation does not reserve records that are returned in a query.</p>

getSupplierV2

Review the information in the `getSupplier` section of this chapter before using the `getSupplierV2` web service operation. All of the information in the `getSupplier` section also applies to the `getSupplierV2` operation.

The `getSupplierV2` operation is a version of the `getSupplier` web service operation. This web service operation is used to retrieve supplier information with associated GLNs. This version executes the `getSupplier (J0100032)` web service operation.

getSupplierV3

Review the information in the `getSupplier` section of this chapter before using the `getSupplierV3` web service operation. All of the information in the `getSupplier` section also applies to the `getSupplierV3` operation.

The `getSupplierV3` operation is a version of the `getSupplier` web service operation. This web service operation is used to retrieve supplier information with associated GLNs and the voucher match automation attributes.

This version executes the `getSupplier (J0100032)` web service operation.

Supplier Master Data Real-Time Events

This section describes the available real-time events associated with the Supplier Master Data business object, which include:

- RTSMOUT, which is a container event for these single events:

- RTSMHDR
- RTSMPHOUT
- RTSMEAOUT
- RTSMOUT2, which is a container event for these single events:
 - RTSMHDR2
 - RTSMPHOUT2
 - RTSMEAOUT2

RTSMOUT

RTSMOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSMOUT	Supplier Master	XAPI	Container Contains these events: <ul style="list-style-type: none"> • RTSMHDR (D0100083A) • RTSMPHOUT (D0100083B) • RTSMEAOUT (D0100083C) 	01

Special Setup

RTSMOUT has some additional special setup requirements due to the logic in the real-time wrapper.

The processing options for version ZJDE0001 of the Supplier Master Real Time Event Wrapper application program (P0100083) determine which phone and email records are sent when records are changed. The purpose of these processing options is to enable filtering of the outbound event, RTSMOUT.

This table explains the processing options:

Configure Contact Information

Enter a code to specify how you want to send contact information. Values are:

0, blank, or null: The system sends only primary contact information for an address book record as single fields. It sends the first telephone number, fax number, and email address of an address book record that matches the type specified in the other processing options.

1: Regardless of the types specified in the other processing options, the system sends the complete collection of contact information, including primary contact information, for an address book record in one group.

2: The system sends primary contact information in single fields, and the complete collection of contact information in a group. This value is a combination of values **0** and **1**.

Phone Number Type

Indicate either the location or use of a telephone number. Values are defined in UDC table 01/PH (Phone Type).

Fax Number Type

Specify the phone type for the telephone number to use for faxes. Values are defined in UDC table 01/PH (Phone Type).

Electronic Address Type

Specify the electronic address type that the system uses for an individual or an entity for internet communications. Values are defined in UDC table 01/ET (Electronic Address Type).

Objects that Can Initiate the Event

These objects can initiate the event, and the single events contained within this event:

- N0100041 – Address Book MBF
- N0100043 – Supplier Master MBF
- B0100090 - Phones MBF
- B0100087 - Who's Who MBF
- B0100089 - Electronic Address MBF
- P01111 - Contact Information E-mail application

RTSMHDR

RTSMHDR is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier additions, changes, or deletions.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMHDR	Supplier Master Details	RTE	Single	01	D0100083A: Supplier Master Real Time Record	Included in container event RTSMOUT.

Conceptual Approach

A supplier record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in a number of tables, such as Supplier Master (F0401) and Address Book Master (F0101). A real-time notification event is triggered in the Supplier Master or Address Book Master Business Function after data is modified in any supplier table.

RTSMPHOUT

RTSMPHOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier phone number changes.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMPHOUT	Supplier Master Phone	RTE	Single	01	D0100083B: Supplier Phones Records	Included in container event RTSMOUT.

Conceptual Approach

A phone number is changed for a supplier. This change is sent out using real-time events. A given supplier may have an infinite number of phone numbers.

RTSMEAOUT

RTSMEAOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier email address changes.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMEAOUT	Supplier Electronic Address	RTE	Single	01	D0100083C: Supplier Electronic Address	Included in container event RTSMOUT.

Conceptual Approach

An email address is added or changed for a supplier. This change is sent out using real-time events. A given supplier may have an infinite number of email addresses.

RTSMOUT2

RTSMOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSMOUT2	Supplier Master Version 2	RTE	Container Contains these events: <ul style="list-style-type: none"> • RTSMHDR2 (D0100101A) • RTSMPHOUT2 (D0100101B) • RTSMEAOUT2 (D0100101C) 	01

Special Setup

RTSMOUT2 has some additional special setup requirements due to the logic in the real-time wrapper.

The processing options for version ZJDE0001 of the Supplier Master Real Time Event Wrapper application program (P0100083) determine which phone and email records are sent when records are changed. The purpose of these processing options is to enable filtering of the outbound event, RTSMOUT2.

This table explains the processing options:

Configure Contact Information

Enter a code to specify how you want to send contact information. Values are:

0, blank, or null: The system sends only primary contact information for an address book record as single fields. It sends the first telephone number, fax number, and email address of an address book record that matches the type specified in the other processing options.

1: Regardless of the types specified in the other processing options, the system sends the complete collection of contact information, including primary contact information, for an address book record in one group.

2: The system sends primary contact information in single fields, and the complete collection of contact information in a group. This value is a combination of values **0** and **1**.

Phone Number Type

Indicate either the location or use of a telephone number. Values are defined in UDC table 01/PH (Phone Type).

Fax Number Type

Specify the phone type for the telephone number to use for faxes. Values are defined in UDC table 01/PH (Phone Type).

Electronic Address Type

Specify the electronic address type that the system uses for an individual or an entity for internet communications. Values are defined in UDC table 01/ET (Electronic Address Type).

RTSMHDR2

RTSMHDR2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier additions, changes, or deletions.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMHDR2	Supplier Header Version 2	RTE	Single	01	D0100101A: Supplier Master Real Time Record	Included in container event RTSMOUT2.

Conceptual Approach

A supplier record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in a number of tables, such as Supplier Master (F0401) and Address Book Master (F0101). A real-time notification event is triggered in the Supplier Master or Address Book Master Business Function after data is modified in any supplier table.

RTSMPHOUT2

RTSMPHOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier phone number changes.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMPHOUT2	Supplier Phone Version 2	RTE	Single	01	D0100101B: Supplier Phones Records	Included in container event RTSMOUT.

Conceptual Approach

A phone number is changed for a supplier. This change is sent out using real-time events. A given supplier may have an infinite number of phone numbers.

RTSMEAOUT2

RTSMEAOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems of supplier email address changes.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSMEAOUT2	Supplier Elec Addr Version 2	RTE	Single	01	D0100101C: Supplier Electronic Address	Included in container event RTSMOUT2.

Conceptual Approach

An email address is added or changed for a supplier. This change is sent out using real-time events. A given supplier may have an infinite number of email addresses.

Supplier Master Data Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Supplier Master Data business object:

Program and Program Type	Description	Available Documentation
Supplier Master Batch Upload program (R04010Z) This is a temporary table program.	<p>An address book number must exist in the Address Book Master table before you can upload converted external supplier information. Enter the address book number in the Address Book field of the Supplier Master - Batch Upload table when you run the Supplier Master Batch Upload program (R04010Z).</p> <p>The Supplier Master - Batch Upload program accesses the processed address book records, retrieves the corresponding supplier information in the F0401Z1 table, and uploads it to the Supplier Master table.</p> <p>If you have supplier records that have an associated parent number, you must manually enter the parent/child relationship. You cannot use the batch process to add parent numbers. Parent numbers are stored in the Address Organization Structure Master table.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Uploading Supplier Master Batch Information

Program and Program Type	Description	Available Documentation

34 FMS: Financial Management Solutions - Additional Interface Components

Additional Financial Management Solutions Interface Component Overview

This chapter provides detailed information about the business interface components available for the Financial Management Solutions (FMS) product suite, that are not associated with other business interfaces.

This table lists all of the additional interface components:

Interface Type	Available Interfaces
Business Services	<p>The FinancialComplianceManager published business service manages these business service operations:</p> <ul style="list-style-type: none"> • J0000041 (getAPProcessingOptions) • J0000042 (getWriteOffProcessingOptions) • J0000043 (getAgingCompanyConstants) • J0000044 (getCustomerCreditLimits) • J0000045 (getGeneralConstants) • J0000046 (getJournalGeneralConstants) • J0000047 (getPolicyEditRules) • J0000048 (getPurchasingToleranceRules) • J0000049 (getSecurityWorkbench)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

FinancialComplianceManager Business Services

The FinancialComplianceManager web service (JP000040) manages the processing of compliance-related web service operations. This table includes a description of the financial compliance web service operations:

Operation	Description
getAPPProcessingOptions (J0000041)	Use this operation to retrieve and review the processing option values for these programs in the JD Edwards EnterpriseOne system: <ul style="list-style-type: none"> Recycle Recurring Vouchers (R048101) Purchase Order Receipts (P4312)
getWriteOffProcessingOptions (J0000042)	Use this operation to retrieve and review the processing options for these JD Edwards EnterpriseOne Accounts Receivable programs: <ul style="list-style-type: none"> Standard Receipts Entry (P03B102) Draft Entry (P03B602) Speed Receipts Entry (P03B0001) Invoice Selection Match (R03B50A) Known Invoice Match With Amount (R03B50D) Known Invoice Match Without Amount (R03B50E)
getAgingCompanyConstants (J0000043)	Use this operation to retrieve and review company constant records from the Company Constants table (F0010) in the JD Edwards EnterpriseOne system.
getCustomerCreditLimits (J0000044)	Use this operation to retrieve and review accounts receivable information for specified customers, such as credit limits, from the JD Edwards EnterpriseOne system.
getGeneralConstants (J0000045)	Use this operation to retrieve and review constants that determine how the system processes duplicate invoice numbers. This operation retrieves general constants from the General Constants table (F0009) in the JD Edwards EnterpriseOne system.
getJournalGeneralConstants (J0000046)	Use this operation to retrieve and review constants that determine how the system processes journal entry information. This operation retrieves journal general constants from the General Constants table (F0009) in the JD Edwards EnterpriseOne system.
getPolicyEditRules (J0000047)	Use this operation to retrieve and review policy edit rule information from the JD Edwards EnterpriseOne Expense Management system.
getPurchasingToleranceRules (J0000048)	Use this operation to retrieve and review tolerance rules from the JD Edwards EnterpriseOne Procurement system.
getSecurityWorkbench (J0000049)	Use this operation to retrieve and review security workbench records from the JD Edwards EnterpriseOne system.

Accessing Javadoc for the Financial Compliance Manager Web Service Operations

To access Javadoc for the Financial Compliance Manager web service and its related operations, review these Javadoc packages:

- JP000040 (FinancialComplianceManager)

- J0000041 (getAPProcessingOptions)
- J0000042 (getWriteOffProcessingOptions)
- J0000043 (getAgingCompanyConstants)
- J0000044 (getCustomerCreditLimits)
- J0000045 (getGeneralConstants)
- J0000046 (getJournalGeneralConstants)
- J0000047 (getPolicyEditRules)
- J0000048 (getPurchasingToleranceRules)
- J0000049 (getSecurityWorkbench)

getAPProcessingOptions

The getAPProcessingOptions web service operation is a business function query operation that enables consumers to retrieve and review the processing option values for these programs in the JD Edwards EnterpriseOne system:

- Recycle Recurring Vouchers (R048101)
- Purchase Order Receipts (P4312)

You can retrieve processing options for selected versions, or for all versions. If the operation encounters errors while processing, those errors are returned to the consumer.

Implementation Details

This table includes information that can help determine whether the getAPProcessingOptions operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns all processing option values for the specified programs and versions.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p>
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This is a query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and try your query again. If the operation does not return any records, verify that a version matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

Prerequisites

Before you can use this operation, you should be familiar with recurring accounts payable vouchers and receipt processing.

See *"Processing Accounts Payable Vouchers, Processing Recurring Vouchers" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide* .

See *"Using Receipt Processing" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide* .

getWriteOffProcessingOptions

The getWriteOffProcessingOptions web service operation is a business function query operation that enables consumers to retrieve and review the processing options for these JD Edwards EnterpriseOne Accounts Receivable programs:

- Standard Receipts Entry (P03B102)
- Draft Entry (P03B602)
- Speed Receipts Entry (P03B0001)
- Invoice Selection Match (R03B50A)
- Known Invoice Match With Amount (R03B50D)
- Known Invoice Match Without Amount (R03B50E)

You can retrieve processing options for selected versions, or for all versions. If the operation encounters errors while processing, those errors are returned to the consumer.

Implementation Details

This table includes information that can help determine whether the getWriteOffProcessingOptions operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns the processing option values for the selected programs and versions. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and try your query again. If the operation does not return any records, verify that a version matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisites

Before you use this operation, you should have an understanding of the Accounts Receivable system, with specific knowledge of drafts and receipts processing.

See *"Setting Up Accounts Receivable Drafts"*, *"Setting Up Automatic Receipts Processing"*, *"Processing Manual Receipts"*, and *"Processing Accounts Receivable Drafts"* in the *JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide*.

getAgingCompanyConstants

The getAgingCompanyConstants web service operation is a database query operation that enables consumers to retrieve and review company constant records from the Company Constants table (F0010) in the JD Edwards EnterpriseOne system. The operation returns zero to many records, based on the selection criteria that the consumer passes in, and on the maximum number of rows to return, as defined in the business service properties.

If the consumer does not pass in selection criteria for the fromCompany and toCompany fields, the operation uses these default values when processing the query:

- fromCompany: **00000**
- toCompany: **99999**

This operation returns to the consumer all errors and warnings encountered during processing.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the getAgingCompanyConstants operation uses:

Group	Key	Description	Default Value
J0000043	J0000043_QUERY_MAX_ROWS	Use this business service property to specify the maximum number of records the operation returns for a query.	100

Note: It is recommended that you configure and use this business service property, and that you send in selection criteria when performing a query. If you leave this property blank, or set it to 0 (zero), the system does not limit the number or records to return. Therefore, sending in a blank query when this property is not set to limit the number of records returned could cause significant performance issues.

Implementation Details

This table includes information that can help determine whether the getAgingCompanyConstants operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system displays company constant records matching the selection criteria. If no records match the selection criteria, the operation returns a response message with an empty array.

Question	Answer
	The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisites

Before using this operation, you must define accounts receivable company constants information in the JD Edwards EnterpriseOne system.

See *"Setting Up Constants for Accounts Receivable" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* .

getCustomerCreditLimits

The getCustomerCreditLimits web service operation is a database query operation that enables consumers to retrieve and review accounts receivable information, such as credit limits, for selected customers in the JD Edwards EnterpriseOne system. The operation retrieves information from the Customer Master by Line of Business table (F03012).

If the operation encounters errors, processing stops and all errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system getCustomerCreditLimits operation retrieves data. This table includes information about the business service properties that the getCustomerCreditLimits operation uses:

Group	Key	Description	Default Value
J0000044	J0000044_MAX_ROWS	Use this business service property to specify the maximum number of records that the operation can return for a query.	100

Note: It is recommended that you configure and use this business service property, and that you send in selection criteria when performing a query. If you leave this property blank, or set it to 0 (zero), the system does not limit the number or records to return. Therefore, sending in a blank query when this property is not set to limit the number of records returned could cause significant performance issues.

Implementation Details

This table includes information that can help determine whether the `getCustomerCreditLimits` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system displays accounts receivable records for the specified customers.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p>
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

Prerequisites

Before you use this operation, verify that credit and collection information for your customers exists in the JD Edwards EnterpriseOne Accounts Receivable system.

See *"Setting Up Customer Master Information"*, and *"Setting Up Credit and Collections Management"* in the *JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* .

getGeneralConstants

The `getGeneralConstants` web service operation is a database query operation that enables consumers to retrieve and review general constants from the General Constants table (F0009) in the JD Edwards EnterpriseOne system. The constant that this operation displays determines how the system handles duplicate invoice numbers.

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Note: The F0009 contains only one record. Therefore, this operation will not return multiple records.

Implementation Details

This table includes information that can help determine whether the `getGeneralConstants` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the record that exists in the F0009 table is returned to the consumer.</p> <p>This record includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns a non-zero value for the <code>duplicateInvoiceNumberEdit</code> field.</p>
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and try your query again. If the operation does not return any records, verify that a record matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you use this operation, you must set up accounts payable and receivable constants in the JD Edwards EnterpriseOne system.

See *"Setting Up Constants for Accounts Receivable" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* .

See *"Setting Up Constants for Accounts Payable" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide* .

getJournalGeneralConstants

The `getJournalGeneralConstants` web service operation is a database query operation that enables consumers to retrieve and review general constants from the General Constants table (F0009) in the JD Edwards EnterpriseOne system. This operation returns constants that determine how the system processes journal entry information. For example, you can use this operation to review whether batch control or management approval is required, whether postings or invalid accounts are allowed, and whether the system processes intercompany settlements or multicurrency transactions.

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Note: The F0009 contains only one record. Therefore, this operation will not return multiple records.

Implementation Details

This table includes information that can help determine whether the `getJournalGeneralConstants` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the record that exists in the F0009 table is returned to the consumer.</p> <p>This record includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • <code>isBatchControlRequired</code> • <code>isBatchManagementApprovalRequired</code> • <code>arePostingsAllowed</code> • <code>areInvalidAccountsAllowed</code> • <code>intercompanySettlements</code> • <code>isMultiCurrencyIntercompanyTransAllowed</code>
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and try your query again. If the operation does not return any records, verify that a record matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you use this operation, you must set up accounts payable and receivable constants in the JD Edwards EnterpriseOne system.

See *"Setting Up Constants for Accounts Receivable" in the JD Edwards EnterpriseOne Applications Accounts Receivable Implementation Guide* .

See *"Setting Up Constants for Accounts Payable" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide* .

getPolicyEditRules

The `getPolicyEditRules` web service operation is a database query operation that enables consumers to retrieve and review policy edit rule information from the JD Edwards EnterpriseOne Expense Management system. The operation retrieves information from the Policy Edit Rules table (F09E108).

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the `getPolicyEditRules` operation uses:

Group	Key	Description	Default Value
J0000047	J0000047_MAX_ROWS	Use this operation to specify the maximum number of records that the system returns for a query.	100

Implementation Details

This table includes information that can help determine whether the `getPolicyEditRules` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, policy edit rule records are returned to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service properties and try your query again. If the operation does not return any records, verify that a record matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you use this operation, you must define policy rules in the JD Edwards EnterpriseOne Expense Management system.

See *"Setting Up Expense Management Policies" in the JD Edwards EnterpriseOne Applications Expense Management Implementation Guide* .

getPurchasingToleranceRules

The `getPurchasingToleranceRules` web service operation is a database query operation that enables consumers to retrieve and review tolerance rules from the JD Edwards EnterpriseOne Procurement system. The operation retrieves data from the Purchasing Tolerance Rules table (F4322).

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the `getPurchasingToleranceRules` operation uses:

Group	Key	Description	Default Value
J0000048	J0000048_MAX_ROWS	Use this operation to specify the maximum number of records that the system returns for a query.	100

Implementation Details

This table includes information that can help determine whether the `getPurchasingToleranceRules` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the processing completes successfully, the operation returns zero to many records, based on the query. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service properties and try your query again. If the operation does not return any records, verify that a record matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you use this operation, you must set up tolerance rules in the JD Edwards EnterpriseOne Procurement system.

See *"Creating Tolerance Rules" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide* .

getSecurityWorkbench

The getSecurityWorkbench web service operation is a database query operation that enables consumers to retrieve and review security workbench records from the JD Edwards EnterpriseOne system. The operation retrieves data from the Security Workbench table (F00950).

The operation returns security records for these JD Edwards EnterpriseOne programs:

- Manual Payments (P0413M)
- Work With Payment Groups (P04571)
- Create Payment Groups (R04570)
- Speed Status Change (P0411S)
- Multi Company Voucher (P041016)
- Multiple Voucher (P041017)
- Standard Voucher (P0411)
- Speed Voucher (P0411SV)
- Supplier Master (P04012)

If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the getSecurityWorkbenchoperation uses:

Group	Key	Description	Default Value
J0000049	J0000049_MAX_ROWS	Use this operation to specify the maximum number of records that the system returns for a query.	0: Return All Rows

Note: The maximum number of rows that this operation can return is minimal. Leaving this option set to zero does not cause performance issues.

Implementation Details

This table includes information that can help determine whether the getSecurityWorkbenchoperation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the processing completes successfully, the operation returns zero to many records, based on the query.

Question	Answer
	The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service properties and try your query again. If the operation does not return any records, verify that a record matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you use this operation, you must set up the security workbench in the JD Edwards EnterpriseOne system.

See *JD Edwards EnterpriseOne Tools Security Administration Guide* .

35 Foundation: Address Book

Address Book Overview

This chapter provides detailed information about the business interfaces that are available for the Address Book business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The AddressBookManager web service (JP010000) manages the processing of these address book-related web service operations:</p> <ul style="list-style-type: none"> • getAddressBook (J0100002) • getAddressBookV2 (J0100002) • processAddressBook (J0100001) • processAddressBookV2 (J0100001) • getContact (J0100004) • processContact (J0100003) • lookupAddressBook (J0100009) <p>(Release 9.2 Update) The LatLongProcessor web service (JC011500) uses the Oracle Location Service to retrieve and store the latitude and longitude of the current mailing address for selected address book records. The following business service is available for the Address Book business service:</p> <ul style="list-style-type: none"> • getProcessLatLong
Real-Time Events	<p>This list includes the real-time events for the Address Book business object:</p> <ul style="list-style-type: none"> • RTABOUT is a container event that includes these single events: <ul style="list-style-type: none"> ○ RTABHDR ○ RTABHOUT ○ RTABEAOUT • RTABOUT2 is a container event that includes these single events: <ul style="list-style-type: none"> ○ RTABHDR2 ○ RTABHOUT2 ○ RTABEAOUT2 • RTPCOUT is a container event that includes these single events: <ul style="list-style-type: none"> ○ RTPCOUTDS

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> • RTPCOUT2 is a container event that includes these single events: <ul style="list-style-type: none"> ○ RTPCOUTDS2 • RTSHPOUT is a container event that includes these single events: <ul style="list-style-type: none"> ○ RTSHPHDR ○ RTSHPPHOUT ○ RTSHPEAOUT
Batch Import Programs	<p>This list includes the batch import programs for the Address Book business object:</p> <ul style="list-style-type: none"> • Address Book Batch Upload program (R01010Z) • Who's Who Batch Upload program (R01110Z)
Additional Interfaces	<p>The Address Book business object also uses the following business interfaces:</p> <ul style="list-style-type: none"> • Address Book MBF (P0100041) <p>Depending on the processing option settings on the Outbound tab, this program can write outbound interoperability records that can be used by an external software system.</p> <p>See Setting Processing Options for Address Book MBF (P0100041).</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Address Book Business Services - AddressBookManager

The AddressBookManager web service (JP010000) manages the processing of the address book-related web service operations. This table includes a description of the address book web service operations:

Operation	Description
getAddressBook (J0100002)	Use this operation to retrieve and review address, phone, and electronic mail address information from the JD Edwards EnterpriseOne Address Book system.
getAddressBookV2 (J0100002)	Use this operation to retrieve and review address, phone, and electronic mail address information, including the global locator number (GLN), from the JD Edwards EnterpriseOne Address Book system.
processAddressBook (J0100001)	Use this operation to add, change, or delete address book records in the JD Edwards EnterpriseOne Address Book system.

Operation	Description
processAddressBookV2 (J0100001)	Use this operation to add, change, or delete address book records, including GLN, from the JD Edwards EnterpriseOne Address Book system.
getContact (J0100004)	Use this operation to retrieve and review contact information from the JD Edwards EnterpriseOne Address Book system.
processContact (J0100003)	Use this operation to add, change, or delete contact records in the JD Edwards EnterpriseOne Address Book system.
lookupAddressBook (J0100009)	Use this operation to retrieve address book information from the EnterpriseOne database.

Accessing Javadoc for the Address Book Manager Web Services

To access Javadoc for the Address Book Manager web service and its related operations, review these Javadoc packages:

- JP010000 (AddressBookManager)
- J0100001 (processAddressBook)
- J010001 (processAddressBookV2)
- J0100002 (getAddressBook)
- J0100002 (getAddressBookV2)
- J0100003 (processContact)
- J0100004 (getContact)

Prerequisite

Before you can use the Address Book Manager web service, or any of the related web service operations, you must first install and configure the JD Edwards EnterpriseOne Address Book system.

See *"Setting Up the JD Edwards EnterpriseOne Address Book System" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide* .

Additionally, you must set up business service properties that are used by all the Address Book web service operations that process phone and electronic mail address information. This table lists the business service properties that must be set up:

Group	Key	Description	Default Value
J0100005	J0100005_PHONE_ERROR_PREFIX	Use this business service property to specify the prefix that the system uses for phone error messages.	Phone Record Sent

Group	Key	Description	Default Value
J0100006	J0100006_ELECADDR_ERROR_PREFIX	Use this business service property to specify the prefix that the system uses for electronic address error messages.	Electronic Address Record Sent

getAddressBook

The getAddressBook web service operation is a database query operation that enables consumers to retrieve and review address, phone, and electronic mail information for specified entities from the JD Edwards EnterpriseOne Address Book system.

The operation returns zero to many records if it completes successfully. If the operation encounters errors while processing address information, processing stops and those errors and warnings are returned to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted to warnings, processing continues, and the warnings are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the getAddressBook operation uses:

Group	Key	Description	Default Value
J0100002	J0100002_MAX_ROWS	Use this operation to specify the maximum number of records that the operation returns.	100

Note:

It is recommended that users configure this business service property, and use a value other than **0** (zero). Setting this property to **0** enables the operation to return all matching records. Additionally, it is recommended that you pass in selection criteria when processing this operation. If you do not pass in selection criteria, and you set this business property to **0**, you might encounter significant performance issues.

Additionally, this operation uses a joined view of the F0101, F0116, and F0150 tables when retrieving data. The system enforces a one-to-one relationship between records from the F0101 and F0116, and a one-to-many relationship between the F0101 and the F0150 during initial record retrieval. After all records have been retrieved, the operation deletes records with duplicate address book numbers (AN8). This business service property is enforced during the initial record retrieval, before duplicate records are deleted. Therefore, the actual number of records that the operation returns might be less than the maximum number of rows specified.

See the [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the getAddressBook operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system displays zero to many records, based on the selection criteria that the consumer passes in. If the operation fails, an error is returned to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityID entityName entityTypeCode
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

getAddressBookV2

Review the information in the getAddressBook section of this chapter before using the getAddressBookV2 web service operation. All of the information in the getAddressBook section also applies to the getAddressBookV2 operation.

The getAddressBookV2 operation is a version of the getAddressBook web service operation. This web service operation is used to retrieve address book records with associated GLNs. This version executes the getAddressBook (J010002) web service operation.

processAddressBook

The processAddressBook web service operation is an inbound transaction operation that enables consumers to:

- Add records to the JD Edwards EnterpriseOne Address Book system.
- Delete records from the JD Edwards EnterpriseOne Address Book system.
- Change records in the JD Edwards EnterpriseOne Address Book system.

The operation uses the AddressBookMasterMBF business function (N0100041) to process information. If the operation encounters errors while processing address book information, processing stops and all errors and warnings are returned

to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted into warnings, processing continues, and the warnings are returned to the consumer.

Supported Functionality

This section discusses the functionality that the processAddressBook operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

This operation enables consumers to add, change, or delete address book records. The operation also enables users to add, change, or delete the Who's Who records in the Address Book. These records contain a value of **0** (zero) in the Who's Who Line Number field. The operation also processes phone and electronic mail information for the Who's Who record with line number zero.

This operation does not support related person or alternate address processing for an address book record.

See *"Adding Who's Who Information to Address Book Records" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide* .

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes address book information. This table includes information about the business service properties that the processAddressBook operation uses:

Group	Key	Description	Default Value
J0100001	J0100001_AB_MBF_VERSION	Use this business service property to specify the version of the AddressBookMasterMBF (N0100041) the operation uses to process address book information.	ZJDE0001

See the *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

This table includes information that can help determine whether the processAddressBook operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the user receives a return message that includes the records that were added, changed or updated. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.

Question	Answer
	<p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • entityID • entityName • entityTypeCode
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. If the system encounters errors while processing address book information, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.</p> <p>If the operation encounters errors while processing phone or electronic mail information, those errors are converted to warnings and the warnings are returned to the consumer. These warnings do not stop the processAddressBook operation from processing, but the phone or electronic address that causes the error is not processed.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

processAddressBookV2

Review the information in the processAddressBook section of this chapter before using the processAddressBookV2 web service operation. All of the information in the processAddressBook section also applies to the processAddressBookV2 operation.

The processAddressBookV2 operation is a version of the processAddressBook web service operation. This web service operation is used to add, update, or delete address book records with associated GLNs. This version executes the processAddressBook (J010001) web service operation.

getContact

The getContact web service operation is a database query operation that enables consumers to retrieve and review contact, alternate address, phone, and electronic mail information for specified contacts from the JD Edwards EnterpriseOne Address Book system.

The operation returns zero to many records if it completes successfully. If the operation encounters errors while processing contact information, processing stops and those errors and warnings are returned to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted to warnings, processing continues, and the warnings are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves data from the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the getContact operation uses:

Group	Key	Description	Default Value
J0100004	J0100004_MAX_ROWS	Use this operation to specify the maximum number of records that the operation returns.	100

Note: It is recommended that users configure this business service property and use a value other than **0** (zero). Setting this constant to 0 enables the operation to return all matching records. Additionally, it is recommended that you pass in selection criteria when processing this operation. If you do not pass in selection criteria, and you set this business property to 0, you might encounter significant performance issues.

See the [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the getAddressBook operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system displays zero to many records, based on the selection criteria that the consumer passes in. If the operation fails, an error is returned to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityID contactID entityNameContact
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

processContact

The processContact web service operation is an inbound transaction operation that enables consumers to add, change, or delete records to and from these JD Edwards EnterpriseOne tables:

- Who's Who table (F0111)

- Phones table (F0115)
- Electronic Addresses table (F01151)
- Alternate Addresses table (F01161)

The operation uses the WhosWhoMBF (N0100087) to process contact information. If the operation encounters errors while processing contact information, processing stops and the errors are returned to the consumer. If the operation encounters errors while processing phone or electronic mail information, the errors are converted to warnings, processing continues, and the warnings are returned to the consumer.

Supported Functionality

This section discusses the functionality that the processContact operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The operation does not support any country-specific localization functionality. Additionally, the operation does not enable users to add, change, or delete a contact's related person information.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes contact data. This table includes information about the business service properties that the processContact operation uses:

Group	Key	Description	Default Value
J0100003	J0100003_WHOSWHO_MBF_VERSION	Use this business service property to specify the version of the Who'sWhoMBF (N0100087) that the operation uses to process contact information.	ZJDE0001

See the [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the processContact operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the user receives a return message that includes the records that were changed or updated.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • entityID

Question	Answer
	<ul style="list-style-type: none"> contactID
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. If the system encounters errors while processing contact information, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.</p> <p>If the operation encounters errors while processing phone or electronic mail information, those errors are converted to warnings and the warnings are returned to the consumer. These warnings do not stop the processContact operation from processing, but the phone or electronic address that causes the error is not processed.</p>
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

lookupAddressBook

The lookupAddressBook web service operation is a database query operation that enables consumers to retrieve and review address book information from the JD Edwards EnterpriseOne database. This operation retrieves these items for each business unit, if the data exists in the database:

- EntityTypeCode
- EntityName
- Entity, which includes:
 - EntityId
 - LongId
 - EntityTaxId
- CategoryCodesAddressBook, which includes CategoryCode001 through CategoryCode030.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes contact data. This table includes information about the business service properties that the lookupAddressBook operation uses:

Group	Key	Description	Default Value
J0100009	J0100009_MAX_ROWS	Use this business service property to specify the maximum number of F0101 records the operation can return for a query.	100

See the [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#).

Implementation Details

This table includes information that can help determine whether the lookupAddressBook operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero or more records are returned, based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • EntityTypeCode • EntityId • EntityName
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Address Book Business Services - LatLongProcessor (Release 9.2 Update)

The LatLongProcessor web service (JC011500) processes the latitude and longitude of the current mailing address for selected address book records. This table includes a description of the address book business service.

Operation	Description
getProcessLatLong	Use this business service to retrieve and update the latitude and longitude information from the JD Edwards EnterpriseOne Address Book system.

Prerequisite

Before using the Address Book web service, or any of the related web service operations, you must first install and configure the JD Edwards EnterpriseOne Address Book system.

You must also set the value of Enable Location Services to Yes in the Work with EnterpriseOne System Control (P99410) program.

See *"Setting Up the JD Edwards EnterpriseOne Address Book System" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide* .

getProcessLatLong

The getProcessLatLong business service runs on the Oracle WebLogic Server (WLS) or the Web Application Server (WAS) platform and reads the data stored in the Address Book Geo Code (F0116) table. The EnterpriseOne Enterprise Server collects the data required from the JD Edwards EnterpriseOne Address Book Geo Code (F0116) table and then calls the getProcessLatLong business service that communicates with Oracle Location Services to convert the address book record's address to a latitude and longitude. The getProcessLatLong business service operation enables users to retrieve address book geocodes for specified entities from the JD Edwards EnterpriseOne Address Book system.

For each record in the F0116 table, when a record is in error, EV01 is set to 1. Otherwise, when a record is successfully processed, EV01 is set to 0 and written to the F0119 table.

Address Book Real-Time Events

This section describes the available real-time events associated with the Address Book business object, which include:

- RTABOUT is a container event that includes these single events:
 - RTABHDR
 - RTABHOUT
 - RTABEAOUT
- RTABOUT2 is a container event that includes these single events:
 - RTABHDR2
 - RTABHOUT2
 - RTABEAOUT2
- RTPCOUT is a container event that includes these single events:
 - RTPCOUTDS
- RTPCOUT2 is a container event that includes these single events:
 - RTPCOUTDS2
- RTSHPOUT is a container event that includes these single events:
 - RTSHPHDR
 - RTSHPPHOUT
 - RTSHPEAOUT

RTABOUT

RTABOUT is a container event generated by the JD Edwards EnterpriseOne system to notify third-party systems about changes to the address book record. RTABOUT is a container event for RTABHDR, RTABPHOUT, and RTABEAOUT.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTABOUT	Address Book Master	RTE	Container Container event for: <ul style="list-style-type: none"> • RTABHDR (D0100085A) • RTABPHOUT (D0100085B) • RTABEAOUT (D0100085C) 	01

See the appropriate real-time events that are contained by this event for information about the event fields and conceptual approach.

These are the objects that can initiate the event and any single event contained by this event:

- B0100087 - Who's Who MBF
- B0100089 - Electronic Address
- B0100090 - Phones
- N0100041 - Address book MBF
- P01111 – Contact Information

SPECIAL SETUP

RTABOUT has some additional special setup requirements due to the logic in the real-time wrapper.

The processing options for version ZJDE0001 of the Address Book Master Real Time Event Wrapper Application program (P0100085) determine which phone and email records are sent. The purpose of these processing options is to enable filtering of the outbound event, RTABOUT.

The below list explains the processing options:

Configure Contact Information

Enter a code to specify how you want to send contact information. Values are:

0, blank, or null: The system sends only primary contact information for an address book record as single fields. It sends the first telephone number, fax number, and email address of an address book record that matches the type specified in the other processing options.

1: Regardless of the types specified in the other processing options, the system sends the complete collection of contact information, including primary contact information, for an address book record in one group.

2: The system sends the primary contact information in single fields and the complete collection of contact information in a group. This value is a combination of values **0** and **1**.

Phone Number Type

Indicate either the location or use of a telephone number. Values are defined in the user-defined code (UDC) table 01/PH (Phone Type).

Fax Number Type

Specify the phone type for the telephone number to use for faxes. Values are defined in the UDC table 01/PH (Phone Type).

Electronic Address Type

Specify the electronic address type that the system uses for an individual or an entity for internet communications. Values are defined in the UDC table 01/ET (Electronic Address Type).

RTABHDR

RTABHDR is a single event that provides address book header information within the RTABOUT container event. RTABHDR is generated only when RTABOUT is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABHDR	AB Header	RTE	Single	01	D0100085A: Address Book Master Real Time	Included in the container event RTABOUT.

Conceptual Approach

An address book record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Address Book tables (Address Book Master, F0101; Address Book - Who's Who, F0111; Contact Information, F01111; Address Book - Contact Phone Numbers F0115; Electronic Address, F01151; Address by Date, F0116).

RTABPHOUT

RTABPHOUT is a single event that provides address book phone information within the RTABOUT container event. RTABPHOUT is generated only when RTABOUT is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABPHOUT	AB Phone	RTE	Single	01	D0100085B: Address Book Master Phones Wrapper Data Structure	Included in the container event RTABOUT.

Conceptual Approach

An address book phone record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, updates, or deletes a record in the Address Book - Contact Phone Numbers (F0115).

RTABEAOUT

RTABEAOUT is a single event that provides address book email information within the RTABOUT container event. RTABEAOUT is generated only when RTABOUT is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABEAOUT	AB Electronic Address	RTE	Single	01	D0100085C: Address Book Master Email Wrapper Data Structure	Included in the container event RTABOUT.

Conceptual Approach

An address book email record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in the Electronic Address (F01151) table.

RTABOUT2

RTABOUT2 is a container event generated by the JD Edwards EnterpriseOne system to notify third-party systems about changes to the address book record. RTABOUT is a container event for RTABHDR2, RTABPHOUT2, and RTABEAOUT2.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTABOUT2	Address Book Master Version 2	RTE	Container Container event for: <ul style="list-style-type: none"> • RTABHDR2 (D0100100A) • RTABPHOUT2 (D0100100B) • RTABEAOUT2 (D0100100C) 	01

See the appropriate real-time events that are contained by this event for information about the event fields and conceptual approach.

These are the processes that can initiate the event and the single events contained by this event:

- B0100087 - Who's Who MBF
- B0100089 - Electronic Address
- B0100090 - Phones
- N0100041 - Address book MBF
- P01111 – Contact Information

SPECIAL SETUP

RTABOUT2 has some additional special setup requirements due to the logic in the real-time wrapper.

The processing options for version ZJDE0001 of the Address Book Master Real Time Event Wrapper Application program (P0100100) determine which phone and email records are sent. The purpose of these processing options is to enable filtering of the outbound event, RTABOUT2.

The below list explains the processing options:

Configure Contact Information

Enter a code to specify how you want to send contact information. Values are:

0, blank, or null: The system sends only primary contact information for an address book record as single fields. It sends the first telephone number, fax number, and email address of an address book record that matches the type specified in the other processing options.

1: Regardless of the types specified in the other processing options, the system sends the complete collection of contact information, including primary contact information, for an address book record in one group.

2: The system sends the primary contact information in single fields and the complete collection of contact information in a group. This value is a combination of values **0** and **1**.

Phone Number Type

Indicate either the location or use of a telephone number. Values are defined in the user-defined code (UDC) table 01/PH (Phone Type).

Fax Number Type

Specify the phone type for the telephone number to use for faxes. Values are defined in the UDC table 01/PH (Phone Type).

Electronic Address Type

Specify the electronic address type that the system uses for an individual or an entity for internet communications. Values are defined in the UDC table 01/ET (Electronic Address Type).

RTABHDR2

RTABHDR2 is a single event that provides address book header information within the RTABOUT2 container event. RTABHDR2 is generated only when RTABOUT2 is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABHDR2	AB Header Version 2	RTE	Single	01	D0100100A: Address Book Master Real Time	Included in the container event RTABOUT

Conceptual Approach

An address book record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Address Book tables (Address Book Master, F0101; Address Book - Who's Who, F0111; Contact Information, F01111; Address Book - Contact Phone Numbers F0115; Electronic Address, F01151; Address by Date, F0116).

RTABPHOUT2

RTABPHOUT2 is a single event that provides address book phone information within the RTABOUT2 container event. RTABPHOUT2 is generated only when RTABOUT2 is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABPHOUT2	AB Phone Version 2	RTE	Single	01	D0100100B: Address Book Master Phones	Included in the container event RTABOUT.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
					Wrapper Data Structure	

Conceptual Approach

An address book phone record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, updates, or deletes a record in the Address Book - Contact Phone Numbers (F0115).

RTABEAOUT2

RTABEAOUT2 is a single event that provides address book email information within the RTABOUT2 container event. RTABEAOUT2 is generated only when RTABOUT2 is generated.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTABEAOUT2	AB Electronic Address Version 2	RTE	Single	01	D0100100C: Address Book Master Email Wrapper Data Structure	Included in the container event RTABOUT.

Conceptual Approach

An address book email record added, modified, or deleted in the JD Edwards EnterpriseOne system creates a new record in the Electronic Address (F01151) table.

RTPCOUT

RTPCOUT is a container event generated by the JD Edwards EnterpriseOne system to provide parent/child information to third-party systems.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTPCOUT	Parent Child Master	RTE	Container Container event for: <ul style="list-style-type: none"> RTPCOUTDS (D0100079A) 	H01

Objects that Can Initiate the Event

A parent child record added, changed, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Address Organization Structure Master table (F0150) (Parent/Child table) This is the only event that initiates the container event, or any single events contained by this event.

RTPCOUTDS

RTPCOUTDS is a single event within the RTPCOUT container event that provides parent/child information to third-party systems. RTPCOUTDS is generated only when RTPCOUT is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPCOUTDS	Parent Child Details	RTE	Single	H01	D0100079A: ParentChild Master Manage Notify Real Time Structure	Included in the container event RTPCOUT.

RTPCOUT2

RTPCOUT2 is a container event generated by the JD Edwards EnterpriseOne system to provide parent/child information to third-party systems.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTPCOUT2	Parent Child Master Version 2	RTE	Container Container event for: <ul style="list-style-type: none">RTPCOUTDS2 (D0100102A)	H01

Conceptual Approach

A parent child record added, changed, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Address Organization Structure Master table (F0150) (Parent/Child table).

Objects that Can Initiate the Event

A parent child record added, changed, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Address Organization Structure Master table (F0150) (Parent/Child table) This is the only event that initiates the container event, or any single events contained by this event.

RTPCOUTDS2

RTPCOUTDS2 is a single event within the RTPCOUT2 container event that provides parent/child information to third-party systems. RTPCOUTDS2 is generated only when RTPCOUT2 is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPCOUTDS2	Parent Child Details Version 2	RTE	Single	H01	D0100102A: ParentChild Master	Included in the container event RTPCOUT.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
					Manage Notify Real Time Structure	

RTSHPOUT

RTSHPOUT is a container event generated by the JD Edwards EnterpriseOne system to notify third-party systems about add or changes to the address book number of a branch plant record.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSHPOUT	Branch Plant Address Book	RTE	Container Container event for: <ul style="list-style-type: none"> • RTSHPHDR (D4102160A) • RTSHPPHOUT (D4102160B) • RTSHPEAOUT (D4102160C) 	H42

See the appropriate real-time events that are contained by this event for information about the event fields associated with this event.

Conceptual Approach

An address book number added or changed in a Branch Plant Constant table (F41001) in the JD Edwards EnterpriseOne system sends out ship to real-time events with address book (F0101), phones (F0115), and email (F01151) information. All phones and email are sent for the first contact only.

Objects that Can Initiate the Event

The P41001 - Branch Plant Constants program can initiate the RTSHPOUT event and all single events contained by the event.

RTSHPHDR

RTSHPHDR is a single event that is generated by the JD Edwards EnterpriseOne system to provide ship to header information for a given branch plant.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPHDR	Branch Plant AB Header	RTE	Single	H41	D4102160A: Ship To Real Time Header	Included in the container event RTSHPOUT.

Conceptual Approach

A ship to address book number added or modified in a branch plant modifies the Branch Plant Constants table and this information is sent out in a real-time event: Address Book Master (F0101).

RTSHPPHOUT

RTSHPPHOUT is a single event generated by the JD Edwards EnterpriseOne system to provide address book phone information.

Event Name	Event Description	Event Type	Event Category	Product Code	Data Structure	Event Relationship
RTSHPPHOUT	Branch Plant AB Phone	RTE	Single	H41	D4102160B: Ship To Real Time Phones	Included in the container event RTSHPOUT.

Conceptual Approach

A ship to address book number added or modified in a branch plant modifies the Branch Plant Constants table and this information is sent out in a real-time event: all Address Book - Contact Phone Numbers (F0115) for the first contact.

RTSHPEAOUT

RTSHPEAOUT is a single event generated by the JD Edwards EnterpriseOne system to provide address book email information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPEAOUT	Branch Plant AB Elec Address	RTE	Single	H41	D4102160C: Ship To Real Time Emails	Included in the container event RTSHPOUT.

Conceptual Approach

A ship to address book number added or modified in a branch plant modifies the Branch Plant Constants table and this information is sent out in a real-time event: all Electronic Address (F01151) records for the first contact.

Address Book Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Address Book business object:

Program and Program Type	Description	Available Documentation
<p>Address Book Batch Upload program (R01010Z)</p> <p>This is a temporary table batch import program.</p>	<p>After you convert address book information from an external source and transfer it to the Address Book - Interoperability table, use the Address Book Batch Upload program (R01010Z) to process the information and update several Address Book tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding the Address Book Batch Upload Process
<p>Who's Who Batch Upload program (R011110Z)</p> <p>This is a temporary table batch import program.</p>	<p>An address book number must exist in the Address Book Master table before you can upload converted external who's who information. Enter the address book number in the Address Book field of the Batch Who's Who Information table (F0111Z1) when you run the Who's Who Batch Upload program.</p> <p>The Who's Who Batch Upload program accesses the processed address book records, retrieves the corresponding who's who information in the F0111Z1 table, and uploads it to several Address Book tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Uploading Who's Who Batch Information

36 Foundation: Business Unit

Business Unit Overview

This chapter provides detailed information about the business interfaces that are available for the Business Unit business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The BusinessUnitManager (JP000006) manages the process of these business unit-related web service operations:</p> <ul style="list-style-type: none"> • processBusinessUnit (J0000120) • getBusinessunit (J0000130)
Real-Time Events	<p>This list includes the real-time events for the Business Unit business object:</p> <ul style="list-style-type: none"> • RTBUOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTBUOUTA ○ RTBUOUTB • RTBUOUT2, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTBUOUTA2 ○ RTBUOUTB2 • RTBUOUT3, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTBUOUTA3 ○ RTBUOUTB3
Batch Import Programs	<p>This list includes the batch import programs for the Business Unit business object:</p> <ul style="list-style-type: none"> • Business Unit Batch Inbound (R0006ZI)
Batch Export Programs	<p>This list includes the batch export programs for the Business Unit business object:</p> <ul style="list-style-type: none"> • Outbound Cost Center Extraction (R0006Z1E)
Additional Components	<p>There are several programs that work together to enable users to manually input, or upload from a spreadsheet, business unit data. The data can be reviewed and modified, and then transferred to a the live EnterpriseOne tables. You can then purge the records in the temporary table.</p> <p>For additional information about these programs, see <i>Importing and Exporting Business Unit and Company Records</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Business Unit Business Services - BusinessUnitManager

The BusinessUnitManager web service (JP000006) manages the processing of the business unit-related web service operations. This table includes a description of the business unit web service operations:

Operation	Description
processBusinessUnit (J0000120)	Use this operation to process the Business Unit information in the JD Edwards Enterprise One system. This operation enables you to add, change, and delete Business Unit records.
getBusinessunit (J0000130)	Use this operation to retrieve Business Unit information from the EnterpriseOne database. The operation uses the user-defined search criteria to retrieve the records.

Accessing Javadoc for the BusinessUnitManager Web Services

To access Javadoc for the BusinessUnitManager web service and its related operations, review these Javadoc packages:

- JP000006 (BusinessUnitManager)
- J0000120 (processBusinessUnit)
- J0000130 (getBusinessunit)

processBusinessUnit

The ProcessBusinessUnitProcessor (J0000120) is a web service operation that calls the processBusinessUnit method, based on the action of the user. This method enables consumers to add, change, or delete Business Unit records in the Enterprise One database. When the processBusinessUnit method is called, the operation calls the BSFNs - ProcessBusinessUnit_Wrapper (B0001440) to add, change, or delete Business records, and calls the ProcessBusinessUnit_Translation (B0001440) to add, change, or delete Business Unit Alternate Description Translations records.

If the operation is successful, the system updates the Business Unit information in the JD Edwards EnterpriseOne system. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes contact data. This table includes information about the business service properties that the processBusinessUnit operation uses:

Group	Key	Description	Default Value
J0000120	J0000120_BU_VERSION	Use this business service property to specify the version of the P0006 Entry program that will be used while processing the data.	ZJDE0001

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the processBusinessUnit operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The message includes all of the fields that are listed in the response interface for this operation.</p> <p>However, some of those fields can contain blank or zero values, depending on the data that exists in the Enterprise One system. At a minimum, the system returns non-zero values for these fields</p> <ul style="list-style-type: none"> • businessUnit • company • description • addressNumber
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors while processing contact information, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	This operation reserves records in the JD Edwards EnterpriseOne system during the change of Business Unit.

getBusinessUnit

The getBusinessunit (J0000130) web service operation is a database query operation that enables consumers to retrieve and review Business Unit records from the JD Edwards EnterpriseOne database. The operation returns one or more records to the consumers from the Business Unit (F0006) table and Business Unit Business Unit Alternate Description (F0006D) table that meet the specified search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes contact data. This table includes information about the business service properties that the getBusinessUnit operation uses:

Group	Key	Description	Default Value
J0000130	J0000130_MAX_ROWS_F0006	<p>Use this business service property to specify the maximum number of BU records the operation can return for a query.</p> <p>The default value is set to 100; however, this might limit the returned data when searching for records. If you find that limiting the number of results returned to 100 does not enable you to find the records you are searching for, it is acceptable to increase this value to a number that fits your business needs.</p>	100

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the `getBusinessUnit` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned, based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards Enterprise One system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> businessUnit description company
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Business Unit Real-Time Events

This section describes the available real-time events associated with the Business Unit business object, which include:

- RTBUOUT, which is a container event for these single events:
 - RTBUOUTA
 - RTBUOUTB
- RTBUOUT2, which is a container event for these single events:
 - RTBUOUTA2
 - RTBUOUTB2
- RTBUOUT3, which is a container event for these single events:
 - RTBUOUTA3
 - RTBUOUTB3

RTBUOUT

RTBUOUT is a container event generated by the JD Edwards EnterpriseOne system to provide business unit information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTBUOUT	Business Unit Master	RTE	Container Container event for: <ul style="list-style-type: none"> • RTBUOUTA (D0000600B) • RTBUOUTB (D4101700B) 	H00

Please see the appropriate real-time events that are contained by this event for information about the event fields associated with this event.

Conceptual Approach

A business unit record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Business Unit table (F0006). If a business unit is added or modified, the information is sent to another third-party system for add or update. If a business unit is deleted, the pertinent information is sent to the third-party, which take the appropriate action.

Objects that Can Initiate the Event

These objects can initiate the RTBUOUT event, and the single events contained within the event:

- P0006 - Business Units

- P41001 - Branch/Plant Constants
- PDG1117A
- R10480 - Journalize Consolidation Balances
- R10862 - Refresh Consolidation
- B0000105 - F0006D Update Business Unit Translation
- B0000107 - F0006D Edit Business Unit Translation
- B0000203 - F0006 Update Business Unit Category Codes from Address Book
- B0900156 - Delete F0006 Business Unit and F0006D Translations
- B1000012 - Business Unit Existence Check
- B1000040 - Build Parent/Child Business Unit Cache

RTBUOUTA

RTBUOUTA is a single event within the RTBUOUT container event that provides business unit information. RTBUOUTA is generated only when RTBUOUT is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTA	Business Unit Details	RTE	Single	H00	D0000600B: Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT

RTBUOUTB

RTBUOUTB is a single event within the RTBUOUT container event that provides business unit information. RTBUOUTB is generated only when RTBUOUT is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTB	Business Unit - Branch Plant	RTE	Single	H00	D4101700B: Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT

RTBUOUT2

RTBUOUT2 is a container event generated by the JD Edwards EnterpriseOne system to provide business unit information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTBUOUT2	Business Unit Master Version 2	RTE	Container Container event for: <ul style="list-style-type: none"> RTBUOUTA2 (D0000610B) RTBUOUTB2 (D4101770B) 	H00

Please see the appropriate real-time events that are contained by this event for information about the event fields associated with this event.

Conceptual Approach

A business unit record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Business Unit table (F0006). If a business unit is added or modified, the information is sent to another third-party system for add or update. If a business unit is deleted, the pertinent information is sent to the third-party, which take the appropriate action.

Objects that Can Initiate the Event

These objects can initiate the RTBUOUT2 event, and the single events contained within the event:

- P0006 - Business Units
- P41001 - Branch/Plant Constants
- PDG1117A
- R10480 - Journalize Consolidation Balances
- R10862 - Refresh Consolidation
- B0000105 - F0006D Update Business Unit Translation
- B0000107 - F0006D Edit Business Unit Translation
- B0000203 - F0006 Update Business Unit Category Codes from Address Book
- B0900156 - Delete F0006 Business Unit and F0006D Translations
- B1000012 - Business Unit Existence Check
- B1000040 - Build Parent/Child Business Unit Cache
- B0000600 - Business Unit Real Time Event Notification
- B4101700 - Business Unit Real Time Event Notify 2

RTBUOUTA2

RTBUOUTA2 is a single event within the RTBUOUT2 container event that provides business unit information. RTBUOUTA2 is generated only when RTBUOUT2 is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTA2	Bus Unit Details Version 2	RTE	Single	H00	D0000610B: Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship

RTBUOUTB2

RTBUOUTB2 is a single event within the RTBUOUT2 container event that provides business unit information. RTBUOUTB is generated only when RTBUOUT2 is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTB2	Bus Unit-Branch Plant Version2	RTE	Single	H00	D4101770B: Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT

RTBUOUT3

RTBUOUT3 is a container event generated by the JD Edwards EnterpriseOne system to provide business unit information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTBUOUT3	RTBUOUT3	RTE	Container Container event for: <ul style="list-style-type: none"> • RTBUOUTA3 (D0000620B) • RTBUOUTB3 (D4101780B) 	H00

Conceptual Approach

A business unit record added, modified, or deleted in the JD Edwards EnterpriseOne system creates, modifies, or deletes a record in the Business Unit table (F0006). If a business unit is added or modified, the information is sent to another third-party system for add or update. If a business unit is deleted, the pertinent information is sent to the third-party, which take the appropriate action.

Objects that Can Initiate the Event

These objects can initiate the RTBUOUT3 event, and the single events contained within the event:

- P0006 - Business Units
- P41001 - Branch/Plant Constants
- PDG1117A
- R10480 - Journalize Consolidation Balances

- R10862 - Refresh Consolidation
- B0000105 - F0006D Update Business Unit Translation
- B0000107 - F0006D Edit Business Unit Translation
- B0000203 - F0006 Update Business Unit Category Codes from Address Book
- B0900156 - Delete F0006 Business Unit and F0006D Translations
- B1000012 - Business Unit Existence Check
- B1000040 - Build Parent/Child Business Unit Cache
- B0000600 - Business Unit Real Time Event Notification
- B4101700 - Business Unit Real Time Event Notify 2

RTBUOUTA3

RTBUOUTA3 is a single event within the RTBUOUT3 container event that provides business unit information. RTBUOUTA3 is generated only when RTBUOUT3 is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTA3	RTBUOUTA3	RTE	Single	H00	D0000620B - Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT3

RTBUOUTB3

RTBUOUTB3 is a single event within the RTBUOUT3 container event that provides business unit information. RTBUOUTB3 is generated only when RTBUOUT3 is generated.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTBUOUTB3	RTBUOUTB3	RTE	Single	H00	D4101780B - Business Unit Real Time Notification Outbound Data	Included in container event RTBUOUT3

Business Unit Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Business Unit business object:

Program and Program Type	Description	Additional Information
<p>Business Unit Batch Inbound (R0006ZI)</p> <p>This is an interoperability batch program.</p>	<p>This program is used to copy information from F0006Z2 table to Business Unit Master table (F0006). The program also performs the below functions:</p> <ul style="list-style-type: none"> • Processes the inbound transactions from F0006Z2 table. • Update the F0006 table. • Deletes records from F0006D table if deleted from F0006. 	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Importing and Exporting Business Unit and Company Records</i>

Business Unit Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Business Unit business object:

Program and Program Type	Description	Available Documentation
<p>Outbound Cost Center Extraction (R0006Z1E)</p> <p>This is an outbound interoperability program.</p>	<p>This program writes business unit information that is retrieved from the Business Unit Master table (F0006) into the Cost Center Unedited Transaction Table (F0006Z1) as outbound transaction records. This information is then intended to be used by systems outside of Enterprise One.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Importing and Exporting Business Unit and Company Records</i>

37 Foundation: Company

Company Overview

This chapter provides detailed information about the business interfaces that are available for the Company business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The Company Manager web service (JP000010) manages the processing of Company constant-related web service operations:</p> <ul style="list-style-type: none"> • getCompany (J0000110) • processCompany (J0000100)
Real-Time Events	<p>This list includes the real-time events for the Company business object:</p> <ul style="list-style-type: none"> • RTCOOUT • RTCOOUT2
Batch Import Programs	<p>This list includes the batch import programs for the Company business object:</p> <ul style="list-style-type: none"> • Process Company Constants Inbound (R0010ZI) • Purge Company Z File (R0010ZP)
Batch Export	<p>This list includes the batch export programs for the Company business object:</p> <ul style="list-style-type: none"> • Company File Export (R0010ZO)
Additional Components	<p>There are several programs that work together to enable users to manually input, or upload from a spreadsheet, company data. The data can be reviewed and modified, and then transferred to a the live EnterpriseOne tables. You can then purge the records in the temporary table.</p> <p>For additional information about these programs, see <i>Importing and Exporting Business Unit and Company Records</i></p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Company Business Services - CompanyManager

This section describes the available business services associated with the Company business object, which include the Company Manager web service (JP000010), which manages the processing of these company-related web service operations:

The Company Manager web service (JP000010) manages the processing of the company-related web service operations. This table includes a description of the company web service operations:

Operation	Description
getCompany (J0000110)	Use this operation to retrieve and review company constants from the JD Edwards EnterpriseOne General Accounting system.
processCompany (J0000100)	Use this operation to add, company constant record in the JD Edwards EnterpriseOne General Accounting system.

Accessing Javadoc for the CompanyManager Web Services

To access Javadoc for the CompanyManager web service and its related operations, review these Javadoc packages:

- JP000010 (CompanyManager)
- getCompany (J0000110)
- processCompany (J0000100)

getCompany

The getCompany web service operation is a database query operation that enables consumers to retrieve company constant records from the JD Edwards EnterpriseOne General Accounting system.

The operation retrieves records based on the selection criteria that is passed in. If no selection criteria is passed in, the operation attempts to retrieve all records from the EnterpriseOne database, using the business service property to limit the number of records returned. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes contact data. This table includes information about the business service properties that the getCompany operation uses:

Group	Key	Description	Default Value
J0000110	J0000110_MAX_ROWS	Use this business service property to	100

Group	Key	Description	Default Value
		specify the maximum number of records the operation returns for the query.	

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Note:

It is recommended that this business service property is configured and used. Also, it is strongly recommended that the consumer pass in selection criteria when performing a query. If these two recommendations are not followed, the operation attempts to fetch all the records from the database, which can significantly impact performance.

Also be aware that this operation performs multiple data retrievals from the EnterpriseOne database. Initially, records are selected from the Company constant tables (F0010,F0010T) based on the selection criteria that is passed in by the consumer.

After the operation retrieves records from the F0010, processing continues, and additional record is retrieved from the Company Constant Tag table(F0010T), based on the data in the F0010 records. This business service property limits the number of records initially retrieved from the F0010. Therefore, the total number of records that are returned to the consumer might be greater than the value that you enter in this property.

Implementation Details

This table includes information that can help determine whether the getCompany operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned to the consumer based on the selection criteria.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • companyNumber • companyName • GeneralAccounting • fiscalDatePattern • numberOfPeriod • currentGLPeriodandBeginFiscalYear • currentPeriod

Question	Answer
	<ul style="list-style-type: none"> • beginFiscalYear • accountPayable • currentAPPeriodandBeginFiscalYear • currentPeriod • beginFiscalYear • accountsReceivable • currentARPeriodandBeginFiscalYear • currentPeriod • beginFiscalYear
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

processCompany

The processCompany web service operation is an inbound transaction operation that enables consumers to add, delete Company Constants in the JD Edwards EnterpriseOne General Accounting system.

If the operation completes successfully, the consumer receives a success message. If the operation encounters errors, processing stops and the errors are returned to the consumer.

Implementation Details

This table includes information that can help determine whether the processCompany operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, a success message is returned to the consumer.</p> <p>The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • companyNumber • companyName • generalAccounting • fiscalDatePattern

Question	Answer
	<ul style="list-style-type: none"> • numberOfPeriod • currentGLPeriodandBeginFiscalYear • currentPeriod • beginFiscalYear • accountPayable • currentAPPeriodandBeginFiscalYear • currentPeriod • beginFiscalYear • accountsReceivable • currentARPeriodandBeginFiscalYear • currentPeriod • beginFiscalYear
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Company Real-Time Events

This section describes the available real-time events associated with the Company business object, which include:

- RTCOOUT
- RTCOOUT2

RTCOOUT

RTCOOUT is a single event generated by the JD Edwards EnterpriseOne system to provide company information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTCOOUT	Company Master	RTE	Single	H00	D0000310B: Company Real Time Notification Outbound Data

Conceptual Approach

A new company added in the JD Edwards EnterpriseOne system creates a new record in the Company table (F0010). If a company is added or modified in a system, the information is sent to a third-party system for add or update. If

a company is deleted, the pertinent information is sent to the third-party, which will take the appropriate action. If applicable, the third-party integration must update the cross-reference table in the process in order to tie together the key values from both systems.

Objects that Can Initiate the Event

These objects can initiate the RTCOOUT event:

- P0000 - System Setup
- P0010 - Companies
- B0000132 - Update Currency Conversion
- B03B0026 - Retrieve/Update A/R Constants

RTCOOUT2

RTCOOUT2 is a single event generated by the JD Edwards EnterpriseOne system to provide company information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTCOOUT2	RTCOOUT2	RTE	Single	H00	D0001460B: Company Real Time Notification Outbound Data 2

Conceptual Approach

A new company added in the JD Edwards EnterpriseOne system creates a new record in the Company tables (F0010,F0010T). If a company is added or modified in a system, the information is sent to a third-party system for add or update. If a company is deleted, the pertinent information is sent to the third-party, which will take the appropriate action. If applicable, the third-party integration must update the cross-reference table in the process in order to tie together the key values from both systems.

Objects that Can Initiate the Event

These objects can initiate the RTCOOUT2 event:

- P0000 - System Setup
- P0010 - Companies
- B0000132 - Update Currency Conversion
- B03B0026 - Retrieve/Update A/R Constants
- R890010E - Currency Conversion
- B0001450 - Process Company Info
- P001001 - Alternate Tax Rate/Area Assign
- P00218 - Invoice and Voucher Company Tax Constants

Company Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Company business object:

Program and Program Type	Description	Available Documentation
Process Company Constants Inbound (R0010ZI)	This program is used to import data from the F0010Z into the F0010 and F0010T tables.	See this topic: <ul style="list-style-type: none"> <i>Importing and Exporting Business Unit and Company Records</i>

Company Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Company business object:

Program and Program Type	Description	Available Documentation
Company File Export (R0010ZO)	Run this program to export data from the F0010 and F0010T tables and create an XML output that can be consumed by a third party system.	See this topic: <ul style="list-style-type: none"> <i>Importing and Exporting Business Unit and Company Records</i>

38 Foundation: Currency Information

Currency Information Overview

This chapter provides detailed information about the business interfaces that are available for the Currency Information business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The Currency Information business object uses the <code>lookupCurrencyCodes (J0000060)</code> method. For additional details on this service, see FoundationEnvironmentManager Business Services .
Real-Time Events	This list includes the real-time events for the Currency Information business object: <ul style="list-style-type: none"> RTCROUT
Batch Import Programs	This list includes the batch import programs for the Currency Information business object: <ul style="list-style-type: none"> External Exchange Rate Processor program (R0015Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Currency Information Real-Time Events

This section describes the available real-time events associated with the Currency Information business object, which include:

- RTCROUT

RTCROUT

RTCROUT is used by the JD Edwards EnterpriseOne system to notify third-party systems of currency exchange rate additions, changes, or deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTCROUT	Currency Information	RTE	Single	H09	D0900200B: Currency Exchange Rates Real Time Event Notification - B

Conceptual Approach

When a user adds, modifies, or deletes a currency exchange rate in the JD Edwards EnterpriseOne system, the system updates the Currency Exchange Rate table (F0015) and publishes the RTCROUT event. This event contains all columns from the modified currency exchange rate record and an action code specifying what action was performed on the data (add, change, or delete).

Objects that Can Initiate the Event

These objects can initiate the RTCROUT event:

- F0015 - Currency Exchange Rates table trigger

Currency Information Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Currency Information business object:

Program and Program Type	Description	Available Documentation
External Exchange Rate Processor program (R0015Z1) This is a temporary table import program.	After you upload exchange rates from an external source into the F0015Z1 and, if necessary, revise the rates, run the External Exchange Rate Processor program (R0015Z1). This program processes information from the F0015Z1 workfile and stores it in the F0015 table.	See this topic: <ul style="list-style-type: none"> • Understanding the External Exchange Rate Processor Program

39 Foundation: Additional Interface Components

Foundation Additional Components Overview

This chapter provides detailed information about the additional business interfaces that are available for the EnterpriseOne Foundation system.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The FoundationEnvironmentManager (JP000000) manages these web service operations:</p> <ul style="list-style-type: none"> • getUserDefinedCode (J0000030) • lookupBusinessUnits (J0000040) • getUDCDescription (J0000050) • lookupCurrencyCodes (J0000060) • setUserValues and getUserValues (J0000070) • getUserProfile (J0000080) • lookupUDC (J0000090)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

FoundationEnvironmentManager Business Services

The Foundation Environment web service (JP000000) manages the processing of the these web service operations:

- getUserDefinedCode (J0000030)
You use the getUserDefinedCode web service operation to retrieve and review user defined codes from the JD Edwards EnterpriseOne system.
- lookupBusinessUnits (J0000040)
You use this operation to retrieve business unit values when searching for valid business units in the EnterpriseOne database.

- `getUDCDescription (J0000050)`
Use this method to retrieve sales order type description values from user defined code (UDC) table 00/DT.
- `lookupCurrencyCodes (J0000060)`
You use this operation to retrieve currency code values when searching for valid currency codes in the EnterpriseOne database.
- `setUserValues` and `getUserValues (J0000070)`
Use the `setUserValues` method to save the user-entered values in the mobile sales application. Use the `getUserValues` method to retrieve previously saved user-entered values.
- `getUserProfile (J0000080)`
Use this operation to retrieve the language preference, date separator and format, and other profile information from the EnterpriseOne database for the signed - on mobile applications user.
- `lookupUDC (J0000090)`
Use this operation to retrieve user defined code details for a specified code.

Before using this web service, you should be familiar with how user defined codes (UDCs) are used in the JD Edwards EnterpriseOne system.

Accessing Javadoc for the FoundationEnvironmentManager Web Service Operations

To access Javadoc for the FoundationEnvironmentManager web service and its related operations, review these Javadoc packages:

- JP000000 (FoundationEnvironmentManager)
- J0000030 (`getUserDefinedCode`)
- J0000040 (`lookupBusinessUnits`)
- J0000050 (`getUDCDescription`)
- J0000060 (`lookupCurrencyCodes`)
- J0000070 (`get` or `setUserValues`)
- J0000080 (`getUserProfile`)
- J0000090 (`lookupUDC`)

`getUserDefinedCode`

The `getUserDefinedCode` web service operation is a database query operation that enables consumers to retrieve and review user-defined code (UDC) tables that are stored in the JD Edwards EnterpriseOne system.

UDC tables are used throughout the EnterpriseOne system. They enable consumers to store, track, calculate, and process information that is specific to their business operations. The `getUserDefinedCode` operation enables consumers to search on UDC tables and review information about the tables and the values that are stored in those tables.

This operation contains the `getExpenseManagementValuesList` method, which is used to retrieve expense management UDC values. This method is used in the Expense Management mobile application.

Note:

- *Expense Management or Sales Inquiries in the JD Edwards EnterpriseOne Applications Functionality for Mobile Devices Implementation Guide .*

Note: When you enter a query, you must specify selection criteria. The operation does not allow a query without selection criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service property that the `getUserDefinedCode` operation uses:

Group	Key	Description	Default Value
J0000030	J0000030_V0004A_MAX_GRID_ROWS_RETURNED	Use this business service property to specify the maximum number of records that the system returns for a query.	100

Note: It is strongly recommended that consumers set this business service property to a value other than **0**. Entering **0** in this property will allow the operation to retrieve all records that match the search criteria, and could cause performance issues.

Implementation Details

This table includes information that can help determine whether the `getUserDefinedCode` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns UDC records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • productCode • userDefinedCodeTable • description1

Question	Answer
<p>If I encounter errors while process a transaction, do I need to reverse the transaction?</p>	<p>This operation does not process transactions, therefore, no updates are necessary if you encounter errors.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p> <p>If you do not receive any matching records, review the data in the EnterpriseOne system to verify that the records for which you are searching exist in the database.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

lookupBusinessUnits

The lookupBusinessUnits web service operation is a database query operation that enables consumers to retrieve and review business unit values that are stored in the JD Edwards EnterpriseOne system.

The operation retrieves this data from the EnterpriseOne system, if it exists in the database:

- BusinessUnit
- BusinessUnitType
- BusinessUnitDescription
- Company
- CategoryCodeBusinessUnit, which includes:
 - Division
 - Region
 - Group
 - BranchOffice
 - DepartmentType
 - PersonResponsible
 - LineOfBusiness
 - CategoryCodeCostCt008 through CategoryCodeCostCt023
 - CategoryCodeCostCenter24 through CategoryCodeCostCenter50

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service property that the lookupBusinessUnit operation uses:

Group	Key	Description	Default Value
J0000040	J0000040_MAX_ROWS	Use this business service property to specify the maximum number of F0006 records the operation can return for a query.	100

Implementation Details

This table includes information that can help determine whether the lookupBusinessUnit operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns zero or more records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • BusinessUnit • BusinessUnitType
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This operation does not process transactions, therefore, no updates are necessary if you encounter errors.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p> <p>If you do not receive any matching records, review the data in the EnterpriseOne system to verify that the records for which you are searching exist in the database.</p>
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

getUDCDescription

The getUDCDescription (J0000050) is a database query operation that enables consumers to retrieve the sales order type description values from user-defined code table 00/DT in the EnterpriseOne database. getUDCDescription is called by the getDocumentTypeDescription method.

This operation retrieves the description001 value if data exists in the database.

This operation is called from within the MobileSalesOrderManager published business service (JP42M000).

Note: *Sales Inquiries, in the JD Edwards EnterpriseOne Applications Functionality for Mobile Devices Implementation Guide.*

Implementation Details

This table includes information that can help determine whether the `getUserDefinedCode` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the document type description is returned, based on the selection criteria.</p> <p>At a minimum, the system returns non-zero values for the <code>description001</code> field.</p>
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This operation does not process transactions, therefore, no updates are necessary if you encounter errors.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p> <p>If you do not receive any matching records, review the data in the EnterpriseOne system to verify that the records for which you are searching exist in the database.</p>
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

lookupCurrencyCodes

The `lookupCurrencyCodes` web service operation is a database query operation that enables consumers to retrieve and review currency code values that are stored in the JD Edwards EnterpriseOne system.

The operation retrieves this data from the EnterpriseOne system, if it exists in the database:

- CurrencyCode
- CurrencyDescription
- EditCode
- CurrencyDecimals
- CheckRoutine

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service property that the `lookupCurrencyCodes` operation uses:

Group	Key	Description	Default Value
J0000060	J0000060_MAX_ROWS	Use this business service property to specify the maximum number of F0013	100

Group	Key	Description	Default Value
		records the operation can return for a query.	

Implementation Details

This table includes information that can help determine whether the lookupCurrencyCodes operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns zero or more records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • CurrencyCode • CurrencyCodeDescription
If I encounter errors while process a transaction, do I need to reverse the transaction?	<p>This operation does not process transactions, therefore, no updates are necessary if you encounter errors.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p> <p>If you do not receive any matching records, review the data in the EnterpriseOne system to verify that the records for which you are searching exist in the database.</p>
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

setUserValues and getUserValues

The setUserVaues web service operation is an inbound transaction operation that enables consumers to save the user-entered values from the mobile sales application in the JD Edwards EnterpriseOne database. This operation retrieves these items, if the insert or update database operation is successful:

- numRowsInserted
- numRowsUpdated

If this operation is successful, one of the above fields will contain a value of 1.

The getUserValues web service operation is a database query operation that enables consumers to retrieve saved user-entered values in the mobile sales application from the JD Edwards EnterpriseOne database. This operation retrieves these items for each mobile user-entered value record, if the data exists in the database:

- addressNumberSoldTo
- addressNumberShipTo

- documentOrderInvoiceE
- itemFreeForm
- documentTypesList
- searchOnDate
- todayMinus
- todayPlus
- salesBranchPlantsList
- availabilityBranchPlantsList
- priceBranchPlantsList
- availabilityItemFreeForm
- priceItemFreeForm
- unitOfMeasure
- reference1
- omitZeroQuantities

Implementation Details

This table includes information that can help determine whether the setUserValues operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, one of these fields will contain a value of 1: <ul style="list-style-type: none"> • numRowsInserted • numRowsUpdated
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database Insert/update operation that does not perform transactions. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

This table includes information that can help determine whether the getUserValues operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, zero to many records are returned, based on the selection criteria. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.

Question	Answer
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

getUserProfile

The `getUserProfile` web service operation retrieves `LanguagePreference`, `DateSeparator`, `DateFormat`, and other data from the JD Edwards EnterpriseOne database. This operation retrieves these items, if the data exists in the database:

- `LanguagePreference`
- `DateSeparator`
- `DateFormat`
- `fup`
- `uprc`
- `pqoh`

Implementation Details

This table includes information that can help determine whether the `getUserProfile` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned, based on the selection criteria. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the operation returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • <code>LanguagePreference</code> • <code>DateSeparator</code> • <code>DateFormat</code> • <code>fup</code> • <code>uprc</code> • <code>pqoh</code>
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

lookupUDC

The lookupUDC web service operation retrieves user defined code details of a specific UDC and description from the JD Edwards EnterpriseOne database. This operation retrieves these items if the data exists in the database:

- UserDefinedCode
- Description1
- Description 2

Setup Considerations

Before you use this operation, you can set business services properties to specify how many records the system returns for a query. This table includes information about the business service property that the lookup IDC operation uses:

Group	Key	Description	Default Value
J0000090	J0000090_V0005A_MAX_GRID_ROWS_RETURNED	Use this business service property to specify the maximum number of header records the operation can return for a query.	100

Implementation Details

This table includes information that can help determine whether the lookupUDC operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, zero to many records are returned, based on the selection criteria. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • UserDefinedCode • Description1 • Description2
If I encounter errors while process a transaction, do I need to reverse the transaction?	This is a database query operation that does not perform transactions. If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

40 HCM: Competency

Competency Overview

This chapter provides detailed information about the business interfaces that are available for the Competency business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Competency business object:</p> <ul style="list-style-type: none"> • RTCPOUT, which is a container event for RTCPOUTB.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Competency Real-Time Events

This section describes the available real-time events associated with the Competency business object, which include:

- RTCPOUT, which is a container event for RTCPOUTB.

RTCPOUT

RTCPOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency master records (F05104).

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTCPOUT	Competency Master	RTE	<p>Container</p> <p>Container event for RTCPOUTB (D0500870B).</p>	05

Conceptual Approach

When a user adds, modifies, or deletes competency master data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTCPOUT contains one single event RTCPOUTB.

Objects that Can Initiate the Event

These objects can initiate the RTCPOUT event:

- F05104 - Competency Master
- P05104 - Competency Master

RTCPOUTB

RTCPOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency master records (F05104).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCPOUTB	Competency Details	RTE	Single	05	D0500870B: Competency Real Time Event Notification Structure B	Included in container event RTCPOUT.

Conceptual Approach

When a user adds, modifies, or deletes competency master data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTCPOUT contains one single event RTCPOUTB. The RTCPOUTB event contains all columns from the modified competency master record (F05104) and an action code specifying what action was performed on the data (added, changed, deleted).

41 HCM: Employee Master

Employee Master Overview

This chapter provides detailed information about the business interfaces that are available for the Employee Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business services	<p>The EmployeeManager web service (JP080020) manages the processing of these web service operations:</p> <ul style="list-style-type: none"> • processEmployee (J0800040) • getEmployee (J0800041) • processPendingEmployee (J0800050)
Real-Time Events	<p>This list includes the real-time events for the Employee Master business object:</p> <ul style="list-style-type: none"> • RTEMOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTEMOUTB ○ RTEMOUTC ○ RTEMOUTD • RTPIOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTPIOUTB ○ RTPIOUTC
Batch Import Programs	<p>This list includes the batch import programs for the Employee Master business object:</p> <ul style="list-style-type: none"> • Process Pending Employees report (R060116P)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

EmployeeManager Business Services

The EmployeeManager web service (JP080020) manages the processing of these web service operations:

Operation	Description
processEmployee	You use the processEmployee web service operation to add, change (net change) or delete employee and employee attachments (media objects) records in JD Edwards EnterpriseOne. You can also add or change address book, phone, and electronic address records, but cannot delete them.
getEmployee	You use the getEmployee web service operation to get employee data from the JD Edwards EnterpriseOne database based on the query criteria.
processPendingEmployee	You use the processPendingEmployee web service operation to insert employee data into table F060116Z (Unedited Quick Hire Transaction File). This service allows inserting multiple employee records at the same time. You cannot update or delete records using this operation.

Accessing Javadoc for the EmployeeManager Web Service Operations

To access Javadoc for the EmployeeManager web service and its related operations, review these Javadoc packages:

- JP080020 (EmployeeManager)
- J0800040 (processEmployee)
- J0800041 (getEmployee)
- J0800050 (processPendingEmployee)

processEmployee

The processEmployee web service operation is an inbound transaction operation that enables consumers to process employee information within the JD Edwards EnterpriseOne system.

The consumer can complete the following tasks in the JD Edwards EnterpriseOne Human Capital Management system. As you work with employee records, you can also work with related address book records for the employee.

- Add, change, or delete employee records.
- Add or change address book records for employees when working with employee records.
- Add or change phone number records for employees when working with employee records.
- Add or change electronic address records for employees when working with employee records.
- Add, change, or delete tax override records of US employees.
- Add or change tax overrides records of Canada, Australia, and New Zealand employees.
- Add, change, or delete additional tax overrides of US and Canada employees.
- Add or delete Media Objects for employees.

The processEmployee web service operation calls the processAddressBook (J0100001) business service (BSSV) to process employee address information. If all validations succeed, the processEmployee operation in Add mode calls the Process Employee Integrations Wrapper BSFN to process the employee record. In Update mode, the web service operation retrieves data from the Employee Master table (F060116), and its related data, for net change. The process then calls the performNetChange utility (J0000020) to perform net change. After the net change is done, the web

service calls the Process Employee Integrations Wrapper BSFN to process the employee record. In Delete mode, the web service calls the Wrapper BSFN to delete the employee and its related data.

The EmployeeManager published business service also uses the processEntity (J0100010) utility business service when it compares values for address book records and uses the getProcessingVersion (J0000010) utility business service to retrieve program versions.

Note: The processAddressBook BSSV processes the Tax ID number of the employee, which may be considered sensitive information. The BSSV does not treat the Tax ID differently than other information. Address Book Data privacy provided in the JD Edwards EnterpriseOne software is not supported by this BSSV. If the processAddressBook BSSV fails, it will not process the address book information and will rollback any address book changes.

The processEmployee web service operation calls the Process Employee Integrations Wrapper BSFN (B0801350) to process an Employee record. B0801350 in turn primarily calls the Employee Master MBF (B0500575) to process the employee record. The wrapper BSFN (B0801350) also has calls to following BSFN's:

- N0800016 - HRM Tax Id Edit US
- N0800017 - HRM Tax Id Edit CA
- N75A0002 - Validate Tax Id - AU
- N75Z0002 - Validate Tax Id - NZ
- N0000604 - Validate Application Id
- N0000602 - F00095 Record Reservation
- B0000603 - Set Object Reservation Text Substitution Error
- N0800002 - Set Employee Current Status
- B0500575 - Employee Master MBF
- B9900410 - Get Environment Value
- N0800030 - Fiscal Year Begin End Dates
- X0005 - Get UDC
- B9800420 - Get Data Dictionary Definition
- B0000564 - Get Internal Next Number
- B9800140 - Get User Address Book Number
- B0800010 - Get Organizational PO
- N0500042 - F08001 Retrieve Job Specific Info
- N0800003 - Verify Job/Retrieve Job Default Information
- N0800204 - Track Date in Current Job
- N0800006 - Calculate Pay Period Salary from Annual Salary Input
- N0500025 - Salary Calculations Server
- N0800004 - Compa-Ratio Server
- N0800146 - Get a specific ESS Common Setting
- N0700040 - F069016 Fetch Tax Area Profile
- B0500543 - Validate Geo Code HR
- N0500020 - Update Payroll Lock Out Code
- B0500132 - F06017 Canadian Tax Info

- N0800008 - Update F06210 to Force Payroll Rerun
- N0500032 - Synchronize EE/Supervisor in EE Master & Org Structure
- N0800020 - Update Employee Job Server

Note: The processEmployee web service and its related web services and business functions perform all of the required validations for country-specific data for Australia (AU), Canada (CA), New Zealand (NZ), and the United States (US) including tax ID, tax overrides, and payroll data.

If all validations are successful, the processEmployee web service operation writes to these JD Edwards tables:

- F060116 (Employee Master)
- F0101 (Address Book Master)
- F0111 (Who's Who)
- F0115 (Phones)
- F0016 (Address by Effective Date)
- F01151 (Electronic Addresses)
- F060117 (Employee Master International Data)
- F060117A (Employee Master International Data Tag)
- F060120 (Employee Master Additional Informational File)
- F06017 (Employee Tax Exceptions/Overrides)
- F0701732 (Additional Tax Overrides)
- F75A0006 (Australian Employee Tax Overrides)
- F75Z0005 (New Zealand Employee Tax Overrides)

Required Fields and Values

You must provide values for the following fields when using the processEmployee web service:

- Action Code
- Address Number
Required for change and delete actions.
- Alpha Name
- Social Security Number
Required in Add mode for US and Canadian employees.
- Rehire
Required when rehiring a former employee.
- Cost Center Home
- Sex Male/Female
- Minority EEO
Values must exist in UDC 06/M.
- RT Salary
A value for either the salary or the hourly field must be provided, depending on the employee's pay class.

- RT Hourly
A value for either the salary or the hourly field must be provided, depending on the employee's pay class.
- Date of Birth
Required for Canada only.
- Tax Area Work
- Tax Area Residence
- Tax File Number
Required for Australia and New Zealand employee in Add mode.

You must pass the listed values for the following fields and options to the BSSV because these values are in drop-down combo boxes or radio buttons:

- JIT Values field in the Additional Tax Overrides (USA) program (P0701732).
You must pass the values as one of these numeric characters: 0, 1, -1, 2, 3, 4, 5, 6, 7, 8, 9.
- Basis of Employment options in the Australian Employee Tax Override program (P75A0005) require the following values:
 - Blank: Full Time
 - **3**: Part Time
 - **4**: Casual
- Variation From Levy option in the Australian Employee Tax Override program (P75A0005) require the following values:
 - **1**: 1.5%
 - **2**: 1.25%
 - **3**: 1%
 - **4**: Exempt
- Extra Emoluments Payment options in the New Zealand Employee Tax ID Number program (P75Z0002) require following values:
 - **1**: No Override Selected
 - **2**: 7.5% override selected
 - **3**: 30% overrides selected
 - **4**: 33% override selected

Setup Considerations

Before you use this operation, you can set business service properties to specify the version of called programs. This table includes information about the business service properties that the processEmployee operation uses:

Group	Key	Description	Default Value
J0800040	J0800040_EMP_MBF_VERSION	Employee Master Processing Version	ZJDE0001

Group	Key	Description	Default Value
		See <i>"Setting Processing Options for Employee Master (P0801)" in the JD Edwards EnterpriseOne Applications Human Capital Management Fundamentals Implementation Guide .</i>	
J0800040	J0800040_AB_MBF_VERSION	Address Book MBF Version when called by Employee BSSV See <i>"Setting Processing Options for Address Book MBF (P0100041)" in the JD Edwards EnterpriseOne Applications Address Book Implementation Guide .</i>	ZJDE0001

Note: The processing options in versions that you specify in the business service properties determine how data is processed and how certain fields are populated in the JD Edwards EnterpriseOne database tables.

Implementation Details

This table includes information that can help determine whether the processEmployee operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the EnterpriseOne system. The system returns non-zero values for certain fields. See <i>Returned Values</i> .
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required. Error and warning messages from EnterpriseOne are returned to the EmployeeManager PBSSV as an E1MessageList. If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.
Does this operation use record reservation?	Yes. This operation calls the F00095 Record Reservation business function (N0000602). This function reserves records in the JD Edwards EnterpriseOne system when the operation is changing or deleting existing data.

Returned Values

At a minimum, the system return values for these fields:

- EntityID
- Entity Type Code
- Alpha Name
- Company
- Home Business Unit
- Gender
- Ethnic Code
- Pay Class
- Pay Frequency
- Salary
- Hourly Rate
- Date – Original Employment
- Date Started
- Pay Start Date
- Date – Current Position
- Country Code
- Change Reason
- Tax Area Work
- Tax Area Residence
- Full Time Equivalent

getEmployee

The getEmployee web service operation is a database query operation that enables consumers to retrieve and review employee information from the JD Edwards EnterpriseOne database.

The getEmployee operation (J0800041) is managed by the EmployeeManager PBSSV and uses the following value objects to fetch employee information based on the query criteria:

- InternalGetEmployee
- InternalGetEmployeeWhereFields
- InternalShowEmployee
- InternalShowAddTaxCAandUS
- InternalShowTaxOverridesAU
- InternalShowTaxOverridesCA
- InternalShowTaxOverridesNZ
- InternalShowTaxOverridesUS

- MOItem_Internal

This getEmployee operation retrieves data from these tables, based on the query:

- F0101 (Address Book Master)
- F0111 (Who's Who)
- F0016 (Address by Effective Date)
- F0115 (Address Book - Phone Numbers)
- F01151 (Electronic Address)
- F060116 (Employee Master)
- F060117 (Employee Master International Data)
- F060117A (Employee Master International Data Tag)
- F060120 (Employee Master Additional Informational File)
- F06017 (Employee Tax Exceptions/Overrides)
- F0701732 (Additional Tax Overrides)
- F75A0006 (Australian Employee Tax Overrides)
- F75Z0005 (New Zealand Employee Tax Overrides)

Setup Considerations

Before you use this operation, you can set business service properties to specify the number of records retrieved. This table includes information about the business service property that the getEmployee operation uses:

Group	Key	Description	Default Value
J0800041	J0800041_MAX_ROWS	Maximum rows retrieved	100

Implementation Details

This table includes information that can help determine whether the getEmployee operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. Some of the fields can include blank or zero values depending on the data that exists in the JD Edwards EnterpriseOne tables.
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation does not process transactions; therefore, no updates are necessary if you encounter errors. Error and warning messages from EnterpriseOne are returned to the EmployeeManager PBSSV as an E1MessageList. If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Question	Answer

processPendingEmployee

The processPendingEmployee web service operation is an inbound operation that enables consumers to insert employee details into the Unedited Quick Hire Transaction table (F060116Z). The web service also provides a success or error message. The processPendingemployee operation will pass the pending employees' information to the ProcessPendingEmployeeWrapper BSFN (B0800670) which will insert the records into the F060116Z table.

Setup Considerations

Before you use this operation, you can set business service properties to specify the versions of programs to use. The processing options in versions that you specify in the business service properties determine how data is processed and how certain fields are populated in the JD Edwards EnterpriseOne database tables.

Group	Key	Description	Default Value
J0800050	None	Version of Employee Quick Hire (P060116Q)	ZJDE0001

Required Fields and Values

You must provide values for the Social Security Number field when using the processPendingEmployee web service.

Additionally, if the version of the Employee Quick Hire program (P060116Q) that you specify in the business service properties has the New Hire processing option activated, you must also provide values for the following fields:

- Alpha Name
- Supervisor Number
- Date Started
- Job Type and Step

Implementation Details

This table includes information that can help determine whether the processPendingEmployee operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns records to the consumer. At a minimum, the system returns a value for the batch number and the number of records inserted. If the operation does not complete successfully, the system returns an error message.
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.

Question	Answer
	Error and warning messages from EnterpriseOne are returned to the EmployeeManager PBSSV as an E1MessageList. If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Employee Master Real-Time Events

This section describes the available real-time events associated with the Employee Master business object, which include:

- RTEMOUT, which is a container event for these single events:
 - RTEMOUTB
 - RTEMOUTC
 - RTEMOUTD
- RTPIOUT, which is a container event for these single events:
 - RTPIOUTB
 - RTPIOUTC

RTEMOUT

RTEMOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060116).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTEMOUT	Employee Master	RTE	Container Container event for: <ul style="list-style-type: none"> • RTEMOUTB (D0500873B) • RTEMOUTC (D0500873C) • RTEMOUTD (D0500873D) 	05	D0500873A: Employee Master RTE Notification A Structure

Conceptual Approach

When a user adds, modifies, or deletes employee master data from the JD Edwards EnterpriseOne system, the system publishes an event. The RTEMOUT event is a container event activated on the F060116, F060117, and F060120 table triggers and contains all columns from the modified records and an action code specifying what action was performed

on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116, F060117, and F060120) is modified, the corresponding record from the other table is automatically fetched and included in the container.

Objects that Can Initiate the Event

These objects can initiate the RTEMOUTB event, and the single events contained within the event:

- F060117 - Employee Master - International Data table trigger
- B0500575 - Employee Master Business Function

RTEMOUTB

RTEMOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060116).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTEMOUTB	Employee Details	RTE	Single	05	D0500873B: Employee Master RTE Notification B Structure	Included in container event RTEMOUT.

Conceptual Approach

When a user adds, modifies, or deletes employee master data, F060116 table only, from the JD Edwards EnterpriseOne system, the system publishes an event. The RTEMOUT event is a container event activated on the F060116 table trigger and contains all columns from the modified employee master record (F060116) and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116, F060117, and F060120) is modified, the corresponding record from the other table is automatically fetched and included in the container.

RTEMOUTC

RTEMOUTC is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060117).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTEMOUTC	Employee International Details	RTE	Single	05	D0500873C: Employee Master RTE Notification C Structure	Included in container event RTEMOUT.

Conceptual Approach

When a user adds, modifies, or deletes employee master data, F060117 table only, from the JD Edwards EnterpriseOne system, the system publishes an event. The RTEMOUT event is a container event activated on the F060117 table trigger and contains all columns from the modified employee master record (F060117) and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by

the container event (F060116, F060117, and F060120) is modified, the corresponding record from the other table is automatically fetched and included in the container.

RTEMOUTD

RTEMOUTD is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060120).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTEMOUTD	Employee Additional Details	RTE	Single	05	D0500873D: Employee Master RTE Notification D Structure	Included in container event RTEMOUT.

Conceptual Approach

When a user adds, modifies, or deletes employee master data, F060120 table only, from the JD Edwards EnterpriseOne system, the system publishes an event. The RTEMOUT event is a container event activated on the F060120 table trigger and contains all columns from the modified employee master record (F060120) and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116, F060117, and F060120) is modified, the corresponding record from the other table is automatically fetched and included in the container.

RTPIOUT

RTPIOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060116). This message is similar to the message issued by RTEMOUT.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTPIOUT	Employee Personal Information	RTE	Container Container event for: <ul style="list-style-type: none"> RTPIOUTB (D0500881B) RTPIOUTC (D0500881C) 	05	D0500873A: Employee Master RTE Notification A Structure

Conceptual Approach

When a user adds, modifies, or deletes employee master data from the JD Edwards EnterpriseOne system, the system publishes an event. The RTPIOUT event is a container event activated within the employee master business function and contains a specific subset of columns, and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116 and F060117) is modified the corresponding record from the other table is automatically fetched and included in the container.

Objects That Can Initiate The Event

These objects can initiate the RTPIOUT event, and the single events contained within it:

- B0500575 - Employee MBF International Data

RTPIOUTB

RTPIOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to employee master records (F060116). This message is similar to the message issued by RTEMOUTB.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPIOUTB	Employee Personal Details	RTE	Single	05	D0500881B: Person Information RTE Notification B Structure	Included in container event RTPIOUT.

Conceptual Approach

When a user adds, modifies, or deletes employee master data, F060116 table only, from the JD Edwards EnterpriseOne system, the system publishes an event out to a third party system. The RTPIOUT event is a container event activated within the employee master business function and contains a specific subset of columns, from the modified employee master record (F060116) and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116 and F060117) is modified, the corresponding record from the other table is automatically fetched and included in the container.

RTPIOUTC

RTPIOUTC is a real-time event notification used by the JD Edwards EnterpriseOne system to notify external systems of changes to employee master records (F060117). This message is similar to the message issued by RTEMOUTC.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPIOUTC	Emp Per International Details	RTE	Single	05	D0500881C: Person Information RTE Notification C Structure	Included in container event RTPIOUT.

Conceptual Approach

When a user adds, modifies, or deletes employee master data, F060117 table only, from the JD Edwards EnterpriseOne system, the system publishes an event out to a third party system. The RTPIOUT event is a container event activated within the employee master business function and contains a specific subset of columns, from the modified employee master record (F060117) and an action code specifying what action was performed on the data (added, changed, deleted). Whenever data from one of the tables handled by the container event (F060116 and F060117) is modified the corresponding record from the other table is automatically fetched and included in the container.

Employee Master Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Employee Master business object:

Program and Program Type	Description	Available Documentation
Process Pending Employees report (R060116P) This is a temporary table import program.	The Process Pending Employees report (R060116P) updates the Employee Master Information table (F060116) and the Address Book Master table (F0101) with information from the Unedited Quick Hire Transaction File table (F060116Z).	See this topic: <ul style="list-style-type: none"><li data-bbox="781 730 1068 779">• <i>Understanding the Process Pending Employees Report</i>

42 HCM: Job

Job Overview

This chapter provides detailed information about the business interfaces that are available for the Job business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Job business object:</p> <ul style="list-style-type: none"> • RTJBOUT, which is a container event for RTJBOUTB. • RTJDOUT, which is a container event for RTJDOUTB. • RTJBUOUT, which is a container event for RTJBUOUTB.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Job Real-Time Events

This section describes the available real-time events associated with the Job business object, which include:

- RTJBOUT, which is a container event for RTJBOUTB.
- RTJDOUT, which is a container event for RTJDOUTB.
- RTJBUOUT, which is a container event for RTJBUOUTB.

RTJBOUT

RTJBOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to job information records (F08001).

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTJBOUT	Job	RTE	Container Container event for RTJBOUTB (D0800630B).	08

Conceptual Approach

When a user adds, modifies, or deletes job information data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTJBOUT contains one single event RTJBOUTB.

Objects that Can Initiate the Event

These objects can initiate the RTJBOUT event:

- P08001 - Job Entry and Evaluation
- P08JC - Category Codes Window
- P08001US - Job Entry U.S. Legislative/Regulatory Data
- P08001CA- Job Entry Canadian Legislative/Regulatory Data
- P050011 - Job Evaluation

RTJBOUTB

RTJBOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to job information records (F08001).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTJBOUTB	Job Entry and Evaluation	RTE	Single	08	D0800630B: Jobs Real Time Event Notification Structure B	Included in container event RTJBOUT.

Conceptual Approach

When a user adds, modifies, or deletes job information data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTJBOUT contains one single event RTJBOUTB. The RTJBOUTB event contains all columns from the modified job information record (F08001) and an action code specifying what action was performed on the data (added, changed, deleted).

RTJDOUT

RTJDOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency based job description records.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTJDOUT	Job Description	RTE	Container Container event for RTJDOUTB (D0800740B).	08

Conceptual Approach

When a user adds, modifies, or deletes competency based job description data in the JD Edwards EnterpriseOne Human Resources system and then runs R080025, Incremental Job Description Changes Outbound Integration program, the system publishes an event. The container event RTJDOUT contains one single event RTJDOUTB.

Objects that Can Initiate the Event

The N0800741 - Send Job Desc Integration Notice object can initiate the RTJDOUT event, or the single event contained by the event.

RTJDOUTB

RTJDOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency based job description records.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTJDOUTB	Jobs in an Organization	RTE	Single	08	D0800740B: Competency Based Job Desc RTE Notify	Included in container event RTJDOUT.

Conceptual Approach

When a user adds, modifies, or deletes competency based job description data in the JD Edwards EnterpriseOne Human Resources system and then runs R080025, Incremental Job Description Changes Outbound Integration program, the system publishes an event. The container event RTJDOUT contains one single event RTJDOUTB. The RTJDOUTB event contains all columns from the modified competency based job description record and an action code specifying what action was performed on the data (added, changed, deleted).

RTJBUOUT

RTJBUOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to the organizational business unit/jobs cross reference (F0800601).

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTJBUOUT	Jobs in Business Unit	RTE	Container Container event for RTJBUOUTB (D0800770B).	08

Conceptual Approach

When a user adds, modifies, or deletes organizational business unit/jobs cross reference data (F0800601) from the JD Edwards EnterpriseOne system, the system publishes an event. The RTJBUOUT event is a container event activated on the F0800601 table trigger and contains the single event RTJBUOUTB.

Objects that Can Initiate the Event

These objects can initiate the RTJBUOUT event, and the single event contained by the event:

- F0800601 - Organizational Business Unit/Job Cross Reference table trigger

RTJBUOUTB

RTJBUOUTB is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to the organizational business unit/jobs cross reference (F0800601).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTJBUOUTB	Org BU /Jobs Cross Reference	RTE	Single	08	D0800770B: Data Structure for B0800770 RTE	Included in container event RTJBUOUT.

Conceptual Approach

When a user adds, modifies, or deletes organizational business unit/jobs cross reference data (F0800601) from the JD Edwards EnterpriseOne system, the system publishes an event. The RTJBUOUT event is a container event activated on the F0800601 table trigger and contains the single event RTJBUOUTB. The RTJBUOUTB event contains all columns from the modified organizational business unit/jobs cross reference record (F0800601) and an action code specifying what action was performed on the data (added, changed, deleted).

43 HCM: Resource Competency

Resource Competency Overview

This chapter provides detailed information about the business interfaces that are available for the Resource Competency business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Resource Competency business object:</p> <ul style="list-style-type: none"> • RTSLOUT, which is a container event for these single events: <ul style="list-style-type: none"> ◦ RTSLDHDR ◦ RTSLDTL • RTRCOUT, which is a container event for RTRCHDR.
Batch Import	<p>This batch import program is available for the Resource Competency business object:</p> <ul style="list-style-type: none"> • Update Resource Competencies (R05100Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Resource Competency Real-Time Events

This section describes the available real-time events associated with the Resource Competency business object, which include:

- RTSLOUT, which is a container event for these single events:
 - RTSLDHDR
 - RTSLDTL
- RTRCOUT, which is a container event for RTRCHDR.

RTSLOUT

RTSLHDR is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency scales records.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSLOUT	Competency Scale	RTE	Container Container event for: <ul style="list-style-type: none"> RTSLHDR (D0500871B) RTSLDTL (D0500871C) 	05

Conceptual Approach

When a user adds, modifies, or deletes scales data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. This container event, RTSLOUT, contains two single events RTSLHDR and RTSLDTL.

Objects that Can Initiate the Event

These objects can initiate the RTSLOUT event, and the single events contained within the event:

- P05103 - Scale Information

RTSLHDR

RTSLHDR is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency scales header records (F05103).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSLHDR	Competency Scale Header	RTE	Single	05	D0500871B: Scales Real Time Event Notification B Structure (F05103)	Included in container event RTSLOUT.

Conceptual Approach

When a user adds, modifies, or deletes scales data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. This container event, RTSLOUT, contains two single events RTSLHDR and RTSLDTL. The RTSLHDR event contains all columns from the modified scales header record (F05103) and an action code specifying what action was performed on the data (added, changed, deleted).

RTSLDTL

RTSLDTL is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to competency scales detail records (F05119).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSLDTL	Competency Scale Detail	RTE	Single	05	D0500871C: Scales Real Time Event Notification C Structure (F05119)	Included in container event RTSLOUT.

Conceptual Approach

When a user adds, modifies, or deletes scales data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTSLOUT contains two single events RTSLHDR and RTSLDTL. The RTSLDTL event contains all columns from the modified scales detail record (F05119) and an action code specifying what action was performed on the data.

RTRCOUT

RTRCOUT is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to resource competency header records.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTRCOUT	Resource Competency	RTE	Container Container event for RTRCHDR (D0500872B).	05

Conceptual Approach

When a user adds, modifies, or deletes resource competency data from the JD Edwards EnterpriseOne Human Resources system, the system publishes the RTRCOUT event. The container event RTRCOUT contains one single event RTRCHDR.

Objects that Can Initiate the Event

These objects can initiate the RTRCOUT event:

- F05100 - Resource Competency

RTRCHDR

RTRCHDR is a real-time event notification used by JD Edwards EnterpriseOne to notify external systems of changes to resource competency header records (F05100).

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTRCHDR	Resource Competency Header	RTE	Single	05	D0500872B: Resource Competency	Included in container event RTRCOUT.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
					Real Time Event Notification B Structure	

Conceptual Approach

When a user adds, modifies, or deletes resource competency data from the JD Edwards EnterpriseOne Human Resources system, the system publishes an event. The container event RTRCOUT contains one single event RTRCHDR. The RTRCHDR event contains all columns from the modified resource competency header record (F05100) and an action code specifying what action was performed on the data (added, changed, deleted). Current integrations do not require synchronization of detail data. However, real-time event business function (B0500872) has been designed to implement this functionality in the future if required.

Resource Competency Batch Import Programs

You use batch import programs to retrieve and format data from external systems so that it can be used in the JD Edwards EnterpriseOne system.

This table lists and describes the available batch import programs that are associated with the Resource Competency business object:

Program and Program Type	Description	Available Documentation
Update Resource Competencies (R05100Z1)	<p>When users manually enter resource competency details using the Manage Resource Competency program (P05100Z) competency), the system updates the Resource Competencies Unedited Transaction table (F05100Z1).</p> <p>After an entry is made in the F05100Z1 table, you run the R05100Z1, which updates the competency records of an employee gets updated in the Resource Competency table (F05100) based on the processing option settings.</p>	You can access this program from the Batch Versions program by entering BV in the Fast Path field.

44 HCM: Human Resource Requisitions

Requisition Components Overview

This chapter provides detailed information about the business interfaces that are available for human resource requisitions.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The RequisitionManager (JP080010) manages the getRequisition (J0800020) and processRequisition (J0800021) operations.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

RequisitionManager Business Service

The RequisitionManager web service (JP080010) manages the processing of these web service operations:

Operation	Description
getRequisition	Use this operation to obtain data from the Requisition Information (F08102) and Job Information (F08001) tables, based on the query criteria.
processRequisition	Use this operation to add, update and delete requisition details into the Requisition Information table (F08102).

Prerequisites

Before using the RequisitionManager web service, or any of the related web service operations, you must install and configure the JD Edwards EnterpriseOne Address Book and JD Edwards EnterpriseOne Human Capital Management systems.

See the [JD Edwards EnterpriseOne Applications Address Book Implementation Guide](#) and the [JD Edwards EnterpriseOne Applications Human Capital Management Fundamentals Implementation Guide](#).

Accessing Javadoc for the RequisitionManager Web Service

To access Javadoc for the RequisitionManager web service and it's related operations, review these Javadoc packages:

- JP080010 (RequisitionManager)
- J0800020 (getRequisition)
- J0800021 (processRequisition)

getRequisition

The getRequisition web service operation is a database query operation that enables consumers to obtain requisition data from the JD Edwards EnterpriseOne system. It is managed by the RequisitionManager PBSSV and uses the InternalGetRequisition, InternalGetRequisitionWhereFields, and InternalShowRequisition value objects. The process obtains data from these tables, based on the query criteria:

- Requisition Information (F08102)
- Job Information (F08001)

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service property that the getRequisition web service operation uses:

Group	Key	Description	Default Value
J0800020	J0800020_MAX_ROWS	Maximum Rows Retrieved	100

Implementation Details

This table includes information that can help determine whether the getRequisition web service operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. Some of the fields can include blank or zero values depending on the data that exists in the JD Edwards EnterpriseOne tables.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • Requisition Number • Home Business Unit • Security Business Unit

Question	Answer
	<ul style="list-style-type: none"> • Fiscal Year • Requested Date • Head Count • Date Updated <p>If errors occur, errors and warning messages are returned from EnterpriseOne to the RequisitionManager PBSSV as an E1MessageList.</p>
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>No. This operation does not process transactions; therefore, no updates are necessary if you encounter errors.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p> <p>If you do not receive any matching records, review the data in the EnterpriseOne system to verify that the records for which you are searching exist in the database.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

processRequisition

The processRequisition web service operation is an inbound transaction operation that enables consumers to add, update, and delete requisition details in the Requisition Information table (F08102). The web service operation also supports add or delete in the Media Objects for Requisition when working with Requisition records. It is managed by the RequisitionManager PBSSV and uses the InternalProcessRequisition value object.

Required Fields

You must provide values for the following fields when using the processEmployee web service:

- Business Unit
- Home Business Unit
- Requested By

Implementation Details

This table includes information that can help determine whether the processRequisition web service operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the operation completes successfully, the web service returns a success message to the RequisitionManager PBSSV as an E1MessageList.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • Requisition Number

Question	Answer
	<ul style="list-style-type: none">• Headcount <p>If errors occur, error and warning messages are returned from EnterpriseOne to the RequisitionManager PBSSV as an E1MessageList.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.</p> <p>If you receive errors while processing this operation, review your selection criteria and business service properties, and then retry your query.</p>
Does this operation use record reservation?	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

45 HCM: Timecard

Timecard Overview

This chapter provides detailed information about the business interfaces that are available for the Timecard business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The BatchTimeSheetManager published business service (JP05001) manages the processing of the BatchTimeSheetInsertProcessor Web service operation (J0500010). See also: <i>Project</i> .
Batch Import Programs	This list includes the batch import programs for the Timecard business object: <ul style="list-style-type: none"> Time Entry Batch Processor (R05116Z11)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Timecard Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Timecard business object:

Program and Program Type	Description	Available Documentation
Time Entry Batch Processor (R05116Z11) This is a temporary table import program.	After you review and revise uploaded timecard information, you can create timecards. You then include those timecards in the usual payroll cycle processing. When you create timecards from uploaded timecard information, the system verifies the information against constants tables in the JD Edwards EnterpriseOne Payroll system.	See this topic: <ul style="list-style-type: none"> <i>Understanding the Creation of Timecards from Uploaded Information</i>

Program and Program Type	Description	Available Documentation
	When you run the Time Entry Batch Processor, the system transfers the uploaded information from the F06116Z1 table to the F06116 table.	

46 Project Management: Bid Master

Bid Master Overview

This chapter provides detailed information about the business interfaces that are available for the Bid Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Bid Master business object:</p> <ul style="list-style-type: none"> • RTHBVAOUT, which contains these single events: <ul style="list-style-type: none"> ◦ RTHBVAEA ◦ RTHBVAHDR ◦ RTHBVAPH

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Bid Master Real-Time Events

This section describes the available real-time events associated with the Bid Master business object which include:

- RTHBVAOUT, which contains these single events:
 - RTHBVAEA
 - RTHBVAHDR
 - RTHBVAPH

RTHBVAOUT

RTHBVAOUT is used by JD Edwards EnterpriseOne to inform or notify third-party systems of changes to the F44H604 table by the vendor assignment application. This includes adds, changes, and deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTHBVAOUT	Vendor Assignment	RTE	Container Container event for: <ul style="list-style-type: none"> RTHBVAEA (D44H9902A) RTHBVAHDR (D44H9902B) RTBVAPH (D44H9902C) 	44H

Conceptual Approach

In the J.D. Edwards Homebuilder management system, a vendor is assigned to a community for a specific trade. A vendor assigned to multiple trades in a community will have multiple records in F44H604. A third party system needs to know if a change is a change to the assignment, or vendor address fields. Wildcards (+) can exist in community and phase fields. Third party systems need to support wildcards on their side. Adds, changes and deletes in a vendor assignment will trigger a real time event.

Objects that Can Initiate the Event

These objects can initiate the RTHBVAOUT event, and the single events contained within it:

- P44H604 - Supplier Assignment

RTHBVAEA

RTHBVAEA is used by JD Edwards EnterpriseOne to inform or notify third-party systems of changes to F44H604 vendor emails by the vendor assignment application. This includes adds, changes, and deletions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBVAEA	Vendor Assignment Elec Address	RTE	Single	44H	D44H9902A - Vendor Assignment Real Time - Header Record	Included in container event RTHBVAOUT.

Conceptual Approach

In the J.D. Edwards Homebuilder management system, a vendor is assigned to a community for a specific trade. A vendor assigned to multiple trades in a community will have multiple records in F44H604. A third party system needs to know if a change is a change to the assignment, or vendor address fields. Wildcards (+) can exist in community and phase fields. Third party systems need to support wildcards on their side. Adds, changes and deletes in a vendor assignment will trigger a real time event.

RTHBVAHDR

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBVAHDR	Vendor Assignment Header	RTE	Single	44H	D44H9902B - Vendor Assignment Real Time - Phones Record	Included in container event RTHBVAOUT.

Conceptual Approach

In the J.D. Edwards Homebuilder management system, a vendor is assigned to a community for a specific trade. A vendor assigned to multiple trades in a community will have multiple records in F44H604. A third party system needs to know if a change is a change to the assignment, or vendor address fields. Wildcards (+) can exist in community and phase fields. Third party systems need to support wildcards on their side. Adds, changes and deletes in a vendor phone number will trigger a real time event.

RTHBVAPH

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBVAPH	Vendor Assignment Phone	RTE	Single	44H	D44H9902C - Vendor Assignment Real Time - E-mail Record	Included in container event RTHBVAOUT.

Conceptual Approach

In the J.D. Edwards Homebuilder management system, a vendor is assigned to a community for a specific trade. A vendor assigned to multiple trades in a community will have multiple records in F44H604. A third party system needs to know if a change is a change to the assignment, or vendor address fields. Wildcards (+) can exist in community and phase fields. Third party systems need to support wildcards on their side. Adds, changes and deletes in a vendor e-mail address will trigger a real time event.

47 Project Management: Community Master

Community Master Overview

This chapter provides detailed information about the business interfaces that are available for the Community Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Community Master business object:</p> <ul style="list-style-type: none"> • RTHBCMOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTHBCMHDR ○ RTHBCMCPH ○ RTHBCMCEA ○ RTHBCMBTPH ○ RTHBCMBTEA ○ RTHBCMSTPH ○ RTHBCMSTEA

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Community Master Real-Time Events

This section describes the available real-time events associated with the Community Master business object, which include:

- RTHBCMOUT, which is a container event for these single events:
 - RTHBCMHDR
 - RTHBCMCPH
 - RTHBCMCEA
 - RTHBCMBTPH
 - RTHBCMBTEA

- RTHBCMSTPH
- RTHBCMSTEA

RTHBCMOUT

RTHBCMOUT is used by JD Edwards EnterpriseOne to inform or notify third-party systems when a community has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTHBCMOUT	Community Master	RTE	Container Container event for: <ul style="list-style-type: none"> • RTHBCMHDR • RTHHBCMPH • RTHBCMCEA • RTHBCMSTPH • RTHBCMSTEA 	44H

Conceptual Approach

If a Community is added or modified in a system, the information is sent through an XPI integration to another third-party system for add or update.

Objects That Can Initiate the Event

This object can initiate this container event, and the events contained within it:

- N44H9901 - Community Master Outbound Notification Wrapper

RTHBCMHDR

RTHBCMHDR is used by JD Edwards EnterpriseOne to inform or notify third-party systems when a Community has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMHDR	Community Master Header	RTE	Single	44H	D44H9901D	Included in container event .RTHBCMOUT

Conceptual Approach

If a Community is added or modified in a system, the information is sent through an XPI integration to another third-party system for add or update.

RTHBCMCPH

RTHBCMCPH is used by EnterpriseOne Homebuilder to inform or notify third-party systems when a Community contact phone has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMCPH	CM Contact Phones	RTE	Single	44H	D44H9901E	Included in container event RTHBCMOUT.

Conceptual Approach

If a Community contact phone is added or modified in a system, the information will be sent through an XPI integration to another third-party system for add or update.

RTHBCMCEA

RTHBCMCEA is used by EnterpriseOne Homebuilder to inform or notify third-party systems when a Community Email Address has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMCEA	CM Contact Electronic Address	RTE	Single	44H	D44H9901F	Included in container event RTHBCMOUT.

Conceptual Approach

If a Community Email Address is added or modified in a system, the information will be sent through an XPI integration to another third-party system for add or update.

RTHBCMTPH

RTHBCMTPH is used by J. D. Edwards Homebuilder to inform or notify third-party systems when a Community has been added or changed in order to keep the two systems synchronized. This single event holds information about community bill to phone.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMTPH	CM Bill To Phones	RTE	Single	44H	D44H9901G	Included in container event RTHBCMOUT.

Conceptual Approach

If a Community is added or modified in a system, the information about bill to phone will be sent through an XPI integration to another third-party system for add or update.

RTHBCMBTEA

RTHBCMBTEA is used by EnterpriseOne Homebuilder to inform or notify third-party systems when a Community has been added or changed in order to keep the two systems synchronized. This single event holds information about community Bill to email address.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMBTEA	CM Bill To Electronic Address	RTE	Single	44H	D44H9901H	Included in container event RTHBCMOUT.

Conceptual Approach

If a Community is added or modified in a system, the information about bill to email address will be sent through an XPI integration to another third-party system for add or update.

RTHBCMSTPH

RTHBCMSTPH is used by EnterpriseOne Homebuilder to inform or notify third-party systems when a Community has been added or changed in order to keep the two systems synchronized. This single event holds information about community ship to phone.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMSTPH	CM Ship to Phones	RTE	Single	44H	D44H9901I	Included in container event RTHBCMOUT.

Conceptual Approach

If a Community is added or modified in a system, the information about ship to phone will be sent through an XPI integration to another third-party system for add or update.

RTHBCMSTEA

RTHBCMSTEA is used by EnterpriseOne Homebuilder to inform or notify third-party systems when a Community has been added or changed in order to keep the two systems synchronized. This single event holds information about community ship to email address.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCMSTEA	CM Ship to Electronic Address	RTE	Single	44H	D44H9901J	Included in container event .

Conceptual Approach

If a Community is added or modified in a system, the information about ship to email address will be sent through an XPI integration to another third-party system for add or update.

48 Project Management: Construction Lot Master

Construction Lot Master Overview

This chapter provides detailed information about the business interfaces that are available for the Construction Lot Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Construction Lot Master business object:</p> <ul style="list-style-type: none"> • RTHBLMOUT, which is a container event for these single events: <ul style="list-style-type: none"> ◦ RTHBLMHDR ◦ RTHBLMBPH ◦ RTHBLMBEA

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Construction Lot Master Real-Time Events

This section describes the available real-time events associated with the Construction Lot Master business object, which include:

- RTHBLMOUT, which is a container event for these single events:
 - RTHBLMHDR
 - RTHBLMBPH
 - RTHBLMBEA

RTHBLMOUT

RTHBLMOUT is used by JD Edwards EnterpriseOne to inform or notify third-party systems when Lot information or sales information has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTHBLMOUT	Home Builder Lot Master	RTE	Container Container event for: <ul style="list-style-type: none"> RTHBLMHDR RTHBLMBPH RTHBLMBEA 	44H

Conceptual Approach

Lot master and related information on a buyer (if one exists) is sent to a third party after the point in time that the lot is released for construction. Trigger points: at generation of lot work file; at lot master maintenance if work file has already been generated; at sales master maintenance if plan/elevation changes.

Objects That Can Initiate the Event

These objects can initiate the container event, and the events contained within it:

- N44H0083 – Home Builder Activity Rule Interface
- P44H201 – Lot Master
- P44H500 – Sales Management
- R44H700 – Lot Start Workfile Generation

RTHBLMHDR

RTHBLMHDR is used by JD Edwards Enterprise One to inform or notify third-party systems when Lot information or sales information has been added or changed in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBLMHDR	HB Lot Master Buyer Header	RTE	Single	44H	D44H9903D	Included in container event RTHBLMOUT.

Conceptual Approach

Lot master and related information on a buyer (if one exists) is sent to a third party after the point in time that the lot is released for construction. Trigger points: at generation of lot work file; at lot master maintenance if work file has already been generated; at sales master maintenance if plan/elevation changes.

RTHBLMBPH

RTHBLMBPH is used by EnterpriseOne Homebuilder to inform or notify third-party systems when Lot information or sales information has been added or changed in order to keep the two systems synchronized. This single event holds information about lot phone number.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBLMBPH	HB Lot Master Buyer Phones	RTE	Single	44H	D44H9903E	Included in container event RTHBLMOUT.

Conceptual Approach

Lot master and related information on a buyer (if one exists) is sent to a third party after the point in time that the lot is released for construction. Trigger points occur at generation of lot work file; at lot master maintenance if work file has already been generated; at sales master maintenance if plan/elevation changes.

RTHBLMBEA

RTHBLMBEA is used by EnterpriseOne Homebuilder to inform or notify third-party systems when Lot information or sales information has been added or changed in order to keep the two systems synchronized. This single event holds information about lot email address.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBLMBEA	HB Lot Master Buyer Address	RTE	Single	44H	D44H9903F	Included in container event RTHBLMOUT.

Conceptual Approach

Lot master and related information on a buyer (if one exists) is sent to a third party after the point in time that the lot is released for construction. Trigger points occur at generation of lot work file; at lot master maintenance if work file has already been generated; at sales master maintenance if plan/elevation changes.

49 Project Management: Project

Project Overview

This chapter provides detailed information about the business interfaces that are available for the Project business object.

This table lists all of the available interface components:

Business Service	Description
ProjectManager (JP510010)	The ProjectManager published business service (JP510010) manages the processing of project-related Web service operations.
processProject (J5100010)	<p>The processProject Web service operation is an inbound transaction operation that processes adds and updates of job cost data in JD Edwards EnterpriseOne.</p> <p>The processProject operation is called from the ProjectManager published business service. The net change utility is called for each project that is being updated. The processProject operation calls the CreateProjectHeader business function (B5100120) for adds and updates to the Extended Job Cost Master record (F5108) and the Business Unit Master record (F0006). You can specify a model business unit in the business service properties of this operation. When a new project is added, the system copies the model business unit that is specified in the properties.</p>
processWBS (J5100030)	<p>The processWBS Web service operation is an inbound transaction operation that processes adds and updates of WBS/task data in JD Edwards EnterpriseOne.</p> <p>The processWBS operation is called from the processProject operation. The processWBS operation calls the CreateProjectAccount business function (B5100410) to add and UpdateProjectAccount business function (B5100140) to update the Account Master record (F0901).</p> <p>After processing the F0901 records, if there are no errors and financial forecast records exist in the value object, the processWBS operation calls the processFinancialForecast operation.</p>
processFinancialForecast (J5100020)	<p>The processFinancialForecast Web service operation is an inbound transaction operation that processes adds and updates of financial forecast data in JD Edwards EnterpriseOne.</p> <p>The processFinancialForecast operation is called from the processWBS operation. The processFinancialForecast operation calls the CreateProjectFinancialForecasts business function (B5100160) to add, and the UpdateProjectFinancialForecasts business function (B5100160) when updating financial forecast data.</p>
BatchTimeSheetManager (JP05001)	The BatchTimeSheetManager published business service (JP05001) manages the processing of the BatchTimeSheetInsertProcessor Web service operation (J0500010).
BatchTimeSheetInsertProcessor (J0500010)	<p>The BatchTimeSheetInsertProcessor Web service operation is a database insert operation that processes inserts into the Employee Transactions Batch File (F06116Z1) in JD Edwards EnterpriseOne.</p> <p>The BatchTimeSheetInsertProcessor operation is called from the BatchTimeSheetManager published business service. The operation assigns a unique identifier using next number processing to each record before inserting it into the table. Any errors that are encountered are returned to you in an EIMessageList.</p>

Business Service	Description

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

50 Project Management: Purchase Order - Homebuilder Extensions

Purchase Order - Homebuilder Extensions Overview

This chapter provides detailed information about the business interfaces that are available for the Purchase Order - Homebuilder Extensions business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	This list includes the real-time events for the Purchase Order - Homebuilder business object: <ul style="list-style-type: none"> RTHBCOMMIT

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Purchase Order - Homebuilder Extensions Real-Time Events

This section describes the available real-time events associated with the Purchase Order - Homebuilder Extensions business object, which include:

- RTHBCOMMIT

RTHBCOMMIT

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTHBCOMMIT	HB Encumbrance Commitment	RTE	Single	44H	D44H9904D	Included in container event .

Conceptual Approach

As commitments are created in the J D Edwards Enterprise One, notification of those commitments is sent to the third party system where the system treats them as tasks that must be completed. For normal commitments (a vendor is specified), the Homebuilder management system first sends out commitments when they are posted by R44H7113 to the procurement system. Commitments for TBD vendors are sent to the third party system when they are generated to the lot start work file (F44H711) by R44H700.

Objects That Can Initiate the Event

These objects can initiate the event:

- N44H0088 - F44H73W1 - Create HomeBuilder
- N44H0113 – Lot Start Workfile Status Code
- R44H700 – Lot Start Workfile Generation

51 REM: Lease (Release 9.2 Update)

Lease Overview

This chapter provides detailed information about the business interfaces that are available for the Lease business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Lease business object:</p> <ul style="list-style-type: none"> • Inbound Bill Code Processor (R1512Z1I) • Inbound Unit Master Processor (R1507Z1I) • Inbound Area Master Processor (R1514Z1I) • Inbound Lease Master Processor (R1501BZ1I) • Inbound Lease Detail Processor (R15017Z1I) • Inbound Recurring Billing Processor (R1502BZ1I) • Inbound Manual Billing Processor (R1511BZ1I)

Lease Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Lease business object:

Program and Program Type	Description	Available Documentation
<p>These are temporary table batch import programs:</p> <ul style="list-style-type: none"> • Inbound Bill Code Processor (R1512Z1I) • Inbound Unit Master Processor (R1507Z1I) • Inbound Area Master Processor (R1514Z1I) • Inbound Lease Master Processor (R1501BZ1I) 	<p>All of these batch import programs are used to upload data from temporary Z tables into the JD Edwards EnterpriseOne Real Estate Management system.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Importing Real Estate Management Data (Release 9.2 Update)

Program and Program Type	Description	Available Documentation
<ul style="list-style-type: none">Inbound Lease Detail Processor (R15017Z1)Inbound Recurring Billing Processor (R1502BZ1)Inbound Manual Billing Processor (R1511BZ1)		

52 REM: Tenant Work Order

Tenant Work Order Overview

This chapter provides detailed information about the business interfaces that are available for the Tenant Work Order business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>These are the published business services available for the Tenant Work Order business object:</p> <ul style="list-style-type: none">• Tenant Work Order (JP150000), which manages these business services:<ul style="list-style-type: none">◦ Work Order Processor (J4800010)◦ Work Order Query Processor (J4800020)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Tenant Work Order Business Services - Tenant Work Order (JP150000)

This published business service processes tenant work order information using these business services:

- Work Order Processor (J4800010)
- Work Order Query Processor (J4800020)

For detailed information about these services, see [Work Order Processor \(J4800010\)](#) and [Work Order Query Processor \(J4800020\)](#).

53 SCM and Manufacturing: Demand

Demand Overview

This chapter provides detailed information about the business interfaces that are available for the Demand business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Demand business object:</p> <ul style="list-style-type: none"> • EDI Inbound Demand Edit/Update Report (R47171)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Demand Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Demand business object:

Program and Program Type	Description	Available Documentation
<p>EDI Inbound Demand Edit/Update Report (R47171)</p> <p>This is an EDI inbound program.</p>	<p>You run the EDI Inbound Demand Edit/Update batch program to update the demand and cumulative tables with EDI information. This process is useful for comparing a supplier and customer's cumulative quantities.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Running the EDI Inbound Demand Edit/Update Report (R47171)</i>

54 SCM and Manufacturing: Engineering Change Orders

Engineering Change Orders Overview

This chapter provides detailed information about the business interfaces that are available for the Engineering Change Orders business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	The EngineeringChangeOrderManager published business service (JP300000) was created as part of the integration between JD Edwards EnterpriseOne and Agile Product Lifecycle Management. See <i>Business Services for Pre-Built Integrations</i> .

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

55 SCM and Manufacturing: Forecast

Forecast Overview

This chapter provides detailed information about the business interfaces that are available for the Forecast business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Forecast business object:</p> <ul style="list-style-type: none"> Forecast Inbound Processor (R3460Z11)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Forecast Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Forecast business object:

Program and Program Type	Description	Available Documentation
<p>Forecast Inbound Processor (R3460Z11)</p> <p>This is an interoperability program.</p>	<p>Running a transaction process, such as Forecast Inbound Processor (R3460Z11), often identifies one or more inbound transactions that contain invalid transactions. The Forecast Inbound Processor program identifies the invalid transaction and sends an error message to the Work Center (P012501). The error message indicates the transaction number for the transaction in error.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> Understanding Reviewing and Revising Interoperability Transactions for Forecast Management

56 SCM and Manufacturing: Inventory Cycle Counts

Inventory Cycle Counts Overview

This chapter provides detailed information about the business interfaces that are available for the Inventory Cycle Counts business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Inventory Cycle Count business objects:</p> <ul style="list-style-type: none"> • Cycle Count Inbound Transaction Process program (R4141Z1I) • Cycle Count Update program (R41413)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Inventory Cycle Counts Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Inventory Cycle Count business objects:

Program and Program Type	Description	Available Documentation
<p>Cycle Count Inbound Transaction Process program (R4141Z1I)</p> <p>This is an interoperability program.</p>	<p>You run the Cycle Count Inbound Transaction Process program (R4141Z1I) to copy information from the F4141Z1 table to the F4141 table.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Cycle Counts from Inbound Transactions
<p>Cycle Count Update program (R41413)</p>	<p>After the cycle count has been entered, reviewed, and approved, run the Cycle</p>	<p>See this topic:</p>

Program and Program Type	Description	Available Documentation
	Count Update program (R41413) to update ledgers and balances. This program records variances to the F4111, F0911, and F41021 tables.	<ul style="list-style-type: none"><li data-bbox="781 281 1047 338">• <i>Running the Cycle Count Update Program</i>

57 SCM and Manufacturing: Inventory Location Master

Inventory Location Master Overview

This chapter provides detailed information about the business interfaces that are available for the Inventory Location Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Inventory Location Master business object:</p> <ul style="list-style-type: none"> • RTIBOUT, which is a container event for RTIBOUTA. • RTIBOUT2, which is a container event for RTIBOUTB.
Batch Import Programs	<p>This list includes the batch import programs for the Inventory Location Master business object:</p> <ul style="list-style-type: none"> • Item Location Extraction (R41021ZX)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Inventory Location Master Real-Time Events

This section describes the available real-time events associated with the Item Master business object, which include:

- RTIBOUT, which is a container event for RTIBOUTA.
- RTIBOUT2, which is a container event for RTIBOUTB.

RTIBOUT

RTIBOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems when an item location has been added, changed, or deleted in order to keep the two systems synchronized. Adding an Item Balance record that does not have quantity information or deleting an Item Balance record will not be published.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTIBOUT	Item Balance by Location	RTE	Container Container event for RTIBOUTA (D4101660A).	H41	D4101660A: Item Balance Real Time Event Notification Template

Conceptual Approach

An inventory location that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Item Location (F41021) table. When an item balance record is modified, an RTIBOUT real-time event notification is published.

Currently, this real-time event is used only by the APS integration.

Objects that Can Initiate the Event

These objects can initiate the RTIBOUT event:

- XT4111Z1 - Inventory Transactions
- XF41021 - Update F41021 Locations
- B4100430 - Update Misc F41021 Fields
- B4101370 - F41021 Update Quantities
- N4202100 - F41021 Commitments Processing
- B4100100 – Location Status Update
- B4101810 – Process F41021WF Records
- XF41021P – Update F41021 Locationsfor OT Orders
- P41024 – Location Revisions
- P4108 - Lot Master
- R41083 – Update Effective Lots

RTIBOUTA

RTIBOUTA is used by the JD Edwards EnterpriseOne system to notify third-party systems when an item location has been added, changed, or deleted in order to keep the two systems synchronized. Adding an Item Balance record that does not have quantity information or deleting an Item Balance record will not be published.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTIBOUTA	Item Balance by Loc Details	RTE	Single	H41	D4101660A: Item Balance Real Time Event Notification Template	Included in container event RTIBOUT

Conceptual Approach

An inventory location that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Item Location (F41021) table. When an item balance record is modified, an RTIBOUT real-time event notification is published. The RTIBOUT real-time event notification contains the RTIBOUTA event.

Currently, this real-time event is used only by the APS integration.

RTIBOUT2

RTIBOUT2 is used by the JD Edwards EnterpriseOne system to notify third-party systems when an item location has been added, changed, or deleted in order to keep the two systems synchronized. Adding an Item Balance record that does not have quantity information or deleting an Item Balance record will not be published.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTIBOUT2	Item Balance by Loc Version 2	RTE	Container Container event for RTIBOUTB (D4101661A).	H41	D4101661A: Item Balance Real Time Event Notification Template

Conceptual Approach

An inventory location that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Item Location (F41021) table. When an item balance record is modified, an RTIBOUT2 real-time event notification is published.

Currently, this real-time event is used only by the APS integration.

Objects that Can Initiate the Event

These objects can initiate the RTIBOUT2 event, and the single event contained within the event:

- XT4111Z1 - Inventory Transactions
- XF41021 - Update F41021 Locations
- B4100430 - Update Misc F41021 Fields
- B4101370 - F41021 Update Quantities
- N4202100 - F41021 Commitments Processing
- B4100100 – Location Status Update
- B4101810 – Process F41021WF Records
- XF41021P – Update F41021 Locationsfor OT Orders
- P41024 – Location Revisions
- P4108 - Lot Master
- R41083 – Update Effective Lots

RTIBOUTB

RTIBOUTB is used by the JD Edwards EnterpriseOne system to notify third-party systems when an item location has been added, changed, or deleted in order to keep the two systems synchronized. Adding an Item Balance record that does not have quantity information or deleting an Item Balance record will not be published.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTIBOUTB	Item Bal by Loc Dtl Version2	RTE	Single	H41	D4101661A: Item Balance Real Time Event Notification Template	Included in container event RTIBOUT

Conceptual Approach

An inventory location that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Item Location (F41021) table. When an item balance record is modified, an RTIBOUT2 real-time event notification is published. The RTIBOUT2 real-time event notification contains the RTIBOUTB event.

Currently, this real-time event is used only by the APS integration.

Inventory Location Master Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Inventory Location Mastger business object:

Program and Program Type	Description	Available Documentation
Item Location Extraction (R41021ZX) This is a temporary table extraction program.	You run this outbound extraction program to retrieve data from the outbound transaction tables and create a flat file if one does not exist, or to append to an existing flat file. Every field is written from the EDI interface tables to the flat file.	See this topic: <ul style="list-style-type: none"> Understanding Outbound Interoperability for Inventory Management

58 SCM and Manufacturing: Inventory Tag Counts

Inventory Tag Counts Overview

This chapter provides detailed information about the business interfaces that are available for the Inventory Tag Counts business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Inventory Tag Count business objects:</p> <ul style="list-style-type: none"> Cycle Count Inbound Transaction Process program (R4141Z1I) <p>For information about this program, see <i>Inventory Cycle Counts Batch Import Programs</i>.</p> <ul style="list-style-type: none"> Tag Inventory Update (R41610)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Inventory Tag Counts Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Inventory Tag Count business objects:

Program and Program Type	Description	Available Documentation
Tag Inventory Update (R41610)	<p>After you have entered the tag count results and reviewed variances, run the Tag Inventory Update program (R41610) to:</p> <ul style="list-style-type: none"> Compare the online on-hand count to the physical count. 	<p>See this topic:</p> <ul style="list-style-type: none"> <i>Running Tag Count Updates</i>

Program and Program Type	Description	Available Documentation
	<ul style="list-style-type: none">• Calculate the quantity and amount of variances.• Update the new quantity information in the item location record and item ledger.• Create entries to the general ledger based on AAls.	

59 SCM and Manufacturing: Inventory Transactions

Inventory Transactions Overview

This chapter provides detailed information about the business interfaces that are available for the Inventory Transactions business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	This list includes the batch import programs for the Inventory Transactions business object: <ul style="list-style-type: none"> • EDI Product Activity Data Edit/Create program (R47121)
Batch Export Programs	This list includes the batch export programs for the Inventory Transactions business object: <ul style="list-style-type: none"> • EDI Product Activity Data Extraction program (R47122)

Inventory Transactions Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Inventory Transactions business object:

Program and Program Type	Description	Available Documentation
EDI Product Activity Data Edit/Create program (R47121) This is an EDI program.	Run the EDI Product Activity Data Edit/Create program (R47121) from the Product Activity Data menu to report inventory activity and to generate replacement orders for distribution centers, warehouses, or retail outlets.	See this topic: <ul style="list-style-type: none"> • Understanding Inbound Product Activity Data Transactions

Inventory Transactions Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Item Branch Master business object:

Program and Program Type	Description	Available Documentation
EDI Product Activity Data Extraction program (R47122) This is an EDI extract program.	Run the EDI Product Activity Data Extraction program (R47122) from the Product Activity Data menu to generate product activity data transactions (852/INVRPT) to send to a central distribution center, warehouse, or the parent company.	See this topic: <ul style="list-style-type: none"><li data-bbox="781 726 1057 800">• <i>Understanding Outbound Product Activity Data Transactions</i>

60 SCM and Manufacturing: Item Branch Master

Item Branch Master Overview

This chapter provides detailed information about the business interfaces that are available for the Item Branch Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The InventoryManager web service (JP410000) manages the processing of these inventory-related web service operations:</p> <ul style="list-style-type: none"> • processSupplierCatalogPrice (J4100002) • processSupplierCatalogPriceV2 (J4100002) • processInventoryItem (J4100003) • processInventoryItemV2 (J4100003) • getBranchPlantItem (J4100004) • getBranchPlantItemV2 (J4100004) • getSupplierCatalogPrice (J4100005) • getSupplierCatalogPriceV2 (J4100005) • getItemAvailability (J4100001) • insertInventoryItemStaging (J4100006)
Batch Import Programs	<p>This list includes the batch import programs for the Item Branch Master business object:</p> <ul style="list-style-type: none"> • Item Master Inbound Transaction Process (R4101Z11) <p>For information on this program, see <i>Item Master Batch Import Programs</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Item Branch Master Business Services - InventoryManager

The InventoryManager web service (JP410000) manages the processing of inventory-related web service operations. This table includes a description of the inventory web service operations:

Operation	Description
processSupplierCatalogPrice (J4100002)	Use this operation to add and change supplier catalog prices within the JD Edwards EnterpriseOne Inventory Management system.
processSupplierCatalogPriceV2 (J4100002)	Use this operation to add and update supplier catalog prices, including global locator number (GLN) within the JD Edwards EnterpriseOne Inventory Management system.
processInventoryItem (J4100003)	Use this operation to add and change inventory item master records within the JD Edwards EnterpriseOne Inventory Management system.
processInventoryItemV2 (J4100003)	Use this operation to add and change inventory item master records, including the country of origin required indicator, within the JD Edwards EnterpriseOne Inventory Management system.
getBranchPlantItem (J4100004)	Use this operation to query the JD Edwards EnterpriseOne database for existing item branch/plant records.
getBranchPlantItemV2 (J4100004)	Use this operation to query the JD Edwards EnterpriseOne database for existing item branch/plant records including the country of origin required indicator.
getSupplierCatalogPrice (J4100005)	Use this operation to query the JD Edwards EnterpriseOne database for existing supplier catalog prices.
getSupplierCatalogPriceV2 (J4100005)	Use this operation to query the JD Edwards EnterpriseOne database for existing supplier catalog prices along with GLN.
getItemAvailability (J4100001)	Use this operation to query the JD Edwards EnterpriseOne database for existing item availability.
insertInventoryItemStaging (J4100006)	Use this operation to insert inventory item staging records within the JD Edwards EnterpriseOne database.
insertInventoryItemStagingV2 (J4100006)	Use this operation to insert inventory item staging records, including country of origin required indicator, within the JD Edwards EnterpriseOne database.
Additional Methods for Mobile Sales Functionality	Additional methods were created for use specifically with the Mobile Sales features. For additional information about those methods, see <i>Mobile Smartphone Applications</i> .

Accessing Javadoc for the Inventory Manager Web Service Operations

To access Javadoc for the Inventory Manager web service and its related operations, review these Javadoc packages:

- JP410000 (InventoryManager)
- J4100002 (processSupplierCatalogPrice)
- J4100002 (processSupplierCatalogPriceV2)
- J4100003 (processInventoryItem)
- J4100003 (processInventoryItemV2)
- J4100004 (getBranchPlantItem)
- J4100004 (getBranchPlantItemV2)
- J4100005 (getSupplierCatalogPrice)
- J4100005 (getSupplierCatalogPriceV2)
- J4100001 (getItemAvailability)
- J4100006 (insertInventoryItemStaging)
- J4100006 (insertInventoryItemStagingV2)

Prerequisite

Before using the Inventory Manager web service, or any of the related web service operations, you must install and configure the JD Edwards EnterpriseOne Inventory Management system.

See [JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide](#) .

processSupplierCatalogPrice

The processSupplierCatalogPrice web service operation is an inbound transaction operation that enables consumers to process supplier catalog price information within the JD Edwards EnterpriseOne system. The consumer can add and change supplier catalog price records in the JD Edwards EnterpriseOne Inventory Management system. The system updates the Supplier Price/Catalog File table (F41061).

If the operation is successful, the system returns a confirmation message to the consumer which includes supplier catalog data.

If the operation fails, the system returns an error message to the consumer. When an error is encountered during processing of the processSupplierCatalogPrice web service operation, any changes to the F41061 that occurred as a result of the web service operation being run are rolled back.

The processSupplierCatalogPrice web service operation verifies that values are provided for both the start and end effective dates. If no value is provided for the start date then today's date is assigned to the start date. If no value is provided for the end date then December 31st of the current year is assigned to the end date. The processSupplierCatalogPrice web service operation retrieves the currency code from the specified supplier or, if the supplier does not exist, the specified business unit's company. If a currency code is specified, the currency code lookup is bypassed. The processSupplierCatalogPrice web service operation retrieves the cost level for a specified item and the

appropriate unit of measure (purchasing or primary) from the Item Master table (F4101) as specified in the distribution constants. If a unit of measure is specified, the unit of measure lookup is bypassed.

The table illustrates the supported action codes for the processSupplierCatalogPrice web service operation:

Action Type	Action Code
Add	1, A, I
Update	2, U, C

Supported Functionality

This section discusses the functionality that the processSupplierCatalogPrice operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The operation supports adding new supplier catalog prices and updating existing supplier catalog prices.

This web service operation does not support deleting supplier catalog prices. The operation also does not support adding or modifying supplier catalog prices for items with a cost level of **3**.

Implementation Details

The following table includes information that can help determine whether the processSupplierCatalogPrice operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • supplier • item • currencyCode • unitOfMeasureCode • dateEffectiveStart • dateEffectiveEnd <p>The web service operation always returns an exception in the response if the operation fails to complete successfully.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. If the supplier catalog price action encounters errors, the transaction stops processing and the system rolls back the information in the tables. No manual update is necessary.</p>

Question	Answer
Does this operation use record reservation?	No. This operation does not reserve records.

processSupplierCatalogPriceV2

Review the information in the processSupplierCatalog section of this chapter before using the processSupplierCatalogPriceV2 web service operation. All of the information in the processSupplierCatalogPrice section also applies to the processSupplierCatalogPriceV2 operation.

The processSupplierCatalogPriceV2 operation is a version of the processSupplierCatalogPrice web service operation. This web service operation is used to add and update supplier catalog prices and associated GLNs. This version executes the processSupplierCatalog (J4100002) web service operation.

processInventoryItem

The processInventoryItem web service operation is an inbound transaction operation that enables consumers to process inventory item information within the JD Edwards EnterpriseOne system. The consumer can add and change inventory item master records in the JD Edwards EnterpriseOne Inventory Management system.

Note: The data provided for item master insertion must be in the JD Edwards EnterpriseOne format as no formatting is performed before item master insertion in JD Edwards EnterpriseOne begins. Also, codes must be values in the JD Edwards EnterpriseOne system.

If the operation is successful, the system returns a confirmation message to the consumer. The messages includes inventory item data.

If the operation fails, the system returns an error message to the consumer.

Prerequisites

Before using the processInventoryItem operation, you must set the processing options for the Item Master program (P4101).

See *"Setting Processing Options for Item Master (P4101)" in the JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide*.

Supported Functionality

This section discusses the functionality that the processInventoryItem operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The consumer can add and change inventory item master records in the JD Edwards EnterpriseOne Inventory Management system.

The processInventoryItem web service operation does not support the cancellation of inventory items.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes inventory item information. This table includes information about the business service properties that the processInventoryItem web service operation uses:

Group	Key	Description	Default Value
J4100003	J4100003_ITEM_MBF_VERSION	Use this business service property to specify which version of the Item Master program (P4101) the operation uses. See <i>"Setting Processing Options for Item Master (P4101)" in the JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide</i> .	ZJDE0001
J4100003	J4100003_ITEM_STOCKING_TYPE_CODE	Use this business service property to specify the stocking type code that the operation uses for an add request if the code is not provided.	S

Implementation Details

The following table includes information that can help determine whether the processInventoryItem operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system. At a minimum, the system returns non-zero values in the itemId field.
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.

Question	Answer
Does this operation use record reservation?	This operation reserves the inventory record immediately when an update is requested by the consumer.

processInventoryItemV2

Review the information in the processInventoryItem section of this chapter before using the processInventoryItemV2 web service operation. All of the information in the processInventoryItem section also applies to the processInventoryItemV2 operation.

The processInventoryItemV2 operation is a version of the processInventoryItem web service operation. This web service operation is used to add, update, and delete inventory item master records along with country of origin required indicator. This version executes the processInventoryItem (J4100003) web service operation.

getBranchPlantItem

The getBranchPlantItem web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve existing branch/plant item records. The operation invokes a database operation to retrieve branch/plant item information from the F4101 and Item Branch File (F4102) tables in JD Edwards EnterpriseOne based on the selection criteria specified in the value object.

Note: The data provided for selection must be in the JD Edwards EnterpriseOne format. No formatting is performed before the query is made. Also, codes must be values in the JD Edward EnterpriseOne system.

If the operation is successful, the system returns zero to many records to the consumer. You can specify the maximum number of records to return during a query using the Max Rows business service property. If the operation fails, the system returns an error message to the consumer.

The getBranchPlantItem web service operation enables source systems to query JD Edwards EnterpriseOne branch/plant item information in a real-time fashion. The query contains fields that can be used to filter the branch/plant item information retrieved by JD Edwards EnterpriseOne. ShowBranchPlant is returned during a successful query.

Supported Functionality

This section discusses the functionality that the getBranchPlantItem operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The getBranchPlantItem web service operation enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve existing branch/plant item records.

The getBranchPlantItem operation does not support the wildcard (asterisk (*)) for search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes inventory item information. This table includes information about the business service properties that the getBranchPlantItem web service operation uses:

Group	Key	Description	Default Value
J4100004	J4100004_V4102XPI2_MAX_GRID_ROWS_RETURNED	Use this business service property to define the maximum number of rows that the operation returns when querying the JD Edwards EnterpriseOne database.	100

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database. The system returns an error if you perform a query with no selection criteria and this business service property is set to **0**.

Implementation Details

The following table includes information that can help determine whether the getBranchPlantItem operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> branchPlant supplier buyer itemId
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

getBranchPlantItemV2

Review the information in the `getBranchPlantItem` section of this chapter before using the `getBranchPlantItemV2` web service operation. All of the information in the `getBranchPlantItem` section also applies to the `getBranchPlantItemV2` operation.

The `getBranchPlantItemV2` operation is a version of the `getBranchPlantItem` web service operation. This web service operation is used to retrieve branch plant item records along with country of origin required indicator. This version executes the `getBranchPlantItem` (J4100004) web service operation.

getSupplierCatalogPrice

The `getSupplierCatalogPrice` web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve existing supplier catalog price information. The operation retrieves supplier catalog price information from fields in the Supplier Price/Catalog File table (F41061).

If the operation is successful, the system returns zero to many records to the consumer. You can specify the maximum number of records to return during a query using the Max Grid Rows Returned business service property. If the operation fails, the system returns an error message to the consumer.

Note: The data provided for selection must be in the JD Edwards EnterpriseOne format. No formatting is performed before the query is made.

Supported Functionality

This section discusses the functionality that the `getSupplierCatalogPrice` operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The `getSupplierCatalogPrice` web service operation enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve existing supplier catalog price information.

The `getSupplierCatalogPrice` operation does not support the wildcard (asterisk (*)) for search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes supplier catalog price information. This table includes information about the business service properties that the `getSupplierCatalogPrice` web service operation uses:

Group	Key	Description	Default Value
J4100005	J4100005_F41061_MAX_GRID_ROWS_RETURNED	Use this business service property to define the maximum number of rows that	100

Group	Key	Description	Default Value
		the operation returns when querying the JD Edward EnterpriseOne database.	

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database.

Implementation Details

The following table includes information that can help determine whether the `getSupplierCatalogPrice` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> entityIdSupplier itemId itemProduct itemCatalog currencyCode unitOfMeasureCode <p>The operation may complete successfully without returning rows because the selection criteria did not match any records in the database or an open query was performed and there were no records. This is considered successful and provides valid information.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

getSupplierCatalogPriceV2

Review the information in the getSupplierCatalogPrice section of this chapter before using the getSupplierCatalogV2 web service operation. All of the information in the getSupplierCatalogPrice section also applies to the getSupplierCatalogPriceV2 operation.

The getSupplierCatalogPriceV2 operation is a version of the getSupplierCatalogPrice web service operation. This web service operation is used to retrieve supplier catalog price records with associated GLNs. This version executes the getSupplierCatalogPrice (J4100005) web service operation.

getItemAvailability

The getItemAvailability web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve item availability information.

The getItemAvailability web service operation calls the InvRealTimeItemAvailability business function (B4101640) to fetch item availability. You must specify itemId, businessUnit, and unitOfMeasure as inputs to the query. If the operation is successful, the system returns item availability data. If the operation fails, the system returns an error message to the consumer.

Supported Functionality

This section discusses the functionality that the getItemAvailability operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The getItemAvailability web service operation enables consumers to query the JD Edwards EnterpriseOne Inventory Management system to retrieve item availability information.

The getItemAvailability operation does not support the wildcard (asterisk (*)) for search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes item availability information. This table includes information about the business service properties that the getItemAvailability web service operation uses:

Group	Key	Description	Default Value
J4100001	J4100001_ERROR_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages.	Get Item Availability Sent In:

Implementation Details

The following table includes information that can help determine whether the `getItemAvailability` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • <code>businessUnit</code> • <code>transactionUnitOfMeasure</code> • <code>quantityAvailable</code> • <code>itemId</code>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

insertInventoryItemStaging

The `insertInventoryItemStaging` web service operation is a database insert operation that enables consumers to insert one to many inventory item records into the staging table in the JD Edwards EnterpriseOne database.

The `insertInventoryItemStaging` web service operation uses auto commit transaction processing. Since inventory item records are autonomous, each inventory item staging record is immediately added to the F4101 Item Master Unedited Transaction Table (F4101Z1) and the entire group is not rolled back if an error occurs. If the operation is successful, the system inserts one to many records. You can specify the maximum number of records to insert using the `Max Rows` business service property.

If the operation fails, the system returns an error message to the consumer. The records that return an error will not be inserted into the database. However, records that do not error are committed to the database even if other records in the same transaction fail.

If all rows passed in were not inserted into the staging table, then the system returns an error message. The `Max Rows` business service property limits the number of inserted records. The consumer then sends the data in groups less than the `Max Rows` business service property or changes the `Max Rows` business service property to **0** (zero) so that all records are inserted.

Note: The data provided for insertion must be in the JD Edwards EnterpriseOne format. No formatting is performed before the insert is made.

Supported Functionality

This section discusses the functionality that the insertInventoryItemStaging operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The operation supports the insert of one to many inventory item records into the staging table in the JD Edwards EnterpriseOne database.

This web service operation does not support changing or deleting inventory item staging records.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes inventory item staging records. This table includes information about the business service properties that the insertInventoryItemStaging web service operation uses:

Group	Key	Description	Default Value
J4100006	J4100006_MAX_ROWS	Use this business service property to define the maximum number of rows that the operation inserts in the JD Edward EnterpriseOne database.	999
J4100006	J4100006_ITEM_STOCKING_TYPE_CODE	Use this business service property to specify the stocking type code that the operation uses for the insert if the code is not provided.	S

Note: The Max Rows business service property limits the number of inserted records. Either send the data in groups less than the Max Rows business service property or change the Max Rows service constant to **0** (zero). If you set this value to **0**, the system inserts all records.

Implementation Details

The following table includes information that can help determine whether the insertInventoryItemStaging operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns a confirmation message to the consumer that includes the number of rows that were inserted into the F4101Z1 table.
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>No, records that return an error are not inserted into the database. Records that do not return an error are committed to the database even if other records in the same transaction fail.</p> <p>If you encounter errors during processing, review your insertion criteria and business service property settings and try your insert again.</p>
Does this operation use record reservation?	No. This operation does not reserve records.

insertInventoryItemStagingV2

Review the information in the insertInventoryItemStaging section of this chapter before using the insertInventoryItemStagingV2 web service operation. All of the information in the insertInventoryItemStaging section also applies to the insertInventoryItemStagingV2 operation.

The insertInventoryItemStagingV2 operation is a version of the insertInventoryItemStaging web service operation. This web service operation is used to insert inventory item staging records, including the country of origin required indicator. This version executes the insertInventoryItemStaging (J4100006) web service operation.

61 SCM and Manufacturing: Item Cost

Item Cost Overview

This chapter provides detailed information about the business interfaces that are available for the Item Cost business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	This list includes the real-time events for the Item Cost business object: <ul style="list-style-type: none"> RTCOSTOUT, which is a container event for RTCOSTOUTA.
Batch Import Programs	This list includes the batch import programs for the Item Cost business object: <ul style="list-style-type: none"> Item Cost Inbound Transaction Process program (R4105Z1I)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Item Cost Real-Time Events

This section describes the available real-time events associated with the Item Cost business object, which include:

- RTCOSTOUT, which is a container event for RTCOSTOUTA.

RTCOSTOUT

RTCOSTOUT is a container event generated by the JD Edwards EnterpriseOne system to notify third-party systems of changes to the unit cost of an item. Any adds or changes to a unit cost are generated as a change real-time event. Deletes are not supported.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTCOSTOUT	Item Cost Master	RTE	Container Contains the RTCOSTOUTA (D4102180B).	41

Conceptual Approach

A unit cost is added or changed in the item cost table (F4105). The JD Edwards EnterpriseOne system sends out item cost real-time events with unit cost information.

Objects that Can Initiate the Event

The P4105 - Work with Item Cost program can initiate the RTCOSTOUT event.

RTCOSTOUTA

RTCOSTOUTA is a single event that is generated by the JD Edwards EnterpriseOne system to provide item information about changes to the unit cost of an item. Any adds or changes to a unit cost are generated as a change real-time event. Deletes are not supported.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTCOSTOUTA	Item Cost Details	RTE	Single	H41	D4102180B: Item Cost Real Time Wrapper	Included in container event RTCOSTOUT.

Conceptual Approach

A unit cost is added or changed in the item cost table (F4105). The JD Edwards EnterpriseOne system sends out item cost real-time events with unit cost information.

Item Cost Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Item Cost business object:

Program and Program Type	Description	Available Documentation
Item Cost Inbound Transaction Process program (R4105Z11) This is an interoperability program.	During processing, the Item Cost Inbound Transaction Process program (R4105Z11) adds, changes, or deletes multiple records that exist within the Unedited Transaction Table - Item Cost table (F4105Z1) directly to the F4105 table. Additionally, the costing selection methods for sales, inventory, and purchasing can be added or changed during processing.	See this topic: <ul style="list-style-type: none"> Item Costs from Inbound Transactions

62 SCM and Manufacturing: Item Master

Item Master Overview

This chapter provides detailed information about the business interfaces that are available for the Item Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The Item Master business object uses these business service methods, which are documented under the InventoryManager web service (JP410000) documentation:</p> <ul style="list-style-type: none"> • getCalculatedAvailability • getItem Availability • getItemSearch • process InventoryItem • process InventoryItemV2 • Insert InventoryItemStaging • Insert InventoryItemStagingV2 <p>See <i>Item Branch Master Business Services - InventoryManager</i>.</p>
Real-Time Events	<p>This list includes the real-time events for the Item Master business object:</p> <ul style="list-style-type: none"> • RTIMOUT, which is a container event for RTIMOUTA. • RTIMOUT2, which is a container event for RTIMOUTB.
Batch Import Programs	<p>This list includes the batch import programs for the Item Master business object:</p> <ul style="list-style-type: none"> • Item Master Inbound Transaction Process (R4101Z1I)
Batch Export Programs	<p>This list includes the batch export programs for the Item Master business object:</p> <ul style="list-style-type: none"> • Item Batch Outbound (R4101ZO)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Item Master Real-Time Events

This section describes the available real-time events associated with the Item Master business object, which include:

- RTIMOUT, which is a container event for RTIMOUTA.
- RTIMOUT2, which is a container event for RTIMOUTB.

RTIMOUT

RTIMOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems of item additions, changes, or deletions in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTIMOUT	Item Master	RTE	Container Container event for RTIMOUTA (D4102150B).	H41

Conceptual Approach

A new item master or item branch added in the JD Edwards EnterpriseOne system creates a new record in the Item Master (F4101) table or the Item Branch (F4102) table. A real-time notification event is triggered in the maintenance application after item master or item branch information is added to either table.

Objects that Can Initiate the Event

These objects can initiate the RTIMOUT event, and the single event contained within the event:

- B4102151 - Item Master Real Time Event Notify 2
- N4101070 - Update Item Process Data
- N4101130 - Workflow Update Approved Item Master Change
- P4101N - Non Stock Item Master
- R4101P - Item Master Purge (F4101)
- P4690 - Warehouse Speed Group Maintenance
- R4690 - Update Process and Item Dimens
- BCW75 - Style Item Purge

RTIMOUTA

RTIMOUTA is used by the JD Edwards EnterpriseOne system to notify third-party systems of item additions, changes, or deletions in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTIMOUTA	Item Master Details	RTE	Single	H41	D4102150B: Item Master Real Time Event Notify Load Data	Included in container event RTIMOUT.

Conceptual Approach

A new item master or item branch added in the JD Edwards EnterpriseOne system creates a new record in the Item Master (F4101) table or the Item Branch (F4102) table. A real-time notification event is triggered in the maintenance application after item master or item branch information is added to either table.

RTIMOUT2

The JD Edwards EnterpriseOne system uses RTIMOUT2 to notify third-party systems of item additions, changes, or deletions in order to keep the two systems synchronized.

Oracle designed this version to support information related to country of origin and global location number.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTIMOUT2	Item Master Version 2	RTE	Container Container event for RTIMOUTB (D4102152B).	H41

Conceptual Approach

A new item master or item branch added in the JD Edwards EnterpriseOne system creates a new record in the Item Master (F4101) table or the Item Branch (F4102) table. A real-time notification event is triggered in the maintenance application after item master or item branch information is added to either table.

Objects that Can Initiate the Event

These objects can initiate the RTIMOUT2 event, and the single event contained within the event:

- N4101070 - Update Item Process Data
- P4101N - Non Stock Item Master

RTIMBOUTB

RTIMOUTB is used by the JD Edwards EnterpriseOne system to notify third-party systems of item additions, changes, or deletions in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTIMOUTB	Item Master Details Version 2	RTE	Single	H41	D4102152B: Item Master Real Time Event Notify Load Data	Included in container event RTIMOUT.

Conceptual Approach

A new item master or item branch added in the JD Edwards EnterpriseOne system creates a new record in the Item Master (F4101) table or the Item Branch (F4102) table. A real-time notification event is triggered in the maintenance application after item master or item branch information is added to either table.

Item Master Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Item Master business object:

Program and Program Type	Description	Available Documentation
Item Master Inbound Transaction Process (R4101Z1I) This is a temporary table import program.	You run the Item Master Inbound Transaction Process program (R4101Z1I) to copy the information from the F4101Z1 and F4101Z1A tables to the related application tables.	See this topic: <ul style="list-style-type: none"> <i>Processing Inbound Interoperability for Inventory Management</i>

Item Master Batch Export Programs

You use batch export programs to extract data from the JD Edwards EnterpriseOne system so that it can be consumed by an external software system.

This table lists and describes the available batch export programs that are associated with the Item Master business object:

Program and Program Type	Description	Available Documentation
Item Batch Outbound (R4101ZO)	You run the Item Batch Outbound program to create an XML output file	Access this program from menu G41313.

Program and Program Type	Description	Available Documentation
This is an outbound batch program.	that contains information from the Item Master table (F4101) for specified items.	

63 SCM and Manufacturing: Kanban

Kanban Overview

This chapter provides detailed information about the business interfaces that are available for the Kanban business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Kanban business object:</p> <ul style="list-style-type: none"> Inbound Kanban Transaction program (R30161Z11)
Orchestration	<p>This list includes the pre-defined orchestrations available for the Kanban business object:</p> <ul style="list-style-type: none"> Kanban Check In Kanban Check Out <p>For additional information, see <i>Kanban Check In and Check Out</i>.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Kanban Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Kanban business object:

Program and Program Type	Description	Available Documentation
Inbound Kanban Transaction program (R30161Z11)	You run the Inbound Kanban Transaction program (R30161Z11) to copy the information from the F30161Z1 table to the related application tables.	<p>See this topic:</p> <ul style="list-style-type: none"> <i>Kanban Inbound Transactions</i>

64 SCM and Manufacturing: Shop Floor

Shop Floor Overview

This chapter provides detailed information about the business interfaces that are available for the Shop Floor business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Shop Floor business object:</p> <ul style="list-style-type: none"> • OPBOMROUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ OPBOMRHDR ○ OPAPSRTG ○ OPAPSRLR

Shop Floor Real-Time Events

This section describes the available real-time events associated with the Shop Floor business unit, which include:

- OPBOMROUT, which is a container event for these single events:
 - OPBOMRHDR
 - OPAPSRTG
 - OPAPSRLR

OPBOMROUT

OPBOMROUT is used by the JD Edwards EnterpriseOne system to inform third-party systems when a work order parts list and routing are added, changed, or deleted.

Event Name	Event Description	Event Type	Event Category	Product Code
OPBOMROUT	Configured Item Trans BOMR	RTE	Container Contains these data structures:	34A

Event Name	Event Description	Event Type	Event Category	Product Code
			<ul style="list-style-type: none"> D34A1140A: Transactional RLR Header Event Notification Canonical D34A1140B: Transactional RLR Routing Event Notification Canonical D34A1140C: Transactional RLR Resource List Event Notification Canonical 	

Conceptual Approach

A work order parts list or routing that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Work Order Parts List (F3111) and the Work Order routing (F3112) tables. When a work order parts list or routing record is modified, an OPBOMROUT real-time event notification is published.

Objects that Can Initiate the Event

These objects can initiate the OPBOMROUT event, and the single events contained within the event:

- B3101250 – F3112 Work Order Routings MBF
- B3101260 - F3111 Parts List MBF
- R31410 – Order Processing
- B3101250 – F3112 Work Order Routings MBF
- B34A2090 - Publish Transactional Work Order
- B34A2070 – Get APS Integration Constants For WO

OPBOMRHDR

OPBOMRHDR is used by the JD Edwards EnterpriseOne system to inform third-party systems when a work order is added, changed, or deleted in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
OPBOMRHDR	Configured Item Header	RTE	Single	34A	D34A1140A: Transactional RLR Header Event Notification Canonical	Included in container event OPBOMROUT.

Conceptual Approach

When a work order is added, modified or deleted in the JD Edwards EnterpriseOne system, and records are created, modified or deleted in the Work Order Master File (F4801) and Work Order Master Tag File (F4801T), this event is used.

OPAPSRTG

OPAPSRTG is used by the JD Edwards EnterpriseOne system to inform third-party systems when a routing is added, changed, or deleted in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
OPAPSRTG	Configured Item Routing	RTE	Single	34A	D34A1140B: Transactional RLR Routing Event Notification Canonical	Included in container event OPBOMROUT.

Conceptual Approach

When a work order routing is added, modified or deleted into JD Edwards EnterpriseOne system, creating, modifying or deleting a record in the Work Order Routing (F3112) table, this event is used.

OPAPSRLR

OPAPSRLR is used by the JD Edwards EnterpriseOne system to inform third-party systems when a work order parts list is added, changed, or deleted in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
OPAPSRLR	Configured Item Parts/ Resource	RTE	Single	34A	D34A1140C: Transactional RLR Resource List Event Notification Canonical	Included in container event OPBOMROUT.

Conceptual Approach

When a work order parts list is added, modified or deleted in the JD Edwards EnterpriseOne system, creating, modifying or deleting a record in the Work Order Parts List table (F3111), this event is used.

65 SCM and Manufacturing: Planning Schedule

Planning Schedule Overview

This chapter provides detailed information about the business interfaces that are available for the Planning Schedule business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	This list includes the batch import programs for the Planning Schedule business object: <ul style="list-style-type: none"> • EDI Planning Schedule Edit/Create program (R47061)
Batch Export Programs	This list includes the batch export programs for the Planning Schedule business object: <ul style="list-style-type: none"> • Outbound Extraction - Forecast program (R47062) • Outbound Extraction - Supplier program (R470621) • Outbound Extraction - Ad Hoc Supplier Release Schedules program (R470622) • EDI Planning Schedule Extraction Conversion program (R47062C)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Planning Schedule Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Planning Schedule business object:

Program and Program Type	Description	Available Documentation
EDI Planning Schedule Edit/Create program (R47061)	Run the EDI Planning Schedule Edit/Create program (R47061) from the Planning Schedule - Forecast menu to receive planning schedule forecast	See this topic:

Program and Program Type	Description	Available Documentation
This is an inbound EDI program.	transactions. Planning schedule forecast transactions are processed and copied into the F3460 table.	<ul style="list-style-type: none"> Understanding Inbound Planning Schedule Forecasts

Planning Schedule Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Planning Schedule business object:

Program and Program Type	Description	Available Documentation
Outbound Extraction - Forecast program (R47062) This is an EDI outbound program.	Run the Outbound Extraction - Forecast program (R47062) from the Planning Schedule Forecast menu to generate planning schedule transactions (830/DELFOR) to send to the supplier and to communicate the expected demand for items over a specified period.	See this topic: <ul style="list-style-type: none"> Understanding Outbound Planning Schedule Forecasts
Outbound Extraction - Supplier program (R470621) This is an EDI outbound program.	When using EDI to communicate planning forecasts and release schedules between buyer and supplier, you run the Outbound Extraction - Supplier program (R470621). The planning schedule records are extracted from the F3430 table into the EDI Planning Schedule Header - Outbound table (F47066) and the EDI Planning Schedule Detail - Outbound table (F47067).	See this topic: <ul style="list-style-type: none"> Capturing Supplier Commitments Using EDI
Outbound Extraction - Ad Hoc Supplier Release Schedules program (R470622) This is an EDI outbound program.	Communication of planned and committed quantities between buyers and suppliers can occur when you are using EDI. For ad hoc schedules, the records are extracted from the Ad hoc Vendor Schedule File table into the EDI 830 outbound tables by running this program.	See this topic: <ul style="list-style-type: none"> Capturing Supplier Commitments Using EDI
EDI Planning Schedule Extraction Conversion program (R47062C)	Communication of planned and committed quantities between buyers	See this topic:

Program and Program Type	Description	Available Documentation
This is an EDI outbound program.	and suppliers can occur when you are using EDI. The records in the EDI 830 outbound tables are translated into flat file format by running this program.	<ul style="list-style-type: none"><li data-bbox="781 285 1040 338">• <i>Capturing Supplier Commitments Using EDI</i>

66 SCM and Manufacturing: Pricing

Pricing Overview

This chapter provides detailed information about the business interfaces that are available for the Pricing business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>These sales order-related business service methods are used by the Pricing business object:</p> <ul style="list-style-type: none"> • get CustomerItem Price • getCustomerItemPriceV2 • getItemListPrice • getItemPriceAndAvailability • getItemPriceAndAvailabilityV2 • getItemPriceAndAvailabilityV3 <p>For information about these methods, see <i>Sales Order Business Services - SalesOrderManager</i>.</p> <p>These inventory-related business service methods are used by the Pricing business object:</p> <ul style="list-style-type: none"> • getSupplierCatalogPrice • getSupplierCatalogPriceV2 • processSupplierCatalogPrice • processSupplierCatalogPriceV2 <p>For information about these methods, see <i>Item Branch Master Business Services - InventoryManager</i>.</p>
Real-Time Events	<p>This list includes the real-time events for the Pricing business object:</p> <ul style="list-style-type: none"> • RTLPOUT, which is a container event for RTLPOUTDS • RTLPOUT2
Batch Import Programs (Release 9.2 Update)	<p>This list includes the batch import programs for the Pricing business object:</p> <ul style="list-style-type: none"> • Item Price Unedited Transaction program (R4106Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Pricing Real-Time Events

This section describes the available real-time events associated with the Pricing business object, which include:

- RTLPOUT, which is a container event for RTLPOUTDS.
- RTLPOUT2

RTLPOUT

The RTLPOUT event is used by the JD Edwards EnterpriseOne system to publish an item's list price information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTLPOUT	Item Base Price	RTE	Container Container event for RTLPOUTDS (D4502700B).	45	D4502700B: F4106 List Price Real Time Event

Conceptual Approach

When a user creates, modifies, or cancels an active price record (F4106) that does not involve a customer group or item group, the system triggers the RTLPOUT notification event.

Objects that Can Initiate the Event

These objects can initiate the RTLPOUT event, and all single events contained by the event:

- P4106 - Work With Preference Base Price
- R41816 - Item Sales Price Level Conversion
- B4002270 - F4106 Update Base Price
- X4078 - Margin Maint Price Update
- X4106C - Copy/Write Price Records

RTLPOUTDS

The RTLPOUTDS event is used by the JD Edwards EnterpriseOne system to publish an item's list price information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTLPOUTDS	Item Base Price Details	RTE	Single	H45	D4502700B: F4106 List Price Real Time Event

RTLPOUT2

The RTLPOUT event is used by the JD Edwards EnterpriseOne system to publish an item's list price information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTLPOUT2	Item Base Price Version 2	RTE	Single	45	D4502710B: F4106 List Price Real Time Event

Conceptual Approach

When a user creates, modifies, or cancels an active price record (F4106) that does not involve a customer group or item group, the system triggers the RTLPOUT2 notification event.

Objects that Can Initiate the Event

These objects can initiate the RTLPOUT2 event:

- B4002270 – F4106 Update Base Price X4078 – Margin Maint Price Update
- X4106C - Copy Write Price Records
- R41816 – Item Sales Price Level Conversion

Item Base Price Batch Import Program

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import program that is associated with the Pricing business object:

Program and Program Type	Description	Available Documentation
Item Price Unedited Transaction program (R4106Z1) This is an interoperability program.	During processing, the Item Price Unedited Transaction program (R4106Z1) adds or changes multiple records that exist within the Unedited Transaction Table - Item Price table (F4106Z1) directly to the F4106 table.	See this topic: <ul style="list-style-type: none"> • <i>Item Base Price from Inbound Transactions</i> in the <i>JD Edwards EnterpriseOne Interoperability Fundamentals Implementation Guide</i>

67 SCM and Manufacturing: Product Data Management

Product Data Management Overview

This chapter provides detailed information about the business interfaces that are available for the Product Data Management business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import and Export Programs	<p>This list includes the batch import and export programs for the Product Data Manufacturing business object:</p> <ul style="list-style-type: none"> Process Inbound Work Center program (R30006Z1I) Process Inbound Routing (R3003Z1I) Process Inbound Bill of Material program (R3002Z1I) <p>Note that both the R3003Z1I and R3002Z1I can function as both import and export programs.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Product Data Management Batch Import and Export Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import and export programs that are associated with the Product Data Management business object:

Program and Program Type	Description	Available Documentation
Process Inbound Work Center program (R30006Z1I)	You run the Process Inbound Work Center program (R30006Z1I) to copy the information from the F30006Z1	<p>See this topic:</p> <ul style="list-style-type: none"> Inbound Work Center Data

Program and Program Type	Description	Available Documentation
This is a temporary table import program.	table to the Work Center Master File table (F30006).	
<p>Process Inbound Routing (R3003Z11)</p> <p>This is an interoperability program.</p> <p>This program can be used as a batch import and a batch export program.</p>	<p>You run the Process Inbound Routing program (R3003Z11) to copy the information from the F3003Z1 table to the Routing Master File table (F3003).</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Processing Inbound Interoperability for Product Data Management</i>
<p>Process Inbound Bill of Material program (R3002Z11)</p> <p>This is an interoperability import program.</p> <p>This program can be used as a batch import and a batch export program.</p>	<p>You run the Process Inbound Bill of Material program (R3002Z11) to copy the information from the F3002Z1 table to the Bill of Material Master File table (F3002).</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Inbound Bill of Material Data</i>

68 SCM and Manufacturing: Sales Invoice

Sales Invoice Overview

This chapter provides detailed information about the business interfaces that are available for the Sales Invoice business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the Sales Invoice business object:</p> <ul style="list-style-type: none"> • RTINVOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTINVHDR ○ RTINVDTL • RTINVOUT2, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTINVHDR* ○ RTINVDTL2
Batch Export Programs	<p>This list includes the batch export programs for the Sales Invoice business object:</p> <ul style="list-style-type: none"> • Invoice Print program (R42565)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Sales Invoice Real-Time Events

This section describes the available real-time events associated with the Sales Invoice business object, which include:

- RTINVOUT, which is a container event for these single events:
 - RTINVHDR
 - RTINVDTL

- RTINVOUT2, which is a container event for these single events:
 - RTINVHDR*
 - RTINVDTL2

*Note that RTINVHDR is contained within this event, and also within RTINVOUT. The documentation for RTINVHDR is located in the RTINVOUT section of this chapter.

RTINVOUT

RTINVOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems that an invoice has been created or changed. This transaction is for the publishing information required to meet the RosettaNet standards.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTINVOUT	Sales Invoice	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTINVOUTA (D4202330B) • RTINVHDR • RTINVOUTB (D4202330C) 	H42

Conceptual Approach

When a new Invoice Number, Invoice Type, or Invoice Company is assigned to a sales order detail record in the Sales Order Detail (F4211) table, the RTINVOUT real-time event is published.

Objects that Can Initiate the Event

These objects can initiate the RTINVOUT event and the single events contained within this event:

- B4208020 - Do Invoice Level Inventory
- R42565 - Invoice Print

RTINVHDR

RTINVHDR is used by the JD Edwards EnterpriseOne system to inform third-party systems that an invoice has been created or changed. This transaction is for the publishing information required to meet the RosettaNet standards.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTINVHDR	Sales Invoice Header	RTE	Single	H42	D4202330B: Invoice Header for Invoice Real Time Event DS	Included in container event RTINVOUT.

Conceptual Approach

When a new Invoice Number, Invoice Type, or Invoice Company is assigned to a sales order detail record in the Sales Order Detail (F4211) table, the RTINVOUT real-time event is published. The RTINVOUT includes the RTINVHDR event. The RTINVHDR event contains sales order header information.

RTINVDTL

RTINVDTL is used by the JD Edwards EnterpriseOne system to inform third-party systems that an invoice has been created or changed. This transaction is for the publishing information required to meet the RosettaNet standards.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTINVDTL	Sales Invoice Detail	RTE	Single	H42	D4202330C: Invoice Detail for Real Time Events DS	Included in container event RTINVOUT.

Conceptual Approach

When a new Invoice Number, Invoice Type, or Invoice Company is assigned to a sales order detail record in the Sales Order Detail (F4211) table, the RTINVOUT real-time event is published. The RTINVOUT includes the RTINVDTL event. The RTINVDTL event contains sales order detail information.

RTINVOUT2

RTINVOUT2 is used by the JD Edwards EnterpriseOne system to inform third-party systems that an invoice has been created or changed. This transaction is for the publishing information required to meet the RosettaNet standards.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTINVOUT2	Sales Invoice Version 2	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTINVHDR (D4202330B) • RTINVDTL2 (D4202430C) 	H42

Conceptual Approach

When a new Invoice Number, Invoice Type, or Invoice Company is assigned to a sales order detail record in the Sales Order Detail (F4211) table, the RTINVOUT2 real-time event is published.

Objects that Can Initiate the Event

These objects can initiate the RTINVOUT event and the single events contained within this event:

- B4208020 - Do Invoice Level Inventory
- R42565 - Invoice Print

RTINVDTL2

RTINVDTL2 is used by the JD Edwards EnterpriseOne system to inform third-party systems that an invoice has been created or changed. This transaction is for the publishing information required to meet the RosettaNet standards.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTINVDTL2	Sales Invoice Detail Version 2	RTE	Single	H42	D4202430C: Invoice Detail for Real Time Events DS	Included in container event RTINVOUT2.

Conceptual Approach

When a new Invoice Number, Invoice Type, or Invoice Company is assigned to a sales order detail record in the Sales Order Detail (F4211) table, the RTINVOUT2 real-time event is published. The RTINVOUT2 includes the RTINVDTL2 event. The RTINVDTL2 event contains sales order detail information.

Sales Invoice Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Sales Invoice business object:

Program and Program Type	Description	Available Documentation
Invoice Print program (R42565)	<p>The Print Invoices program (R42565) updates this information in the Sales Order Detail File table (F4211):</p> <ul style="list-style-type: none"> • Invoice number. • Invoice date. • Invoice document type. • Status codes (the program sets the next status code to run the Sales Update program (R42800)). 	<p>See this topic:</p> <ul style="list-style-type: none"> • Printing Standard Invoices

69 SCM and Manufacturing: Sales Order

Sales Order Overview

This chapter provides detailed information about the business interfaces that are available for the Sales Order business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The SalesOrderManager web service (JP420000) manages the execution of these sales order-related web service operations:</p> <ul style="list-style-type: none"> • processSalesOrder (J4200010) • processSalesOrderV2 (J4200010) • processSalesOrderV3 (J4200010) • processSalesOrderV5 (J4200010) • processSalesPriceAdjustment (J4200020) • processSalesPriceAdjustmentV2 (J4200020) • getItemPriceAndAvailability (J4200030) • getItemPriceAndAvailabilityV2 (J4200030) • getItemPriceAndAvailabilityV3 (J4200030) • getCustomerItemPrice (J4200040) • getCustomerItemPriceV2 (J4200040) • getSalesOrder (J4200050) • getSalesOrderV2 (J4200050) • getSalesOrderV3 (J4200050) • getSalesOrderV4 (J4200050) • getItemListPrice (J4200060) • getSalesOrderPriceHistory (J4200070)
Real-Time Events	<p>This list includes the real-time events for the Sales Order business object:</p> <ul style="list-style-type: none"> • RTSOOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSOHDR ○ RTSODTL • RTSOOUT2, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSOHDR2 ○ RTSODTL2

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> • RTSOOUT3, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSOHDR3 ○ RTSODTL3
Batch Import	<p>This list includes the batch import programs for the Sales Order business object:</p> <ul style="list-style-type: none"> • Recurring Orders Edited Creation Report (R40211Z) • EDI Inbound Purchase Order Edit/Create (R47011) • EDI Inbound Purchase Order Change (R47131) • EDI Request for Quote Edit/Create program (R47091)
Batch Export Programs	<p>This list includes the batch export programs for the Sales Order business object:</p> <ul style="list-style-type: none"> • Outbound Extraction program (R47142) • Invoice Print program (R42565) <p>For information on this program, see Sales Invoice Batch Export Programs.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Sales Order Business Services - SalesOrderManager

The SalesOrderManager web service (JP420000) manages the execution of sales order-related web service operations. This table includes a description of the sales order web service operations:

Operation	Description
processSalesOrder (J4200010)	Use this operation to process sales order information in the JD Edwards EnterpriseOne Sales Order Management system. This operation enables you to add, change, or cancel a sales order.
processSalesOrderV2 (J4200010)	Use this operation to process sales order information with taxed prices in the JD Edwards EnterpriseOne Sales Order Management system when the Enable Sales Taxed Prices constant is selected. This operation enables you to add, change, or cancel a sales order.
processSalesOrderV3 (J4200010)	Use this operation to add, update, or cancel sales order information along with global locator number (GLN), in the JD Edwards EnterpriseOne Sales Order Management system.
processSalesOrderV5 (J4200010)	Use this operation to add, update, or cancel sales order information along with the Opportunity ID in the JD Edwards EnterpriseOne Sales Order Management system.

Operation	Description
processSalesPriceAdjustment (J4200020)	<p>Use this operation to process sales price adjustment data in the JD Edwards EnterpriseOne Advanced Pricing and Sales Order Management systems. This operation enables you to:</p> <ul style="list-style-type: none"> • Add or update price adjustment definition records. • Add or update price adjustment detail records. • Add or update adjustment schedule records.
processSalesPriceAdjustmentV2 (J4200020)	<p>Use this operation to add or update price adjustment records including GLN, in the JD Edwards EnterpriseOne Advanced Pricing and Sales Order Management system.</p>
getItemPriceAndAvailability (J4200030)	<p>Use this operation to retrieve item pricing and availability information from the JD Edwards EnterpriseOne Inventory Management system.</p> <p>Note: There are several operations that provide item pricing information to the consumer. Use this operation to review item availability information in addition to item pricing.</p>
getItemPriceAndAvailabilityV2 (J4200030)	<p>Use this operation to retrieve item pricing, including taxed prices using the base price or advanced pricing for the specific branch plant, and item availability information from the JD Edwards EnterpriseOne Inventory Management system when the Enable Sales Taxed Prices constant is selected.</p>
getItemPriceAndAvailabilityV3 (J4200030)	<p>Use this operation to retrieve item pricing and item availability information, along with GLN, from the JD Edwards EnterpriseOne Inventory Management system.</p>
getCustomerItemPrice (J4200040)	<p>Use this operation to retrieve item pricing for a specified customer from the JD Edwards EnterpriseOne Inventory Management system.</p> <p>Note: There are several operations that provide item pricing information to the consumer. Use this operation to review item prices, including line-level discounts and markups, for specific customers.</p>
getCustomerItemPriceV2 (J4200040)	<p>Use this operation to retrieve item pricing for a specified customer, including the taxed prices and taxed amounts with line level discounts and/or markups using the base price or advanced pricing, from the JD Edwards EnterpriseOne Inventory Management system, when the Enable Sales Taxed Prices constant is selected.</p> <p>Note: There are several operations that provide item pricing information to the consumer. Use this operation to review item prices, including line-level discounts and markups, for specific customers.</p>
getSalesOrder (J4200050)	<p>Use this operation to review sales orders that exist in the JD Edwards EnterpriseOne Sales Order Management system.</p>
getSalesOrderV2 (J4200050)	<p>Use this operation to review sales orders, including taxed prices and taxed amounts, that exist in the JD Edwards EnterpriseOne Sales Order Management system.</p>
getSalesOrderV3 (J4200050)	<p>Use this operation to review sales orders, including GLNs, that exist in the JD Edwards EnterpriseOne Sales Order Management system.</p>
getSalesOrderV4 (J4200050)	<p>Use this operation to review sales orders, including GLNs, and opportunity information, that exist in the JD Edwards EnterpriseOne Sales Order Management system.</p>

Operation	Description
getItemListPrice (J4200060)	Use this operation to retrieve a list of base prices from the JD Edwards EnterpriseOne Inventory Management system for a specified item. Note: There are several operations that provide item pricing information to the consumer. Use this operation to review base prices for specified items. This operation does not provide customer-specific prices or availability information.
getSalesOrderPriceHistory (J4200070)	Use this operation to retrieve historical pricing information from the JD Edwards EnterpriseOne Sales Order Management system.

Accessing Javadoc for the Sales Order Manager Web Service Operations

To access Javadoc for the Sales Order Manager web service and its related operations, review these Javadoc packages:

- JP420000 (SalesOrderManager)
- J4200010 (processSalesOrder)
- J4200010 (processSalesOrderV2)
- J4200010 (processSalesOrderV3)
- J4200010 (processSalesOrderV5)
- J4200020 (processSalesPriceAdjustment)
- J4200030 (getItemPriceAndAvailability)
- J4200030 (getItemPriceAndAvailabilityV2)
- J4200040 (getCustomerItemPrice)
- J4200040 (getCustomerItemPriceV2)
- J4200050 (getSalesOrder)
- J4200050 (getSalesOrderV2)
- J4200050 (getSalesOrderV3)
- J4200050 (getSalesOrderV4)
- J4200060 (getItemListPrice)
- J4200070 (getSalesOrderPriceHistory)

Prerequisites

Before you use the Sales Order Manager web service, or any of the related operations, you must install and set up these JD Edwards EnterpriseOne systems:

- Sales Order Management
- Inventory Management

Note: Many of these operations managed by the SalesOrderManager web service process pricing information. If you want these operations to process pricing information using advanced pricing functions, you must install and set up the JD Edwards EnterpriseOne Advanced Pricing system. If you do not set up this system, these operations will process, but they will not include advanced pricing calculations.

See [JD Edwards EnterpriseOne Applications Advanced Pricing Implementation Guide](#) .

See [JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide](#) .

See [JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide](#) .

processSalesOrder

The processSalesOrder web service operation is an inbound transaction operation that enables consumers to add, change, and cancel these order types in the JD Edwards EnterpriseOne Sales Order Management system:

- Sales orders
- Blanket orders
- Credit orders
- Direct ship orders
- Quote orders
- Transfer orders
- Transportation orders

If the operation encounters errors during processing, the system returns those errors to the consumer.

Prerequisites

Before using this operation, you must be familiar with entering, updating and deleting sales orders in the JD Edwards EnterpriseOne system.

See ["Entering Sales Order Header and Detail Information" in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide](#) .

Supported Functionality

This section discusses the functionality that the processSalesOrder operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The processSalesOrder operation supports inventory/stock items, configured items, non-stock items such as miscellaneous, text lines and freight, and direct ship items. An order may contain one or more of these line types.

The processSalesOrder operation supports retrieval of the alternate tax rate/area assignment if the functionality is enabled for the entities involved in the transaction.

See ["Sales Order Management" in the JD Edwards EnterpriseOne Applications Tax Processing Implementation Guide](#) .

Blanket Orders

Though the solution does support the addition, change or cancellation of blanket orders, the solution does not support automatic blanket order release. Additionally, the solution does not support blanket orders for configured items.

Advanced Pricing

The solution supports advanced pricing for line, basket, and order-level pricing. The operation does not support advanced pricing when an override price is passed in by the consumer.

Kit Items

The solution does not support kit items. If the consumer enters a kit item, the system will process the information, but kit processing is not initiated in the JD Edwards EnterpriseOne system. The operation returns a *Kit Components Are Not Processed* message to the consumer.

If a kit parent item is entered by the consumer, the system processes the parent, but does not initiate kit processing for components. To resolve the issue, the consumer must use the JD Edwards EnterpriseOne Sales Order Entry program (P4210) to choose the kits components.

Configured Items

The solution does support the processing of configured items. The operation provides the consumer with a field structure that enables the specification of configured items and their segments.

Configured items may be pre-evaluated and passed into the JD Edwards EnterpriseOne system as they are, which is known as *bypass*. Alternatively, consumers can pass items into the JD Edwards EnterpriseOne system and have the item evaluated using JD Edwards EnterpriseOne defined rules and logic.

Configured parent items may contain child items of varying types, such as stock, non-stock and configurable.

Configured items are enabled for sales orders and quote orders only at the configured item parent level.

The solution does not support blanket orders for configured items.

When processing configured items, be aware of the following items:

- The stocking type for a configured item is hard-coded as **C**.
- The consumer must provide segment answers when adding or changing a configured item.
- To utilize the JD Edwards EnterpriseOne rules and logic for evaluation, segment answers and configured components are required when adding or changing a configured item.
- To bypass evaluation in the JD Edwards EnterpriseOne system, the consumer must provide the segment answers and all components when adding or changing a configured item.
- The solution does not validate segment answers and cross-segment editing rules.
- The consumer must sequence the configuration by children before siblings.

For example, configured item A has components A1 and A2. Component A1 has another child component, A11. Using this example, the input to the operation should follow this sequence:

- a. parentItem = null, childItem = A
- b. parentItem = A, childItem = A1
- c. parentItem = A, childItem = A2
- d. parentItem = A1, childItem = A11

- The solution does not support the changing or cancellation of individual components of a configured item.
- When processing the cancellation of a configured item, the operation also cancels the components from the sales order details.
- The solution does not support back orders, transfer orders, direct ship orders, and inter-branch orders for configured items.
- The price will be overridden when the isZeroPriceOverride field is set to **true**, or there is a price (domestic/foreign) in the input document.
- The solution does not support:
 - Dynamic rules validation during configuration.
 - The limitation of configured selections to include only valid responses.
 - Calculation of pricing rules based on a user-defined date.

Setup Considerations

Before using this operation, you can set business service properties to specify how the system processes sales order information. This table includes information about the business service properties that the processSalesOrder web service operation uses:

Group	Key	Description	Default Value
J4200010	J4200010_SOE_MBF_VERSION	Use this business service property to specify the version of the Sales Order Entry program (P4210) that the operation uses when processing data.	ZJDE0001
J4200010	J4200010_BYPASS_WARNINGS	<p>Use this business service property to specify whether the operation bypasses warnings that occur during processing.</p> <p>If you choose to bypass warnings, the operation continues processing to the end, and returns all warnings to the consumer.</p> <p>If you choose not to bypass warnings, change the default value to 2. Using this setting, the operation stops processing when a warning is encountered and the warning is returned to the consumer.</p>	1: Bypass Warnings

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Note:

- *"Setting Processing Options for Sales Order Entry (P4210)" in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide .*

Implementation Details

The following table includes information that can help determine whether the processSalesOrder operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany • businessUnit • entity – soldTo • itemId • documentLineNumber
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. In the event that the operation encounters errors, all JD Edwards EnterpriseOne tables are returned to their original state, and no updates are performed. No manual updates are necessary.</p>
Does this operation use record reservation?	<p>This operation reserves records in the JD Edwards EnterpriseOne system during the change or cancellation of a sales order.</p>

processSalesOrderV2

Review the information in the processSalesOrder section of this chapter before using the processSalesOrderV2 web service operation. All of the information in the processSalesOrder section also applies to processSalesOrderV2.

processSalesOrderV2 is a new version of the processSalesOrder web service operation. This web service operation is used to add, update, or cancel sales orders with taxed prices when the Sales Taxed Prices constant is selected. This new version executes the processSalesOrder (J4200010) web service operation.

processSalesOrderV3

Review the information in the processSalesOrder section of this chapter before using the processSalesOrderV3 web service operation. All of the information in the processSalesOrder section also applies to processSalesOrderV3.

The processSalesOrderV3 operation is a version of the processSalesOrderV2 web service operation. This web service operation is used to add, update, or cancel sales orders along with GLNs. This version executes the processSalesOrder (J4200010) web service operation.

processSalesOrderV5

Review the information in the processSalesOrder V1, V2, and V3 sections of this chapter before using the processSalesOrderV5 web service operation. All of the information in the other versions of the processSalesOrder methods also applies to processSalesOrderV5.

The processSalesOrderV3 operation is a version of the processSalesOrder web service operation. This web service operation is used to add, update, or cancel sales orders along with Opportunity IDs. This version executes the processSalesOrder (J4200010) web service operation.

processSalesPriceAdjustment

The processSalesPriceAdjustment operation is an inbound transaction operation that enables consumers to process sales price adjustment information within the JD Edwards EnterpriseOne system.

The consumer can add this information to the JD Edwards EnterpriseOne Sales Order Management system:

- Price adjustment definition records.
- Price adjustment detail records.
- Price adjustment schedule records.

Consumers are also able to modify existing price adjustment detail records, provided that values for the adjustment name and adjustment ID are passed in.

If the system encounters errors while processing transactions using this operation, error messages are generated and returned to the consumer.

Be aware that this operation updates only the pricing setup records in the JD Edwards EnterpriseOne system. Existing sales orders are not updated with pricing details that are entered into the system using this operation. To reprice existing sales orders using the updated pricing information, you must process the Sales Order Batch Price/Cost Update program (R42950).

See *"Updating Prices for a Customer" in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide* .

Note: To add a price adjustment detail record, you must include a unit of measure (UoM). If you do not include a unit of measure, the system returns a hard error to the consumer. If currency processing is turned on, you must include a currency code. If no currency code is passed in, and currency processing is enabled, the system returns a hard error to the consumer. If no dates are passed in, the system uses the system date as the effective date, and an expiration date that is generated by business function B4000630.

Prerequisite

Before using the processSalesPriceAdjustment operation, you must install and configure the JD Edwards EnterpriseOne Advanced Pricing system.

See *"Setting Up EnterpriseOne Advanced Pricing" in the JD Edwards EnterpriseOne Applications Advanced Pricing Implementation Guide* .

Supported Functionality

This section discusses the functionality that the processSalesPriceAdjustment operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

When adding price adjustment definition records:

- Item price groups, customer price groups, and order detail groups are not supported.
- Basket, order, and volume level adjustments are not supported.
Only line level adjustments are supported.
- Weight, amount, and quality level breaks are not supported.
Only quantity level breaks are supported.
- These adjustment control codes are supported:
 - Blank
 - **2:** Print on Invoice
 - **4:** Accruals
- When making changes to price adjustment detail records:
 - The From Level value always starts at **1**.
 - Basis code **5** (Add on Amount) is supported.

Setup Considerations

Before you use this operation, you can set a business service property to define which version of the Live Promotions Interface Processing program (R45720Z) to use. This table lists the business service properties that the processSalesPriceAdjustment operation uses:

Group	Key	Description	Default Value
J4200020	J4200020_PROM_MBF_VERSION	Use this business service property to	ZJDE0002

Group	Key	Description	Default Value
		specify the version of the Live Promotions Interface Processing program (R45720Z) that this operation uses.	

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

The following table includes information that can help determine whether the processSalesPriceAdjustment operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • adjustmentTypeCode • priceAdjustmentId • unitOfMeasureTransaction • actionType • processingVersion
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.
Does this operation use record reservation?	No. This operation does not reserve records in the JD Edwards EnterpriseOne system during processing.

Setting Processing Options for Live Promotions Interface Processing (R45720Z)

You use processing options to set up default values and processing data for a program.

Preference Hierarchy

1. Enter the preference hierarchy name to be used. (Required)

Use this processing option to specify a preference type or a price adjustment hierarchy defined in user-defined code (UDC) table 40/PR. This value gets stored in the Price Adjustment Type table (F4071) in field ATPRFR (Preference) and its associated hierarchy is used for resolving pricing adjustments.

Adjustment Definition

The processing options on this tab are used to define the adjustment definition. One of the input parameters, such as costTypeCode, determines whether the adjustment is a Bill Back or an Off Invoice adjustment. The value sent in

becomes part of the adjustment definition record that is stored in the F4071 table. Adjustment types include 1(Bill Back: Adjustment Control Code = 4, Accruals) or 2 (Off Invoice: Adjustment Control Code = 2, Print on Invoice). If no value is passed in for costTypeCode field, Adjustment Control Code, stored in field ATACNT of the F4071 table, is blank.

1. Enter GL Offset for Bill Back adjustment definition.

Use this processing option to specify the trade account that the system uses as the offset when you post invoices or vouchers. This value gets stored in the Price Adjustment Type table (F4071) in field ATGLC (GLClass).

2. Enter Subledger for Bill Back adjustment definition.

Use this processing option to identify a detailed, auxiliary account within a general ledger account. A subledger can be an equipment item number or an address book number. If you enter a subledger, you must also specify the subledger type.

This value is stored in the Price Adjustment Type table (F4071) in field ATSBIF (SubledgerInformation).

3. Enter GL Offset for Off Invoice adjustment definition.

Use this processing option to specify the trade account that the system uses as the offset when you post invoices or vouchers. This value gets stored in the Price Adjustment Type table (F4071) in field ATGLC (GLClass).

4. Enter Subledger for Off Invoice adjustment definition.

Use this processing option to identify a detailed, auxiliary account within a general ledger account. A subledger can be an equipment item number or an address book number. If you enter a subledger, you must also specify the subledger type.

This value is stored in the Price Adjustment Type table (F4071) in field ATSBIF (SubledgerInformation).

Schedule Option

1. Enter Schedule Option.

Use this processing option to specify the schedule option. Values are:

1: Single

2: Customer

3: Manual

Single Schedule Option

1. Enter schedule name. (Required)

Use this processing option to specify the name of the schedule.

2. Enter prefix (1 char) for adjustment name. (Required)

Use this processing option to specify the one-character prefix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

3. Enter suffix (1 char) for adjustment name. (Required)

Use this processing option to specify the one-character suffix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

4. Enter increment for sequence in schedule. If blank, 1.0 will be used.

Use this processing option to specify the numeric increment for sequence in schedule. If you leave this option blank, the system uses an increment of 1.0.

5. Enter start reserved range of sequence numbers for promotions.

Use this processing option to specify the beginning value in a range of sequence numbers that are reserved for promotions.

6. Enter end reserved range of sequence numbers for promotions.

Use this processing option to specify the ending value in a range of sequence numbers that are reserved for promotions.

Customer Schedule Option

1. Enter prefix (1 char) for adjustment name. (Required)

Use this processing option to specify the one-character prefix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

2. Enter suffix (1 char) for adjustment name. (Required)

Use this processing option to specify the one-character suffix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

3. Enter increment for sequence in schedule. If blank, 1.0 will be used.

Use this processing option to specify the numeric increment for sequence in schedule. If you leave this option blank, the system uses an increment of 1.0.

4. Enter start reserved range of sequence numbers for promotions.

Use this processing option to specify the beginning value in a range of sequence numbers that are reserved for promotions.

5. Enter end reserved range of sequence numbers for promotions.

Use this processing option to specify the ending value in a range of sequence numbers that are reserved for promotions.

Manual Schedule Maintenance Option

1. Enter Adjustment name for Bill Back spending method.

Use this processing option to specify the adjustment name for Bill Back spending method.

Note: If you leave options 3 and 4 blank, you must enter a value in this processing option and in option 2.

2. Enter Adjustment name for Off Invoice spending method.

Use this processing option to specify the adjustment name for Off Invoice spending method.

Note: If you leave options 3 and 4 blank, you must enter a value in this processing option and in option 1.

3. Enter prefix (1 char) for adjustment name.

Use this processing option to specify the one-character prefix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

Note: If you leave options 1 and 2 blank, you must enter a value in this option and in option 4.

4. Enter suffix (1 char) for adjustment name.

Use this processing option to specify the one-character suffix that is used for the adjustment name. The adjustment name is the prefix plus the next number plus the suffix for Bill Back and Off Invoice adjustment names. For example, if the prefix is **S** and the suffix is **B** (Bill Back), the adjustment name might be S123456B.

Note: If you leave options 1 and 2 blank, you must enter a value in this option and in option 3.

processSalesPriceAdjustmentV2

Review the information in the processSalesPriceAdjustment section of this chapter before using the processSalesPriceAdjustmentV2 web service operation. All of the information in the processSalesPriceAdjustment section also applies to processSalesPriceAdjustmentV2.

The processSalesPriceAdjustmentV2 operation is a version of the processSalesPriceAdjustment web service operation. This web service operation is used to add or update price adjustment details along with GLNs. This version executes the processSalesPriceAdjustment (J4200020) web service operation.

getItemPriceAndAvailability

The getItemPriceAndAvailability web service operation is an inbound transaction operation that enables consumers to retrieve pricing, availability, and branch/plant information for items that are stored in the JD Edwards EnterpriseOne Inventory Management system.

Note: There are several operations that return item prices to the consumer. Use this operation if you want to review item availability information in addition to pricing information.

This operation uses the information that the consumer passes in, along with the Price and Availability Header business function (B4204100), to simulate the creation of a sales order header and sales order detail information. The system then uses this simulated sales order to retrieve pricing information. The price is based on the branch/plant that the consumer passes in. If the consumer does not pass in a branch/plant, the system calculates the price using the branch/plant from the processing options of the P4210. The operation then returns to the consumer the unit price and extended price for the item.

Additionally, the operation uses the Retrieve F41021 Records business function (B4204120) to calculate the availability of the item in the specified branch/plant. The operation then returns to the consumer the availability information, along with the address book number and mailing address of the branch/plant.

If the operation encounters errors or warnings during processing, the system returns those errors and warnings to the consumer.

Supported Functionality

This section discusses the functionality that the getItemPriceAndAvailability operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

This operation does not support configured items, as configured items typically do not have availability information.

If you are using advanced pricing, the operation applies markups and discounts. The operation does not apply free goods and rebates.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves and calculates price and availability information. This table includes information about the business service properties that the `getItemPriceAndAvailability` operation uses:

Group	Key	Description	Default Value
J4200030	J4200030_SOE_MBF_VERSION	Use this business service property to specify the version of the Sales Order Entry program (P4210) that the operation uses when processing price and availability information.	ZJDE0001
J4200030	J4200030_BYPASS_BSFN_WARNINGS	<p>Use this business service property to specify whether the operation bypasses warnings that occur during processing.</p> <p>If you choose to bypass warnings, the operation continues processing to the end, and returns all warnings to the consumer.</p> <p>If you choose not to bypass warnings, change the default value to 2. Using this setting, the operation stops processing when a warning is encountered and the warning is returned to the consumer.</p>	1: Bypass Warnings

The `getItemPriceAndAvailability` operation uses the Sales Order Entry program (P4210) to simulate the creation of a sales order for the purposes of retrieving price and availability information. Typically, the version of the Sales Order Entry program that is used to create and process sales orders is set up to include full editing and validations. The `getItemPriceAndAvailability` operation does not need to complete these processes. Therefore, to improve performance, you might consider using a different version of the Sales Order Entry program when processing this operation.

In addition to using a separate version of the P4210, it is also recommended that you use these guidelines when setting the processing options for the version used by this operation:

- Order Holds must be blank.

- Activate Availability Checking must be blank.
- Blanket or Quote Processing must be blank.
- Inventory Commitment Preferences must be blank.
- If Preferences are turned on, the following preferences must not be used:
 - Product Allocation
 - Grade and Potency
 - Next Order Status
 - Sales Commissions

See *"Setting Processing Options for Sales Order Entry (P4210)"* in the *JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide* .

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

This table includes information that can help determine whether the `getItemPriceAndAvailability` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system does not return errors to the consumer. In some instances, the consumer does not receive a return message. For example, if the item in the query does not have price or availability information, the operation does not return data to the consumer, even though it has successfully processed.</p> <p>If the operation is successful in finding matching data, price and availability information for the items included in the query are returned to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • priceUnit • priceExtended • quantityAvailable
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a query operation. No transactions occur during the processing of this operation. Therefore, no data updates are necessary if errors are encountered.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisite

Before you can process this operation successfully, you must enter items into the JD Edwards EnterpriseOne Inventory Management system.

See *"Setting Up the Inventory Management System"* and *"Entering Item Information"* in the *JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide*.

getItemPriceAndAvailabilityV2

Review the information in the getItemPriceAndAvailability section of this chapter before using the getItemPriceAndAvailabilityV2 web service operation. All of the information in the getItemPriceAndAvailability section also applies to getItemPriceAndAvailabilityV2.

getItemPriceAndAvailabilityV2 is a new version of the getItemPriceAndAvailability web service operation. This web service operation is used to retrieve the sales price of the item, including taxed prices using the base price or advanced pricing for the specific branch plant and item availability, when the Sales Taxed Prices constant is selected. This new version executes the getItemPriceAndAvailability (J4200030) web service operation.

getItemPriceAndAvailabilityV3

Review the information in the getItemPriceAndAvailability section of this chapter before using the getItemPriceAndAvailabilityV3 web service operation. All of the information in the getItemPriceAndAvailability section also applies to getItemPriceAndAvailabilityV3.

The getItemPriceAndAvailabilityV3 operation is a version of the getItemPriceAndAvailabilityV2 web service operation. This web service operation is used to retrieve the sales price of the item, including GLNs, using the base price or advanced pricing for the specific branch plant and item availability. This version executes the getItemPriceAndAvailability (J4200030) web service operation.

getCustomerItemPrice

The getCustomerItemPrice web service operation is an inbound transaction operation that enables consumers to retrieve item pricing information that is based on the base pricing or on advanced pricing with line-level discounts or markups.

Note: There are several operations that return item prices to consumers. Use this operation if you want to review the cost of an item for a particular customer.

This operation uses the information that the consumer passes in, along with the Price and Availability Header business function (B4204100), to simulate the creation of a sales order header and sales order detail information. The system then uses this simulated sales order to retrieve pricing information for the specified item and customer.

You can include only one item in each request.

Supported Functionality

This section discusses the functionality that the getCustomerItemPrice operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The `getCustomerItemPrice` operation does not include basket or order-level adjustments when calculating the final price of the item.

Consumers can request pricing information for kit and configured items. However, the operation returns prices only for the parent item if one has been specified.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system retrieves and calculates price and availability information. This table includes information about the business service properties that the `getCustomerItemPrice` operation uses:

Group	Key	Description	Default Value
J4200030	J4200030_SOE_MBF_VERSION	Use this business service property to specify the version of the Sales Order Entry program (P4210) that the operation uses when processing price and availability information.	ZJDE0001
J4200030	J4200030_BYPASS_BSFN_WARNINGS	<p>Use this business service property to specify whether the operation bypasses warnings that occur during processing.</p> <p>If you choose to bypass warnings, the operation continues processing to the end, and returns all warnings to the consumer.</p> <p>If you choose not to bypass warnings, change the default value to 2. Using this setting, the operation stops processing when a warning is encountered and the warning is returned to the consumer.</p>	1: Bypass Warnings

The `getCustomerItemPrice` operation uses the Sales Order Entry program (P4210) to simulate the creation of a sales order for the purposes of retrieving price and availability information. Typically, the version of the Sales Order Entry program that is used to create and process sales orders is set up to include full editing and validations. The `getCustomerItemPrice` operation does not need to complete these processes. Therefore, to improve performance, you might consider using a different version of the Sales Order Entry program when processing this operation.

In addition to using a separate version of the P4210, it is also recommended that you use these guidelines when setting the processing options for the version used by this operation:

- Order Holds must be blank.
- Activate Availability Checking must be blank.
- Blanket or Quote Processing must be blank.
- Inventory Commitment Preferences must be blank.
- If Preferences are turned on, the following preferences must not be used:
 - Product Allocation
 - Grade and Potency
 - Next Order Status
 - Sales Commissions

See *"Setting Processing Options for Sales Order Entry (P4210)"* in the *JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide* .

See *JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide* .

Implementation Details

This table includes information that can help determine whether the `getCustomerItemPrice` operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the operation completes successfully, the system does not return errors to the consumer. In some instances, the consumer does not receive a return message. For example, if the item in the query does not have price or availability information, the operation does not return data to the consumer, even though it has successfully processed.</p> <p>If the operation is successful in finding matching data, price and availability information for the items included in the query are returned to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • priceUnitDomestic • priceUnitForeign • priceExtendedDomestic • priceExtendedForeign • costUnitDomestic • costUnitForeign • costExtendedDomestic • costExtendedForeign
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This is a query operation. No transactions occur during the processing of this operation. Therefore, no data updates are necessary if errors are encountered.</p>

Question	Answer
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisites

Before you can use this operation, you must enter item and pricing information into the JD Edwards EnterpriseOne Inventory Management system.

See *"Entering Item Information" in the JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide*.

getCustomerItemPriceV2

Review the information in the `getCustomerItemPrice` section of this chapter before using the `getCustomerItemPriceV2` web service operation. All of the information in the `getCustomerItemPrice` section also applies to `getCustomerItemPriceV2`.

`getCustomerItemPriceV2` is a new version of the `getCustomerItemPrice` web service operation. This web service operation is used to retrieve the sales price of the item, including the taxed prices and taxed amounts with line level discounts and/or markups using the base price or advanced pricing, when the Sales Taxed Prices constant is selected. This new version executes the `getCustomerItemPrice (J4200040)` web service operation.

getSalesOrder

The `getSalesOrder` web service operation is a database query operation that enables consumers to review sales orders that exist in the JD Edwards EnterpriseOne Sales Order Management system.

The `getSalesOrder` operation uses the Sales Order Inquiry business view (V4211XBP) to select records from the JD Edwards EnterpriseOne Sales Order Management system based on the information that the consumer enters. If no errors are encountered, the operation searches the database for records that match the consumer's query, adding those records to the result set.

The operation returns one or more sales order header records to the consumer, based on the information that was passed in. In addition, the operation returns all sales order detail lines associated with those sales order header records.

If the operation encounters errors while querying the database, it stops processing and returns the errors to the consumer.

Note: The consumer must enter search criteria in order for the operation to process successfully. If the consumer attempts to query the database without specifying any search criteria, the operation returns an error to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns when you query the JD Edwards EnterpriseOne database. This table includes information about the business service properties that the getSalesOrder operation uses:

Group	Key	Description	Default Value
J4200050	J4200050_MAX_ROWS	Use this business service property to specify the maximum number of rows that the query will return. Note: The number of rows includes both sales order header records and sales order detail records.	0: Return all records.

Note: If you leave this constant set to **0**, the system returns all records that match the specified search criteria. To improve processing time, it is recommended that you set the default value for this business service property to a value other than **0**.

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help you determine whether the getSalesOrder operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns records that match your search criteria. The records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system. At a minimum, if the query finds matching records, the system returns non-zero values for these fields: <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany • Entity – Sold To • Entity – Ship To
If I encounter errors while processing the operation, do I need to reverse the transaction?	This is a query operation. No transactions occur during the processing of this operation. Therefore, no data updates are necessary if errors are encountered.
Does this operation use record reservation.	No. The operation does not reserve records that are returned in the query.

Question	Answer

Prerequisite

Before you can use this operation, sales orders must exist in the JD Edwards EnterpriseOne database.

See *"Entering Sales Orders"* and *"Entering Sales Orders for Customer Service Representatives"* in the *JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide* .

getSalesOrderV2

Review the information in the `getSalesOrder` section of this chapter before using the `getSalesOrderV2` web service operation. All of the information in the `getSalesOrder` section also applies to `getSalesOrderV2`.

`getSalesOrderV2` is a new version of the `getSalesOrder` web service operation. This web service operation is used to retrieve the sales order information in real-time, including taxed prices and taxed amounts. This new version executes the `getSalesOrder (J4200050)` web service operation.

getSalesOrderV3

Review the information in the `getSalesOrder` section of this chapter before using the `getSalesOrderV3` web service operation. All of the information in the `getSalesOrder` section also applies to `getSalesOrderV3`.

The `getSalesOrderV3` operation is a version of the `getSalesOrderV2` web service operation. This web service operation is used to retrieve the sales order information in real-time, including GLNs. This version executes the `getSalesOrder (J4200050)` web service operation.

getSalesOrderV4

Review the information in the `getSalesOrder` section, and all additional versions of the `getSalesOrder` method, before using the `getSalesOrderV4` web service operation. All of the information in the previous versions of `getSalesOrder` also apply to `getSalesOrderV4`. Additionally, the `getSalesOrderV4` web service has been modified to search by opportunity ID, and to include opportunity information.

The `getSalesOrderV4` operation is a version of the `getSalesOrder` web service operation.

getItemListPrice

The `getItemListPrice` web service operation is a database query operation that enables consumers to view a list of base prices for a specified item. Consumers can also specify the business unit in order to view a list of base prices for an item by business unit.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service properties that the getItemListPrice operation uses:

Group	Key	Description	Default Value
J4200060	J4200060_MAX_ROWS	Use this business service property to specify the maximum number of records that the system returns for a query.	0: Return All Records

Note: If you leave this constant set to **0**, the system returns all records that match the specified search criteria. To improve processing time, it is recommended that you set the default value for this business service property to a value other than **0**.

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the getItemListPrice operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns a list of prices for the item and or business unit specified by the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system. At a minimum, the system returns non-zero values for the priceList field.
If I encounter errors while processing a transaction, do I need to reverse the transaction?	This is a query operation. No transactions occur during the processing of this operation. Therefore, no data updates are necessary if errors are encountered.
Does this operation use record reservation?	No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.

Prerequisites

Before using this operation, you must enter items and base pricing information in the JD Edwards EnterpriseOne Inventory Management system.

See ["Entering Item Master Information"](#) and ["Entering Sales Price Information"](#) in the [JD Edwards EnterpriseOne Applications Inventory Management Implementation Guide](#) .

getSalesOrderPriceHistory

The getSalesOrderPriceHistory web service operation is a database query operation that enables consumers to inquire on price history information that resides in the JD Edwards EnterpriseOne Price Adjustment Ledger table (F4074). The consumer can search for price history by passing in any combination of these fields:

- Order Number
- Order Type
- Order Company
- Line Number

Note: You must pass in at least one of these fields when entering a query. If you enter a blank query, the system returns an error message.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns for a query. This table includes information about the business service properties that the getSalesOrderPriceHistory operation uses:

Group	Key	Description	Default Value
J4200070	J4200070_MAX_ROWS	Use this business service property to specify the maximum number of records that the system returns for a query.	0: Return all matching records.

Note: If you enter a default value of **0** for this business service property, the system does not limit the number of rows returned during a query. To improve processing time, it is recommended that you set the default value for this business service property to a value other than **0**.

See [JD Edwards EnterpriseOne Tools Business Services Development Methodology Guide](#) .

Implementation Details

This table includes information that can help determine whether the getSalesOrderPriceHistory operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the system returns a list of price history records to the consumer. The records that are returned to the consumer include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.

Question	Answer
	<p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany • documentLineNumber
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>If you encounter errors while processing this operation, verify your business service properties and selection criteria. Then perform your query again.</p> <p>If the system does not return any matching records, verify that matching records exist in the JD Edwards EnterpriseOne database.</p>
<p>Does this operation use record reservation?</p>	<p>No. The operation does not reserve records within the JD Edwards EnterpriseOne system during processing.</p>

Sales Order Real-Time Events

This section discusses the real-time events generated by Sales Order Management system. See *Setting Up Real-Time Event Filtering in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide*.

This list describes the available real-time events associated with the Sales Order business object, which include:

- RTSOOUT, which is a container event for these single events:
 - RTSOHDR
 - RTSODTL
- RTSOOUT2, which is a container event for these single events:
 - RTSOHDR2
 - RTSODTL2
- RTSOOUT3, which is a container event for these single events:
 - RTSOHDR3
 - RTSODTL3

RTSOOUT

The RTSOOUT event is used by the JD Edwards EnterpriseOne system to publish sales order header and detail information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSOOUT	Sales Order	RTE	Container Contains events for: <ul style="list-style-type: none"> RTSOHDR (D4202310A) RTSODTL (D4202310B) 	H42

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT, contains two separate events: RTSOHDR and RTSODTL. The RTSOHDR event contains all information pertinent to the sales order header. The RTSODTL event contains all information pertinent to each sales order detail line.

Objects that Can Initiate the Event

These objects can initiate the RTSOOUT event, and the single events contained within the event:

- B3104430 - F4211 Update Miscellaneous Fields for KIT Items
- B3201180 - WO Update Configured Item Costs to F4211 and F3215
- B3201400 - Update Conig. Driven F4211 Fields
- B4003000 - XPI Publish Events From Cache
- B4003020 - XPI Publish Events From Cache 2
- B4200310 - F4211 FS End Doc
- B4201220 - Update F4211 Sales Detail and F42199 S.O. Ledger
- B4208060 - XPI Work Flow Post Commit For TP
- B4300250 - Release Held Order
- N3200840 - F4211 Update Configured Item Costs
- N4600455 - Change F4211 SO Detail File

RTSOHDR

RTSOHDR is used by the JD Edwards EnterpriseOne system to publish sales order header information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSOHDR	Sales Order Header	RTE	Single	H42	D4202310A: Sales Order Header	Included in container event RTSOOUT.

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT, contains two separate events: RTSOHDR and RTSODTL. The RTSOHDR event contains all information pertinent to the sales order header. The RTSODTL event contains all information pertinent to each sales order detail line.

RTSODTL

RTSODTL event is used by the JD Edwards EnterpriseOne system to publish sales order detail line information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSODTL	Sales Order Detail	RTE	Single	H42	D4202310B:Sales Order Detail	Included in container event RTSOOUT.

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT, contains two separate events: RTSOHDR and RTSODTL. The RTSOHDR event contains all information pertinent to the sales order header. The RTSODTL event contains all information pertinent to each sales order detail line.

RTSOOUT2

The RTSOOUT2 event is a new version of the RTSOOUT event. The RTSOOUT2 event is used by the JD Edwards EnterpriseOne system to publish sales order header and detail information with taxed prices information.

There are no changes to the RTSOOUT event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSOOUT2	Sales Order Version 2	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTSOHDR2(D4202410A) • RTSODTL2 (D4202410B) 	H42

All of the objects that can initiate the RTSOOUT event can also initiate the RTSOOUT2 event, with the addition of the B4202420 - Sales Order Real-Time Event Notify Function 2 object.

RTSOHDR2

The RTSOHDR2 event is a new version of the RTSOHDR event. The RTSOHDR2 is used by the JD Edwards EnterpriseOne system to publish sales order header information with taxed prices.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSOHDR2	Sales Order Header Version 2	RTE	Single	H42	D4202410A: Sales Order Header2	Included in container event RTSOOUT2.

RTSODTL2

The RTSODTL2 event is a new version of the RTSODTL event. The RTSODTL2 is used by the JD Edwards EnterpriseOne system to publish sales order detail line information with taxed prices.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSODTL2	Sales Order Detail Version 2	RTE	Single	H42	D4202410B:Sales Order Detail2	Included in container event RTSOOUT2.

RTSOOUT3

The RTSOOUT3 event is used by the JD Edwards EnterpriseOne system to publish sales order header and detail information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSOOUT3	Sales Order Version 3	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTSOHDR3(D4202440A) • RTSODTL3 (D4202440B) 	H42

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT3, contains two separate events: RTSOHDR3 and RTSODTL3. The RTSOHDR3 event contains all information pertinent to the sales order header. The RTSODTL3 event contains all information pertinent to each sales order detail line.

Objects that Can Initiate the Event

These objects can initiate the RTSOOUT3 event, and the single events contained within the event:

- B4003000 - XPI Publish Events From Cache
- B4003020 - XPI Publish Events From Cache 2
- B4200310 - F4211 FS End Doc

- B4202320 - Sales Order Real Time Event Notification
- B4208060 - XPI Work Flow Post Commit For TP

RTSOHDR3

RTSOHDR3 is used by the JD Edwards EnterpriseOne system to publish sales order header information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSOHDR3	Sales Order Header Version 3	RTE	Single	H42	D4202440A: Sales Order Header	Included in container event RTSOOUT.

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT3, contains two separate events: RTSOHDR3 and RTSODTL3. The RTSOHDR3 event contains all information pertinent to the sales order header. The RTSODTL3 event contains all information pertinent to each sales order detail line.

RTSODTL3

RTSODTL3 event is used by the JD Edwards EnterpriseOne system to publish sales order detail line information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSODTL3	Sales Order Detail Version 3	RTE	Single	H42	D4202440B: Sales Order Detail	Included in container event RTSOOUT.

Conceptual Approach

When a user creates, modifies, or cancels a sales order, the system triggers a notification event. This notification event, RTSOOUT3, contains two separate events: RTSOHDR3 and RTSODTL3. The RTSOHDR3 event contains all information pertinent to the sales order header. The RTSODTL3 event contains all information pertinent to each sales order detail line.

Sales Order Batch Import Programs

You use batch import programs to import data into the JD Edwards EnterpriseOne database from an external software system.

This table lists and describes the available batch import programs that are associated with the Sales Order business object:

Program and Program Type	Description	Additional Information
Recurring Orders Edited Creation Report (R40211Z)	If you use taxed prices and access the Recurring Orders Edited Creation report (R40211Z), the system edits the information from the Recurring Order Header Tax (F4001ZTX) and Recurring Order Detail Tax (F4011ZTX) tables and updates the records in the Sales Order Header Tax (F4201TX) and Sales Order Detail Tax (F4211TX) tables.	See this topic: <ul style="list-style-type: none"> Taxed Prices in the Recurring Orders Edited Creation Report (R40211Z)
EDI Inbound Purchase Order Edit/Create (R47011) EDI Inbound Purchase Order Change (R47131) These are EDI programs.	The system supports creating and updating sales orders through electronic data interchange (EDI) using these programs.	See this topic: <ul style="list-style-type: none"> Understanding Sales Order Entry for Production Controlled Items
EDI Request for Quote Edit/Create program (R47091) This is an EDI program.	The EDI Request for Quote Edit/Create program (R47091) access report features that are specific to each EDI Standard document supported by the JD Edwards EnterpriseOne system. The edit/update programs use the data in the EDI inbound interface tables to update applicable application tables. For example, when you receive a purchase order from a trading partner, the Inbound Edit/Update program for customer orders updates the sales application and creates a sales order, based on the data that it received.	See this topic: <ul style="list-style-type: none"> Understanding the Inbound Edit/Update Program (R47091)

Sales Order Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Sales Order business object:

Program and Program Type	Description	Additional Information
Outbound Extraction program (R47142)	Run the Outbound Extraction program (R47142) for order change acknowledgments (865/ORDRSP) to confirm that an order was changed and to convey pricing and availability	See this topic:

Program and Program Type	Description	Additional Information
	information to the customer. Run this program after receiving and processing an inbound customer order change.	<ul style="list-style-type: none">• <i>Understanding Outbound Order Change Acknowledgments</i>

70 SCM and Manufacturing: Ship and Debit

Ship and Debit Overview

This chapter provides detailed information about the business interfaces that are available for the Ship and Debit business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Ship and Debit business object:</p> <ul style="list-style-type: none"> • Ship and Debit Update program (R47762Z1) • Ship and Debit Claim Output program (R47761Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Ship and Debit Batch Import Programs

You use batch import programs to load data from external systems into the EnterpriseOne database.

This table lists and describes the available batch import programs that are associated with the Ship and Debit business object:

Program and Program Type	Description	Available Documentation
Ship and Debit Update program (R47762Z1)	When you receive approval from the supplier, run the Ship and Debit Update program (R47762Z1) to update the F4576 table.	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Working with Ship and Debit Adjustments</i>
Ship and Debit Claim Output program (R47761Z1)	Run the Ship and Debit Claim Output program (R47761Z1) to submit a claim to a supplier with whom you have an agreement. When you run this program, the system extracts data from the F4576 table and creates records in the F4576Z1 table.	<p>See this topic:</p> <ul style="list-style-type: none"> • <i>Understanding Ship and Debit Claims</i>

71 SCM and Manufacturing: Style Item Master

Style Item Master Overview

This chapter provides detailed information about the business interfaces that are available for the Style Item Master business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Style Item Master business object:</p> <ul style="list-style-type: none"> Style Item Inbound Processor report (RCW02Z1)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Style Item Master Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Style Item Master business object:

Program and Program Type	Description	Available Documentation
Style Item Inbound Processor report (RCW02Z1)	When you run the Style Item Inbound Processor report (RCW02Z1), the system copies all the details that are stored in the FCW02Z1 table to the FCW02 table.	<p>See this topic:</p> <ul style="list-style-type: none"> Tables Used by JD Edwards EnterpriseOne Apparel Management System

72 SCM and Manufacturing: Transportation Shipment

Transportation Shipment Overview

This chapter provides detailed information about the business interfaces that are available for the Transportation Shipment business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>These are the business services available for the transportation shipment business object:</p> <ul style="list-style-type: none"> • RateWare XL Processor (J4900010) • RateWare XL JAXRPC Processor (JC490015) • RateWare XL JAXWS Processor (JC490020)
Real-Time Events	<p>This list includes the real-time events for the Transportation Shipment business object:</p> <ul style="list-style-type: none"> • RTSHPNOUT, which is a container event for these single events: <ul style="list-style-type: none"> ○ RTSHPNOUTA ○ RTSHPNOUTB ○ RTSHPNOUTC ○ RTSHPNOUTD ○ RTSHPNOUTE ○ RTSHPNOUTF ○ RTSHPNOUTG ○ RTSHPNOUTH ○ RTSHPNOUTI ○ RTSHPNOUTJ ○ RTSHPNOUTK • RTSHPNOUT2, which is a container event for all of the single events contained within RTSHPNOUT, along with these additional single events: <ul style="list-style-type: none"> ○ RTSHPNOUTL ○ RTSHPNOUTM ○ RTSHPNOUTP
Batch Import Programs	<p>This list includes the batch import programs for the Transportation Shipment business object:</p>

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> • EDI Shipping Notice Edit/Create program (R47031)
Batch Export Programs	<p>This list includes the batch export programs for the Transportation Shipment business object:</p> <ul style="list-style-type: none"> • EDI Advance Ship Notice Extraction program (R47032)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Transportation Shipment Business Services

This section describes the transportation shipment business services.

Rateware XL Processor Business Service (J4900010)

The RateWare XL Processor business service (J4900010) runs on the Oracle Application Server (OAS) platform or the Websphere Application Server (WAS) platform and reads the data stored in the RateWare XL Rate Constants (F4974) table. The E1 Enterprise server collects the data required for rating a shipment from the JD Edwards EnterpriseOne tables and then calls the J4900010 business service. The J4900010 business service makes the necessary calls to the RateWare® XL web methods and returns the freight charges received from RateWare XL to the E1 Enterprise server. The system displays the data in the Work with Routing Options (P4980) and Shipment/Load Charges (P4945) programs; and writes the charges to the Shipment Charges (F4945) table.

Before you use J4900010, you can set business service properties, which the system uses to process user authentication information. You must set up these properties before you can use the business service. This table describes each of the properties that are associated with this business service:

Group	Key	Description	Value
J4900010	J4900010_URL_RWXL_WS	Use this business service property to indicate the value of the url used to access to the RateWare® XL web service.	Provided by SMC ³ .
J4900010	J4900010_LICENSEKEY_WS	Use this business service property to indicate the value of the valid license that gains access to the RateWare® XL web service.	User Defined.
J4900010	J4900010_USERNAME_WS	Use this business service property to indicate the value of the user identification that gains	User defined.

Group	Key	Description	Value
		access to the RateWare® XL web service.	
J4900010	J4900010_PASSWORD_WS	Use this business service property to indicate the value of the password that gains access to the RateWare® XL web service.	User defined.

RateWare XL JAXRPC Processor Service (JC490015)

The RateWare XL JARPC Processor business service (JC490015) runs on the Oracle Weblogic Server (WLS) platform and reads the data stored in the RateWare XL Rate Constants table (F4974). The E1 Enterprise server collects the data required for rating a shipment from the JD Edwards EnterpriseOne tables and then calls the JC490015 business service. The JC490015 business service makes the necessary calls to the RateWare® XL web methods and returns the freight charges received from RateWare XL to the E1 Enterprise server. The system displays the data in the Work with Routing Options (P4980) and Shipment/Load Charges (P4945) programs; and writes the charges to the Shipment Charges (F4945) table.

Before you use JC490015, you can set business service properties, which the system uses to process user authentication information. You must set up these properties before you can use the business service. This table describes each of the properties that are associated with this business service:

Group	Key	Description	Value
JC490015	JC490015_URL_RWXL_WS	Use this business service property to indicate the value of the URL used to access to the RateWare® XL web service.	Provided by SMC ³ .
JC490015	JC490015_LICENSEKEY_WS	Use this business service property to indicate the value of the valid license that gains access to the RateWare® XL web service.	User Defined.
JC490015	JC490015_USERNAME_WS	Use this business service property to indicate the value of the user identification that gains access to the RateWare® XL web service.	User defined.
JC490015	JC490015_PASSWORD_WS	Use this business service property to indicate the value of the password that gains access to the RateWare® XL web service.	User defined.

RateWare XL JAXWS Processor Business Service (JC490020)

The RateWare XL JAXWS Processor business service (JC490020) runs on the Oracle Weblogic Server (WLS) or the Web Application Server (WAS) platform and reads the data stored in the RateWare XL Rate Constants (F4974) table. The E1 Enterprise server collects the data required for rating a shipment from the JD Edwards EnterpriseOne tables and then calls the JC490020 business service. The JC490020 business service makes the necessary calls to the RateWare® XL web methods and returns the freight charges received from RateWare XL to the E1 Enterprise server. The system displays the data in the Work with Routing Options (P4980) and Shipment/Load Charges (P4945) programs; and writes the charges to the Shipment Charges table (F4945).

Before you use JC490020, you can set business service properties, which the system uses to process user authentication information. You must set up these properties before you can use the business service. This table describes each of the properties that are associated with this business service:

Group	Key	Description	Value
JC490020	JC490020_URL_RWXL_WS	Use this business service property to indicate the value of the URL used to access to the RateWare® XL web service.	Provided by SMC ³ .
JC490020	JC490020_LICENSEKEY_WS	Use this business service property to indicate the value of the valid license that gains access to the RateWare® XL web service.	User defined.
JC490020	JC490020_USERNAME_WS	Use this business service property to indicate the value of the user identification that gains access to the RateWare® XL web service.	User defined.
JC490020	JC490020_PASSWORD_WS	Use this business service property to indicate the value of the password that gains access to the RateWare® XL web service.	User defined.

Transportation Shipment Real-Time Events

This section describes the available real-time events associated with the Transportation Shipment business object, which include:

- RTSHPNOUT, which is a container event for these single events:
 - RTSHPNOUTA
 - RTSHPNOUTB
 - RTSHPNOUTC
 - RTSHPNOUTD
 - RTSHPNOUTE
 - RTSHPNOUTF
 - RTSHPNOUTG
 - RTSHPNOUTH
 - RTSHPNOUTI
 - RTSHPNOUTJ
 - RTSHPNOUTK
- RTSHPNOUT2, which is a container event for all of the single events contained within RTSHPNOUT, along with these additional single events:
 - RTSHPNOUTL
 - RTSHPNOUTM
 - RTSHPNOUTP

Note that all of the single events contained within both the RTSHPNOUT and RTSHPNOUT2 are documented under the RTSHPNOUT section.

RTSHPNOUT

RTSHPNOUT is used to inform or notify third-party systems of shipment confirmation in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSHPNOUT	Ship Notice	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTSHPNOUTA (D4901400A) • RTSODTL3 (D4202440B) • RTSHPNOUTB (D4901400B) 	49

Event Type	Event Description	Event Category	Event Aggregate	Product Code
			<ul style="list-style-type: none"> • RTSHPNOUTC (D4901400C) • RTSHPNOUTD (D4901400D) • RTSHPNOUTE (D4901400E) • RTSHPNOUTF (D4901400F) • RTSHPNOUTG (D4901400G) • RTSHPNOUTH (D4901400H) • RTSHPNOUTI (D4901400I) • RTSHPNOUTJ (D4901400J) • RTSHPNOUTK (D4901400K) 	

Conceptual Approach

When a user confirms an outbound shipment within the JD Edwards EnterpriseOne system, the system triggers a notification. The RTSHPNOUT notification event contains eleven separate events:

- RTSHPNOUTA
- RTSHPNOUTB
- RTSHPNOUTC
- RTSHPNOUTD
- RTSHPNOUTE
- RTSHPNOUTF
- RTSHPNOUTG
- RTSHPNOUTH
- RTSHPNOUTI
- RTSHPNOUTJ
- RTSHPNOUTK

Objects that Can Initiate the Event

These objects can initiate the RTSHPNOUT event, and all of the single events contained within this event:

- B4003000 – XPI Distribution Event Notification
- B4003020 - XPI Distribution Event Notification Func 2

Special Setup

To process UCC-128 information, you must set up processing options for P40XPI.

RTSHPNOUTA

RTSHPNOUTA is used to supply shipment information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTA	Ship Notice Details	RTE	Single	49	D4901400A: Shipment Notice Real Time Event Notification	Included in container event RTSHPNOUT.

RTSHPNOUTB

RTSHPNOUTB is used to supply shipment header information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTB	Ship Notice Header	RTE	Single	49	D4901400B: Shipment Notice - Shipment Header	Included in container event RTSHPNOUT.

RTSHPNOUTC

RTSHPNOUTC is used to supply global carrier code information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTC	Ship Notice Global Carrier Cd	RTE	Single	49	D4901400C: Shipment Notice - Global Carrier Code	Included in container event RTSHPNOUT.

RTSHPNOUTD

RTSHPNOUTD is used to supply shipment detail information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTD	Ship Notice Tare Detail	RTE	Single	49	D4901400D: Shipment Notice - Shipment Detail	Included in container event RTSHPNOUT.

RTSHPNOUTE

RTSHPNOUTE is used to supply container order information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTE	Ship Notice Tare Order	RTE	Single	49	D4901400E: Shipment Notice - Shipment Container Order	Included in container event RTSHPNOUT.

RTSHPNOUTF

RTSHPNOUTF is used to supply container item information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTF	Ship Notice Tare Item	RTE	Single	49	D4901400F: Shipment Notice - Shipment Container Item	Included in container event RTSHPNOUT.

RTSHPNOUTG

RTSHPNOUTG is used to supply serial number information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTG	Ship Notice Tare Serial Number	RTE	Single	49	D4901400G: Shipment Notice - Container Item Serial Number	Included in container event RTSHPNOUT.

RTSHPNOUTH

RTSHPNOUTH is used to supply sub-container detail information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTH	Ship Notice Pack Detail	RTE	Single	49	D4901400H: Shipment Notice - SubContainer Detail	Included in container event RTSHPNOUT.

RTSHPNOUTI

RTSHPNOUTI is used to supply sub-container order information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTI	Ship Notice Pack Order	RTE	Single	49	D4901400I: Shipment Notice - SubContainer Order	Included in container event RTSHPNOUT.

RTSHPNOUTJ

RTSHPNOUTJ is used to supply sub-container item information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTJ	Ship Notice Pack Item	RTE	Single	49	D4901400J: Shipment Notice - SubContainer Item	Included in container event RTSHPNOUT.

RTSHPNOUTK

RTSHPNOUTK is used to supply sub-container serial number information to the RTSHPNOUT container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTK	Ship Notice Pack Serial Number	RTE	Single	49	D4901400K: Shipment Notice - SubContainer Item Serial Number	Included in container event RTSHPNOUT.

RTSHPNOUT2

RTSHPNOUT2 is used to inform or notify third-party systems of shipment confirmation in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTSHPNOUT2	Ship Notice Version 2	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTSHPNOUTA (D4901400A) • RTSHPNOUTB (D4901400B) • RTSHPNOUTC (D4901400C) • RTSHPNOUTD (D4901400D) 	49

Event Type	Event Description	Event Category	Event Aggregate	Product Code
			<ul style="list-style-type: none"> • RTSHPNOUTE (D4901400E) • RTSHPNOUTF (D4901400F) • RTSHPNOUTG (D4901400G) • RTSHPNOUTH (D4901400H) • RTSHPNOUTI (D4901400I) • RTSHPNOUTJ (D4901400J) • RTSHPNOUTK (D4901400K) • RTSHPNOUTL (D4901400L) • RTSHPNOUTM (D4901400M) 	

Conceptual Approach

When a user confirms an outbound shipment within the JD Edwards EnterpriseOne system, the system triggers a notification. The RTSHPNOUT2 notification event contains thirteen separate events:

- RTSHPNOUTA
- RTSHPNOUTB
- RTSHPNOUTC
- RTSHPNOUTD
- RTSHPNOUTE
- RTSHPNOUTF
- RTSHPNOUTG
- RTSHPNOUTH
- RTSHPNOUTI
- RTSHPNOUTJ
- RTSHPNOUTK
- RTSHPNOUTL
- RTSHPNOUTM

Objects that Can Initiate the Event

These objects can initiate the RTSHPNOUT2 event, and the single events contained within the event:

- B4003000 - XPI Distribution Event Notification

Special Setup

To process UCC-128 information, you must set up processing options for P40XPI.

RTSHPNOUTL

RTSHPNOUTL is used to supply container item information to the RTSHPNOUT2 container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTL	Ship Notice Tare Item Cty Orig	RTE	Single	49	D4901400L: Shipment Notice - SubContainer Item	Included in container event RTSHPNOUT.

RTSHPNOUTM

RTSHPNOUTM is used to supply sub-container item information to the RTSHPNOUT2 container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTM	Ship Notice Pack Item Cty Orig	RTE	Single	49	D4901400M: Shipment Notice - SubContainer Container Item	Included in container event RTSHPNOUT.

RTSHPNOUTP

RTSHPNOUTP is used to supply shipment header information to the RTSHPNOUT2 container event.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTSHPNOUTP	Ship Notice Header GLN	RTE	Single	49	D4901400P: Shipment Notice - Shipment Header	Included in container event RTSHPNOUT.

Transportation Shipment Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Transportation Shipment business object:

Program and Program Type	Description	Available Documentation
EDI Shipping Notice Edit/Create program (R47031) This is an EDI import program.	Run the EDI Shipping Notice Edit/Create program (R47031) to update the status of a purchase order after you have received the ASN. The system uses the information that is contained	See this topic: <ul style="list-style-type: none"> <i>Receiving Inbound Shipping Notices (856)</i>

Program and Program Type	Description	Available Documentation
	in the F47031 table and the F47032 table to perform the status update.	

Transportation Shipment Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Transportation Shipment business object:

Program and Program Type	Description	Available Documentation
EDI Advance Ship Notice Extraction program (R47032) This is an EDI extract program.	After running the EDI Advance Ship Notice Extraction program (R47032) to update the EDI ASN tables, you can use the Work With Ship Notice program (P47036) to revise all of the shipment-related information for the ASN.	See this topic: <ul style="list-style-type: none"> <li data-bbox="781 940 1068 993">• Understanding Ship Notice Information

73 SCM and Manufacturing: Test Result

Test Result Overview

This chapter provides detailed information about the business interfaces that are available for the Test Result business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Test Result business object:</p> <ul style="list-style-type: none"> Batch Test Results program (R3711Z1I)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Test Result Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Test Result business object:

Program and Program Type	Description	Available Documentation
Batch Test Results program (R3711Z1I)	You run the Batch Test Results program (R3711Z1I) to copy the information from the unedited transaction table to the Test Results table (F3711).	<p>See this topic:</p> <ul style="list-style-type: none"> Understanding Inbound Interoperability for Quality Management

74 SCM and Manufacturing: UOM Conversions

UOM Conversions Overview

This chapter provides detailed information about the business interfaces that are available for the UOM Conversions business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Real-Time Events	<p>This list includes the real-time events for the UOM Conversions business object:</p> <ul style="list-style-type: none"> RTUOMOUT, which is a container event for RTUOMDTL.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

UOM Conversions Real-Time Events

This section describes the available real-time events associated with the Item Master business object, which include:

- RTUOMOUT, which is a container event for RTUOMDTL.

RTUOMOUT

RTUOMOUT is a container event generated by the JD Edwards EnterpriseOne system to notify third-party systems of adds and changes to unit of measure conversions.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTUOMOUT	Item UOM Conversion Master	RTE	<p>Container</p> <p>Container event for RTUOMDTL (D4102170A).</p>	H41

Conceptual Approach

Unit of measure conversion information that is added or modified for a given item in the Unit of Measure table F41002 is sent out in a real-time event.

Special Setup

If the system constant Unit of Measure Conversion by Branch is Off, all unit of measure (UOM) conversions for the item and a blank branch plant are sent in the real-time event.

If the system constant Unit of Measure Conversion by Branch is On and the Default Branch Plant processing option on version ZJDE0001 of UOM Conversion Real Time Wrapper Application (P4102170) is blank, the UOM conversions for the item and all branch plants are sent in the real-time event.

If the system constant Unit of Measure Conversion by Branch is On and the Default Branch Plant processing option on version ZJDE0001 of UOM Conversion Real Time Wrapper Application (P4102170) is 1, the UOM conversions for the item and the current branch plant are sent in the real-time event.

Objects that Can Initiate the Event

The P41002 - Unit of Measure Conversions - Item program can initiate the RTUOMOUT event.

RTUOMDTL

RTUOMDTL is a single event that is generated by the JD Edwards EnterpriseOne system to provide unit of measure conversion information.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTUOMDTL	Item UOM Conversion Details	RTE	Single	H41	D4102170A: UOM Conversion Real Time Data	Included in container event RTUOMOUT.

Conceptual Approach

Unit of measure conversion information added or modified for a given item in the Unit of Measure table F41002 is sent out in a real-time event.

75 SCM and Manufacturing: Warehouse Suggestion

Warehouse Suggestion Overview

This chapter provides detailed information about the business interfaces that are available for the Warehouse Suggestion business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Warehouse Suggestion business object:</p> <ul style="list-style-type: none"> Inbound Suggestion Processor (R4611Z1I)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Warehouse Suggestion Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Warehouse Suggestion business object:

Program and Program Type	Description	Available Documentation
Inbound Suggestion Processor (R4611Z1I)	<p>This program uses interoperability table F4611Z1 to confirm and or cancel existing suggestions through inbound interoperability.</p> <p>Data in the suggestions table (F4611) must be manually populated into the F4611Z1 table using the application P4611Z1.</p> <p>You specify whether the record is confirmed or canceled by populating</p>	<p>Refer to document ID659204.1 in My Oracle Support for additional details.</p> <p>This program can be accessed using fast path BV (batch versions.).</p>

Program and Program Type	Description	Available Documentation
	<p>the Transaction action Field in P4611Z1 with appropriate value.</p> <p>You then run the R4611Z1I to process the suggestions. If an invalid transaction action code is used, the suggestions are not updated.</p>	

76 SCM and Manufacturing: Work Order - Food and Beverage

Work Order - Food and Beverage Overview

This chapter provides detailed information about the business interface components that are available for the Work Order - Food and Beverage business object.

This table lists all of the additional interface components:

Interface Type	Available Interfaces
Batch Import Programs	<p>This list includes the batch import programs for the Work Order - Food and Beverage business object:</p> <ul style="list-style-type: none"> Inbound Work Order for Blend program (R31B65AZ1) Process Inbound Samples for Blend program (R31BSPZ1)
Orchestration	<p>Add Blend Operation</p> <p>For additional information about the Add Blend Operation orchestration, see Add Blend Operation.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Work Order - Food and Beverage Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the Work Order - Food and Beverage batch import programs that are available:

Program and Program Type	Description	Available Documentation
Inbound Work Order for Blend program (R31B65AZ1)	When you run the Inbound Work Order for Blend program (R31B65AZ1), the system validates the records and creates operations. Based on the	<p>See this topic:</p> <ul style="list-style-type: none"> Creating and Modifying Inbound Transactions

Program and Program Type	Description	Available Documentation
This is a temporary table import program.	operation information in the Z file, the system creates a work order and attaches the operation details from the flat file to the work order.	
Process Inbound Samples for Blend program (R31BSPZ1) This is a temporary table import program.	Run the Process Inbound Samples for Blend program (R31BSPZ1) to update barrels with the same sample number in the VBT Detail table (F31B101) with sample information, such as the value from Sampled check box, the sample sequence and the barrel location.	See this topic: <ul style="list-style-type: none"><li data-bbox="781 495 971 520">• <i>Inbound Samples</i>

77 SCM and Manufacturing: Work Order - Manufacturing

Work Order - Manufacturing Overview

This chapter provides detailed information about the business interfaces that are available for the Work Order - Manufacturing business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>These are the published business services available for the Work Order - Manufacturing business object:</p> <ul style="list-style-type: none"> • WorkOrderManager (JP480000), which manages the lookupWorkOrders (J4800001) business service. <p>See <i>lookupWorkOrders</i>.</p> <ul style="list-style-type: none"> • Manufacturing Work Order (JP310000), which manages these business services: <ul style="list-style-type: none"> ○ Work Order Processor (J4800010) Work Order Query Processor (J4800020) Parts List Processor (J3100010) Routing Processor (J3100020) ○ Completion Processor (J3100030)
Real-Time Events	<p>This list includes the real-time events for the Work Order - Manufacturing business object:</p> <ul style="list-style-type: none"> • RTWOOUT, which is a container event for RTWOHDR.
Batch Import Programs	<p>This list includes the batch import programs for the Work Order - Manufacturing business object:</p> <ul style="list-style-type: none"> • Inbound Super Backflush Processor program (R31123Z1I) • Inbound Completion Processor (R31114Z1I) • Inbound Hours and Quantity Processor (R31122Z1I) <p>Note that this program can function as a batch export program as well as an import program.</p> <ul style="list-style-type: none"> • Inbound Inventory Issues Processor (R31113Z1I) <p>Note that this program can function as a batch export program as well as an import program.</p> <ul style="list-style-type: none"> • Process Inbound Work Day Calendar DC program (R0007Z1I)
Batch Export Programs	<p>This list includes the batch export programs for the Work Order - Manufacturing business object:</p>

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> • Outbound Work Order Extraction (R4801Z1X) • Hours and Quantities Update program (R31422) • Order Processing (R31410) <p>Also note that the following programs, documented under the import section of this chapter, can also function as export programs:</p> <ul style="list-style-type: none"> • Inbound Hours and Quantity Processor (R31122Z1I) <p>Note that this program can function as a batch export program as well as an import program.</p> <ul style="list-style-type: none"> • Inbound Inventory Issues Processor (R31113Z1I) <p>Note that this program can function as a batch export program as well as an import program.</p>

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Work Order - Manufacturing Business Services - WorkOrderManager (JP480000)

This section describes the WorkOrderManager (JP480000) published business service, which manages the lookupWorkOrders (J4800001) business service.

This business service is used primarily for Mobile Expense Management functionality. The business service is documented with the mobile processing documentation.

See [lookupWorkOrders](#).

Work Order Business Services - Manufacturing Work Order (JP310000)

This section describes the Manufacturing Work Order published business service, which manages these business services:

- Work Order Processor (J4800010)
- Work Order Query Processor (J4800020)
- Parts List Processor (J3100010)
- Routing Processor (J3100020)

Work Order Processor (J4800010)

Published method processManufacturingWorkOrder (JP310000-J4800010) an inbound transaction operation that enables consumers to add, delete, or change work orders, parts lists and routings in JD Edwards EnterpriseOne.

The operation uses the ProcessWorkOrder business function (B4800470) to process work order information. If the operation encounters errors while processing work order information, processing stops and all errors and warnings are returned to the consumer.

Depending on the published business service that is calling this service, the Work Order Processor can be used to process standard, service, equipment, or tenant work order information.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the Work Order Processor operation uses:

Group	Key	Description	Default Value
J4800010	J4800010_EQUIPMENT_VERSION	Use this property to specify the version of the Manage Equipment Work Orders program (P13714).	ZJDE0001
J4800010	J4800010_MANUFACTURING_VERSION	Use this property to specify the version of Manufacturing Work Order Processing program (P48013).	ZJDE0001
J4800010	J4800010_PREFIX_1	Use this property to specify the prefix for error in Process Work Order wrapper function.	B4800470 Process Work Order:
J4800010	J4800010_SERVICE_VERSION	Use this property to specify the default version of the Manage Service Orders program (P90CD0002).	ZJDE0001
J4800010	J4800010_TENANT_VERSION	Use this property to specify the default version of the Tenant Work Order Entry program (P15248).	ZJDE0001

Implementation Details

This table includes information that can help determine whether the Work Order Processor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the service returns a message to the user with the processed data.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	Yes.

Work Order Query Processor (J4800020)

The Work Order Query Processor is used to retrieve all manufacturing work order records from the F4801, F4801T, and the associated records from the Parts List table F3111 and the Routing table F3112. If the service encounters errors, a message containing the text of any errors or warnings is sent to the consumer.

Depending on the published business service that is calling this service, the Work Order Query Processor can be used to query standard, service, equipment, or tenant work order information.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the Work Order Query Processor operation uses:

Group	Key	Description	Default Value
J4800020	J4800020_MAX_ROWS	Use this property to specify the maximum number of rows that can be returned by the query.	100

Implementation Details

This table includes information that can help determine whether the Work Order Query Processor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the service returns the results of the query.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation queries the EnterpriseOne database. No updates are made to data in the database.
Does this operation use record reservation?	No.

Parts List Processor (J3100010)

The Parts List Processor is used to process work order parts lists. This is called after the Manufacturing WO header has been processed, to add, update and delete parts lists. If the system encounters errors during processing, the service returns any error messages or warnings to the consumer.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the Parts List Processor operation uses:

Group	Key	Description	Default Value
J3100010	J3100010_PREFIX_1	Use this property to specify the prefix for errors in the Process Parts List wrapper function.	B3103140 Process Parts List - Line #

Implementation Details

This table includes information that can help determine whether the Parts List Processor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the service returns a confirmation message with the results.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation uses standard transaction processing. If the system encounters errors during processing, no data in the JD Edwards EnterpriseOne system is updated. No manual updates are required.
Does this operation use record reservation?	Yes.

Question	Answer

Routing Processor (J3100020)

The Routing Processor is used to process work order routings. This is called after the manufacturing work order header has been processed, to add, update and delete routing.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the Routing Processor operation uses:

Group	Key	Description	Default Value
J3100020	J3100020_PREFIX_1	Use this property to specify the prefix for errors in the Commit Routing wrapper function.	B3103150 Commit Routing

Implementation Details

This table includes information that can help determine whether the Routing Processor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, the service returns a confirmation message to the user.
If I encounter errors while process a transaction, do I need to reverse the transaction?	This operation queries the EnterpriseOne database. No updates are made to data in the database.
Does this operation use record reservation?	No.

Completion Processor (J3100030)

The Completion Processor is an inbound processor service that is used to process work order completions, and enables users to complete a manufacturing work order. Completions is the final step of a work order where the finished goods

are completed into inventory. The Completion Processor calls the Process Completion business function (B3103160), which then calls the W.O. Completions MBF (B3100770).

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes equipment data. This table includes information about the business service properties that the Routing Processor operation uses:

Group	Key	Description	Default Value
J3100030	J3100030_COMPLETION_VERSION	Use this property to specify the version of the Work Order Completions program (P31114) to use.	ZJDE0001
J3100030	J3100030_PREFIX_1	Use this option to specify the prefix for error in the Process Completion wrapper function.	B3103160 Process Completion - Location #

Implementation Details

This table includes information that can help determine whether the Completion Processor operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	If the operation completes successfully, a response xml message is returned showing the result of the processing, along with any warnings. If there is an error, the error text is returned, but the processing data is not returned, as the process did not complete successfully.
If I encounter errors while process a transaction, do I need to reverse the transaction?	No. If the process ends in error, the data being sent or the Completion configuration should be corrected before running the process again.
Does this operation use record reservation?	Yes.

Work Order - Manufacturing Real-Time Events

This section describes the available real-time events associated with the Work Order - Manufacturing business unit, which include:

- RTWOOUT, which is a container event for RTWOHDR.

RTWOOUT

RTWOOUT is used by the JD Edwards EnterpriseOne system to inform third-party systems when a work order has been added, changed, or deleted in order to keep the two systems synchronized.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTWOOUT	Manufacturing Work Order	RTE	Container Contains the RTWOHDR (D3102360A) event.	H31	D3102360A: Work Order Event Notification Canonical

Conceptual Approach

A work order that has been added, changed, or deleted in the JD Edwards EnterpriseOne system updates the Work Order Header (F4801) table. When a work order record has been modified, a RTWOOUT real-time event notification is published.

Objects that Can Initiate the Event

These objects can initiate the RTWOOUT event, and the single events contained within the event:

- B3101030 - Purge Work Order Data
- N4202020 - F4211 Update Work Order Status
- X4801 - X4801 Work Order Entry
- N3104330 – Work Order Status Update With Interop
- B3100770 – Work Order Completions MBF

RTWOHDR

RTWOHDR is used by the JD Edwards EnterpriseOne system to pass work order information to third-party systems when a work order has been changed.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTWOHDR	Manufacturing WO Header	RTE	Single	H31	D3102360A: Work Order Event Notification Canonical	Included in container event RTWOOUT.

Conceptual Approach

This real-time event is triggered when a work order header is changed.

Work Order - Manufacturing Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Work Order - Manufacturing business object:

Program and Program Type	Description	Available Documentation
<p>Inbound Super Backflush Processor program (R31123Z11)</p> <p>This is an interoperability import program.</p>	<p>You run the Inbound Super Backflush Processor program (R31123Z11) to copy the information from the F3112Z1 table to the related application tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Super Backflush Inbound Transactions
<p>Inbound Completion Processor (R31114Z11)</p> <p>This is an interoperability import program.</p>	<p>You run the Inbound Completion Processor program (R31114Z11) to copy the information from the F4801Z1 table to the related application tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding Inbound Interoperability for Shop Floor Management
<p>Inbound Hours and Quantity Processor (R31122Z11)</p> <p>Note that this program can function as a batch export program as well as an import program.</p>	<p>The interoperability interface table for hours and quantities inbound transactions is the Work Order Time Transactions Unedited Transaction table (F31122Z1).</p> <p>You run the Inbound Hours and Quantity Processor program (R31122Z11) to copy the information from the F31122Z1 table to the related application tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding Inbound Interoperability for Shop Floor Management
<p>Inbound Inventory Issues Processor (R31113Z11)</p> <p>Note that this program can function as a batch export program as well as an import program.</p>	<p>The interoperability interface table for inventory issues inbound transactions is the Outbound Work Order Parts List table (F3111Z1).</p> <p>You run the Inbound Inventory Issues Processor program (R31113Z11) to copy the information from the F3111Z1 table to the related application tables.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding Inbound Interoperability for Shop Floor Management
<p>Process Inbound Work Day Calendar DC program (R0007Z11)</p>	<p>You run the Process Inbound Work Day Calendar DC program (R0007Z11) to copy the information from the F0007Z1 table to the Workday Calendar table (F0007).</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Inbound Work Day Calendar Data

Program and Program Type	Description	Available Documentation
This is an interoperability program.		

Work Order - Manufacturing Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Work Order - Manufacturing business object:

Program and Program Type	Description	Available Documentation
Outbound Work Order Extraction (R4801Z1X) This is an interoperability export program.	Use this program to send work order transactions to an external system.	See this topic: <ul style="list-style-type: none"> <i>Understanding Outbound Interoperability for Shop Floor Management</i>
Hours and Quantities Update program (R31422)	When you change the status of a routing operation, the change does not take effect until you run the Hours and Quantities Update program (R31422) to update the Work Order Routing table (F3112).	See this topic: <ul style="list-style-type: none"> <i>Understanding Hours and Quantities</i>
Order Processing (R31410)	After you create configured item work order headers through an order entry program such as Sales Order Entry (P4210) or Manufacturing Work Order Processing (P48013), you must run Order Processing (R31410) to generate the work order parts list, routing instructions, commit inventory, and back schedule configured routings.	See this topic: <ul style="list-style-type: none"> <i>Work Order Processing for Configured Items</i>

78 SCM and Manufacturing: Additional Interface Components

SCM Additional Components Overview

This chapter provides detailed information about the business interface components that are available for the Supply Chain Management (SCM) product area, but are not contained within a specified business object.

This table lists all of the additional interface components:

Interface Type	Available Interfaces
Business Services	The IntegrationTimeStampManager PBSSV manages the processing of the ProcessIntegrationTimeStamp (J3000030) business service.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

SCM Business Services - IntegrationTimeStampManager

The IntegrationTimeStampManager PBSSV manages the processing of the ProcessIntegrationTimeStamp (J3000030) business service.

The IntegrationTimeStampManager PBSSV calls the ProcessIntegrationTimeStamp web service. This processor adds and modifies the Integration Time Stamp table (F0095) with the last successful runtime. See *Business Services for Pre-Built Integrations*.

79 Supply Management: Purchase Orders

Purchase Orders Overview

This chapter provides detailed information about the business interfaces that are available for the Purchase Orders business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The ProcurementManager web service (JP430000) manages the processing of these procurement-related web service operations:</p> <ul style="list-style-type: none"> • processPurchaseOrder (J4300010) • processPurchaseOrderV2 (J4300010) • processPurchaseOrderAcknowledge (J4300020) • getPurchaseOrder (J4300030) • getPurchaseOrderV2 (J4300030)
Real-Time Events	<p>This list includes the real-time events for the Purchase Orders business object:</p> <ul style="list-style-type: none"> • RTPOOUT, which is a container event for: <ul style="list-style-type: none"> ○ RTPOHDR ○ RTPODTL ○ RTPODIST • RTPOOUT2, which is a container event for: <ul style="list-style-type: none"> ○ RTPOHDR2 ○ RTPODTL2 ○ RTPODIST* ○ RTPOHLDOUT <p>*Note that RTPODIST is contained within this event, and also the RTPOOUT event. RTPODIST is documented under the RTPOOUT section of this chapter.</p>
Batch Import Programs	<p>This list includes the batch import programs for the Purchase Orders business object:</p> <ul style="list-style-type: none"> • Inbound Purchase Order (R4311Z11) • EDI Inbound PO Change Acknowledgment program (R47141) • EDI Inbound PO Acknowledgment program (R47021)
Batch Export Programs	<p>This list includes the batch export programs for the Purchase Orders business object:</p>

Interface Type	Available Interfaces
	<ul style="list-style-type: none"> • EDI Purchase Order Change Extraction program (R47132) • Purchase Order Print program (R43500) • Open Purchase Orders (R43632) • EDI RFQ Extraction (R47102)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Purchase Orders Business Services - ProcurementManager

The ProcurementManager web service (JP430000) manages the processing of procurement-related web service operations. This table includes a description of the procurement web service operations:

Operation	Description
processPurchaseOrder (J4300010)	Use this operation to add, change or cancel a purchase order within the JD Edwards EnterpriseOne Procurement system.
processPurchaseOrderV2 (J4300010)	Use this operation to add, update, or delete a purchase order, along with global locator number (GLN), within the JD Edwards EnterpriseOne Procurement system.
processPurchaseOrderAcknowledge (J4300020)	Use this operation to synchronize sales orders in the source system to the corresponding purchase orders in the JD Edwards EnterpriseOne Procurement system.
getPurchaseOrder (J4300030)	Use this operation to query the JD Edwards EnterpriseOne database for existing purchase order information.
getPurchaseOrderV2 (J4300030)	Use this operation to query the JD Edwards EnterpriseOne database for existing purchase order information, including GLN.
Additional methods for Mobile Purchase Order Approval.	<p>Several additional methods were created to support the Mobile Purchase Order Approval process. These methods are specific to the mobile approval process.</p> <p>See <i>Purchase Order Approvals Business Services</i>.</p>

Accessing Javadoc for Business Services

To access Javadoc for the Procurement Manager web service and its related operations, review these Javadoc packages:

- JP430000 (ProcurementManager)
- J4300010 (processPurchaseOrder)
- J4300010 (processPurchaseOrderV2)
- J4300020 (processPurchaseOrderAcknowledge)
- J4300030 (getPurchaseOrder)
- J4300030 (getPurchaseOrderV2)

Prerequisites

Before using the Procurement Manager web service, or any of the related web service operations, you must install and configure the JD Edwards EnterpriseOne Procurement system.

See [JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide](#) .

processPurchaseOrder

The processPurchaseOrder web service operation is an inbound transaction operation that enables consumers to process purchase order information within the JD Edwards EnterpriseOne system. The consumer can add, change or cancel a purchase order in the JD Edwards EnterpriseOne Procurement system.

If the operation is successful, the system returns a confirmation message which includes purchase order data to the consumer. If the operation fails, the system returns an error message to the consumer.

Supported Functionality

This section discusses the functionality that the processPurchaseOrder operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The processPurchaseOrder operation supports add, change or cancel of purchase orders, blanket orders, quote orders, requisition orders, and subcontracts. In Change mode the processPurchaseOrder operation supports add, change or cancel of detail lines for purchase orders, blanket orders, quote orders, requisition orders, and subcontracts.

The processPurchaseOrder operation supports retrieval of the alternate tax rate/area assignment if the functionality is set up for the entities in a transaction.

See ["Purchase Orders" in the JD Edwards EnterpriseOne Applications Tax Processing Implementation Guide](#) .

The processPurchaseOrder web service operation does not support:

- Requisition self service.

- Purchase order generation.
- Suffix input at the header level.

Although the suffix field is supported at the header level and is part of the key in JD Edwards World, the value in the suffix field for purchase orders in JD Edwards EnterpriseOne is always **000**. By preventing input of the suffix at the header level, the integration may not be compatible when integrating with JD Edwards World.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes purchase order information. This table includes information about the business service properties that the processPurchaseOrder web service operation uses:

Group	Key	Description	Default Value
J4300010	J4300010_PO_MBF_VERSION	Use this business service property to specify which version of the Purchase Orders program (P4310) the operation uses. <i>See "Setting Processing Options for Purchase Orders (P4310)" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide .</i>	ZJDE0001
J4300010	J4300010_BYPASS_BSFN_WARNINGS	Use this business service property to specify whether the system ignores warnings during processing.	1 Note: The system does not treat warnings as errors so the system keeps processing.
J4300010	J4300010_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages on purchase order detail lines.	Line No. Sent in:

Implementation Details

The following table includes information that can help determine whether the processPurchaseOrder operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentCompany • documentTypeCode • documentSuffix • purchaseOrderLineNumber <p>If the operation completes successfully, the system returns the confirm value object and no exception is thrown.</p>
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.</p>
<p>Does this operation use record reservation?</p>	<p>This operation calls the PurchaseOrderInit business function (B4305320). This function reserves records in the JD Edwards EnterpriseOne system when the operation is changing or deleting existing data.</p>

processPurchaseOrderV2

Review the information in the processPurchaseOrder section of this chapter before using the processPurchaseOrderV2 web service operation. All of the information in the processPurchaseOrder section also applies to the processPurchaseOrderV2 operation.

The processPurchaseOrderV2 operation is a version of the processPurchaseOrder web service operation. This web service operation is used to add, update, and delete purchase order information along with associated GLNs. This version executes the processPurchaseOrder (J4300010) web service operation.

processPurchaseOrderAcknowledge

The processPurchaseOrderAcknowledge web service operation is an inbound transaction operation that enables consumers to process purchase order acknowledgements within the JD Edwards EnterpriseOne system. The consumer can synchronize sales orders in the source system to the corresponding purchase orders in the JD Edwards EnterpriseOne Procurement system.

If the operation is successful, the system returns a confirmation message which includes purchase order acknowledgement data to the consumer. If the operation fails, the system returns an error message to the consumer.

If the operation is successful, the system returns purchase order header and line keys to the caller through the ConfirmProcurementOrderAcknowledge web service operation. The system returns exceptions to the caller. The minimum required fields to acknowledge a purchase order in JD Edwards EnterpriseOne are:

- Header
 - statusOrderCode
 - purchaseOrderKey
- Detail (optional)
 - statusOrderCode
 - purchaseOrderLineKey

The processPurchaseOrderAcknowledge operation enables the consumer to specify the statusOrderCode at the header and detail level. The statusOrderCode indicates the type of operation that is being performed on the purchase order and is required. The status order codes are:

1: Accept

2: Accept with change

3: Pending

4: Product already shipped

5: Reject

This table includes information about the JD Edwards EnterpriseOne purchase order acknowledgement scenarios used by the processPurchaseOrderAcknowledge operation:

Status Order Code in Header	Status Order Code in Detail	Comment	Result
1	1	Accept	Only the detail status is changed and no data is updated.
1	2	Accept	The detail status is changed and other data is updated.
2	1	Accept	Only the detail status is changed and the data in the header is updated.
2	2	Accept with change	The detail status is changed, and the header and detail data is updated.
3	3	Pending	The status is changed to Pending as set in the processing options.

Status Order Code in Header	Status Order Code in Detail	Comment	Result
4	4	Product already shipped	There is no change in the EnterpriseOne system.
5	1, 2, 3, 4, 5	Reject	All the lines are rejected.
1	3		The header is not updated, the status is changed to Pending as set in the processing options, and the VR01 field in the detail is changed.
2	3		The header data is updated, the status is changed to Pending as set in the processing options, and the VR01 field in the detail is changed.
2	4		The header data is updated but no fields in the detail are updated.
2	5		The header data is updated and the status is changed.
3	2		The header is not updated, the detail status is changed, and other data is updated.
4	5		The data in the header is not updated and the status fields are changed.

Prerequisites

Before using the processPurchaseOrderAcknowledge web service operation, you must:

- Set the 3. Acknowledged Order Status Code, 4. Acknowledged With Change Order Status Code, and 5. Pending Order Status Code processing options on the Order Statuses tab of the Purchase Order XPI program (P43XPI). See *Setting Processing Options for Purchase Order XPI (P43XPI)*.

- Set the Order Type to **OP** and Line Type for stock, non-stock or service lines in the Order Activity Rules from the Procurement System Setup menu (G43A41).

See *"Setting Up Order Activity Rules" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide* .

Supported Functionality

This section discusses the functionality that the processPurchaseOrderAcknowledge operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The processPurchaseOrderAcknowledge operation enables a purchase order to be acknowledged, acknowledged and changed, appended, or rejected. The operation also supports sales order line splits. When the operation sends line split information to the JD Edwards EnterpriseOne system, it must be sent as an acknowledge with change so that a line is added to the purchase order. Two lines must be sent to split a line, both as an acknowledge with change. If a related sales order is not added, the operation performs a call in rejection mode and the corresponding JD Edwards EnterpriseOne purchase order is cancelled.

The processPurchaseOrderAcknowledge web service operation does not allow the status of cancelled lines to change.

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes purchase order acknowledgment information. This table includes information about the business service properties used by the processPurchaseOrderAcknowledge operation:

Group	Key	Description	Default Value
J4300020	J4300020_PO_MBF_VERSION	Use this business service property to specify which version of the Purchase Orders program (P4310) the operation uses. See <i>"Setting Processing Options for Purchase Orders (P4310)" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide</i> .	ZJDE0001
J4300020	J4300020_BYPASS_BSFN_WARNINGS	Use this business service property to specify whether the system ignores warnings during processing.	1 The system does not treat warnings as errors so the system keeps processing.

Group	Key	Description	Default Value
J4300020	J4300020_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages on purchase order detail lines.	Line No. Sent in:

Implementation Details

The following table includes information that can help determine whether the processPurchaseOrderAcknowledge operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentCompany • documentTypeCode • documentLineNumber • documentSuffix <p>If the operation completes successfully, the system returns the confirm value object and no exception is thrown.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.</p>
Does this operation use record reservation?	<p>This operation calls the PurchaseOrderAcknowledgeNotify business function (B4302190). This function reserves records in the JD Edwards EnterpriseOne system when the operation synchronizes sales orders in the source system with their corresponding purchase orders in the JD Edwards EnterpriseOne Procurement system.</p>

getPurchaseOrder

The getPurchaseOrder web service operation is a database query operation that enables consumers to query the JD Edwards EnterpriseOne Procurement system to retrieve existing purchase order information.

If the operation is successful, the system returns zero to many records to the consumer. You can specify the maximum number of records to return during a query using the Max Rows business service property. If the operation fails, the system returns an error message to the consumer.

The database query operation uses the V4301XPI business view which returns only the fields from the Purchase Order Header (F4301) and Purchase Order Detail File (F4311) tables that are needed for processing. Use of the business view allows retrieval of all required fields in a single optimized call.

Field retrieval is based on the selection criteria specified in the value object. The query published value object contains fields which filter the purchase order information retrieved by the JD Edwards EnterpriseOne operation.

This flow supports querying purchase order information in JD Edwards EnterpriseOne. The `getPurchaseOrder` web service operation enables source systems to query JD Edwards EnterpriseOne purchase order information in a real-time fashion. The `selectPurchaseOrder` adapter service is invoked to retrieve purchase order information from the V4301XPI business view in JD Edwards EnterpriseOne based on the selection criteria specified in the `GetPurchaseOrder` published interface document. Retrieved results are grouped together with header and detail information and mapped to the `ShowPurchaseOrder` published interface document and delivered back to the source application.

The query published interface contains fields that can be used to filter the purchase order information that is retrieved. At minimum, any of the purchase order keys (`documentNumber`, `documentCompany`, or `documentType`) or the item (short item number), or any combination of these fields, is required to process the `getPurchaseOrder` integration.

Optional fields to include in the filter are:

- `BusinessUnit`
- `CurrencyCode`
- `Buyer`
- `Supplier`
- `ShipTo`
- `PaymentTermsCode`

Note: Preceding spaces are already padded to the `businessUnit` value in the `getPurchaseOrder` web service operation. The JD Edwards EnterpriseOne date format is used to query purchase orders based on dates in the `getPurchaseOrder` operation.

Supported Functionality

This section discusses the functionality that the `getPurchaseOrder` operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The `getPurchaseOrder` web service operation enables consumers to query the JD Edwards EnterpriseOne Procurement system to retrieve existing purchase order information.

The `getPurchaseOrder` operation does not support the wildcard (asterisk (*)) for search criteria.

Setup Considerations

Before you use this operation, you can set business service properties to specify how many records the system returns when you query the JD Edwards EnterpriseOne database. This table includes information about the business service properties used by the `getPurchaseOrder` operation:

Group	Key	Description	Default Value
J4300030	J4300030_MAX_ROWS	Use this business service property to define the maximum number of rows that the operation returns when querying the JD Edwards EnterpriseOne database.	100

Note: It is strongly recommended that you set this business service property to a value other than **0** (zero). If you leave this value set to **0**, the system returns all matching records. Additionally, it is recommended that you specify selection criteria when you query the JD Edwards EnterpriseOne database. The system returns an error if you perform a query with no selection criteria and this business service property is set to **0**.

Implementation Details

The following table includes information that can help determine whether the `getPurchaseOrder` operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If the operation completes successfully, the system returns records that match your search criteria. These records include all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, if the query finds matching records, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany • orderSuffix • businessUnit • orderedBy • orderTakenBy • documentLineNumber • documentSuffix • itemId • itemProduct • itemCatalog <p>If a business service exception is not thrown, then the operation completed successfully.</p>
If I encounter errors while processing a transaction, do I need to reverse the transaction?	<p>This is a database query operation that does not perform transactions.</p> <p>If you encounter errors during processing, review your search criteria and business service property settings and try your query again. If the operation does not return any records, verify that records matching your query exist in the JD Edwards EnterpriseOne database.</p>

Question	Answer
Does this operation use record reservation?	No. This operation does not reserve records that are returned in a query.

getPurchaseOrderV2

Review the information in the getPurchaseOrder section of this chapter before using the getPurchaseOrderV2 web service operation. All of the information in the getPurchaseOrder section also applies to the getPurchaseOrderV2 operation.

The getPurchaseOrderV2 operation is a version of the getPurchaseOrder web service operation. This web service operation is used to query purchase order information along with associated GLNs. This version executes the getPurchaseOrder (J4300030) web service operation.

Purchase Orders Real-Time Events

This section describes the available real-time events associated with the Purchase Orders business until, which include:

- RTPOOUT, which is a container event for:
 - RTPOHDR
 - RTPODTL
 - RTPODIST
- RTPOOUT2, which is a container event for:
 - RTPOHDR2
 - RTPODTL2
 - RTPODIST*
 - RTPOHLDOUT

*Note that RTPODIST is contained within this event, and also the RTPOOUT event. RTPODIST is documented under the RTPOOUT section of this chapter.

RTPOOUT

RTPOOUT is a container event used by the JD Edwards EnterpriseOne system that notifies third-party systems when purchase order information changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTPOOUT	Purchase Order	RTE	Container Contains events for: <ul style="list-style-type: none"> • RTPOHDR (D4302470A) • RTPODTL (D4302470B) • RTPODIST (D4302470C) RTPODIST is included in RTPOOUT if the multiple accounts flag is selected for the order line.	H43

Conceptual Approach

A new purchase order creates a new record in the Purchase Order Header table (F4301) and the Purchase Order Detail table (F4311). It also creates a new record in Purchase Order Multiple Accounts (F4316) file multiple account records are associated with the purchase order. A real-time notification is triggered whenever there is an insert in these tables. Also, this event is triggered whenever an exiting purchase order is updated and there is an update to the Purchase Order Header and Purchase Order Detail tables.

Objects that Can Initiate the Event

These objects can initiate the RTPOOUT event and the single events contained within the event:

- N4301780 - Update Promised Date – Workflow
- B4300330 - Update PO Next Status
- B4302153 - Purchase Order Real Time Wrapper Function 3
- B4302160 - Purchase Order External Publish Function
- B4302590 - Update Order Status F4301 OSTs
- B4305420 - Purchase Order Reject Approve
- XT4311Z1 - Purchase Order Entry
- P43081 - Purchase Order Approval
- P4310 - Purchase Orders
- P4312 - PO Receipts
- P43360 - Release Open Quotations
- R00993 - Create F43199 commitment Audit Trail
- R11843 - Load Domestic currency codes - Procurement
- R4317 - Encumbrance Rollover
- R43960 - Close Text Lines
- N4002540 - F4211 Direct Ship Order Integrity

- B4300230 - Update PO Order Total
- B4300360 - Update Purchase Blanket Order
- B4301230 - F4311 Update Original PO Line
- B4302152 - Purchase Order Real Time Wrapper Function 2
- B4302154 - Purchase Order Real Time Wrapper Function 4
- B4302155 - Purchase Order Real Time Wrapper Function 5
- B4302640 - Receipts Blanket Order Processing
- XT4312Z1 - Purchase Order Receipts
- P4312 - PO Receipts

Special Setup

To activate the RTPOOUT event, these three important areas are required for data setup:

- The Suppliers Correspondence Method (F0401.CRMD, sometimes called Send Method) must be set to 5.
- The Document Type of the purchase order must be set up to publish externally. Use the P40040 application to inquire on a document type and select the publish externally check box on the Purchasing tab of the Document Type Revisions form.
- Currently all procurement integrations that need processing options use the processing options for P43XPI-ZJDE0001. This event uses the first two processing options on the Order Statuses tab. With these options you can set the range of allowed order statuses that can be published externally. Order Status 999 is valid outside of the dictated range. The canonical has a grouping titled Miscellaneous, which is inside the Header grouping. Within this grouping there is an element named ExternalEvent. This element is set to **1** if these three conditions are met when the purchase order is being processed.

RTPOHDR

RTPOHDR is used by the JD Edwards EnterpriseOne system to notify third-party systems about purchase order changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPOHDR	Purchase Order Header	RTE	Single	H43	D4302470A: Purchase Order Header	Included in container event RTPOOUT.

Conceptual Approach

This real-time event is contained in the RTPOOUT event and is triggered whenever that event is triggered. The RTPOHDR event contains information from the Purchase Order Header table (F4301).

RTPODTL

RTPODTL is used by the JD Edwards EnterpriseOne system to notify third-party systems about purchase order changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPODTL	Purchase Order Detail	RTE	Single	H43	D4302470B: Purchase Order Detail	Included in container event RTPOOUT.

Conceptual Approach

This real-time event is contained in the RTPOOUT event and is triggered whenever that event is triggered. The RTPOHDR event contains information from the Purchase Order Detail table (F4311).

RTPODIST

RTPODIST is used by the JD Edwards EnterpriseOne system to notify third-party systems when purchase order header and detail information is created or changed.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPODIST	PO multiple accounts	RTE	Single	43	D4302470C: Purchase Order Multiple Accounts	Included in container event RTPOOUT.

Conceptual Approach

This real-time event is contained in the RTPOOUT event and is triggered whenever that event is triggered. The RTPOHDR event contains information from the Purchase Order Multiple Accounts table (F4316).

RTPOOUT2

RTPOOUT2 is a container event used by the JD Edwards EnterpriseOne system that notifies third-party systems when purchase order information changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code
RTPOOUT2	Purchase Order Version 2	RTE	Container Contains events for: <ul style="list-style-type: none"> RTPOHDR2 (D4302471A) RTPODTL2 (D4302471B) 	H43

Conceptual Approach

A new purchase order creates a new record in the Purchase Order Header table (F4301) and the Purchase Order Detail table (F4311). It also creates a new record in Purchase Order Multiple Accounts (F4316) file multiple account records are associated with the purchase order. A real-time notification is triggered whenever there is an insert in these tables. Also, this event is triggered whenever an exiting purchase order is updated and there is an update to the Purchase Order Header and Purchase Order Detail tables.

Objects that Can Initiate the Event

These objects can initiate the RTPOOUT2 event, and the single events contained within the event:

- N4900570 - Update F4211 and F49211 with Shipment Number
- N4900640 - F4215 UnApprove Shipment
- N4900330 - F4215 Approve Shipment
- B4300250 - Release Held Order
- B4300330 - Update PO Next Status
- B4302150 - Purchase Order Real Time Wrapper Function 0
- B4302151 - Purchase Order Real Time Wrapper Function 1
- B4302160 - Purchase Order External Publish Function
- XT4311Z1 - Purchase Order Entry
- XT4312Z1 - Purchase Order Receipts
- P4310 - Purchase Orders
- P4312 - PO Receipts
- P4314 - Voucher match
- P43360 - Release Open Quotations
- R00993 - Create F43199 commitment Audit Trail
- R11843 - Load Domestic currency codes – Procurement
- R43500 – Purchase Order Print
- R43510 – Purchase Receiver Print
- R44H6008 – Print Purchase Order
- R47021 – EDI Inbound P.O. Acknowledgement
- R47132 – EDI Purchase Order Change Extraction
- R47141 - EDI Inbound P.O. Change Acknowledgment

Special Setup

To activate the RTPOOUT2 event, these three important areas are required for data setup:

- The Suppliers Correspondence Method (F0401.CRM, sometimes called Send Method) must be set to 5.
- The Document Type of the purchase order must be set up to publish externally. Use the P40040 application to inquire on a document type and select the publish externally check box on the Purchasing tab of the Document Type Revisions form.
- Currently all procurement integrations that need processing options use the processing options for P43XPI-ZJDE0001. This event uses the first two processing options on the Order Statuses tab. With these options you can set the range of allowed order statuses that can be published externally. Order Status 999 is valid outside of the dictated range. The canonical has a grouping titled Miscellaneous, which is inside the Header grouping. Within this grouping there is an element named ExternalEvent. This element is set to **1** if these three conditions are met when the purchase order is being processed.

RTPOHDR2

RTPOHDR2 is used by the JD Edwards EnterpriseOne system to notify third-party systems about purchase order changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPOHDR2	PO Header Version 2	RTE	Single	H43	D4302471A: Purchase Order Header	Included in container event RTPOOUT.

Conceptual Approach

This real-time event is contained in the RTPOOUT2 event and is triggered whenever that event is triggered. The RTPOHDR2 event contains information from the Purchase Order Header table (F4301).

RTPODTL2

RTPODTL2 is used by the JD Edwards EnterpriseOne system to notify third-party systems about purchase order changes.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure	Event Relationship
RTPODTL2	PO Detail Version 2	RTE	Single	H43	D4302471B: Purchase Order Detail	Included in container event RTPOOUT2.

Conceptual Approach

This real-time event is contained in the RTPOOUT2 event and is triggered whenever that event is triggered. The RTPODTL2 event contains information from the Purchase Order Detail table (F4311).

RTPOHLDOUT

RTPOHLDOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems when a purchase order hold is released.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTPOHLDOUT	PO Hold Status	RTE	Single	43	D4302700B: PO Hold Status Real Time Event

Conceptual Approach

When a held purchase order is released from the Held Order Release application (P43070) in the JD Edwards EnterpriseOne system, the Release Held Order business function (B4300250) is called to removed the order hold code from the Purchase Order Header table (F4301) and the held order is released in the JD Edwards EnterpriseOne system. After the order hold code is removed from the F4301 table, the PO Hold Status Real Time Event Notification business function (B4302700) is called in B4300250 to generate a real-time notification event (PO Hold Status Real Time Event). This event contains purchase order key, PO business unit, and a hold status flag that indicates the held order is released.

Objects that Can Initiate the Event

The Release Held Orders program (P4307070) can initiate the RTPOHLDOUT event.

Purchase Orders Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Purchase Orders business object:

Program and Program Type	Description	Available Documentation
Inbound Purchase Order (R4311Z1I)	You run the Inbound Purchase Order program (R4311Z1I) to copy the information from these unedited transaction tables to the Purchase Order Header table (F4301) and the Purchase Order Detail File table (F4311).	See this topic: <ul style="list-style-type: none"> Processing Inbound Interoperability for Procurement
EDI Inbound PO Change Acknowledgment program (R47141)	Run the EDI Inbound PO Change Acknowledgment program (R47141) for purchase order change acknowledgments (865/ORDRSP) to	See this topic:

Program and Program Type	Description	Available Documentation
This is an EDI import program.	receive acknowledgments of changes to an existing purchase order.	<ul style="list-style-type: none"> Understanding Inbound PO Change Acknowledgments (865/ORDRSP)
EDI Inbound PO Acknowledgment program (R47021)	<p>Run the EDI Inbound PO Acknowledgment program (R47021) from the Purchase Order menu to receive purchase order acknowledgments from vendors. When you run EDI Inbound PO Acknowledgment for purchase order acknowledgments, the program:</p> <ul style="list-style-type: none"> Changes the status of a purchase order based on the way processing options are set. Acknowledges vendor receipt of the order. Generates a discrepancy report that lists orders in which the acknowledgment does not match the original order. 	<p>See this topic:</p> <ul style="list-style-type: none"> Understanding Inbound PO Acknowledgments (855/ORDRSP)

Purchase Orders Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Purchase Orders business object:

Program and Program Type	Description	Available Documentation
EDI Purchase Order Change Extraction program (R47132)	Run the EDI Purchase Order Change Extraction program (R47132) from the Purchase Order Change menu to generate EDI purchase order change transactions (860/ORDCHG). The system sends these transactions to their trading partner to communicate changes that you have made to a purchase order.	<p>See this topic:</p> <ul style="list-style-type: none"> Understanding Outbound Purchase Order Changes (860/ORDCHG)
Purchase Order Print program (R43500)	If you have set up your system to use Direct Connect and PO Dispatch, when you click the Dispatch Order button, or run the R43500, the system calls the PO Dispatch Processor business service (J43E0030), which translates the data into cXML 1.2.016 format and	<p>See this topic:</p> <ul style="list-style-type: none"> Dispatching Purchase Orders using PO Dispatch

Program and Program Type	Description	Available Documentation
	posts that cXML message so that it can be received by the vendor's website.	
EDI RFQ Extraction (R47102). This is an EDI program.	This batch program extracts records from the Sales Order tables and creates Responses to Requests for Quotes within the EDI outbound tables, which can be sent to the trading partner. When EDI Response to RFQ Extraction (R47102) program is processed for sales orders, the program extracts data from tables F4201 and F4211 and creates records in these EDI outbound interface tables: <ul style="list-style-type: none"> • F47106 - EDI Response to RFQ Header - Outbound • F471061 -EDI Response to RFQ Additional Header - Outbound • F47107 -EDI Response to RFQ Detail - Outbound • F471071 -EDI Response to RFQ Additional Detail - Outbound • F4714 - EDI Order Text Lines - Header • F4715 - EDI Order Text Lines - Detail • F4706 - EDI Order Address Information 	You can access this program from menu G47221.

80 Supply Management: Purchase Order Receipts

Purchase Order Receipts Overview

This chapter provides detailed information about the business interfaces that are available for the Purchase Order Receipts business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The PurchaseOrderReceiptManager web service (JP43A000) manages the processing of these purchase order receipt-related web service operations:</p> <ul style="list-style-type: none"> • processPurchaseOrderReceipt (J43A0010) • processPurchaseOrderReceiptV2 (J43A0010)
Real-Time Events	<p>This list includes the real-time events for the Purchase Order Receipts business object:</p> <ul style="list-style-type: none"> • RTRECOU
Batch Import Programs	<p>This list includes the batch import programs for the Purchase Order Receipts business object:</p> <ul style="list-style-type: none"> • Receipt Routing Inbound Processor (R43092Z11) • EDI Receiving Advice Edit/Create – Purchasing (R47071) • EDI Shipping Notice Edit/Create program (R47031)
Batch Export Programs	<p>This list includes the batch export programs for the Purchase Order Receipts business object:</p> <ul style="list-style-type: none"> • PO Receipts (P4312) • EDI Receiving Advice Extraction program (R47072)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

Purchase Order Receipts Business Services - PurchaseOrderReceiptsManager

The PurchaseOrderReceiptManager web service (JP43A000) manages the processing of purchase order receipt-related web service operations. This table includes a description of the purchase order receipt web service operation:

Operation	Description
processPurchaseOrderReceipt (J43A0010)	Use this operation to create partial and full purchase order receipts within the JD Edwards EnterpriseOne Procurement system. The operation supports basic and advanced serial number receipts.
processPurchaseOrderReceiptV2 (J43A0010)	Use this operation to create partial and full purchase order receipts, including country of origin information, within the JD Edwards EnterpriseOne Procurement system.

Accessing Javadoc for the Purchase Order Receipt Manager Web Service Operations

To access Javadoc for the Purchase Order Receipt Manager web service and its related operations, review these Javadoc packages:

- JP43A000 (PurchaseOrderReceiptManager)
- J43A0010 (processPurchaseOrderReceipt)
- J43A0010 (processPurchaseOrderReceiptV2)

Prerequisite

Before using the Purchase Order Receipt manager web service, or the related web service operation, you must install and configure the JD Edwards EnterpriseOne Procurement system.

See [JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide](#) .

See ["Understanding Serial Number Processing in the Distribution System" in the JD Edwards EnterpriseOne Applications Sales Order Management Implementation Guide](#) .

processPurchaseOrderReceipt

The processPurchaseOrderReceipt web service operation is an inbound transaction operation that enables consumers to process purchase order receipt information within the JD Edwards EnterpriseOne system. The consumer can create full and partial purchase order receipts in the JD Edwards EnterpriseOne Procurement system. Therefore, you can

choose which lines in a purchase order to send. You can send and receive all lines in a purchase order, or you can send and receive only selected lines in a purchase order.

The operation also supports basic and advanced serial number receipts.

The processPurchaseOrderReceipt web service operation receives goods and services on a purchase order. The processPurchaseOrderReceipt operation calls the XPIProcessInboundReceipt business function (B4302240) to process the purchase order receipt. If there are no errors during processing, the business function may return information and warnings depending upon the scenario and report back to the processPurchaseOrderReceipt web service operation.

Supported Functionality

This section discusses the functionality that the processPurchaseOrderReceipt operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

When creating purchase order receipts:

- Full and partial purchase order receipts are supported.
- Basic and advanced serial number receipts are supported.
- Receipt by purchase order and receipt by purchase order line item are supported.
- Receipt of one purchase order at a time is supported.

The processPurchaseOrderReceipt web service operation does not support the following functionality:

- Receipts of multiple purchase orders at a time.
- Update purchase order receipts.
- Receipts by account.
- Receive into multiple locations.
- Receipts for transportation.
- Receive and close purchase order and cancel purchase order receipt line.
- Receipt routing.
- Reverse receipt.
- Cascading receipts.
- Desktop receipt.
- Weight tag receipt (for wine industry).

Setup Considerations

Before you use this operation, you can set business service properties to specify how the system processes purchase order information. This table includes information about the business service properties that the processPurchaseOrderReceipt web service operation uses:

Group	Key	Description	Default Value
J43A0010	J43A0010_PR_MBF_VERSION	Use this business service property to specify which version	ZJDE0001

Group	Key	Description	Default Value
		of the PO Receipts program (P4312) the operation uses. See " <i>Setting Processing Options for PO Receipts (P4312)</i> " in the <i>JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide</i> .	
J43A0010	J43A0010_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages when JD Edwards EnterpriseOne runs the SerialNumberCacheProc business function (B4302180).	Receipt Serial Number Sent in:
J43A0010	J43A0010_PREFIX_2	Use this business service property to specify the prefix value the operation uses for error messages when JD Edwards EnterpriseOne runs the XPIProcessInboundRecei business function (B4302240) for edit doc and edit line.	Receipt Sent in:
J43A0010	J43A0010_PREFIX_3	Use this business service property to specify the prefix value the operation uses for error messages when JD Edwards EnterpriseOne runs the XPIProcessInboundRecei business function (B4302240) for end doc.	Receipt Process WF Sent in:

Implementation Details

The following table includes information that can help determine whether the processPurchaseOrderReceipt operation is functioning correctly:

Question	Answer
<p>How can I tell if the operation completes successfully?</p>	<p>If the web service call does not throw a business service exception, then the operation has completed successfully. There may be some warning messages in the returned value object which the consumer can review.</p> <p>If the operation completes successfully, the system returns a confirmation message to the consumer. The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> • documentNumber • documentTypeCode • documentCompany • receiptNumber • businessUnit • documentLineNumber • receiptLineNumber
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system.</p> <p>The consumer only needs to reverse the addition of records to the Item Location File (F41021) table in the event that a fatal server or connection error occurred when the rollback was occurring since the rollback of the F41021 table is handled manually with a business function call. All other transactions for the receipt are rolled back in the normal transaction processing handling.</p>
<p>Does this operation use record reservation?</p>	<p>This operation calls the XPIProcessInboundReceipt business function (B4302240). This function reserves purchase order records in the JD Edwards EnterpriseOne system when the operation is processing receipts so that no other user can update the purchase order.</p> <p>If the purchase order for which the operation is creating a receipt is reserved by some other application, then the call to the B4302240 business function returns an error and stops any further processing.</p>

processPurchaseOrderReceiptV2

Review the information in the processPurchaseOrderReceipt section of this chapter before using the processPurchaseOrderReceiptV2 web service operation. All of the information in the processPurchaseOrderReceipt section also applies to the processPurchaseOrderReceiptV2 operation.

The processPurchaseOrderReceiptV2 operation is a version of the processPurchaseOrderReceipt web service operation. This web service operation is used to create partial and full purchase order receipts, including country of origin information. This version executes the processPurchaseOrderReceipt (J43A0010) web service operation.

Purchase Order Receipts Real-Time Events

This section describes the available real-time events associated with the Purchase Order Receipts business object, which include:

- RTRECOUT

RTRECOUT

RTRECOUT is used by the JD Edwards EnterpriseOne system to notify third-party systems when a purchase order receipt is created.

Event Type	Event Description	Event Category	Event Aggregate	Product Code	Data Structure
RTRECOUT	Purchase Order Receipt	RTE	Single	43B	D4303420: Receipt Realtime Event Template

Conceptual Approach

This event is triggered when a purchase order receipt is created in the JD Edwards EnterpriseOne system.

Objects that Can Initiate the Event

These objects can initiate the RTRECOUT event:

- B4301300 - F43121 Update Landed Cost Records
- B4301440 - F43121 Copy Row With New Receipt Line Number
- B7600400 - Receive NF Records - Brazil
- B7602020 - Add Lines to Receipt File - Brazil
- NXT43092 - Receipt Routing
- X43091 - Receipt Routing Initiator
- XF43121 - Purchase Order Item Ledger Functions
- N4301430 - F43121 Update Line Split In Receipt Routing
- B4300720 - F43121, Update Receipt Date
- B4300870 - F43092, Update Disposition Information
- B7602080 - LC - Update GL, Cardex and Cost Difference CT - BRA - 43
- XT4312Z1 - Purchase Order Receipts

Purchase Order Receipts Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Purchase Order Receipts business object:

Program and Program Type	Description	Available Documentation
<p>Receipt Routing Inbound Processor (R43092Z11)</p> <p>This is an inbound interoperability program.</p>	<p>Running a transaction process, such as Receipt Routing Inbound Processor (R43092Z11), often identifies one or more inbound transactions that contain invalid transactions. For example, if you are in receipt routing and you try to move inventory to a step that was not defined in the receipt route, the Receipt Routing Inbound Processor program identifies the invalid transaction and sends an error message to the Work Center (P012501). The error message indicates the transaction number for the transaction in error.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding Reviewing and Revising Interoperability Transactions for Procurement
<p>EDI Receiving Advice Edit/Create – Purchasing (R47071)</p> <p>This is an EDI import program.</p>	<p>Run the EDI Receiving Advice Edit/Update Purchasing program (R47071) from the Receiving Advice Purchasing menu to communicate the receipt of goods or services at a customer site and to record the receipt against the purchase order in the JD Edwards EnterpriseOne Procurement system.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Understanding Inbound Advices into Purchasing (861/RECADV)
<p>EDI Shipping Notice Edit/Create program (R47031)</p> <p>This is an EDI import program.</p>	<p>Run the EDI Shipping Notice Edit/Create program (R47031) to update the status of a purchase order after you have received the ASN. The system uses the information that is contained in the F47031 table and the F47032 table to perform the status update.</p>	<p>See this topic:</p> <ul style="list-style-type: none"> • Receiving Inbound Shipping Notices (856)

Purchase Order Receipts Batch Export Programs

You use batch export programs to retrieve and format data from the JD Edwards EnterpriseOne database so that it can be used by an external software system.

This table lists and describes the available batch export programs that are associated with the Purchase Order Receipts business object:

Program and Program Type	Description	Available Documentation
PO Receipts (P4312) This is a standard interactive program that can create outbound interoperability data.	You can set the processing options of this program to create outbound interoperability data in the F43121Z1 table.	See this topic: <ul style="list-style-type: none"> • Entering Receipts
EDI Receiving Advice Extraction program (R47072) This is an EDI extract program.	Run the EDI Receiving Advice Extraction program (R47072) from the Receiving Advice Purchasing menu to report the receipt of goods or services and to report any quantities that are rejected or in question. Depending upon the warehousing environment, the transaction is inbound to sales or purchasing and outbound from purchasing.	See this topic: <ul style="list-style-type: none"> • Understanding Outbound Receiving Advice (861/IFTMAN)

81 Supply Management: Procurement Requisition Approval

Procurement Requisition Approval Overview

This chapter provides detailed information about the business interfaces that are available for the Procurement Requisition Approval business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The RequisitionSelfServiceManager published business service (JP43E010) manages these business services:</p> <ul style="list-style-type: none"> • RSSApproverQueryProcessor (J43E0050) • RSSDetailQueryProcessor (J43E0060) • processRSSApproveReject (J43E0080) <p>This business service was created as part of the Mobile Requisition Self Service feature. For details about this service, see</p> <ul style="list-style-type: none"> • Business Services for Pre-Built Integrations.

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see [Accessing Additional Information for Business Interface Components](#).

82 Supply Management: Procurement Voucher

Procurement Voucher Overview

This chapter provides detailed information about the business interfaces that are available for the Procurement Voucher business object.

This table lists all of the available interface components:

Interface Type	Available Interfaces
Business Services	<p>The VoucherMatchManager published business service (JP43B000) manages these web service operations:</p> <ul style="list-style-type: none"> • J43B0010 (processVoucherMatch) • J43B0010 (processVoucherMatchV2)
Batch Import Programs	<p>This list includes the batch import programs for the Procurement Voucher business object:</p> <ul style="list-style-type: none"> • EDI Inbound Invoice/Match to PO Edit/Create program (R470412) • Batch Voucher Processor Report (R04110ZA) <p>For detailed information about this program, see <i>Accounts Payable Voucher Batch Import Programs</i>.</p> <ul style="list-style-type: none"> • Voucher Match Automation Driver program (R4304010)

Note: You can find additional technical information about each type of business interface component. For additional descriptions of the information available, and instructions for finding this information, see *Accessing Additional Information for Business Interface Components*.

Procurement Voucher Business Services - VoucherMatchManager

This section describes the available business services associated with the VoucherMatchManager published business service (JP43B000), which is part of the Procurement Voucher business object. The business services include:

Operation	Description
processVoucherMatch (J43B0010)	Use this operation to add voucher match records for logged vouchers and both two-way and three-way voucher match within the JD Edwards EnterpriseOne Procurement system. This operation also supports additional charge information lines for a voucher along with purchase order lines being vouchered.
processsVoucherMatchV2 (J43B0010)	Use this operation to add voucher match records for logged vouchers and both two-way and three-way voucher match within the JD Edwards EnterpriseOne Procurement system. This operation also supports additional charge information lines for a voucher along with purchase order lines being vouchered. Additionally, this version includes global location number (GLN).

Accessing Javadoc for the VoucherMatchManager Web Service Operations

To access Javadoc for the VoucherMatchManager web service and its related operations, review these Javadoc packages:

- JP43B000 (VoucherMatchManager)
- J43B0010 (processVoucherMatch)
- J43B0010 (processVoucherMatchV2)

Prerequisites

Before you use the Voucher Match Manager web service, or any of the related operations, you must install and set up the JD Edwards EnterpriseOne Procurement and JD Edwards EnterpriseOne Accounts Payable systems.

See *"Setting Up the Procurement System" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide* .

See *"Setting Up the Accounts Payable System" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide* .

processVoucherMatch

This section provides an overview of the processVoucherMatch web service operation, lists prerequisites, and discusses how to set processing options for Purchase Order XPI (P43XPI).

The processVoucherMatch operation is an inbound transaction operation that enables consumers to process voucher match information within the JD Edwards EnterpriseOne system. The consumer can add voucher match records to the JD Edwards EnterpriseOne Procurement system for logged vouchers and both two-way and three-way voucher match. This operation also supports additional charge lines for purchase order detail lines on a voucher.

If the operation completes successfully, the system returns a confirmation message to the consumer. This message contains voucher match data. The system also returns to the consumer any error messages that the operation encounters during processing.

The processVoucherMatch operation processes a voucher match in the JD Edwards EnterpriseOne system. The first call of the XPIInboundVoucherMatch business function (B4302300) checks for record reservation. If the business function finds that any record is reserved, the system stores the error in the Message List and returns from flow.

For an additional charge line, the XPIInboundVoucherMatch business function passes tax information into the taxRateArea, taxExplanationCode, additionalChargeQty, amount, lineTaxableCode, description1, lineTaxAmount, and expenseAcctNumber fields.

The system uses the following processing logic to handle tax for additional charge lines:

1. Pass the following information to the VM Edit Line if the operation passes a non-zero value to the lineTaxAmount field:
 - a. Set the lineTaxableCode to **Y**.
 - b. Use the taxRateArea and taxExplanationCode if the operation passes these values.
 - c. Use the values from the first PO Detail record fetched for the passed in PO header information if either the taxRateArea or the taxExplanationCode value passed in is blank.
 - d. Use the values from the supplier purchasing information's default tax values if either the taxRateArea or the taxExplanationCode value passed in is blank.
2. Pass the following information to the VM Edit Line if the operation passes a non-empty and non **N** value for the lineTaxableCode field:
 - a. Use the values the operation passes in for the taxRateArea and taxExplanationCode fields.
 - b. Use the values from the first PO Detail record fetched for the passed in PO header information if either the taxRateArea or the taxExplanationCode value passed in is blank.
 - c. Use the values from the supplier purchasing information's default tax values if either the taxRateArea or the taxExplanationCode value passed in is blank.
3. Do not pass in the tax information to the VM Edit Line function if the integration passes in **N** or blank lineTaxableCode value.

The system uses the expense account number if the B4302300 business function passes it into the system. If the expense account number is not passed in then the system uses the default account number from the P43XPI processing options.

Prerequisites

Before using the processVoucherMatch operation, you must:

- Set the Voucher Type processing option on the Voucher Match tab of the P43XPI program.

Note: It is recommended that you set the Voucher Type processing option to **1**. This processing option setting enables the operation to process matched and logged vouchers.

- Set the 2. Voucher Match Processing Option Version (P4314) processing option on the Versions tab of the P43XPI program.
- Set the 2. Expense Account for Additional Charges processing option on the Voucher Match tab of the P43XPI program for additional charges.
- Set the processing options for the version of P4314 called by the P43XPI program.

See *"Setting Processing Options for Voucher Match (P4314)" in the JD Edwards EnterpriseOne Applications Procurement Management Implementation Guide* .

- Set the 8. Orders Per Voucher processing option on the Process tab of the P4314 program to **1**.

- Set the 1. Order Line Entry processing option on the New Order Line tab of the P4314 program to **2** for additional charges.
- Set the 2. Line Type processing option on the New Order Line tab of the P4314 program to **J** for additional charges.
- Set the 3. Last Status Code processing option on the New Order Line tab of the P4314 program to **400** for additional charges.
- Set the 3. AP Master Business Function Processing Option Version for Logged Vouchers (P0400047) processing option on the Versions tab of the P43XPI program.
- Set the processing options for the version of P0400047 called by the P43XPI program.

See *"Setting Processing Options for Voucher Entry MBF (P0400047)" in the JD Edwards EnterpriseOne Applications Accounts Payable Implementation Guide* .

Note: If logged vouchers are enabled in the Voucher Type processing option on the Voucher Match tab of the P43XPI program, then you must set the 1. Voucher Logging processing option on the Logging tab of the P0400047 program.

Supported Functionality

This section discusses the functionality that the processVoucherMatch operation supports.

Note: If functionality is not explicitly documented as supported functionality, it is to be understood that the functionality is not supported by the integration solution.

The consumer can add voucher match records to the JD Edwards EnterpriseOne Procurement system. The operation supports both two-way and three-way voucher match. The consumer can also create a logged voucher. The operation processes only one line for the logged voucher.

The processVoucherMatch operation does not support the following functionality:

- Change voucher
- Delete voucher
- Reverse voucher
- Voucher redistribution
- Recost vouchers

Setup Considerations

Before you use this operation, you can set business service properties to define how the system processes voucher match data. If you do not set these business service properties, the system uses default values for each constant. This table includes information about the business service properties used by the processVoucherMatch operation:

Group	Key	Description	Default Value
J43B0010	J43B0010_P43XPI_VERSION	Use this business service property to specify which version of the Purchase Order XPI program (P43XPI) the operation uses.	ZJDE0001

Group	Key	Description	Default Value
J43B0010	J43B0010_BYPASS_BSFN_WARNINGS	Use this business service property to specify whether the system converts warning messages to errors during processing.	1 Note: The system does not treat warnings as errors so the system keeps processing.
J43B0010	J43B0010_PREFIX_1	Use this business service property to specify the prefix value the operation uses for error messages when JD Edwards EnterpriseOne processes detail lines.	VoucherMatchRecordSentin
J43B0010	J43B0010_PREFIX_2	Use this business service property to specify the prefix value the operation uses for error messages when JD Edwards EnterpriseOne processes additional charges.	VoucherMatchAdditionalRecordSentin

The processVoucherMatch web service operation calls the XPIInboundVoucherMatch business function (B4302300) which calls the Purchase Order XPI program (P43XPI). You must set the processing options for the P43XPI program, which provides the input for the voucher type. You must set the Voucher Type processing option on the Voucher Match tab.

See *Setting Processing Options for Purchase Order XPI (P43XPI)*..

Implementation Details

The following table includes information that can help determine whether the processVoucherMatch operation is functioning correctly:

Question	Answer
How can I tell if the operation completes successfully?	<p>If a business service exception is not thrown, then the operation completed successfully and the system returns a confirmation message to the consumer.</p> <p>The return message includes all of the fields that are listed in the response interface for this operation. However, some of those fields can contain blank or zero values, depending on the data that exists in the JD Edwards EnterpriseOne system.</p> <p>At a minimum, the system returns non-zero values for these fields:</p> <ul style="list-style-type: none"> supplierInvoiceNumber documentNumber documentTypeCode

Question	Answer
	<ul style="list-style-type: none"> • documentCompany • documentCompany • messages • documentLineNumber • documentLineNumberSuffix
<p>If I encounter errors while processing a transaction, do I need to reverse the transaction?</p>	<p>This operation uses standard transaction processing. Therefore, if you encounter errors during processing, the system does not update any information in the JD Edwards EnterpriseOne system. No manual update is necessary.</p>
<p>Does this operation use record reservation?</p>	<p>This operation calls the XPIInboundVoucherMatch business function (B4302300). This function reserves records in the JD Edwards EnterpriseOne system when the operation adds voucher match records for both two-way and three-way voucher match.</p>

Setting Processing Options for Purchase Order XPI (P43XPI)

Processing options enable you to specify the default processing for programs and reports.

For programs, you can specify options such as the default values for specific transactions, whether fields appear on a form, and the version of the program that you want to run.

Order Statuses

1. Beginning Order Status allowed for External Publish

Specify a value from UDC 40/AT that indicates the last step in the processing cycle that this order line has successfully completed.

2. Ending Order Status allowed for External Publish

Specify a value from UDC 40/AT that indicates the next step in the order flow of the line type.

3. Acknowledged Order Status Code

Specify a value from UDC 40/AT that indicates the next step in the order flow of the line type.

4. Acknowledged With Change Order Status Code

Specify a value from UDC 40/AT that indicates the next step in the order flow of the line type.

5. Pending Order Status Code

Specify a value from UDC 40/AT indicates the next step in the order flow of the line type.

Versions

1. Purchase Order Entry Processing Option Version (P4310)

Specify the version that the system uses when you enter a purchase order.

2. Voucher Match Processing Option Version (P4314)

Specify the version that the system uses when you match an invoice to a purchase order line or receipt line. If the Voucher Match processing option, which is located on the Voucher Match tab, is set to 1 or 2, the system validates the version that you specify.

3. AP Master Business Function Processing Option Version for Logged Vouchers (P0400047)

Specify the version that the system uses when you log a voucher for an invoice. If the Voucher Match processing option, which is located on the Voucher Match tab, is set to blank or 1, the system validates the version that you specify to ensure that the voucher logging process is active.

Receipts

1. Shipped Order Status Code

Specify a value from UDC 40/AT that indicates the next step in the order flow of the line type.

2. Advance Status Only

Specify whether the system advances the shipment status of the line or order. Values are:

Blank: Do not advance to Shipped Status.

1: Advance to shipped status (no receipt).

Voucher Match

1. Voucher Type

Specify whether the system creates a logged voucher or matched voucher. Values are:

Blank: The system always creates a logged voucher.

1: If the invoice includes specific purchase order information, the system creates a matched voucher. If the invoice does not include specific purchase order information, the system creates a logged voucher.

2: The system always creates a matched voucher.

2. Expense Account for Additional Charges

Specify the account that the system uses to expense the additional charges that are specified on the invoice. You set up the account format in the General Accounting Constants program (P0000), using one of the following formats for account numbers:

- Structured account (business unit.object.subsidiary)
- 25-digit unstructured number
- 25-digit unstructured number
- 8-digit short account ID number
- Speed code

processVoucherMatchV2

Review the information in the processVoucherMatch section of this chapter before using the processVoucherMatchV2 web service operation. All of the information in the processVoucherMatch section also applies to the processVoucherMatchV2 operation.

The processVoucherMatchV2 operation is a version of the processPurchaseVoucherMatch web service operation. This web service operation is used to add voucher match records for logged vouchers along with GLNs. This version executes the processPurchaseOrder (J43B0010) web service operation.

Procurement Voucher Batch Import Programs

You use batch import programs to import data from an external software system into the JD Edwards EnterpriseOne system, where it can then be processed by EnterpriseOne programs.

This table lists and describes the available batch import programs that are associated with the Procurement Voucher business object:

Program and Program Type	Description	Available Documentation
EDI Inbound Invoice/Match to PO Edit/Create program (R470412) This is an EDI program.	The EDI Inbound Invoice/Match to PO Edit/Create program (R470412) updates EDI documents that are successfully processed in the EDI inbound interface tables to prevent duplicates. The program updates EDI documents by entering Y in the Processed column (EDSP) in all of the EDI inbound interface tables.	See this topic: <ul style="list-style-type: none"> • Receiving Invoices with Receipt Matches
Voucher Match Automation Driver program (R4304010)	To use the voucher match automation process, you run the Voucher Match Automation Driver program (R4304010), which launches all of the Voucher Match Automation Matching UBE programs.	See this topic: <ul style="list-style-type: none"> • Understanding the Voucher Match Automation Process

getCapitalAssetConditionBasedAlert (J1300002) *96*
 getVoucher (J0400002) *142*
 overview *7, 8*
 processCustomerV2 *183*
 processCustomerV3 *183*
 processVoucher (J0400001) *144*
 business unit
 batch export *258*
 Business Unit Batch Inbound (R0006ZI) *249, 258*
 business unit information *253, 254, 254, 254, 255, 256, 257*
 business unit information RTBUOUT3 *256*
 Business Unit Master record (F0006) *323*
 business unit/jobs cross reference *297, 298*
 BusinessUnitManager (JP000006) *249, 250, 260*

C

Capital Asset Manager (JP130000) *93*
 category code information for customers *187*
 company information *263*
 competency based job descriptions *296, 297*
 competency information
 competency master records *279, 280*
 resource competency header records *301, 301*
 competency scales information
 container event *300*
 detail records *300*
 header records *300*
 configured items *394*
 Construction Lot Master business object *313, 319, 325*
 construction lot master overview *313, 319, 325*
 container item information *432, 434*
 container order information *431*
 createCapitalAssetConditionBasedAlert (J1300001) *93, 94*
 CreateProjectAccount business function (B5100410) *323*
 CreateProjectFinancialForecasts business function (B5100160) *323*
 CreateProjectHeader business function (B5100120) *323*
 CreditCardTransactionProcessor (J09E0004) *71*
 CRM Service Order Integration Processing Options (P1702650) *114*
 currency exchange rate information *267*
 customer additions *188*
 customer category code *189*
 customer category code information *187*
 customer email address *188*
 customer email addresses *186*
 customer information *184, 185, 190, 191*
 Customer Master Batch Upload (R03010Z) *176*
 Customer Master Batch Upload program (R03010Z) *191*
 Customer Master MBF - PO (P0100042) *176*
 customer phone number *189*
 customer phone numbers *186*
 Customer Service Work Order *109*
 CustomerAndContactManager (JP010050) *175, 183*
 CustomerManager (JP010020) *176*
 CustomerServiceManager (JP170000) *109, 111*
 Cycle Count Inbound Transaction Process program (R4141Z11) *337, 337, 343*
 Cycle Count Update program (R41413) *337, 337*

D

deduction information *149, 150*
 Direct Connect *74*
 Direct Connect Authentication Processor *74*
 Draft Entry (P03B602) *214*

E

EDI (Electronic Data Interchange) *15*
 EDI Advance Ship Notice Extraction program (R47032) *426, 436*
 EDI Inbound Demand Edit/Update Report (R47171) *331, 331*
 EDI Inbound Invoice/Match to PO Edit/Create program (R470412) *487, 494*
 EDI Inbound PO Acknowledgment program (R47021) *457, 475*
 EDI Inbound PO Change Acknowledgment program (R47141) *457, 474*
 EDI Planning Schedule Edit/Create program (R47061) *375, 375*
 EDI Planning Schedule Extraction Conversion program (R47062C) *375, 376*
 EDI Product Activity Data Edit/Create program (R47121) *345, 345*
 EDI Product Activity Data Extraction program (R47122) *345, 346*
 EDI Purchase Order Change Extraction program (R47132) *458, 475*
 EDI Receiving Advice Edit/Create – Purchasing (R47071) *483*
 EDI Receiving Advice Edit/Create (R47071) *477*
 EDI Receiving Advice Extraction program (R47072) *477, 484*
 EDI RFQ Extraction (R47102) *458*
 EDI Shipping Notice Edit/Create program (R47031) *426, 435, 477, 483*
 Edit/Create Voucher (R47041) *141, 146*
 Electronic Data Interchange (EDI) *15*
 email addresses
 for customers *186*
 for supplier *208, 210*
 email information *240, 243, 246*
 employee information
 container event *290*
 F060116 detail *291*
 F060117 detail *291*
 F060120 detail *292*
 EmployeeManager web service (JP080020) *281, 281*
 EngineeringChangeOrderPartsListProcessor (J3000020) *69*
 EngineeringChangeOrdersProcessor (J3000010) *69*
 Entity Processor (J0100010) *8*
 equipment information *101, 102*
 EquipmentManager (JP170001) *87*
 equipmentProcessor (J1700001) *87, 88*
 equipmentQueryProcessor (J1700002) *87*
 EquipmentQueryProcessor (J1700002) *89*
 EU VAT ID Validation *76*
 Event Definition Workbench program (P90701A) *12*
 events, defining *12*
 exchange rate information *267*
 ExpenseManagementEmployeeProfileQueryProcessor (J09E0002) *71*
 ExpenseReportProcessor (J09E0005) *71*
 ExpenseReportQueryProcessor (J09E0001) *70*
 ExpenseReportRecordReservation (J09E0006) *71*
 ExpenseReportWorkflowProcessor (J09E0003) *71*
 Extended Job Cost Master record (F5108) *323*
 External Cost Analyzer Balances Processor program (R1602Z1) *171, 171*
 External Exchange Rate Processor program (R0015Z1) *267, 268*

F

F0006 (Business Unit Master record) *323*
 F0411 Interoperability Processing Options program (P0400048) *141*
 F5108 (Extended Job Cost Master record) *323*
 F90701 table *12*
 Financial Compliance Manager
 javadoc *214*
 Financial Management Solutions (FMS) *213*
 FinancialComplianceManager (JP000040) *213*
 Fixed Assets
 batch export *165*
 batch import *165*
 FixedAssetManager
 javadoc *163*

FixedAssetManager (JP120000) *163, 163*
 Forecast Inbound Processor (R3460Z11) *335, 335*
 Foundation Environment web service (JP000000) *269*
 FoundationEnvironment *72*
 functionality matrix *17*

G

General Ledger Account Ledger *127*
 batch import *129*
 General Ledger Account Master *131*
 getAddressBook (J0100002) *227, 228, 237*
 getAddressBookV2 (J0100002) *227, 228*
 getAgingCompanyConstants (J0000043) *213, 214*
 getAllExpenseReports *71*
 getAPProcessingOptions (J0000041) *213*
 getBranchPlantItem (J4100004) *347, 348*
 getBranchPlantItemV2 (J4100004) *347, 348*
 getCalculatedAvailability (J4100007) *73*
 getCapitalAssetConditionBasedAlert (J1300002) *93, 96*
 getCommitmentDataTimeV2 (J1700030) *109, 111*
 getCommitmentDateTime (J1700030) *109, 110, 110, 111*
 getContact (J0100004) *227, 229*
 getCreditCardTransactions *71*
 getCustomer (J0100022) *175*
 getCustomerCreditInformation (J0100023) *175*
 getCustomerCreditInformationV2 (J0100023) *175, 181*
 getCustomerCreditLimits (J0000044) *213, 214*
 getCustomerItemPrice (J4200040) *389, 391*
 getCustomerItemPriceV2 (J4200040) *389, 391*
 getCustomerV2 (J0100022) *175, 175, 179*
 getCustomerV3 (J0100022) *180*
 getDocumentTypeDescription *73*
 getEmployee operation (J0800041) *287*
 getExpenseManagementPolicy (J09E0007) *71*
 getExpenseReportHeaders *71*
 getExpenseReports *71*
 getExpenseReportsToApprove *71*
 getFixedAsset
 business service *164*
 getGeneralConstants (J0000045) *213, 214*
 getItemAvailability (J4100001) *347, 348*
 getItemBranchSearch (J4100060) *73*
 getItemListPrice (J4200060) *389, 392*
 getItemPrice (J4100040) *73*
 getItemPriceAndAvailability (J4200030) *389, 391*
 getItemPriceAndAvailabilityV2 (J4200030) *389, 391*
 getItemPriceAndAvailabilityV3 (J4200030) *389, 391*
 getItemSearch (J4100050) *73*
 getJournalGeneralConstants (J0000046) *213, 214*
 GetMobilePurchaseOrderCountsProcessor (J4300070) *72, 73*
 getPolicyEditRules (J0000047) *213, 214*
 getPurchaseOrder (J4300030) *457, 458*
 getPurchaseOrderCounts *73*
 getPurchaseOrderDetailForApprover *73*
 GetPurchaseOrderDetailForApprover (J4300050) *72*
 GetPurchaseOrderDetailForApprover (J4300050). *73*
 getPurchaseOrderEmployeeProfile *73*
 getPurchaseOrdersForApprover *73*
 getPurchaseOrderV2 (J4300030) *457, 458*
 getPurchasingToleranceRules (J0000048) *213, 214*
 getSalesOrder (J4200050) *389, 391*
 getSalesOrderPriceHistory (J4200070) *389, 392*
 getSalesOrderV2 (J4200050) *389, 391*
 getSalesOrderV3 *389, 391, 391*
 getSecurityWorkbench (J0000049) *213, 214*

getServiceOperationSequence (J1700060) *74*
 getServiceOrder (J1700020) *109, 110, 111*
 getServiceOrderV2 (1700020) *109, 111*
 getServiceOrderV2 (J1700020) *110*
 getServiceTimeCardProcessingOptions (J1700070) *74*
 getServiceTimeCards (J1700050) *74*
 getSupplier (J0100032) *199*
 getSupplierCatalogPrice (J4100005) *347, 348*
 getSupplierCatalogPriceV2 (J4100005) *347, 348*
 getSupplierV2 (J0100032) *199*
 getSupplierV3 (J0100032) *199*
 getUDCDescription (J0000050) *269, 270*
 getUDCDescription (J0000050) *73*
 getUserDefinedCode *72*
 getUserDefinedCode (J0000030) *269, 269*
 getUserProfile (J0000080) *269, 270*
 getVoucher (J0400002) *141, 142*
 getWriteOffProcessingOptions (J0000042) *213*
 GL Account Processor (J0900010) *8*
 global carrier code *431*

H

hold release *474*
 Hours and Quantities Update program (R31422) *446, 454*

I

implementation steps *9*
 Inbound AP Claim program (R04110ZB) *141, 146*
 Inbound Asset Master Processor (R1201Z11) *163, 165*
 Inbound Completion Processor (R31114Z11) *445, 453*
 Inbound Condition-Based Alerts Processor program (R1310Z11) *93, 99, 99*
 Inbound Hours and Quantity Processor (R31122Z11) *445, 446, 453*
 Inbound Inventory Issues Processor (R31113Z11) *445, 446, 453*
 Inbound Kanban Transaction program (R30161Z11) *369, 369*
 Inbound Meter Readings Processor program (R12120Z11) *105, 105*
 Inbound Purchase Order (R4311Z11) *457, 474*
 Inbound Suggestion Processor (R4611Z11) *441, 441*
 Inbound Super Backflush Processor program (R31123Z11) *445, 453*
 Inbound Warranty Claim (R174801Z21) *123*
 Inbound Warranty Claim Parts (R173111Z21) *123*
 Inbound Warranty Claim Parts Detail Processing program (R173111Z21) *123*
 Inbound Warranty Claim Processing program (R174801Z21) *124*
 Inbound Work Order for Blend program (R31B65AZ11) *443, 443*
 insertInventoryItemStaging (J4100006) *347, 348*
 insertInventoryItemStagingV2 (J4100006) *348*
 integration process *8*
 Interoperability Event Definition table (F90701) *12*
 Interoperability Programs *15*
 Inventory Item ID Processor (J4100010) *8*
 Inventory Manager (JP410000) *347, 348*
 InventoryItemBranchProcessor (J4100030) *69*
 InventoryItemsProcessor (J4100020) *69*
 Invoice Entry MBF Processing Options (P03B0011) *151*
 invoice information
 container event *386, 387*
 detail information *387, 388*
 header information *386*
 Invoice Print program (R42565) *385, 388, 390*
 Invoice Selection Match (R03B50A) *214*
 Item Batch Outbound (R4101ZO) *363*
 Item Cost Inbound Transaction Process program (R4105Z11) *361, 362*
 item information *364, 364, 365, 365*
 item list price information *380, 380, 381*
 Item Location Extraction (R41021ZX) *339, 342*

item location information [339](#), [340](#), [341](#), [342](#)
 Item Master Inbound Transaction Process (R4101Z11) [347](#), [363](#), [366](#)
 item unit cost information [361](#), [362](#)

J

J0000010 (Processing Version Processor) [8](#)
 J0000020 (Net Change Processor) [8](#)
 J0000030 (getUserDefinedCode) [72](#), [269](#), [269](#)
 J0000040 (lookupBusinessUnits) [72](#), [269](#), [269](#)
 J0000041 (getAPPProcessingOptions) [213](#)
 J0000042 (getWriteOffProcessingOptions) [213](#)
 J0000043 (getAgingCompanyConstants) [213](#), [214](#)
 J0000044 (getCustomerCreditLimits) [213](#), [214](#)
 J0000045 (getGeneralConstants) [213](#), [214](#)
 J0000046 (getJournalGeneralConstants) [213](#), [214](#)
 J0000047 (getPolicyEditRules) [213](#), [214](#)
 J0000048 (getPurchasingToleranceRules) [213](#), [214](#)
 J0000049 (getSecurityWorkbench) [213](#), [214](#)
 J0000050 (getUDCDescription) [269](#), [270](#)
 J0000060 (lookupCurrencyCodes) [267](#), [269](#), [270](#)
 J0000070 (setUserValues and getUserValues) [269](#), [270](#)
 J0000080 (getUserProfile) [269](#), [270](#)
 J0000090 (lookupUDC) [269](#), [270](#)
 J0000120 [249](#)
 J0000130 [249](#)
 J0100001 (processAddressBook) [227](#), [228](#)
 J0100001 (processAddressBookV2) [227](#), [229](#)
 J0100002 (getAddressBook) [227](#), [228](#), [237](#)
 J0100002 (getAddressBookV2) [227](#), [228](#)
 J0100003 (processContact) [175](#), [227](#), [229](#)
 J0100004 (getContact) [227](#), [229](#)
 J0100009 (lookupAddressBook) [227](#), [229](#)
 J0100010 (Entity Processor) [8](#)
 J0100021 (processCustomer) [175](#), [175](#)
 J0100021 (processCustomerV2) [175](#), [175](#)
 J0100022 (getCustomer) [175](#)
 J0100022 (getCustomerV2) [175](#), [175](#), [179](#)
 J0100022 (getCustomerV3) [180](#)
 J0100023 (getCustomerCreditInformation) [175](#)
 J0100023 (getCustomerCreditInformationV2) [175](#), [181](#)
 J0100031 (processSupplier) [199](#)
 J0100031 (processSupplierV2) [199](#)
 J0100031 (processSupplierV3) [199](#)
 J0100032 (getSupplier) [199](#)
 J0100032 (getSupplierV2) [199](#)
 J0100032 (getSupplierV3) [199](#)
 J03B0001 (processARInvoiceProcessor) [152](#)
 J03B0001 (ProcessARInvoiceProcessor) [151](#)
 J03B0002 (ARInvoiceQueryProcessor) [151](#), [154](#)
 J03B0003 (ARInvoiceRevisionsQueryProcessor) [151](#), [156](#)
 J03B0004 [151](#)
 J0400001 (processVoucher) [141](#), [144](#)
 J0400002 (getVoucher) [141](#), [142](#)
 J0500010 (BatchTimeSheetInsertProcessor) [323](#)
 J0800040 (processEmployee web service operation) [282](#)
 J0800041 (getEmployee operation) [287](#)
 J0800050 (processPendingEmployee) [289](#)
 J0900010 (GL Account Processor) [8](#)
 J09E0001 (ExpenseReportQueryProcessor) [70](#)
 J09E0001 (getAllExpenseReports) [71](#)
 J09E0001 (getExpenseReportHeaders) [71](#)
 J09E0001 (getExpenseReports) [71](#)
 J09E0001 (getExpenseReportsToApprove) [71](#)
 J09E0002 (ExpenseManagementEmployeeProfileQueryProcessor) [71](#)
 J09E0003 (approveExpenseReport) [71](#)
 J09E0003 (ExpenseReportWorkflowProcessor) [71](#)
 J09E0003 (rejectExpensesReport) [71](#)
 J09E0003 (submitExpenseReport) [71](#)
 J09E0004 [173](#)
 J09E0004 (CreditCardTransactionProcessor) [71](#)
 J09E0004 (getCreditCardTransactions) [71](#)
 J09E0005 [197](#)
 J09E0005 (ExpenseReportProcessor) [71](#)
 J09E0005 (processExpenseReportDetail) [72](#)
 J09E0005 (processExpenseReportHeader) [72](#)
 J09E0005 (processExpenseReports) [72](#)
 J09E0006 (ExpenseReportRecordReservation) [71](#)
 J09E0006 (releaseExpenseReport) [72](#)
 J09E0006 (reserveExpenseReport) [72](#)
 J09E0007 [193](#)
 J09E0007 (getExpenseManagementPolicy) [71](#)
 J1300001 (createCapitalAssetConditionBasedAlert) [93](#), [94](#)
 J1300002 (getCapitalAssetConditionBasedAlert) [93](#), [96](#)
 J1700001 (equipmentProcessor) [87](#), [88](#)
 J1700002 (equipmentQueryProcessor) [87](#)
 J1700002 (EquipmentQueryProcessor) [89](#)
 J1700010 (processServiceOrder) [109](#), [110](#), [111](#)
 J1700010 (processServiceOrderV2) [109](#), [111](#)
 J1700010 (processServiceOrderV2) [110](#)
 J1700020 (getServiceOrder) [109](#), [110](#), [111](#)
 J1700020 (getServiceOrderV2) [110](#)
 J1700030 (getCommitmentDataTimeV2) [109](#), [111](#)
 J1700030 (getCommitmentDateTime) [109](#), [110](#), [110](#), [111](#)
 J1700050 (getServiceTimeCards) [74](#)
 J1700060 (getServiceOperationSequence) [74](#)
 J1700070 (getServiceTimeCardProcessingOptions) [74](#)
 J1700080 (processServiceTimeCard) [74](#)
 J3000010 (EngineeringChangeOrdersProcessor) [69](#)
 J3000020 (EngineeringChangeOrderPartsListProcessor) [69](#)
 J3000030 (ProcessIntegrationTimeStamp) [455](#)
 J3100010 [445](#), [446](#)
 J3100020 [445](#), [446](#)
 J34A0010 (OrderPromisingProcessor business service) [75](#)
 J4100001 (getItemAvailability) [347](#), [348](#)
 J4100002 (processSupplierCatalogPrice) [347](#), [348](#)
 J4100002 (processSupplierCatalogPriceV2) [347](#), [348](#)
 J4100003 (processInventoryItem) [347](#), [348](#)
 J4100003 (processInventoryItemV2) [347](#), [348](#)
 J4100004 (getBranchPlantItem) [347](#), [348](#)
 J4100004 (getBranchPlantItemV2) [347](#), [348](#)
 J4100005 (getSupplierCatalogPrice) [347](#), [348](#)
 J4100005 (getSupplierCatalogPriceV2) [347](#), [348](#)
 J4100006 (insertInventoryItemStaging) [347](#), [348](#)
 J4100006 (insertInventoryItemStagingV2) [348](#)
 J4100007 (getCalculatedAvailability) [73](#)
 J4100010 (Inventory Item ID Processor) [8](#)
 J4100020 (InventoryItemsProcessor) [69](#)
 J4100030 (InventoryItemBranchProcessor) [69](#)
 J4100040 (getItemPrice) [73](#)
 J4100050 (getItemSearch) [73](#)
 J4100060 (getItemBranchSearch) [73](#)
 J4200010 (processSalesOrder) [389](#), [390](#)
 J4200010 (processSalesOrderV2) [389](#), [390](#)
 J4200010 (processSalesOrderV3) [389](#), [389](#), [390](#)
 J4200010 (processSalesOrderV5) [390](#)
 J4200020 (processSalesPriceAdjustment) [389](#), [391](#)
 J4200020 (processSalesPriceAdjustmentV2) [389](#), [391](#)
 J4200030 (getItemPriceAndAvailability) [389](#), [391](#)
 J4200030 (getItemPriceAndAvailabilityV2) [389](#), [391](#)
 J4200030 (getItemPriceAndAvailabilityV3) [389](#), [391](#)
 J4200040 (getCustomerItemPrice) [389](#), [391](#)

J4200040 (getCustomerItemPriceV2) *389, 391*
 J4200050 (getSalesOrder) *389, 391*
 J4200050 (getSalesOrderV2) *389, 391*
 J4200060 (getItemListPrice) *389, 392*
 J4200070 (getSalesOrderPriceHistory) *389, 392*
 J4300010 (processPurchaseOrder) *457, 458*
 J4300010 (processPurchaseOrderV2) *457, 458*
 J4300020 (processPurchaseOrderAcknowledge) *457, 458*
 J4300030 (getPurchaseOrder) *457, 458*
 J4300030 (getPurchaseOrderV2) *457, 458*
 J4300040 (PurchaseOrderApprovalQueryProcessor (J4300040)) *72*
 J4300050 (GetPurchaseOrderDetailForApprover) *72*
 J4300060 (PurchaseOrderEmployeeProfileQueryProcessor) *72*
 J4300070 (GetMobilePurchaseOrderCountsProcessor) *72*
 J4300080 (PurchaseOrderApproveRejectProcessor) *73*
 J43A0010 (processPurchaseOrderReceipt) *477, 478*
 J43A0010 (processPurchaseOrderReceiptV2) *477, 478*
 J43B0010 (processPurchaseOrder) *493*
 J43B0010 (processVoucherMatch) *487*
 J43B0010 (processVoucherMatchV2) *487*
 J43E0010 (Direct Connect Authentication Processor) *74*
 J43E0020 (Shopping Cart Processor) *74*
 J43E0030 (PO Dispatch Processor) *75*
 J43E0050 (RSSApproverQueryProcessor) *485*
 J43E0060 (RSSDetailQueryProcessor) *485*
 J43E0080 (processRSSApproveReject) *485*
 J4800001 (lookupWorkOrders) *445*
 J4800001 (WorkOrders) *446*
 J4800010 *107, 107, 109, 119, 329, 329, 445, 446*
 J4800020 *107, 107, 109, 119, 329, 329, 445, 446*
 J49T0010 (Rate Shopping Processor) *76*
 J5100010 (processProject) *323*
 J5100020 (processFinancialForecast) *323*
 J5100030 (processWBS) *323*
 J740105 (VIESAppServerOASOrWLS) *77*
 J76B0021 (NF-e Lot Processing Processor) *77*
 javadoc
 access instructions *38*
 accessing *36*
 Accounts Payable Manager *142*
 Address Book Manager *229, 250, 260*
 Capital Asset Manager *94*
 class hierarchy *38*
 Customer Manager *177*
 CustomerServiceManager *111*
 display and navigation *37*
 Equipment Manager *88*
 Financial Compliance Manager *214*
 FixedAssetManager *163*
 Inventory Manager *349*
 overview *35*
 package naming conventions *37*
 Procurement Manager *459*
 Purchase Order Receipt Manager *478*
 Sales Order Manager *392*
 Supplier Manager *201*
 uses of a business service *38*
 viewing hierarchy information *64*
 viewing uses of an object *59*
 viewing value object and field information *47*
 Voucher Match Manager *488*
 JC76B001 (NfeRecepcao) *78*
 JC76B002 (NfeRetRecepcao) *78*
 JC76B003 (NfeCancelamento) *78*
 JC76B005 (NfeInutilizacao) *78*
 job information *295, 296*

JobCostManager *72*
 Journal Entries Batch Processor program (R09110Z) *127, 129*
 Journal Entry Master Business Function (P0900049) *127*
 JP000000 (Foundation Environment web service) *269*
 JP000006 *249*
 JP000006 (BusinessUnitManager) *250, 260*
 JP000040 (FinancialComplianceManager) *213*
 JP010000 (Address Book Manager) *227, 228*
 JP010020 (CustomerManager) *176*
 JP010030 (Supplier Manager) *200*
 JP010050 (CustomerAndContactManager) *175, 183*
 JP03B000 (ARInvoiceManager) *152*
 JP040000 (AccountsPayableManager) *142*
 JP05001 (BatchTimeSheetManager) *323*
 JP080020 (EmployeeManager web service) *281, 281*
 JP120000 (FixedAssetManager) *163, 163*
 JP130000 (Capital Asset Manager) *93*
 JP130001 *107*
 JP170000 (CustomerServiceManager) *109, 111*
 JP170001 (EquipmentManager) *87*
 JP170002 (Service Time Card Manager published business service) *74*
 JP310000 *445*
 JP410000 (Inventory Manager) *347, 348*
 JP420000 (Sales Order Manager) *389, 390*
 JP430000 (Procurement Manager) *457, 458*
 JP43A000 (Purchase Order Receipt Manager) *477, 478*
 JP43B000 (VoucherMatchManager) *487*
 JP480000 (WorkOrderManager) *445*
 JP510010 (ProjectManager) *323*
 JP90CD00 *109*

K

kit items *394*
 Known Invoice Match With Amount (R03B50D) *214*
 Known Invoice Match Without Amount (R03B50E) *214*

L

labor information *120, 120*
 list price information *380, 380, 381*
 Live Promotions Interface Processing (R45720Z) *399*
 Load Bank Statement report (R09612) *167, 167*
 location information *339, 340, 341, 342*
 lookupAddress Book (J0100009) *72*
 lookupAddressBook (J0100009) *227, 229*
 lookupBusinessUnits (J0000040) *72, 269, 269*
 lookupCurrencyCodes (J0000060) *72, 267, 269, 270*
 lookupJobCost (J5100001) *72*
 lookupUDC (J0000090) *269, 270*
 lookupWorkOrders (J4800001) *72, 445*

M

ManageAddressBook *72*
 master
 RTLPOUT *379*
 RTLPOUTDS *379*

N

naming conventions *11*
 Net Change Processor (J0000020) *8*
 NF-e Lot Processing Processor (J76B0021) *77*
 NfeCancelamento (JC76B003) *78*

NfeInutilizacao (JC76B005) 78
 NfeRecepcao (JC76B001) 78
 NfeRetRecepcao (JC76B002) 78

O

Object Management Workbench 7
 OPAPSRLR 371
 OPAPSRTG 371
 OPBOMRHDR 371
 OPBOMROUT 371
 OPBOMROUT real-time event 371
 Open Purchase Orders (R43632) 458
 Oracle Technical Catalog 19
 Order Processing (R31410) 446, 454
 OrderPromisingProcessor business service (J34A0010) 75
 organizational business unit/jobs cross reference 297, 298
 Outbound Asset Master Processor program (R1201Z1O) 87, 91, 163, 166
 Outbound Extraction - Ad Hoc Supplier Release Schedules program (R470622)
 375, 376
 Outbound Extraction - Forecast program (R47062) 375, 376
 Outbound Extraction - Supplier program (R470621) 375, 376
 Outbound Extraction program (R47142) 390, 418
 Outbound Supplier Recovery Processing (R174801Z2O) 124
 Outbound Work Order Extraction (R4801Z1X) 446, 454
 overview
 Accounts Payable Supplier 199
 General Ledger Account Ledger 127
 General Ledger Account Master 131
 Service Timecard 121
 Supplier Recovery Claim 123
 Overview
 Customer Service Work Order 109

P

P0100041 228
 P0100042 176
 P0100043 199
 P03B0001 (Speed Receipts Entry) 214
 P03B0011 151
 P03B102 (Standard Receipts Entry) 214
 P03B602 (Draft Entry) 214
 P0400048 141
 P0900049 127
 P1301150 (XPI Condition-Based Alerts) 96
 P1702650 (CRM Service Order Integration Processing Options) 114
 P4312 (PO Receipts) 477, 484
 P4312 (Purchase Order Receipts) 214
 P43XPI (Purchase Order XPI) 492
 P90701A program 12
 parent child information 239, 241, 243, 244, 244, 244
 parts list 371
 payment remittance advice
 container event 137
 detail information 139
 header information 138
 phone information 240, 242, 246
 phone numbers
 for customers 186
 for supplier 207, 209
 PLM (Agile Product Lifecycle Management) 69
 PO Dispatch Processor (J43E0030) 75
 PO Receipts (P4312) 477, 484
 price information 380, 380, 381
 Pricing business services 379

Process Inbound Bill of Material program (R3002Z1I) 383, 384
 Process Inbound Routing (R3003Z1I) 383, 384
 Process Inbound Samples for Blend program (R31BSPZ1) 443, 444
 Process Inbound Work Center program (R30006Z1I) 383, 383
 Process Inbound Work Day Calendar DC program (R0007Z1I) 445, 453
 process integration pack (PIP) 187
 Process Pending Employees report (R060116P) 281, 294
 processAddressBook (J0100001) 227, 228
 processAddressBookV2 (J0100001) 227, 229
 processARInvoiceProcessor (J03B0001) 152
 ProcessARInvoiceProcessor (J03B0001) 151
 ProcessARInvoiceRecurringInformationProcessor 151
 processContact (J0100003) 175, 227, 229
 processCustomer (J0100021) 175, 175
 processCustomerV2 183
 processCustomerV2 (J0100021) 175, 175
 processCustomerV3 183
 processEmployee web service operation (J0800040) 282
 processExpenseReportDetail 72
 processExpenseReportHeader 72
 processExpenseReports 72
 processFinancialForecast (J5100020) 323
 Processing Version Processor (J0000010) 8
 ProcessIntegrationTimeStamp (J3000030) 455
 processInventoryItem (J4100003) 347, 348
 processInventoryItemV2 (J4100003) 347, 348
 processPendingEmployee (J0800050) 289
 processProject (J5100010) 323
 processPurchaseOrder (J4300010) 457, 458
 processPurchaseOrder (J43B0010) 493
 processPurchaseOrderAcknowledge (J4300020) 457, 458
 processPurchaseOrderApproveReject 73
 processPurchaseOrderReceipt (J43A0010) 477, 478
 processPurchaseOrderReceiptV2 (J43A0010) 477, 478
 processPurchaseOrderV2 (J4300010) 457, 458
 processRSSApproveReject (J43E0080) 485
 processSalesOrder (J4200010) 389, 390
 processSalesOrderV2 (J4200010) 389, 390
 processSalesOrderV3 (J4200010) 389, 389, 390
 processSalesOrderV5 (J4200010) 390
 processSalesPriceAdjustment (J4200020) 389, 391
 processSalesPriceAdjustmentV2 (J4200020) 389, 391
 processServiceOrder (J1700010) 109, 110, 111
 processServiceOrderV2 (J1700010) 109, 110, 111
 processServiceTimeCard (J1700080) 74
 processSupplier (J0100031) 199
 processSupplierCatalogPrice (J4100002) 347, 348
 processSupplierCatalogPriceV2 (J4100002) 347, 348
 processSupplierV2 (J0100031) 199
 processSupplierV3 (J0100031) 199
 processVoucher (J0400001) 141, 144
 processVoucherMatch (J43B0010) 487
 processVoucherMatchV2 493
 processVoucherMatchV2 (J43B0010) 487
 processWBS (J5100030) 323
 Procurement Manager (JP430000) 457, 458
 ProjectManager (JP510010) 323
 Published business service 7
 published business services
 Address Book Manager (JP010000) 227, 228
 BusinessUnitManager (JP000006) 250, 260
 Capital Asset Manager (JP130000) 93
 CustomerServiceManager (JP170000) 109
 EquipmentManager (JP170001) 87
 FinancialComplianceManager (JP000040) 213
 FixedAssetManager (JP120000) 163

- Inventory Manager (JP410000) *347, 348*
 - overview *7*
 - Procurement Manager (JP430000) *457, 458*
 - Purchase Order Receipt Manager (JP43A000) *477, 478*
 - Sales Order Manager (JP420000) *389, 390*
 - Supplier Manager (JP010030) *200*
 - purchase order
 - RTRECOUT *477*
 - Purchase Order (PO) Dispatch *75*
 - Purchase Order Dispatch for Requisition Self Service *75*
 - purchase order hold release *474*
 - purchase order information *468, 470, 471, 471, 471, 473, 473*
 - Purchase Order Print program (R43500) *458, 475*
 - purchase order receipt information *482*
 - Purchase Order Receipt Manager (JP43A000) *477, 478*
 - Purchase Order Receipts (P4312) *214*
 - Purchase Order XPI (P43XPI) *492*
 - purchase orders
 - RTPODIST *457, 468*
 - RTPODTL *457, 468*
 - RTPODTL2 *457, 468*
 - RTPOHDR *457, 468*
 - RTPOHDR2 *457, 468*
 - RTPOHLDOUT *457, 468*
 - RTPOOUT2 *457, 468*
 - Purchase Orders
 - RTPOOUT *457, 468*
 - PurchaseOrderApprovalQueryProcessor (J4300040) *72, 73*
 - PurchaseOrderApproveRejectProcessor (J4300080) *73, 73*
 - PurchaseOrderEmployeeProfileQueryProcessor (J4300060) *72, 73*
- ## R
- R0006ZI *249, 258*
 - R0007Z1I *445, 453*
 - R0015Z1 (External Exchange Rate Processor program) *267, 268*
 - R01010Z (Address Book Batch Upload program) *228, 247*
 - R011110Z (Who's Who Batch Upload program) *228, 247*
 - R03010Z (Customer Master Batch Upload program) *191*
 - R03010Z (Customer Master Batch Upload) *176*
 - R03B11Z1A (Batch Invoice Processor program) *151, 160*
 - R03B50A (Invoice Selection Match) *214*
 - R03B50D (Known Invoice Match With Amount) *214*
 - R03B50E (Known Invoice Match Without Amount) *214*
 - R03B551 (Update Receipts Header Report) *161, 162*
 - R04010Z (Supplier Master Batch Upload program) *210*
 - R04010Z (Supplier Master Batch Upload) *199*
 - R04110Z (Voucher Batch Processor program) *146*
 - R04110Z (Voucher Batch Processor) *141*
 - R04110Z2 (Store and Forward Voucher Batch Processor Report) *146*
 - R04110Z2 (Store and Forward Voucher Batch Processor) *141*
 - R04110ZA (Batch Voucher Processor Report) *146*
 - R04110ZA (Batch Voucher Processor) *141, 487*
 - R04110ZB *141, 146*
 - R0411Z1 *141, 147*
 - R048101 (Recycle Recurring Vouchers) *214*
 - R05100Z1 *299, 302*
 - R05116Z1I (Time Entry Batch Processor) *307, 307*
 - R060116P (Process Pending Employees report) *281, 294*
 - R09110Z *127, 129*
 - R0911Z1 *127, 129*
 - R09612 *167, 167*
 - R1201Z1I (Inbound Asset Master Processor) *163, 165*
 - R1201Z1O (Outbound Asset Master Processor program) *87, 91, 163, 166*
 - R12120Z1I (Inbound Meter Readings Processor program) *105, 105*
 - R1310Z1I (Inbound Condition-Based Alerts Processor program) *93, 99, 99*
 - R14110 (Upload/Conversion program) *169, 169*
 - R1602Z1 *171, 171*
 - R173111Z2I (Inbound Warranty Claim Parts Detail Processing program) *123*
 - R173111Z2I (Inbound Warranty Claim Parts) *123*
 - R174801Z2I (Inbound Warranty Claim Processing program) *124*
 - R174801Z2I (Inbound Warranty Claim) *123*
 - R174801Z2O (batch export program) *123*
 - R174801Z2O (Outbound Supplier Recovery Processing) *124*
 - R30006Z1I (Process Inbound Work Center program) *383, 383*
 - R3002Z1I (Process Inbound Bill of Material program) *383, 384*
 - R3003Z1I *383, 384*
 - R30161Z1I (Inbound Kanban Transaction program) *369, 369*
 - R31113Z1I *445, 446, 453*
 - R31114Z1I (Inbound Completion Processor) *445, 453*
 - R31122Z1I *445, 446, 453*
 - R31123Z1I (Inbound Super Backflush Processor program) *445, 453*
 - R31410 *446, 454*
 - R31422 *446, 454*
 - R31B65AZ1I (Inbound Work Order for Blend program) *443, 443*
 - R31BSPZ1 *443, 444*
 - R3460Z1I (Forecast Inbound Processor) *335, 335*
 - R34A530 *176, 192*
 - R3711Z1I (Batch Test Results program) *437*
 - R4101Z1I (Item Master Inbound Transaction Process) *347, 363, 366*
 - R4101ZO *363*
 - R41021ZX (Item Location Extraction) *339, 342*
 - R4105Z1I (Item Cost Inbound Transaction Process program) *361, 362*
 - R41413 *337, 337*
 - R4141Z1I (Cycle Count Inbound Transaction Process program) *337, 337, 343*
 - R41610 *343, 343*
 - R42565 (Invoice Print program) *385, 388, 390*
 - R4304010 *487, 494*
 - R43092Z1I (Receipt Routing Inbound Processor) *477, 483*
 - R4311Z1I (Inbound Purchase Order) *457, 474*
 - R43500 (Purchase Order Print program) *458, 475*
 - R43632 *458*
 - R45720Z (Live Promotions Interface Processing) *399*
 - R4611Z1I *441, 441*
 - R47021 *457, 475*
 - R47031 (EDI Shipping Notice Edit/Create program) *426, 435, 477, 483*
 - R47032 (EDI Advance Ship Notice Extraction program) *426, 436*
 - R47041 (Edit/Create Voucher) *141, 146*
 - R470412 *487, 494*
 - R47058 *133, 135*
 - R47058 (Update as Sent program) *139*
 - R47058 (Update as Sent) *137*
 - R47061 (EDI Planning Schedule Edit/Create program) *375, 375*
 - R47062 (Outbound Extraction - Forecast program) *375, 376*
 - R470621 (Outbound Extraction - Supplier program) *375, 376*
 - R470622 (Outbound Extraction - Ad Hoc Supplier Release Schedules program) *375, 376*
 - R47062C (EDI Planning Schedule Extraction Conversion program) *375, 376*
 - R47071 (EDI Receiving Advice Edit/Create – Purchasing) *483*
 - R47071 (EDI Receiving Advice Edit/Create) *477*
 - R47072 (EDI Receiving Advice Extraction program) *477, 484*
 - R47102 *458*
 - R47111 (Update to Electronic Receipts program) *161, 161*
 - R47121 (EDI Product Activity Data Edit/Create program) *345, 345*
 - R47122 (EDI Product Activity Data Extraction program) *345, 346*
 - R47132 (EDI Purchase Order Change Extraction program) *458, 475*
 - R47141 (EDI Inbound PO Change Acknowledgment program) *457, 474*
 - R47142 *418*
 - R47171 (EDI Inbound Demand Edit/Update Report) *331, 331*
 - R47761Z1 (Ship and Debit Claim Output program) *421, 421*
 - R47762Z1 *421, 421*
 - R4801Z1X (Outbound Work Order Extraction) *446, 454*

Rate Shopping Processor (J49T0010) *76*

RCW02Z1 (Style Item Inbound Processor report) *423, 423*

Real-time event

- RTEQORDHDR *107, 108*
- RTEQORDOUT *107, 108*
- RTEQPHDR *87, 101, 101*
- RTEQPOUT *87, 101, 101*
- RTLBRHDR *109, 120*
- RTLBROUT *109, 120*

real-time events (RTes)

- defining *12*
- overview *11*

receipt information *482*

Receipt Routing Inbound Processor (R43092Z11) *477, 483*

Recycle Recurring Vouchers (R048101) *214*

rejectExpenesReport *71*

releaseExpenseReport *72*

remittance advice

- container event *137*
- detail information *139*
- header information *138*

Requisition Self Service *74*

reserveExpenseReport *72*

resource competency

- batch import *302*

RosettaNet standards *386, 387*

routing information *371*

RSSApproverQueryProcessor (J43E0050) *485*

RSSDetailQueryProcessor (J43E0060) *485*

RTABEAOUT *227, 238*

RTABEAOUT real-time event *240, 243*

RTABEAOUT2 *227, 238*

RTABHDR *227, 238*

RTABHDR real-time event *240, 242*

RTABHDR2 *227, 238*

RTABHOUT *227, 238*

RTABHOUT2 *227, 238*

RTABOUT *227, 238*

RTABOUT real-time event *227, 238, 239, 241*

RTABOUT2 *227, 238*

RTABOUT2 real-time event *227, 238*

RTABPHOUT real-time event *240, 242*

RTAMOUT *127, 127, 131*

RTAMOUT real-time event *127, 127, 131, 131*

RTARIVOUT *151, 151, 158, 158*

RTARIVOUTB *151, 158*

RTAVIROUTA *159*

RTAVIROUTB *160*

RTBJESOUT *127, 127*

RTBJESOUT real-time event *127, 127, 128*

RTBUOUT *249, 253*

RTBUOUT real-time event *249, 253, 253, 254*

RTBUOUT3 *249, 253, 256*

RTBUOUTA *249, 249, 253, 253*

RTBUOUTA real-time event *249, 249, 253, 253, 254, 255*

RTBUOUTA3 *249, 253, 257*

RTBUOUTB *249, 249, 253, 253*

RTBUOUTB real-time event *249, 249, 253, 253, 254, 256*

RTBUOUTB2 *249, 253*

RTBUOUTB2 real-time event *249, 253*

RTBUOUTB3 *249, 253*

RTCMAAOUT real-time event *189*

RTCMCCOUT *175, 184*

RTCMCCOUT real-time event *175, 184, 187*

RTCMCCOUT2 real-time event *189*

RTCMEAOUT *175, 184*

RTCMEAOUT real-time event *175, 184, 186*

RTCMEAOUT2 real-time event *188*

RTCMHDR *175, 184*

RTCMHDR real-time event *185, 191*

RTCMHDR2 *176, 184*

RTCMHDR2 real-time event *176, 184, 188*

RTCMOUT *175, 184*

RTCMOUT real-time event *175, 175, 184, 184, 184, 190*

RTCMOUT2 *175, 184*

RTCMOUT2 real-time event *175, 184, 187*

RTCMOUT3 *176, 184*

RTCMOUT3 real-time event *176, 184*

RTCMPHOUT *175, 184*

RTCMPHOUT real-time event *175, 184, 186*

RTCMPHOUT2 real-time event *189*

RTCOOUT *259, 263*

RTCOOUT real-time event *263*

RTCOOUT2 *259, 264*

RTCOSTOUT *361, 361*

RTCOSTOUT real-time event *361, 361, 361*

RTCOSTOUTA *361, 361*

RTCOSTOUTA real-time event *361, 361, 362*

RTCPOUT *279, 279*

RTCPOUT real-time event *279, 279, 279*

RTCPOUTB *279, 279*

RTCPOUTB real-time event *279, 279, 280*

RTCROUT *267*

RTCROUT real-time event *267, 267*

RTDADTL real-time event *150*

RTDAOUT real-time event *149*

RTDAOUT real-time event *149*

RTEMOUT *281, 290*

RTEMOUT real-time event *281, 290, 290*

RTEMOUTB *281, 290*

RTEMOUTB real-time event *281, 290, 291*

RTEMOUTC *281, 290*

RTEMOUTC real-time event *281, 290, 291*

RTEMOUTD *281, 290*

RTEMOUTD real-time event *281, 290, 292*

RTEQORDHDR *107, 108*

RTEQORDHDR real-time event *108*

RTEQORDOUT *107, 108*

RTEQORDOUT real-time event *108*

RTEQPHDR *87, 101, 101*

RTEQPHDR real-time event *101, 102*

RTEQPOUT *87, 101, 101*

RTesreal-time events (RTes) *11*

RTHBCMBTEA *313, 313*

RTHBCMBTPH *313, 313*

RTHBCMCEA *313, 313*

RTHBCMHDR *313, 313*

RTHBCMOUT *313, 313, 319, 319*

RTHBCMSTEA *313, 314*

RTHBCMSTPH *313, 314*

RTHBCOMMIT *325, 325*

RTHBLMBEA *319, 319*

RTHBLMBPH *319, 319*

RTHBLMHDR *319, 319*

RTHBVAEA *309, 309, 310*

RTHBVAHDR *309, 309*

RTHBVAAOUT *309, 309*

RTHBVAPH *309, 309*

RTHHBCMCPH *313, 313*

RTIBHDR real-time event *340, 342*

RTIBOUT *339, 339*

RTIBOUT real-time event *339, 339, 339, 341*

RTIBOUT2 339, 339
RTIBOUT2 real-time event 339, 339
RTIBOUTA 339, 339
RTIBOUTA real-time event 339, 339
RTIBOUTB 339, 339
RTIBOUTB real-time event 339, 339
RTIMOUT 363, 364
RTIMOUT real-time event 363, 364, 364, 365
RTIMOUT2 363, 364
RTIMOUT2 real-time event 363, 364
RTIMOUTA 363, 364
RTIMOUTA real-time event 363, 364, 364, 365
RTIMOUTB 363, 364
RTIMOUTB real-time event 363, 364
RTINVDTL 385, 385
RTINVDTL real-time event 385, 385
RTINVDTL2 385, 386
RTINVDTL2 real-time event 385, 386
RTINVHDR 385, 385
RTINVHDR real-time event 385, 385
RTINVOUT 385, 385
RTINVOUT real-time event 385, 385, 386, 387
RTINVOUT2 385, 386
RTINVOUT2 real-time event 385, 386
RTINVOUTA real-time event 386
RTINVOUTB real-time event 387, 388
RTJBOUT 295, 295
RTJBOUT real-time event 295, 295, 295
RTJBOUTB 295, 295
RTJBOUTB real-time event 295, 295, 296
RTJBUOUT 295, 295
RTJBUOUT real-time event 295, 295, 297
RTJBUOUTB 295, 295
RTJBUOUTB real-time event 295, 295, 298
RTJDOUT 295, 295
RTJDOUT real-time event 295, 295, 296
RTJDOUTB 295, 295
RTJDOUTB real-time event 297
RTJDOUTB real-time event 295, 295
RTLBRHDR 109, 120
RTLBRHDR real-time event 120
RTLBROUT 109, 120
RTLBROUT real-time event 120
RTLPOUT real-time event 379, 380, 381
RTLPOUT2 379
RTLPOUTDS real-time event 379, 380
RTPCOUT 227, 238
RTPCOUT real-time event 227, 238, 243, 244
RTPCOUT2 228, 238
RTPCOUT2 real-time event 228, 238
RTPCOUTDS 227, 238
RTPCOUTDS real-time event 244, 244
RTPCOUTDS2 228, 238
RTPODIST real-time event 457, 468, 471
RTPODTL real-time event 457, 468, 471, 473
RTPODTL2 real-time event 457, 468
RTPOHDR real-time event 457, 468, 470, 473
RTPOHDR2 real-time event 457, 468
RTPOHLDOUT real-time event 457, 468, 474
RTPOOUT real-time event 457, 468, 468, 471
RTPOOUT2 real-time event 457, 468
RTPOSTOUT real-time event 128
RTRADTL 133, 133
RTRADTL real-time event 137, 137, 139
RTRAHDR 133, 133
RTRAHDR real-time event 137, 137, 138
RTRAOUT 133, 133
RTRAOUT real-time event 137, 137, 137
RTRCHDR 299, 299
RTRCHDR real-time event 299, 299, 301
RTRCOUT 299, 299
RTRCOUT real-time event 299, 299, 301
RTRECOUT real-time event 477, 482
RTSHPEAOUT 228, 238
RTSHPEAOUT real-time event 246
RTSHPHDR 228, 238
RTSHPHDR real-time event 245
RTSHPNOUT real-time event 425, 429, 429, 433
RTSHPNOUT2 real-time event 425, 429
RTSHPNOUTA real-time event 425, 429, 430
RTSHPNOUTB real-time event 425, 429, 431, 435
RTSHPNOUTC real-time event 425, 429, 431
RTSHPNOUTD real-time event 425, 429, 431
RTSHPNOUTE real-time event 425, 429, 431
RTSHPNOUTF real-time event 425, 429, 432, 434
RTSHPNOUTG real-time event 425, 429, 432
RTSHPNOUTH real-time event 425, 429, 432
RTSHPNOUTI real-time event 425, 429, 432
RTSHPNOUTJ real-time event 425, 429, 433, 435
RTSHPNOUTK real-time event 425, 429, 433
RTSHPNOUTL real-time event 425, 429
RTSHPNOUTM real-time event 425, 429
RTSHPNOUTP real-time event 425, 429
RTSHPOUT 228, 238
RTSHPOUT real-time event 245
RTSHPPHOUT 228, 238
RTSHPPHOUT real-time event 246
RTSLDTL real-time event 300
RTSLHDR real-time event 300
RTSLOUT 299, 299
RTSLOUT real-time event 299, 299, 300
RTSMEAOUT real-time event 208, 210
RTSMHDR real-time event 207, 209
RTSMOUT 199, 205
RTSMOUT real-time event 199, 205, 206, 208
RTSMOUT2 199, 206
RTSMOUT2 real-time event 199, 206
RTSMPHOUT real-time event 207, 209
RTSODTL 389, 413
RTSODTL real-time event 389, 413, 415, 417
RTSODTL2 389, 413
RTSODTL2 real-time event 389, 413
RTSODTL3 390, 413
RTSODTL3 real-time event 390, 413
RTSOHDR 389, 413
RTSOHDR real-time event 389, 413, 414, 417
RTSOHDR2 389, 413
RTSOHDR2 real-time event 389, 413
RTSOHDR3 390, 413
RTSOHDR3 real-time event 390, 413
RTSOOUT 389, 413
RTSOOUT real-time event 389, 413, 413, 416
RTSOOUT2 389, 413
RTSOOUT2 real-time event 389, 413
RTSOOUT3 390, 413
RTSOOUT3 real-time event 390, 413
RTUOMDTL 439, 439
RTUOMDTL real-time event 439, 439, 440
RTUOMOUT 439, 439
RTUOMOUT real-time event 439, 439, 439
RTWOHDR real-time event 445, 451, 452
RTWOOUT real-time event 445, 451, 452

S

sales order information
 container event *413, 416*
 detail information *415, 417*
 header information *414, 417*

Sales Order Manager (JP420000) *389, 390*

sales orders *393*

SCP Customer Master Information Extract program (R34A530) *176, 192*

serial number information *432*

Service Time Card Manager published business service (JP170002) *74*

Service Timecard business object *121*

setup *9*

setUserValues and getUserValues (J0000070) *269, 270*

Ship and Debit Claim Output program (R47761Z1) *421, 421*

Ship and Debit Update program (R47762Z1) *421, 421*

ship to header information for a branch plant *245*

shipment
 RTSHPNOUT *425, 429*
 RTSHPNOUT2 *425, 429*
 RTSHPNOUTA *425, 429*
 RTSHPNOUTB *425, 429*
 RTSHPNOUTC *425, 429*
 RTSHPNOUTD *425, 429*
 RTSHPNOUTE *425, 429*
 RTSHPNOUTF *425, 429*
 RTSHPNOUTG *425, 429*
 RTSHPNOUTH *425, 429*
 RTSHPNOUTI *425, 429*
 RTSHPNOUTJ *425, 429*
 RTSHPNOUTK *425, 429*
 RTSHPNOUTL *425, 429*
 RTSHPNOUTM *425, 429*
 RTSHPNOUTP *425, 429*

shipment confirmation
 container event *429, 433*
 container item information *432, 434*
 container order information *431*
 global carrier code *431*
 serial number information *432*
 shipment detail *431*
 shipment header *431, 435*
 shipment information *430*
 sub-container detail information *432*
 sub-container item information *433, 435*
 sub-container order information *432*
 sub-container serial number information *433*

shipment detail *431*

shipment header *431, 435*

Shopping Cart Processor *74*

Speed Receipts Entry (P03B0001) *214*

Standard Receipts Entry (P03B102) *214*

Store and Forward Journal Entry Upload *127, 129*

Store and Forward Upload (R0411Z1) *141, 147*

Store and Forward Voucher Batch Processor (R04110Z2) *141*

Store and Forward Voucher Batch Processor Report (R04110Z2) *146*

Style Item Inbound Processor report (RCW02Z1) *423, 423*

sub-container
 detail information *432*
 item information *433, 435*
 order information *432*
 sub-container *433*

submitExpenseReport *71*

supplier email addresses *208, 210*

supplier information *206, 207, 208, 209*

Supplier Manager (JP010030) *200*

Supplier Master Batch Upload (R04010Z) *199*

Supplier Master Batch Upload program (R04010Z) *210*

Supplier Master MBF - PO (P0100043) *199*

supplier phone number *207, 209*

Supplier Recovery Claim *123, 124*

system setup *9*

T

Tag Inventory Update (R41610) *343, 343*

Technical Catalog *19*

Temporary Table Batch Processing *16*

Time Entry Batch Processor (R05116Z11) *307, 307*

U

unit cost information *361, 362*

unit of measure conversions *439, 440*

Update as Sent (R47058) *137*

Update as Sent program (R47058) *133, 135, 139*

Update Receipts Header Report (R03B551) *161, 162*

Update Resource Competencies (R05100Z1) *299, 302*

Update to Electronic Receipts program (R47111) *161, 161*

UpdateProjectAccount business function (B5100140) *323*

UpdateProjectFinancialForecasts business function (B5100160) *323*

Upload/Conversion program (R14110) *169, 169*

utilities
 Entity Processor (J0100010) *8*
 GL Account Processor (J0900010) *8*
 Inventory Item ID Processor (J4100010) *8*
 Net Change Processor (J0000020) *8*
 overview *8*
 Processing Version Processor (J0000010) *8*

V

VAT Information Exchange System (VIES) *76*

VIESAppServerOASOrWLS (J740105) *77*

viewing hierarchy information in javadoc *64*

viewing uses of an object in javadoc *59*

viewing value object and field information for javadoc *47*

Voucher Batch Processor (R04110Z) *141*

Voucher Batch Processor program (R04110Z) *146*

Voucher Match Automation Driver program (R4304010) *487, 494*

VoucherMatchManager (JP43B000) *487*

W

Warranty Claim *123*

Warranty Claim Batch Export Program *124*

Warranty Claim Batch Import Program *123*

Web Services Callout *75*

Who's Who Batch Upload program (R011110Z) *228, 247*

work order
 RTWOHDR *445, 451*
 RTWOOUT *445, 451*

work order information *108, 108, 452, 452*

work order labor information *120, 120*

work order parts list and routing *371*

WorkOrderManager *72*

WorkOrderManager (JP480000) *445*

WorkOrders (J4800001) *446*

X

XAPI events *11*

XAPI Condition-Based Alerts (P1301150) *96*

